



Red Hat OpenStack Platform 9

Configuration Reference

Configuring Red Hat OpenStack Platform environments

Red Hat OpenStack Platform 9 Configuration Reference

Configuring Red Hat OpenStack Platform environments

OpenStack Documentation Team

Red Hat Customer Content Services

rhos-docs@redhat.com

Legal Notice

Copyright © 2016 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution-Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This document is for system administrators who want to look up configuration options. It contains lists of configuration options available with OpenStack and uses auto-generation to generate options and the descriptions from the code for each project. It includes sample configuration files.

Table of Contents

CHAPTER 1. BARE METAL	4
CHAPTER 2. BLOCK STORAGE	29
2.1. VOLUME DRIVERS	29
2.2. BACKUP DRIVERS	125
2.3. BLOCK STORAGE SAMPLE CONFIGURATION FILES	129
2.4. LOG FILES USED BY BLOCK STORAGE	187
2.5. FIBRE CHANNEL ZONE MANAGER	187
2.6. ADDITIONAL OPTIONS	191
2.7. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR OPENSTACK BLOCK STORAGE	226
CHAPTER 3. COMPUTE	240
3.1. OVERVIEW OF NOVA.CONF	240
3.2. CONFIGURE LOGGING	242
3.3. CONFIGURE AUTHENTICATION AND AUTHORIZATION	242
3.4. CONFIGURE RESIZE	242
3.5. DATABASE CONFIGURATION	243
3.6. CONFIGURE THE OSLO RPC MESSAGING SYSTEM	243
3.7. CONFIGURE THE COMPUTE API	249
3.8. CONFIGURE THE EC2 API	251
3.9. FIBRE CHANNEL SUPPORT IN COMPUTE	251
3.10. ISCSI INTERFACE AND OFFLOAD SUPPORT IN COMPUTE	251
3.11. HYPERVISORS	253
3.12. SCHEDULING	258
3.13. CELLS	276
3.14. CONDUCTOR	281
3.15. EXAMPLE NOVA.CONF CONFIGURATION FILES	281
3.16. COMPUTE LOG FILES	285
3.17. COMPUTE SAMPLE CONFIGURATION FILES	285
3.18. NEW, UPDATED AND DEPRECATED OPTIONS IN KILO FOR OPENSTACK COMPUTE	354
CHAPTER 4. DASHBOARD	366
4.1. CONFIGURE THE DASHBOARD	366
4.2. ADDITIONAL SAMPLE CONFIGURATION FILES	371
4.3. DASHBOARD LOG FILES	389
CHAPTER 5. DATABASE SERVICE	390
5.1. CONFIGURE THE DATABASE	407
5.2. CONFIGURE THE RPC MESSAGING SYSTEM	410
5.3. NEW, UPDATED AND DEPRECATED OPTIONS IN LIBERTY FOR DATABASE SERVICE	415
CHAPTER 6. DATA PROCESSING SERVICE	423
6.1. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR DATA PROCESSING SERVICE	447
CHAPTER 7. IDENTITY SERVICE	449
7.1. IDENTITY SERVICE CONFIGURATION FILE	449
7.2. IDENTITY SERVICE SAMPLE CONFIGURATION FILES	485
7.3. NEW, UPDATED AND DEPRECATED OPTIONS IN KILO FOR OPENSTACK IDENTITY	528
CHAPTER 8. IMAGE SERVICE	534
8.1. CONFIGURE THE API	550
8.2. CONFIGURE THE RPC MESSAGING SYSTEM	555
8.3. CONFIGURE IMAGE CACHE	563

8.4. SUPPORT FOR ISO IMAGES	566
8.5. CONFIGURE BACK ENDS	567
8.6. IMAGE SERVICE SAMPLE CONFIGURATION FILES	573
8.7. NEW, UPDATED AND DEPRECATED OPTIONS IN LIBERTY FOR OPENSTACK IMAGE SERVICE	603
CHAPTER 9. NETWORKING	605
9.1. NETWORKING CONFIGURATION OPTIONS	605
9.2. LOG FILES USED BY NETWORKING	705
9.3. NETWORKING SAMPLE CONFIGURATION FILES	706
9.4. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR OPENSTACK NETWORKING	736
CHAPTER 10. OBJECT STORAGE	741
10.1. INTRODUCTION TO OBJECT STORAGE	741
10.2. OBJECT STORAGE GENERAL SERVICE CONFIGURATION	741
10.3. OBJECT SERVER CONFIGURATION	743
10.4. OBJECT EXPIRER CONFIGURATION	758
10.5. CONTAINER SERVER CONFIGURATION	764
10.6. CONTAINER SYNC REALMS CONFIGURATION	775
10.7. CONTAINER RECONCILER CONFIGURATION	778
10.8. ACCOUNT SERVER CONFIGURATION	781
10.9. PROXY SERVER CONFIGURATION	791
10.10. PROXY SERVER MEMCACHE CONFIGURATION	817
10.11. RSYNCD CONFIGURATION	818
10.12. CONFIGURE OBJECT STORAGE FEATURES	819
10.13. NEW, UPDATED AND DEPRECATED OPTIONS IN LIBERTY FOR OPENSTACK OBJECT STORAGE	837
CHAPTER 11. ORCHESTRATION	838
11.1. CONFIGURE APIS	852
11.2. CONFIGURE CLIENTS	859
11.3. CONFIGURE THE RPC MESSAGING SYSTEM	865
11.4. ORCHESTRATION LOG FILES	873
11.5. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR ORCHESTRATION SERVICE	874
CHAPTER 12. TELEMETRY	875
12.1. TELEMETRY SAMPLE CONFIGURATION FILES	898
12.2. NEW, UPDATED AND DEPRECATED OPTIONS IN KILO FOR TELEMETRY	931
APPENDIX A. THE POLICY.JSON FILE	939
A.1. EXAMPLES	939
A.2. SYNTAX	941
A.3. OLDER SYNTAX	942
APPENDIX B. FIREWALLS AND DEFAULT PORTS	943

CHAPTER 1. BARE METAL

The Bare metal service is capable of managing and provisioning physical machines. The configuration file of this module is `/etc/ironic/ironic.conf`.

The following tables provide a comprehensive list of the Bare metal service configuration options.

Table 1.1. Description of agent configuration options

Configuration option = Default value	Description
[agent]	
agent_api_version = <i>v1</i>	(StrOpt) API version to use for communicating with the ramdisk agent.
agent_erase_devices_priority = <i>None</i>	(IntOpt) Priority to run in-band erase devices via the Ironic Python Agent ramdisk. If unset, will use the priority set in the ramdisk (defaults to 10 for the GenericHardwareManager). If set to 0, will not run during cleaning.
agent_pxe_append_params = <i>nofb nomodeset vga=normal</i>	(StrOpt) Additional append parameters for baremetal PXE boot.
agent_pxe_bootfile_name = <i>pxelinux.0</i>	(StrOpt) Neutron bootfile DHCP parameter.
agent_pxe_config_template = <i>\$pybasedir/drivers/modules/agent_config.template</i>	(StrOpt) Template file for PXE configuration.
heartbeat_timeout = <i>300</i>	(IntOpt) Maximum interval (in seconds) for agent heartbeats.
manage_tftp = <i>True</i>	(BoolOpt) Whether Ironic will manage TFTP files for the deploy ramdisks. If set to False, you will need to configure your own TFTP server that allows booting the deploy ramdisks.

Table 1.2. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>openstack</i>	(StrOpt) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
notification_driver = <i>[]</i>	(MultiStrOpt) Driver or drivers to handle sending notifications.

Configuration option = Default value	Description
notification_topics = <i>notifications</i>	(ListOpt) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(StrOpt) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the <code>rpc_backend</code> option and driver specific configuration.

Table 1.3. Description of AMT configuration options

Configuration option = Default value	Description
[amt]	
action_wait = <i>10</i>	(IntOpt) Amount of time (in seconds) to wait, before retrying an AMT operation
max_attempts = <i>3</i>	(IntOpt) Maximum number of times to attempt an AMT operation, before failing
protocol = <i>http</i>	(StrOpt) Protocol used for AMT endpoint, support http/https

Table 1.4. Description of API configuration options

Configuration option = Default value	Description
[api]	
host_ip = <i>0.0.0.0</i>	(StrOpt) The listen IP for the Ironic API server.
max_limit = <i>1000</i>	(IntOpt) The maximum number of items returned in a single response from a collection resource.
port = <i>6385</i>	(IntOpt) The port for the Ironic API server.

Table 1.5. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(StrOpt) Service user password.
admin_tenant_name = <i>admin</i>	(StrOpt) Service tenant name.

Configuration option = Default value	Description
admin_token = <i>None</i>	(StrOpt) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(StrOpt) Service username.
auth_admin_prefix =	(StrOpt) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
auth_host = <i>127.0.0.1</i>	(StrOpt) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_plugin = <i>None</i>	(StrOpt) Name of the plugin to load
auth_port = <i>35357</i>	(IntOpt) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_protocol = <i>https</i>	(StrOpt) Protocol of the admin Identity API endpoint (http or https). Deprecated, use <code>identity_uri</code> .
auth_section = <i>None</i>	(StrOpt) Config Section from which to load plugin specific options
auth_uri = <i>None</i>	(StrOpt) Complete public Identity API endpoint.
auth_version = <i>None</i>	(StrOpt) API version of the admin Identity API endpoint.
cache = <i>None</i>	(StrOpt) Env key for the swift cache.
cafile = <i>None</i>	(StrOpt) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate
check_revocations_for_cached = <i>False</i>	(BoolOpt) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.

Configuration option = Default value	Description
delay_auth_decision = <i>False</i>	(BoolOpt) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(StrOpt) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(ListOpt) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(IntOpt) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(IntOpt) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(StrOpt) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/
include_service_catalog = <i>True</i>	(BoolOpt) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate

Configuration option = Default value	Description
<code>memcache_pool_conn_get_timeout = 10</code>	(IntOpt) (Optional) Number of seconds that an operation will wait to get a memcache client connection from the pool.
<code>memcache_pool_dead_retry = 300</code>	(IntOpt) (Optional) Number of seconds memcached server is considered dead before it is tried again.
<code>memcache_pool_maxsize = 10</code>	(IntOpt) (Optional) Maximum total number of open connections to every memcached server.
<code>memcache_pool_socket_timeout = 3</code>	(IntOpt) (Optional) Socket timeout in seconds for communicating with a memcache server.
<code>memcache_pool_unused_timeout = 60</code>	(IntOpt) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
<code>memcache_secret_key = None</code>	(StrOpt) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
<code>memcache_security_strategy = None</code>	(StrOpt) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. Acceptable values are MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.
<code>memcache_use_advanced_pool = False</code>	(BoolOpt) (Optional) Use the advanced (eventlet safe) memcache client pool. The advanced pool will only work under python 2.x.
<code>memcached_servers = None</code>	(ListOpt) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.
<code>revocation_cache_time = 10</code>	(IntOpt) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
<code>signing_dir = None</code>	(StrOpt) Directory used to cache files related to PKI tokens.

Configuration option = Default value	Description
token_cache_time = 300	(IntOpt) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 1.6. Description of authorization configuration options

Configuration option = Default value	Description
[DEFAULT]	
auth_strategy = <i>keystone</i>	(StrOpt) Method to use for authentication: noauth or keystone.

Table 1.7. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
bindir = <i>/usr/local/bin</i>	(StrOpt) Directory where ironic binaries are installed.
enabled_drivers = <i>pxe_ipmitool</i>	(ListOpt) Specify the list of drivers to load during service initialization. Missing drivers, or drivers which fail to initialize, will prevent the conductor service from starting. The option default is a recommended set of production-oriented drivers. A complete list of drivers present on your system may be found by enumerating the "ironic.drivers" entryptpoint. An example may be found in the developer documentation online.
fatal_deprecations = <i>False</i>	(BoolOpt) Enables or disables fatal status of deprecations.
force_raw_images = <i>True</i>	(BoolOpt) Force backing images to raw format.
grub_config_template = <i>\$pybasedir/common/grub_conf.template</i>	(StrOpt) Template file for grub configuration file.
hash_distribution_replicas = <i>1</i>	(IntOpt) [Experimental Feature] Number of hosts to map onto each hash partition. Setting this to more than one will cause additional conductor services to prepare deployment environments and potentially allow the Ironic cluster to recover more quickly if a conductor instance is terminated.

Configuration option = Default value	Description
hash_partition_exponent = 5	(IntOpt) Exponent to determine number of hash partitions to use when distributing load across conductors. Larger values will result in more even distribution of load and less load when rebalancing the ring, but more memory usage. Number of partitions per conductor is $(2^{\text{hash_partition_exponent}})$. This determines the granularity of rebalancing: given 10 hosts, and an exponent of the 2, there are 40 partitions in the ring. A few thousand partitions should make rebalancing smooth in most cases. The default is suitable for up to a few hundred conductors. Too many partitions has a CPU impact.
host = <i>sd-52009.dedibox.fr</i>	(StrOpt) Name of this node. This can be an opaque identifier. It is not necessarily a hostname, FQDN, or IP address. However, the node name must be valid within an AMQP key.
isolinux_bin = <i>/usr/lib/syslinux/isolinux.bin</i>	(StrOpt) Path to isolinux binary file.
isolinux_config_template = <i>\$pybasedir/common/isolinux_config.template</i>	(StrOpt) Template file for isolinux configuration file.
memcached_servers = <i>None</i>	(ListOpt) Memcached servers or None for in process cache.
my_ip = <i>10.0.0.1</i>	(StrOpt) IP address of this host.
parallel_image_downloads = <i>False</i>	(BoolOpt) Run image downloads and raw format conversions in parallel.
periodic_interval = <i>60</i>	(IntOpt) Seconds between running periodic tasks.
pybasedir = <i>/usr/lib/python/site-packages/ironic/ironic</i>	(StrOpt) Directory where the ironic python module is installed.
rootwrap_config = <i>/etc/ironic/rootwrap.conf</i>	(StrOpt) Path to the rootwrap configuration file to use for running commands as root.
run_external_periodic_tasks = <i>True</i>	(BoolOpt) Some periodic tasks can be run in a separate process. Should we run them here?
state_path = <i>\$pybasedir</i>	(StrOpt) Top-level directory for maintaining ironic's state.
tempdir = <i>None</i>	(StrOpt) Explicitly specify the temporary working directory.

Table 1.8. Description of conductor configuration options

Configuration option = Default value	Description
[conductor]	
api_url = <i>None</i>	(StrOpt) URL of Ironic API service. If not set ironic can get the current value from the keystone service catalog.
check_provision_state_interval = <i>60</i>	(IntOpt) Interval between checks of provision timeouts, in seconds.
clean_nodes = <i>True</i>	(BoolOpt) Cleaning is a configurable set of steps, such as erasing disk drives, that are performed on the node to ensure it is in a baseline state and ready to be deployed to. This is done after instance deletion, and during the transition from a "managed" to "available" state. When enabled, the particular steps performed to clean a node depend on which driver that node is managed by; see the individual driver's documentation for details. NOTE: The introduction of the cleaning operation causes instance deletion to take significantly longer. In an environment where all tenants are trusted (eg, because there is only one tenant), this option could be safely disabled.
configdrive_swift_container = <i>ironic_configdrive_container</i>	(StrOpt) Name of the Swift container to store config drive data. Used when configdrive_use_swift is True.
configdrive_use_swift = <i>False</i>	(BoolOpt) Whether to upload the config drive to Swift.
deploy_callback_timeout = <i>1800</i>	(IntOpt) Timeout (seconds) for waiting callback from deploy ramdisk. 0 - unlimited.
force_power_state_during_sync = <i>True</i>	(BoolOpt) During sync_power_state, should the hardware power state be set to the state recorded in the database (True) or should the database be updated based on the hardware state (False).
heartbeat_interval = <i>10</i>	(IntOpt) Seconds between conductor heart beats.
heartbeat_timeout = <i>60</i>	(IntOpt) Maximum time (in seconds) since the last check-in of a conductor.
inspect_timeout = <i>1800</i>	(IntOpt) Timeout (seconds) for waiting for node inspection. 0 - unlimited.
node_locked_retry_attempts = <i>3</i>	(IntOpt) Number of attempts to grab a node lock.

Configuration option = Default value	Description
<code>node_locked_retry_interval = 1</code>	(IntOpt) Seconds to sleep between node lock attempts.
<code>periodic_max_workers = 8</code>	(IntOpt) Maximum number of worker threads that can be started simultaneously by a periodic task. Should be less than RPC thread pool size.
<code>power_state_sync_max_retries = 3</code>	(IntOpt) During sync_power_state failures, limit the number of times Ironic should try syncing the hardware node power state with the node power state in DB
<code>send_sensor_data = False</code>	(BoolOpt) Enable sending sensor data message via the notification bus
<code>send_sensor_data_interval = 600</code>	(IntOpt) Seconds between conductor sending sensor data message to ceilometer via the notification bus.
<code>send_sensor_data_types = ALL</code>	(ListOpt) List of comma separated metric types which need to be sent to Ceilometer. The default value, "ALL", is a special value meaning send all the sensor data.
<code>sync_local_state_interval = 180</code>	(IntOpt) When conductors join or leave the cluster, existing conductors may need to update any persistent local state as nodes are moved around the cluster. This option controls how often, in seconds, each conductor will check for nodes that it should "take over". Set it to a negative value to disable the check entirely.
<code>sync_power_state_interval = 60</code>	(IntOpt) Interval between syncing the node power state to the database, in seconds.
<code>workers_pool_size = 100</code>	(IntOpt) The size of the workers greenthread pool.

Table 1.9. Description of console configuration options

Configuration option = Default value	Description
[console]	
<code>subprocess_checking_interval = 1</code>	(IntOpt) Time interval (in seconds) for checking the status of console subprocess.
<code>subprocess_timeout = 10</code>	(IntOpt) Time (in seconds) to wait for the console subprocess to start.

Configuration option = Default value	Description
terminal = <i>shellinaboxd</i>	(StrOpt) Path to serial console terminal program
terminal_cert_dir = <i>None</i>	(StrOpt) Directory containing the terminal SSL cert(PEM) for serial console access
terminal_pid_dir = <i>None</i>	(StrOpt) Directory for holding terminal pid files. If not specified, the temporary directory will be used.

Table 1.10. Description of database configuration options

Configuration option = Default value	Description
[database]	
backend = <i>sqlalchemy</i>	(StrOpt) The back end to use for the database.
connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(BoolOpt) Add Python stack traces to SQL as comment strings.
db_inc_retry_interval = <i>True</i>	(BoolOpt) If True, increases the interval between retries of a database operation up to db_max_retry_interval .
db_max_retries = <i>20</i>	(IntOpt) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = <i>10</i>	(IntOpt) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.
db_retry_interval = <i>1</i>	(IntOpt) Seconds between retries of a database transaction.
idle_timeout = <i>3600</i>	(IntOpt) Timeout before idle SQL connections are reaped.

Configuration option = Default value	Description
max_overflow = <i>None</i>	(IntOpt) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = <i>None</i>	(IntOpt) Maximum number of SQL connections to keep open in a pool.
max_retries = <i>10</i>	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = <i>1</i>	(IntOpt) Minimum number of SQL connections to keep open in a pool.
mysql_engine = <i>InnoDB</i>	(StrOpt) MySQL engine to use.
mysql_sql_mode = <i>TRADITIONAL</i>	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode=
pool_timeout = <i>None</i>	(IntOpt) If set, use this value for pool_timeout with SQLAlchemy.
retry_interval = <i>10</i>	(IntOpt) Interval between retries of opening a SQL connection.
slave_connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = <i>oslo.sqlite</i>	(StrOpt) The file name to use with SQLite.
sqlite_synchronous = <i>True</i>	(BoolOpt) If True, SQLite uses synchronous mode.
use_db_reconnect = <i>False</i>	(BoolOpt) Enable the experimental use of database reconnect on connection lost.

Table 1.11. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
backdoor_port = <i>None</i>	(StrOpt) Enable eventlet backdoor. Acceptable values are 0, <port>, and <start>:<end>, where 0 results in listening on a random tcp port number; <port> results in listening on the specified port number (and not enabling backdoor if that port is in use); and <start>:<end> results in listening on the smallest unused port number within the specified range of port numbers. The chosen port is displayed in the service's log file.
pecan_debug = <i>False</i>	(BoolOpt) Enable pecan debug mode. WARNING: this is insecure and should not be used in production.

Table 1.12. Description of deploy configuration options

Configuration option = Default value	Description
[deploy]	
dd_block_size = <i>1M</i>	(StrOpt) Block size to use when writing to the nodes disk.
efi_system_partition_size = <i>200</i>	(IntOpt) Size of EFI system partition in MiB when configuring UEFI systems for local boot.
iscsi_verify_attempts = <i>3</i>	(IntOpt) Maximum attempts to verify an iSCSI connection is active, sleeping 1 second between attempts.

Table 1.13. Description of DHCP configuration options

Configuration option = Default value	Description
[dhcp]	
dhcp_provider = <i>neutron</i>	(StrOpt) DHCP provider to use. "neutron" uses Neutron, and "none" uses a no-op provider.

Table 1.14. Description of discovered configuration options

Configuration option = Default value	Description
[discovered]	
enabled = <i>False</i>	(BoolOpt) whether to enable inspection using ironic-discovered

Configuration option = Default value	Description
service_url = <i>None</i>	(StrOpt) ironic-discoverd HTTP endpoint. If this is not set, the ironic-discoverd client default (http://127.0.0.1:5050) will be used.
status_check_period = <i>60</i>	(IntOpt) period (in seconds) to check status of nodes on inspection

Table 1.15. Description of disk partitioner configuration options

Configuration option = Default value	Description
[disk_partitioner]	
check_device_interval = <i>1</i>	(IntOpt) After Ironic has completed creating the partition table, it continues to check for activity on the attached iSCSI device status at this interval prior to copying the image to the node, in seconds
check_device_max_retries = <i>20</i>	(IntOpt) The maximum number of times to check that the device is not accessed by another process. If the device is still busy after that, the disk partitioning will be treated as having failed.

Table 1.16. Description of glance configuration options

Configuration option = Default value	Description
[glance]	
allowed_direct_url_schemes =	(ListOpt) A list of URL schemes that can be downloaded directly via the direct_url. Currently supported schemes: [file].
auth_strategy = <i>keystone</i>	(StrOpt) Authentication strategy to use when connecting to glance. Only "keystone" and "noauth" are currently supported by ironic.
glance_api_insecure = <i>False</i>	(BoolOpt) Allow to perform insecure SSL (https) requests to glance.
glance_api_servers = <i>None</i>	(ListOpt) A list of the glance api servers available to ironic. Prefix with https:// for SSL-based glance API servers. Format is [hostname IP]:port.
glance_host = <i>\$my_ip</i>	(StrOpt) Default glance hostname or IP address.

Configuration option = Default value	Description
<code>glance_num_retries = 0</code>	(IntOpt) Number of retries when downloading an image from glance.
<code>glance_port = 9292</code>	(IntOpt) Default glance port.
<code>glance_protocol = http</code>	(StrOpt) Default protocol to use when connecting to glance. Set to https for SSL.
<code>swift_account = None</code>	(StrOpt) The account that Glance uses to communicate with Swift. The format is "AUTH_uuid". "uuid" is the UUID for the account configured in the glance-api.conf. Required for temporary URLs. For example: "AUTH_a422b2-91f3-2f46-74b7-d7c9e8958f5d30". Swift temporary URL format: "endpoint_url/api_version/account/container/object_id"
<code>swift_api_version = v1</code>	(StrOpt) The Swift API version to create a temporary URL for. Defaults to "v1". Swift temporary URL format: "endpoint_url/api_version/account/container/object_id"
<code>swift_container = glance</code>	(StrOpt) The Swift container Glance is configured to store its images in. Defaults to "glance", which is the default in glance-api.conf. Swift temporary URL format: "endpoint_url/api_version/account/container/object_id"
<code>swift_endpoint_url = None</code>	(StrOpt) The "endpoint" (scheme, hostname, optional port) for the Swift URL of the form "endpoint_url/api_version/account/container/object_id". Do not include trailing "/". For example, use "https://swift.example.com". Required for temporary URLs.
<code>swift_store_multiple_containers_seed = 0</code>	(IntOpt) This should match a config by the same name in the Glance configuration file. When set to 0, a single-tenant store will only use one container to store all images. When set to an integer value between 1 and 32, a single-tenant store will use multiple containers to store images, and this value will determine how many containers are created.
<code>swift_temp_url_duration = 1200</code>	(IntOpt) The length of time in seconds that the temporary URL will be valid for. Defaults to 20 minutes. If some deploys get a 401 response code when trying to download from the temporary URL, try raising this duration.

Configuration option = Default value	Description
swift_temp_url_key = <i>None</i>	(StrOpt) The secret token given to Swift to allow temporary URL downloads. Required for temporary URLs.

Table 1.17. Description of iLO configuration options

Configuration option = Default value	Description
[ilo]	
clean_priority_clear_secure_boot_keys = 0	(IntOpt) Priority for clear_secure_boot_keys clean step. This step is not enabled by default. It can be enabled to clear all secure boot keys enrolled with iLO.
clean_priority_erase_devices = <i>None</i>	(IntOpt) Priority for erase devices clean step. If unset, it defaults to 10. If set to 0, the step will be disabled and will not run during cleaning.
clean_priority_reset_bios_to_default = 10	(IntOpt) Priority for reset_bios_to_default clean step.
clean_priority_reset_ilo = 1	(IntOpt) Priority for reset_ilo clean step.
clean_priority_reset_ilo_credential = 30	(IntOpt) Priority for reset_ilo_credential clean step. This step requires "ilo_change_password" parameter to be updated in nodes's driver_info with the new password.
clean_priority_reset_secure_boot_keys_to_default = 20	(IntOpt) Priority for reset_secure_boot_keys clean step. This step will reset the secure boot keys to manufacturing defaults.
client_port = 443	(IntOpt) Port to be used for iLO operations
client_timeout = 60	(IntOpt) Timeout (in seconds) for iLO operations
power_retry = 6	(IntOpt) Number of times a power operation needs to be retried
power_wait = 2	(IntOpt) Amount of time in seconds to wait in between power operations
swift_ilo_container = <i>ironic_ilo_container</i>	(StrOpt) The Swift iLO container to store data.
swift_object_expiry_timeout = 900	(IntOpt) Amount of time in seconds for Swift objects to auto-expire.

Table 1.18. Description of IPMI configuration options

Configuration option = Default value	Description
[ipmi]	
<code>min_command_interval = 5</code>	(IntOpt) Minimum time, in seconds, between IPMI operations sent to a server. There is a risk with some hardware that setting this too low may cause the BMC to crash. Recommended setting is 5 seconds.
<code>retry_timeout = 60</code>	(IntOpt) Maximum time in seconds to retry IPMI operations. There is a tradeoff when setting this value. Setting this too low may cause older BMCs to crash and require a hard reset. However, setting too high can cause the sync power state periodic task to hang when there are slow or unresponsive BMCs.

Table 1.19. Description of iRMC configuration options

Configuration option = Default value	Description
[irmc]	
<code>auth_method = basic</code>	(StrOpt) Authentication method to be used for iRMC operations, either "basic" or "digest"
<code>client_timeout = 60</code>	(IntOpt) Timeout (in seconds) for iRMC operations
<code>port = 443</code>	(IntOpt) Port to be used for iRMC operations, either 80 or 443
<code>sensor_method = ipmitool</code>	(StrOpt) Sensor data retrieval method, either "ipmitool" or "scci"

Table 1.20. Description of keystone configuration options

Configuration option = Default value	Description
[keystone]	
<code>region_name = None</code>	(StrOpt) The region used for getting endpoints of OpenStackservices.

Table 1.21. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(BoolOpt) Print debugging output (set logging level to DEBUG instead of default WARNING level).
default_log_levels = <i>amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, keystonemiddleware=WARN, routes.middleware=WARN, stevedore=WARN</i>	(ListOpt) List of logger=LEVEL pairs.
fatal_exception_format_errors = <i>False</i>	(BoolOpt) Make exception message format errors fatal.
instance_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation.
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(StrOpt) Format string for %(asctime)s in log records. Default: %(default)s .
log_dir = <i>None</i>	(StrOpt) (Optional) The base directory used for relative --log-file paths.
log_file = <i>None</i>	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout.
log_format = <i>None</i>	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecate, use logging_context_format_string and logging_default_format_string instead.
logging_context_format_string = <i>%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%(message)s</i>	(StrOpt) Format string to use for log messages with context.

Configuration option = Default value	Description
logging_debug_format_suffix = % (funcName)s %(pathname)s:%(lineno)d	(StrOpt) Data to append to log format when level is DEBUG.
logging_default_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [-] %(instance)s%(message)s	(StrOpt) Format string to use for log messages without context.
logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d TRACE %(name)s % (instance)s	(StrOpt) Prefix each line of exception output with this format.
publish_errors = <i>False</i>	(BoolOpt) Enables or disables publication of error events.
syslog_log_facility = <i>LOG_USER</i>	(StrOpt) Syslog facility to receive log lines.
use_stderr = <i>True</i>	(BoolOpt) Log output to standard error.
use_syslog = <i>False</i>	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED during I, and will change in J to honor RFC5424.
use_syslog_rfc_format = <i>False</i>	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in I, and will be removed in J.
verbose = <i>False</i>	(BoolOpt) Print more verbose output (set logging level to INFO instead of default WARNING level).

Table 1.22. Description of neutron configuration options

Configuration option = Default value	Description
[neutron]	
auth_strategy = <i>keystone</i>	(StrOpt) Default authentication strategy to use when connecting to neutron. Can be either "keystone" or "noauth". Running neutron in noauth mode (related to but not affected by this setting) is insecure and should only be used for testing.
cleaning_network_uuid = <i>None</i>	(StrOpt) UUID of the network to create Neutron ports on when booting to a ramdisk for cleaning/zapping using Neutron DHCP

Configuration option = Default value	Description
retries = 3	(IntOpt) Client retries in the case of a failed request.
url = <i>http://\$my_ip:9696</i>	(StrOpt) URL for connecting to neutron.
url_timeout = 30	(IntOpt) Timeout value for connecting to neutron in seconds.

Table 1.23. Description of policy configuration options

Configuration option = Default value	Description
[oslo_policy]	
policy_default_rule = <i>default</i>	(StrOpt) Default rule. Enforced when a requested rule is not found.
policy_dirs = <i>['policy.d']</i>	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the <code>config_dir</code> option, or absolute paths. The file defined by <code>policy_file</code> must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(StrOpt) The JSON file that defines policies.

Table 1.24. Description of PXE configuration options

Configuration option = Default value	Description
[pxe]	
default_ephemeral_format = <i>ext4</i>	(StrOpt) Default file system format for ephemeral partition, if one is created.
disk_devices = <i>cciss/c0d0,sda,hda,vda</i>	(StrOpt) The disk devices to scan while doing the deploy.
http_root = <i>/httpboot</i>	(StrOpt) Ironic compute node's HTTP root path.
http_url = <i>None</i>	(StrOpt) Ironic compute node's HTTP server URL. Example: <code>http://192.1.2.3:8080</code>
image_cache_size = <i>20480</i>	(IntOpt) Maximum size (in MiB) of cache for master images, including those in use.

Configuration option = Default value	Description
image_cache_ttl = <i>10080</i>	(IntOpt) Maximum TTL (in minutes) for old master images in cache.
images_path = <i>/var/lib/ironic/images/</i>	(StrOpt) Directory where images are stored on disk.
instance_master_path = <i>/var/lib/ironic/master_images</i>	(StrOpt) Directory where master instance images are stored on disk.
ipxe_boot_script = <i>\$pybasedir/drivers/modules/boot.ipxe</i>	(StrOpt) The path to the main iPXE script file.
ipxe_enabled = <i>False</i>	(BoolOpt) Enable iPXE boot.
pxe_append_params = <i>nofb nomodeset vga=normal</i>	(StrOpt) Additional append parameters for baremetal PXE boot.
pxe_bootfile_name = <i>pxelinux.0</i>	(StrOpt) Bootfile DHCP parameter.
pxe_config_template = <i>\$pybasedir/drivers/modules/pxe_config.template</i>	(StrOpt) Template file for PXE configuration.
tftp_master_path = <i>/tftpboot/master_images</i>	(StrOpt) Directory where master tftp images are stored on disk.
tftp_root = <i>/tftpboot</i>	(StrOpt) Ironic compute node's tftp root path.
tftp_server = <i>\$my_ip</i>	(StrOpt) IP address of Ironic compute node's tftp server.
uefi_pxe_bootfile_name = <i>elilo.efi</i>	(StrOpt) Bootfile DHCP parameter for UEFI boot mode.
uefi_pxe_config_template = <i>\$pybasedir/drivers/modules/elilo_efi_pxe_config.template</i>	(StrOpt) Template file for PXE configuration for UEFI boot loader.

Table 1.25. Description of Redis configuration options

Configuration option = Default value	Description
[matchmaker_redis]	
host = <i>127.0.0.1</i>	(StrOpt) Host to locate redis.
password = <i>None</i>	(StrOpt) Password for Redis server (optional).

Configuration option = Default value	Description
port = 6379	(IntOpt) Use this port to connect to redis host.
[matchmaker_ring]	
ringfile = <i>/etc/oslo/matchmaker_ring.json</i>	(StrOpt) Matchmaker ring file (JSON).

Table 1.26. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
matchmaker_heartbeat_freq = 300	(IntOpt) Heartbeat frequency.
matchmaker_heartbeat_ttl = 600	(IntOpt) Heartbeat time-to-live.
rpc_backend = <i>rabbit</i>	(StrOpt) The messaging driver to use, defaults to rabbit. Other drivers include qpid and zmq.
rpc_cast_timeout = 30	(IntOpt) Seconds to wait before a cast expires (TTL). Only supported by impl_zmq.
rpc_response_timeout = 60	(IntOpt) Seconds to wait for a response from a call.
rpc_thread_pool_size = 64	(IntOpt) Size of RPC thread pool.
[oslo_concurrency]	
disable_process_locking = <i>False</i>	(BoolOpt) Enables or disables inter-process locks.
lock_path = <i>None</i>	(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(BoolOpt) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>	(StrOpt) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(StrOpt) Name for the AMQP container

Configuration option = Default value	Description
group_request_prefix = <i>unicast</i>	(StrOpt) address prefix when sending to any server in group
idle_timeout = 0	(IntOpt) Timeout for inactive connections (in seconds)
server_request_prefix = <i>exclusive</i>	(StrOpt) address prefix used when sending to a specific server
ssl_ca_file =	(StrOpt) CA certificate PEM file for verifying server certificate
ssl_cert_file =	(StrOpt) Identifying certificate PEM file to present to clients
ssl_key_file =	(StrOpt) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(StrOpt) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>	(BoolOpt) Debug: dump AMQP frames to stdout

Table 1.27. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
fake_rabbit = <i>False</i>	(BoolOpt) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
heartbeat_rate = 2	(IntOpt) How often times during the <code>heartbeat_timeout_threshold</code> we check the heartbeat.
heartbeat_timeout_threshold = 0	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disables the heartbeat, >0 enables it. Enabling heartbeats requires <code>kombu>=3.0.7</code> and <code>amqp>=1.4.0</code>). EXPERIMENTAL

Configuration option = Default value	Description
kombu_reconnect_delay = <i>1.0</i>	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
rabbit_ha_queues = <i>False</i>	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
rabbit_host = <i>localhost</i>	(StrOpt) The RabbitMQ broker address where a single node is used.
rabbit_hosts = <i>\$rabbit_host:\$rabbit_port</i>	(ListOpt) RabbitMQ HA cluster host:port pairs.
rabbit_login_method = <i>AMQPLAIN</i>	(StrOpt) The RabbitMQ login method.
rabbit_max_retries = <i>0</i>	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = <i>guest</i>	(StrOpt) The RabbitMQ password.
rabbit_port = <i>5672</i>	(IntOpt) The RabbitMQ broker port where a single node is used.
rabbit_retry_backoff = <i>2</i>	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
rabbit_retry_interval = <i>1</i>	(IntOpt) How frequently to retry connecting with RabbitMQ.
rabbit_use_ssl = <i>False</i>	(BoolOpt) Connect over SSL for RabbitMQ.
rabbit_userid = <i>guest</i>	(StrOpt) The RabbitMQ userid.
rabbit_virtual_host = <i>/</i>	(StrOpt) The RabbitMQ virtual host.

Configuration option = Default value	Description
<code>rpc_conn_pool_size = 30</code>	(IntOpt) Size of RPC connection pool.

Table 1.28. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
<code>amqp_auto_delete = False</code>	(BoolOpt) Auto-delete queues in AMQP.
<code>amqp_durable_queues = False</code>	(BoolOpt) Use durable queues in AMQP.
<code>qpid_heartbeat = 60</code>	(IntOpt) Seconds between connection keepalive heartbeats.
<code>qpid_hostname = localhost</code>	(StrOpt) Qpid broker hostname.
<code>qpid_hosts = \$qpid_hostname:\$qpid_port</code>	(ListOpt) Qpid HA cluster host:port pairs.
<code>qpid_password =</code>	(StrOpt) Password for Qpid connection.
<code>qpid_port = 5672</code>	(IntOpt) Qpid broker port.
<code>qpid_protocol = tcp</code>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
<code>qpid_receiver_capacity = 1</code>	(IntOpt) The number of prefetched messages held by receiver.
<code>qpid_sasl_mechanisms =</code>	(StrOpt) Space separated list of SASL mechanisms to use for auth.
<code>qpid_tcp_nodelay = True</code>	(BoolOpt) Whether to disable the Nagle algorithm.
<code>qpid_topology_version = 1</code>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
<code>qpid_username =</code>	(StrOpt) Username for Qpid connection.
<code>rpc_conn_pool_size = 30</code>	(IntOpt) Size of RPC connection pool.

Table 1.29. Description of SeaMicro configuration options

Configuration option = Default value	Description
[seamicro]	
action_timeout = 10	(IntOpt) Seconds to wait for power action to be completed
max_retry = 3	(IntOpt) Maximum retries for SeaMicro operations

Table 1.30. Description of SNMP configuration options

Configuration option = Default value	Description
[snmp]	
power_timeout = 10	(IntOpt) Seconds to wait for power action to be completed

Table 1.31. Description of SSH configuration options

Configuration option = Default value	Description
[ssh]	
libvirt_uri = <i>qemu:///system</i>	(StrOpt) libvirt uri

Table 1.32. Description of swift configuration options

Configuration option = Default value	Description
[swift]	
swift_max_retries = 2	(IntOpt) Maximum number of times to retry a Swift request, before failing.

Table 1.33. Description of VirtualBox configuration options

Configuration option = Default value	Description
[virtualbox]	
port = 18083	(IntOpt) Port on which VirtualBox web service is listening.

CHAPTER 2. BLOCK STORAGE

The OpenStack Block Storage service provides persistent storage for Compute instances, working with many different storage drivers that you can configure.

2.1. VOLUME DRIVERS

To use different volume drivers for the `cinder-volume` service, use the parameters described in these sections.

To set a volume driver, use the `volume_driver` flag. The default is:

```
volume_driver = cinder.volume.drivers.lvm.LVMISCSIDriver
```

2.1.1. Ceph RADOS Block Device (RBD)

If you use KVM or QEMU as your hypervisor, you can configure the Compute service to use [Ceph RADOS block devices \(RBD\)](#) for volumes.

Ceph is a massively scalable, open source, distributed storage system. It is comprised of an object store, block store, and a POSIX-compliant distributed file system. The platform can auto-scale to the exabyte level and beyond. It runs on commodity hardware, is self-healing and self-managing, and has no single point of failure. Ceph is in the Linux kernel and is integrated with the OpenStack cloud operating system. Due to its open-source nature, you can install and use this portable storage platform in public or private clouds.

RADOS

Ceph is based on *RADOS: Reliable Autonomic Distributed Object Store*. RADOS distributes objects across the storage cluster and replicates objects for fault tolerance. RADOS contains the following major components:

- *Object Storage Device (OSD) Daemon* The storage daemon for the RADOS service, which interacts with the OSD (physical or logical storage unit for your data).

You must run this daemon on each server in your cluster. For each OSD, you can have an associated hard drive disk. For performance purposes, pool your hard drive disk with raid arrays, logical volume management (LVM), or B-tree file system (`Btrfs`) pooling. By default, the following pools are created: data, metadata, and RBD.

- *Meta-Data Server (MDS)* Stores metadata. MDSs build a POSIX file system on top of objects for Ceph clients. However, if you do not use the Ceph file system, you do not need a metadata server.
- *Monitor (MON)*. A lightweight daemon that handles all communications with external applications and clients. It also provides a consensus for distributed decision making in a Ceph/RADOS cluster. For instance, when you mount a Ceph shared on a client, you point to the address of a MON server. It checks the state and the consistency of the data. In an ideal setup, you must run at least three `ceph-mon` daemons on separate servers.

Ceph developers recommend XFS for production deployments, `Btrfs` for testing, development, and any non-critical deployments. `Btrfs` has the correct feature set and roadmap to serve Ceph in the long-term, but XFS and `ext4` provide the necessary stability for today's deployments.

**NOTE**

If using **Btrfs**, ensure that you use the correct version (see [Ceph Dependencies](#)).

For more information about usable file systems, see ceph.com/ceph-storage/file-system/.

Ways to store, use, and expose data

To store and access your data, you can use the following storage systems:

- **RADOS**. Use as an object, default storage mechanism.
- **RBD**. Use as a block device. The Linux kernel RBD (RADOS block device) driver allows striping a Linux block device over multiple distributed object store data objects. It is compatible with the KVM RBD image.
- **CephFS**. Use as a file, POSIX-compliant file system.

Ceph exposes RADOS; you can access it through the following interfaces:

- **RADOS Gateway**. OpenStack Object Storage and Amazon-S3 compatible RESTful interface (see [RADOS_Gateway](#)).
- **librados**, and its related C/C++ bindings.
- **RBD and QEMU-RBD**. Linux kernel and QEMU block devices that stripe data across multiple objects.

Driver options

The following table contains the configuration options supported by the Ceph RADOS Block Device driver.

**DEPRECATION NOTICE**

The **volume_tmp_dir** option has been deprecated and replaced by **image_conversion_dir**.

Table 2.1. Description of Ceph storage configuration options

Configuration option = Default value	Description
[DEFAULT]	
rados_connect_timeout = -1	(IntOpt) Timeout value (in seconds) used when connecting to ceph cluster. If value < 0, no timeout is set and default librados value is used.
rados_connection_interval = 5	(IntOpt) Interval value (in seconds) between connection retries to ceph cluster.

Configuration option = Default value	Description
rados_connection_retries = 3	(IntOpt) Number of retries if connection to ceph cluster failed.
rbd_ceph_conf =	(StrOpt) Path to the ceph configuration file
rbd_cluster_name = <i>ceph</i>	(StrOpt) The name of ceph cluster
rbd_flatten_volume_from_snapshot = <i>False</i>	(BoolOpt) Flatten volumes created from snapshots to remove dependency from volume to snapshot
rbd_max_clone_depth = 5	(IntOpt) Maximum number of nested volume clones that are taken before a flatten occurs. Set to 0 to disable cloning.
rbd_pool = <i>rbd</i>	(StrOpt) The RADOS pool where rbd volumes are stored
rbd_secret_uuid = <i>None</i>	(StrOpt) The libvirt uuid of the secret for the rbd_user volumes
rbd_store_chunk_size = 4	(IntOpt) Volumes will be chunked into objects of this size (in megabytes).
rbd_user = <i>None</i>	(StrOpt) The RADOS client name for accessing rbd volumes - only set when using cephx authentication
volume_tmp_dir = <i>None</i>	(StrOpt) Directory where temporary image files are stored when the volume driver does not write them directly to the volume. Warning: this option is now deprecated, use image_conversion_dir instead.

2.1.2. Dell EqualLogic volume driver

The Dell EqualLogic volume driver interacts with configured EqualLogic arrays and supports various operations.

Supported operations

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.
- Clone a volume.

The OpenStack Block Storage service supports:

- Multiple instances of Dell EqualLogic Groups or Dell EqualLogic Group Storage Pools and multiple pools on a single array.

- Multiple instances of Dell EqualLogic Groups or Dell EqualLogic Group Storage Pools or multiple pools on a single array.

The Dell EqualLogic volume driver's ability to access the EqualLogic Group is dependent upon the generic block storage driver's SSH settings in the `/etc/cinder/cinder.conf` file (see [Section 2.3, “Block Storage sample configuration files”](#) for reference).

Table 2.2. Description of Dell EqualLogic volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>eqlx_chap_login = admin</code>	(StrOpt) Existing CHAP account name. Note that this option is deprecated in favour of "chap_username" as specified in <code>cinder/volume/driver.py</code> and will be removed in next release.
<code>eqlx_chap_password = password</code>	(StrOpt) Password for specified CHAP account name. Note that this option is deprecated in favour of "chap_password" as specified in <code>cinder/volume/driver.py</code> and will be removed in the next release
<code>eqlx_cli_max_retries = 5</code>	(IntOpt) Maximum retry count for reconnection. Default is 5.
<code>eqlx_cli_timeout = 30</code>	(IntOpt) Timeout for the Group Manager cli command execution. Default is 30. Note that this option is deprecated in favour of "ssh_conn_timeout" as specified in <code>cinder/volume/drivers/san/san.py</code> and will be removed in M release.
<code>eqlx_group_name = group-0</code>	(StrOpt) Group name to use for creating volumes. Defaults to "group-0".
<code>eqlx_pool = default</code>	(StrOpt) Pool in which volumes will be created. Defaults to "default".
<code>eqlx_use_chap = False</code>	(BoolOpt) Use CHAP authentication for targets. Note that this option is deprecated in favour of "use_chap_auth" as specified in <code>cinder/volume/driver.py</code> and will be removed in next release.

The following sample `/etc/cinder/cinder.conf` configuration lists the relevant settings for a typical Block Storage service using a single Dell EqualLogic Group:

Example 2.1. Default (single-instance) configuration

[DEFAULT]

```

#Required settings

volume_driver = cinder.volume.drivers.eqlx.DellEQLSanISCSIDriver
san_ip = IP_EQLX
san_login = SAN_UNAME
san_password = SAN_PW
eqlx_group_name = EQLX_GROUP
eqlx_pool = EQLX_POOL

#Optional settings

san_thin_provision = true|false
eqlx_use_chap = true|false
eqlx_chap_login = EQLX_UNAME
eqlx_chap_password = EQLX_PW
eqlx_cli_max_retries = 5
san_ssh_port = 22
ssh_conn_timeout = 30
san_private_key = SAN_KEY_PATH
ssh_min_pool_conn = 1
ssh_max_pool_conn = 5

```

In this example, replace the following variables accordingly:

IP_EQLX

The IP address used to reach the Dell EqualLogic Group through SSH. This field has no default value.

SAN_UNAME

The user name to login to the Group manager via SSH at the **san_ip**. Default user name is **grpadmin**.

SAN_PW

The corresponding password of **SAN_UNAME**. Not used when **san_private_key** is set. Default password is **password**.

EQLX_GROUP

The group to be used for a pool where the Block Storage service will create volumes and snapshots. Default group is **group-0**.

EQLX_POOL

The pool where the Block Storage service will create volumes and snapshots. Default pool is **default**. This option cannot be used for multiple pools utilized by the Block Storage service on a single Dell EqualLogic Group.

EQLX_UNAME

The CHAP login account for each volume in a pool, if **eqlx_use_chap** is set to **true**. Default account name is **chapadmin**.

EQLX_PW

The corresponding password of *EQLX_UNAME*. The default password is randomly generated in hexadecimal, so you must set this password manually.

SAN_KEY_PATH (optional)

The filename of the private key used for SSH authentication. This provides password-less login to the EqualLogic Group. Not used when *san_password* is set. There is no default value.

In addition, enable thin provisioning for SAN volumes using the default *san_thin_provision = true* setting.

Example 2.2. Multi back-end Dell EqualLogic configuration

The following example shows the typical configuration for a Block Storage service that uses two Dell EqualLogic back ends:

```
enabled_backends = backend1, backend2
san_ssh_port = 22
ssh_conn_timeout = 30
san_thin_provision = true

[backend1]
volume_driver = cinder.volume.drivers.eqLx.DellEQLSanISCSIDriver
volume_backend_name = backend1
san_ip = IP_EQLX1
san_login = SAN_UNAME
san_password = SAN_PW
eqLx_group_name = EQLX_GROUP
eqLx_pool = EQLX_POOL

[backend2]
volume_driver = cinder.volume.drivers.eqLx.DellEQLSanISCSIDriver
volume_backend_name = backend2
san_ip = IP_EQLX2
san_login = SAN_UNAME
san_password = SAN_PW
eqLx_group_name = EQLX_GROUP
eqLx_pool = EQLX_POOL
```

In this example:

- Thin provisioning for SAN volumes is enabled (*san_thin_provision = true*). This is recommended when setting up Dell EqualLogic back ends.
- Each Dell EqualLogic back-end configuration ([*backend1*] and [*backend2*]) has the same required settings as a single back-end configuration, with the addition of *volume_backend_name*.
- The *san_ssh_port* option is set to its default value, 22. This option sets the port used for SSH.
- The *ssh_conn_timeout* option is also set to its default value, 30. This option sets the timeout in seconds for CLI commands over SSH.

- The `IP_EQLX1` and `IP_EQLX2` refer to the IP addresses used to reach the Dell EqualLogic Group of `backend1` and `backend2` through SSH, respectively.

For information on configuring multiple back ends, see [Configure a multiple-storage back end](#).

2.1.3. Dell Storage Center Fibre Channel and iSCSI drivers

The Dell Storage Center volume driver interacts with configured Storage Center arrays.

The Dell Storage Center driver manages Storage Center arrays through Enterprise Manager. Enterprise Manager connection settings and Storage Center options are defined in the `cinder.conf` file.

Prerequisite: Dell Enterprise Manager 2015 R1 or later must be used.

Supported operations

The Dell Storage Center volume driver provides the following Cinder volume operations:

- Create, delete, attach (map), and detach (unmap) volumes.
- Create, list, and delete volume snapshots.
- Create a volume from a snapshot.
- Copy an image to a volume.
- Copy a volume to an image.
- Clone a volume.
- Extend a volume.

Extra spec options

Volume type extra specs can be used to select different Storage Profiles.

Storage Profiles control how Storage Center manages volume data. For a given volume, the selected Storage Profile dictates which disk tier accepts initial writes, as well as how data progression moves data between tiers to balance performance and cost. Predefined Storage Profiles are the most effective way to manage data in Storage Center.

By default, if no Storage Profile is specified in the volume extra specs, the default Storage Profile for the user account configured for the Block Storage driver is used. The extra spec key `storagetype:storageprofile` with the value of the name of the Storage Profile on the Storage Center can be set to allow to use Storage Profiles other than the default.

For ease of use from the command line, spaces in Storage Profile names are ignored. As an example, here is how to define two volume types using the **High Priority** and **Low Priority** Storage Profiles:

```
$ cinder type-create "GoldVolumeType"
$ cinder type-key "GoldVolumeType" set
storagetype:storageprofile=highpriority
```

```
$ cinder type-create "BronzeVolumeType"
$ cinder type-key "BronzeVolumeType" set
storageprofile=lowpriority
```

iSCSI configuration

Use the following instructions to update the configuration file for iSCSI:

Example 2.3. Sample iSCSI Configuration

```
default_volume_type = delliscsi
enabled_backends = delliscsi

[delliscsi]
# Name to give this storage backend
volume_backend_name = delliscsi
# The iSCSI driver to load
volume_driver =
cinder.volume.drivers.dell.dell_storagecenter_iscsi.DellStorageCenterISCSIDriver
# IP address of Enterprise Manager
san_ip = 172.23.8.101
# Enterprise Manager user name
san_login = Admin
# Enterprise Manager password
san_password = secret
# The Storage Center iSCSI IP address
iscsi_ip_address = 192.168.0.20
# The Storage Center serial number to use
dell_sc_ssn = 64702

# ==Optional settings==
# The Enterprise Manager API port
dell_sc_api_port = 3033
# Server folder to place new server definitions
dell_sc_server_folder = devstacksrv
# Volume folder to place created volumes
dell_sc_volume_folder = devstackvol/Cinder
# The iSCSI IP port
iscsi_port = 3260
```

Fibre Channel configuration

Use the following instructions to update the configuration file for fibre channel:

Example 2.4. Sample FC configuration

```
default_volume_type = dellfc
enabled_backends = dellfc

[dellfc]
# Name to give this storage backend
volume_backend_name = dellfc
# The FC driver to load
volume_driver =
```



```

cinder.volume.drivers.dell.dell_storagecenter_fc.DellStorageCenterFCDriver
# IP address of Enterprise Manager
san_ip = 172.23.8.101
# Enterprise Manager user name
san_login = Admin
# Enterprise Manager password
san_password = secret
# The Storage Center serial number to use
dell_sc_ssn = 64702

# Optional settings

# The Enterprise Manager API port
dell_sc_api_port = 3033
# Server folder to place new server definitions
dell_sc_server_folder = devstacksrv
# Volume folder to place created volumes
dell_sc_volume_folder = devstackvol/Cinder

```

Driver options

The following table contains the configuration options specific to the Dell Storage Center volume driver.

Table 2.3. Description of Dell Storage Center volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
dell_sc_api_port = 3033	(IntOpt) Dell API port
dell_sc_server_folder = <i>openstack</i>	(StrOpt) Name of the server folder to use on the Storage Center
dell_sc_ssn = 64702	(IntOpt) Storage Center System Serial Number
dell_sc_verify_cert = <i>False</i>	(BoolOpt) Enable HTTPS SC certificate verification.
dell_sc_volume_folder = <i>openstack</i>	(StrOpt) Name of the volume folder to use on the Storage Center

2.1.4. EMC ScaleIO Block Storage driver configuration

ScaleIO is a software-only solution that uses existing servers' local disks and LAN to create a virtual SAN that has all of the benefits of external storage, but at a fraction of the cost and complexity. Using the driver, Block Storage hosts can connect to a ScaleIO Storage cluster.

This section explains how to configure and connect the block storage nodes to a ScaleIO storage cluster.

2.1.4.1. Support matrix

2.1.4.2. Deployment prerequisites

- ScaleIO Gateway must be installed and accessible in the network. For installation steps, refer to the Preparing the installation Manager and the Gateway section in ScaleIO Deployment Guide. See [Section 2.1.4.2.1, “Official documentation”](#).
- ScaleIO Data Client (SDC) must be installed on all OpenStack nodes.

2.1.4.2.1. Official documentation

To find the ScaleIO documentation:

1. Go to the [ScaleIO product documentation page](#).
2. From the left-side panel, select the relevant version (1.32 or 2.0).
3. Search for "ScaleIO Installation Guide 1.32" or "ScaleIO 2.0 Deployment Guide" accordingly.

2.1.4.3. Supported operations

- Create, delete, clone, attach, and detach volumes
- Create and delete volume snapshots
- Create a volume from a snapshot
- Copy an image to a volume
- Copy a volume to an image
- Extend a volume
- Get volume statistics
- Manage and unmanage a volume
- Create, list, update, and delete consistency groups
- Create, list, update, and delete consistency group snapshots

2.1.4.4. ScaleIO QoS support

QoS support for the ScaleIO driver includes the ability to set the following capabilities in the Block Storage API `cinder . api . contrib . qos _ specs _ manage` QoS specs extension module:

- `minBWS`
- `maxBWS`

The QoS keys above must be created and associated with a volume type. For information about how to set the key-value pairs and associate them with a volume type, run the following commands:

```
$ cinder help qos-create
```

```
$ cinder help qos-key
$ cinder help qos-associate
```

maxBWS

The QoS I/O issue bandwidth rate limit in KBs. If not set, the I/O issue bandwidth rate has no limit. The setting must be a multiple of 1024.

maxIOPS

The QoS I/O issue bandwidth rate limit in MBs. If not set, the I/O issue bandwidth rate has no limit. The setting must be larger than 10.

Since the limits are per SDC, they will be applied after the volume is attached to an instance, and thus to a compute node/SDC.

2.1.4.5. ScaleIO thin provisioning support

The Block Storage driver supports creation of thin-provisioned volumes, in addition to thick provisioning. The provisioning type settings should be added as an extra specification of the volume type, as follows:

```
sio:provisioning_type = thin\thick
```

If the provisioning type value is not specified, the default value of "thick" will be used.

2.1.4.6. ScaleIO Block Storage driver configuration

Edit the `cinder.conf` file by adding the configuration below under the **[DEFAULT]** section of the file in case of a single back end, or under a separate section in case of multiple back ends (for example **[ScaleIO]**). The configuration file is usually located at `/etc/cinder/cinder.conf`.

For a configuration example, refer to [Section 2.1.4.8, “Configuration example”](#).

2.1.4.6.1. ScaleIO driver name

Configure the driver name by adding the following parameter:

```
volume_driver = cinder.volume.drivers.emc.scaleio.ScaleIODriver
```

2.1.4.6.2. ScaleIO MDM server IP

The ScaleIO Meta Data Manager monitors and maintains the available resources and permissions.

To retrieve the MDM server IP address, use the `drv_cfg - -query_mdms` command.

Configure the MDM server IP address by adding the following parameter:

```
san_ip = ScaleIO GATEWAY IP
```

2.1.4.6.3. ScaleIO Protection Domain name

ScaleIO allows multiple Protection Domains (groups of SDSs that provide backup for each other).

To retrieve the available Protection Domains, use the command `scli --query_all` and search for the Protection Domains section.

Configure the Protection Domain for newly created volumes by adding the following parameter:

```
sio_protection_domain_name = ScaleIO Protection Domain
```

2.1.4.6.4. ScaleIO Storage Pool name

A ScaleIO Storage Pool is a set of physical devices in a Protection Domain.

To retrieve the available Storage Pools, use the command `scli --query_all` and search for available Storage Pools.

Configure the Storage Pool for newly created volumes by adding the following parameter:

```
sio_storage_pool_name = ScaleIO Storage Pool
```

2.1.4.6.5. ScaleIO Storage Pools

Multiple Storage Pools and Protection Domains can be listed for use by the virtual machines.

To retrieve the available Storage Pools, use the command `scli --query_all` and search for available Storage Pools.

Configure the available Storage Pools by adding the following parameter:

```
sio_storage_pools = Comma-separated list of protection domain:storage pool  
name
```

2.1.4.6.6. ScaleIO user credentials

Block Storage requires a ScaleIO user with administrative privileges. ScaleIO recommends creating a dedicated OpenStack user account that has an administrative user role.

Refer to the ScaleIO User Guide for details on user account management.

Configure the user credentials by adding the following parameters:

```
san_login = ScaleIO username  
  
san_password = ScaleIO password
```

2.1.4.7. Multiple back ends

Configuring multiple storage back ends allows you to create several back-end storage solutions that serve the same Compute resources.

When a volume is created, the scheduler selects the appropriate back end to handle the request, according to the specified volume type.

2.1.4.8. Configuration example

cinder.conf example file

You can update the `cinder.conf` file by editing the necessary parameters as follows:

```
[Default]
enabled_backends = scaleio

[scaleio]
volume_driver = cinder.volume.drivers.emc.scaleio.ScaleIODriver
volume_backend_name = scaleio
san_ip = GATEWAY_IP
sio_protection_domain_name = Default_domain
sio_storage_pool_name = Default_pool
sio_storage_pools = Domain1:Pool1,Domain2:Pool2
san_login = SIO_USER
san_password = SIO_PASSWD
```

2.1.4.9. Configuration options

The ScaleIO driver supports these configuration options:

Table 2.4. Description of EMC SIO volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>sio_force_delete = False</code>	(BoolOpt) Whether to allow force delete.
<code>sio_protection_domain_id = None</code>	(StrOpt) Protection domain id.
<code>sio_protection_domain_name = None</code>	(StrOpt) Protection domain name.
<code>sio_rest_server_port = 443</code>	(StrOpt) REST server port.
<code>sio_round_volume_capacity = True</code>	(BoolOpt) Whether to round volume capacity.
<code>sio_server_certificate_path = None</code>	(StrOpt) Server certificate path.
<code>sio_storage_pool_id = None</code>	(StrOpt) Storage pool id.
<code>sio_storage_pool_name = None</code>	(StrOpt) Storage pool name.
<code>sio_storage_pools = None</code>	(StrOpt) Storage pools.
<code>sio_unmap_volume_before_deletion = False</code>	(BoolOpt) Whether to unmap volume before deletion.

Configuration option = Default value	Description
<code>sio_verify_server_certificate = False</code>	(BoolOpt) Whether to verify server certificate.

2.1.5. EMC VMAX iSCSI and FC drivers

The EMC VMAX drivers, `EMCVMAXISCSIDriver` and `EMCVMAXFCDriver`, support the use of EMC VMAX storage arrays under OpenStack Block Storage. They both provide equivalent functions and differ only in support for their respective host attachment methods.

The drivers perform volume operations by communicating with the backend VMAX storage. It uses a CIM client in Python called PyWBEM to perform CIM operations over HTTP.

The EMC CIM Object Manager (ECOM) is packaged with the EMC SMI-S provider. It is a CIM server that enables CIM clients to perform CIM operations over HTTP by using SMI-S in the back-end for VMAX storage operations.

The EMC SMI-S Provider supports the SNIA Storage Management Initiative (SMI), an ANSI standard for storage management. It supports the VMAX storage system.

2.1.5.1. System requirements

EMC SMI-S Provider V4.6.2.8 and higher is required. You can download SMI-S from the [EMC's support](#) web site (login is required). See the EMC SMI-S Provider release notes for installation instructions.

EMC storage VMAX Family is supported.

2.1.5.2. Supported operations

VMAX drivers support these operations:

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.
- Copy an image to a volume.
- Copy a volume to an image.
- Clone a volume.
- Extend a volume.
- Retype a volume.
- Create a volume from a snapshot.

VMAX drivers also support the following features:

- FAST automated storage tiering policy.
- Dynamic masking view creation.
- Striped volume creation.

2.1.5.3. Set up the VMAX drivers

Procedure 2.1. To set up the EMC VMAX drivers

1. Install the `python-pywbem` package for your distribution. To install the `python-pywbem` package for Red Hat Enterprise Linux, CentOS, or Fedora:

```
# yum install pywbem
```

2. Download SMI-S from PowerLink and install it. Add your VMAX arrays to SMI-S.

For information, see [Section 2.1.5.3.1, “Set up SMI-S”](#) and the SMI-S release notes.

3. Change configuration files. See [Section 2.1.5.3.2, “cinder.conf configuration file”](#) and [Section 2.1.5.3.3, “cinder_emc_config_CONF_GROUP_ISCSI.xml configuration file”](#).
4. Configure connectivity. For FC driver, see [Section 2.1.5.3.4, “FC Zoning with VMAX”](#). For iSCSI driver, see [Section 2.1.5.3.5, “iSCSI with VMAX”](#).

2.1.5.3.1. Set up SMI-S

You can install SMI-S on a non-OpenStack host. Supported platforms include different flavors of Windows, Red Hat, and SUSE Linux. SMI-S can be installed on a physical server or a VM hosted by an ESX server. Note that the supported hypervisor for a VM running SMI-S is ESX only. See the EMC SMI-S Provider release notes for more information on supported platforms and installation instructions.



NOTE

You must discover storage arrays on the SMI-S server before you can use the VMAX drivers. Follow instructions in the SMI-S release notes.

SMI-S is usually installed at `/opt/emc/ECIM/ECOM/bin` on Linux and `C:\Program Files\EMC\ECIM\ECOM\bin` on Windows. After you install and configure SMI-S, go to that directory and type `TestSmiProvider.exe`.

Use `addsys` in `TestSmiProvider.exe` to add an array. Use `dv` and examine the output after the array is added. Make sure that the arrays are recognized by the SMI-S server before using the EMC VMAX drivers.

2.1.5.3.2. cinder.conf configuration file

Make the following changes in `/etc/cinder/cinder.conf`.

Add the following entries, where `10.10.61.45` is the IP address of the VMAX iSCSI target:

```
enabled_backends = CONF_GROUP_ISCSI, CONF_GROUP_FC
[CONF_GROUP_ISCSI]
iscsi_ip_address = 10.10.61.45
volume_driver =
cinder.volume.drivers.emc.emc_vmax_iscsi.EMCVMAXISCSIDriver
cinder_emc_config_file =
/etc/cinder/cinder_emc_config_CONF_GROUP_ISCSI.xml
volume_backend_name=ISCSI_backend
[CONF_GROUP_FC]
```

```

volume_driver = cinder.volume.drivers.emc.emc_vmax_fc.EMCVMAXFCDriver
cinder_emc_config_file = /etc/cinder/cinder_emc_config_CONF_GROUP_FC.xml
volume_backend_name=FC_backend

```

In this example, two backend configuration groups are enabled: **CONF_GROUP_ISCSI** and **CONF_GROUP_FC**. Each configuration group has a section describing unique parameters for connections, drivers, the `volume_backend_name`, and the name of the EMC-specific configuration file containing additional settings. Note that the file name is in the format `/etc/cinder/cinder_emc_config_[confGroup].xml`.

Once the `cinder.conf` and EMC-specific configuration files have been created, cinder commands need to be issued in order to create and associate OpenStack volume types with the declared `volume_backend_names`:

```

$ cinder type-create VMAX_ISCSI
$ cinder type-key VMAX_ISCSI set volume_backend_name=ISCSI_backend
$ cinder type-create VMAX_FC
$ cinder type-key VMAX_FC set volume_backend_name=FC_backend

```

By issuing these commands, the Block Storage volume type **VMAX_ISCSI** is associated with the `ISCSI_backend`, and the type **VMAX_FC** is associated with the `FC_backend`.

Restart the `cinder-volume` service.

2.1.5.3.3. `cinder_emc_config_CONF_GROUP_ISCSI.xml` configuration file

Create the `/etc/cinder/cinder_emc_config_CONF_GROUP_ISCSI.xml` file. You do not need to restart the service for this change.

Add the following lines to the XML file:

```

<?xml version="1.0" encoding="UTF-8" ?>
<EMC>
  <EcomServerIp>1.1.1.1</EcomServerIp>
  <EcomServerPort>00</EcomServerPort>
  <EcomUserName>user1</EcomUserName>
  <EcomPassword>password1</EcomPassword>
  <PortGroups>
    <PortGroup>OS-PORTGROUP1-PG</PortGroup>
    <PortGroup>OS-PORTGROUP2-PG</PortGroup>
  </PortGroups>
  <Array>111111111111</Array>
  <Pool>FC_GOLD1</Pool>
  <FastPolicy>GOLD1</FastPolicy>
</EMC>

```

Where:

- **EcomServerIp** and **EcomServerPort** are the IP address and port number of the ECOM server which is packaged with SMI-S.
- **EcomUserName** and **EcomPassword** are credentials for the ECOM server.
- **PortGroups** supplies the names of VMAX port groups that have been pre-configured to expose volumes managed by this backend. Each supplied port group should have sufficient

number and distribution of ports (across directors and switches) as to ensure adequate bandwidth and failure protection for the volume connections. **PortGroups** can contain one or more port groups of either iSCSI or FC ports. When a dynamic masking view is created by the VMAX driver, the port group is chosen randomly from the **PortGroup** list, to evenly distribute load across the set of groups provided. Make sure that the **PortGroups** set contains either all FC or all iSCSI port groups (for a given backend), as appropriate for the configured driver (iSCSI or FC).

- The **Array** tag holds the unique VMAX array serial number.
- The **Pool** tag holds the unique pool name within a given array. For backends not using FAST automated tiering, the pool is a single pool that has been created by the administrator. For backends exposing FAST policy automated tiering, the pool is the bind pool to be used with the FAST policy.
- The **FastPolicy** tag conveys the name of the FAST Policy to be used. By including this tag, volumes managed by this backend are treated as under FAST control. Omitting the **FastPolicy** tag means FAST is not enabled on the provided storage pool.

2.1.5.3.4. FC Zoning with VMAX

Zone Manager is recommended when using the VMAX FC driver, especially for larger configurations where pre-zoning would be too complex and open-zoning would raise security concerns.

2.1.5.3.5. iSCSI with VMAX

- Make sure the `iscsi-initiator-utils` package is installed on the host (use `apt-get`, `zypper`, or `yum`, depending on Linux flavor).
- Verify host is able to ping VMAX iSCSI target ports.

2.1.5.4. VMAX masking view and group naming info

Masking view names

Masking views are dynamically created by the VMAX FC and iSCSI drivers using the following naming conventions:

```
OS-[shortHostName][poolName]-I-MV (for Masking Views using iSCSI)
```

```
OS-[shortHostName][poolName]-F-MV (for Masking Views using FC)
```

Initiator group names

For each host that is attached to VMAX volumes using the drivers, an initiator group is created or re-used (per attachment type). All initiators of the appropriate type known for that host are included in the group. At each new attach volume operation, the VMAX driver retrieves the initiators (either WWNNs or IQNs) from OpenStack and adds or updates the contents of the Initiator Group as required. Names are of the following format:

```
OS-[shortHostName]-I-IG (for iSCSI initiators)
```

```
OS-[shortHostName]-F-IG (for Fibre Channel initiators)
```

**NOTE**

Hosts attaching to VMAX storage managed by the OpenStack environment cannot also be attached to storage on the same VMAX not being managed by OpenStack. This is due to limitations on VMAX Initiator Group membership.

FA port groups

VMAX array FA ports to be used in a new masking view are chosen from the list provided in the EMC configuration file.

Storage group names

As volumes are attached to a host, they are either added to an existing storage group (if it exists) or a new storage group is created and the volume is then added. Storage groups contain volumes created from a pool (either single-pool or FAST-controlled), attached to a single host, over a single connection type (iSCSI or FC). Names are formed:

```
OS-[shortHostName][poolName]-I-SG (attached over iSCSI)
```

```
OS-[shortHostName][poolName]-F-SG (attached over Fibre Channel)
```

2.1.5.5. Concatenated or striped volumes

In order to support later expansion of created volumes, the VMAX Block Storage drivers create concatenated volumes as the default layout. If later expansion is not required, users can opt to create striped volumes in order to optimize I/O performance.

Below is an example of how to create striped volumes. First, create a volume type. Then define the extra spec for the volume type **storagetype:stripecount** representing the number of meta members in the striped volume. The example below means that each volume created under the **GoldStriped** volume type will be striped and made up of 4 meta members.

```
$ cinder type-create GoldStriped
$ cinder type-key GoldStriped set volume_backend_name=GOLD_BACKEND
$ cinder type-key GoldStriped set storagetype:stripecount=4
```

2.1.6. EMC VNX driver

EMC VNX driver consists of **EMCCLIISCSIDriver** and **EMCCLIFCDriver**, and supports both iSCSI and FC protocol. **EMCCLIISCSIDriver** (VNX iSCSI driver) and **EMCCLIFCDriver** (VNX FC driver) are separately based on the **ISCSIDriver** and **FCDriver** defined in Block Storage.

2.1.6.1. Overview

The VNX iSCSI driver and VNX FC driver perform the volume operations by executing Navisphere CLI (NaviSecCLI) which is a command line interface used for management, diagnostics, and reporting functions for VNX.

2.1.6.1.1. System requirements

- VNX Operational Environment for Block version 5.32 or higher.
- VNX Snapshot and Thin Provisioning license should be activated for VNX.
- Navisphere CLI v7.32 or higher is installed along with the driver.

2.1.6.1.2. Supported operations

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.
- Create a volume from a snapshot.
- Copy an image to a volume.
- Clone a volume.
- Extend a volume.
- Migrate a volume.
- Retype a volume.
- Get volume statistics.
- Create and delete consistency groups.
- Create, list, and delete consistency group snapshots.
- Modify consistency groups.
- Efficient non-disruptive volume backup.

2.1.6.2. Preparation

This section contains instructions to prepare the Block Storage nodes to use the EMC VNX driver. You install the Navisphere CLI, install the driver, ensure you have correct zoning configurations, and register the driver.

2.1.6.2.1. Install Navisphere CLI

Navisphere CLI needs to be installed on all Block Storage nodes within an OpenStack deployment. You need to download different versions for different platforms.

- For all other variants of Linux, Navisphere CLI is available at [Downloads for VNX2 Series](#) or [Downloads for VNX1 Series](#).
- After installation, set the security level of Navisphere CLI to low:

```
$ /opt/Navisphere/bin/naviseccli security -certificate -setLevel low
```

2.1.6.2.2. Check array software

Make sure you have following software installed for certain features.

Table 2.5. Required software

Feature	Software Required
All	ThinProvisioning
All	VNXSnapshots
FAST cache support	FASTCache
Create volume with type compressed	Compression
Create volume with type deduplicated	Deduplication

2.1.6.2.3. Install EMC VNX driver

Both **EMCCLIISCSIDriver** and **EMCCLIFCDriver** are included in the Block Storage installer package:

- `emc_vnx_cli.py`
- `emc_cli_fc.py` (for **EMCCLIFCDriver**)
- `emc_cli_iscsi.py` (for **EMCCLIISCSIDriver**)

2.1.6.2.4. Network configuration

For FC Driver, FC zoning is properly configured between hosts and VNX. Check [Section 2.1.6.8.2, “Register FC port with VNX”](#) for reference.

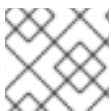
For iSCSI Driver, make sure your VNX iSCSI port is accessible by your hosts. Check [Section 2.1.6.8.3, “Register iSCSI port with VNX”](#) for reference.

You can use `initiator_auto_registration=True` configuration to avoid register the ports manually. Check the detail of the configuration in [Section 2.1.6.3, “Backend configuration”](#) for reference.

If you are trying to setup multipath, see *Multipath Setup* in [Section 2.1.6.6.1, “Multipath setup”](#).

2.1.6.3. Backend configuration

Make the following changes in `/etc/cinder/cinder.conf` file:



NOTE

Changes to your configuration won't take effect until you restart your cinder service.

2.1.6.3.1. Minimum configuration

Here is a sample of minimum backend configuration. See following sections for the detail of each option. Replace **EMCCLIFCDriver** to **EMCCLIISCSIDriver** if you are using the iSCSI driver.

[DEFAULT]

```

enabled_backends = vnx_array1

[vnx_array1]
san_ip = 10.10.72.41
san_login = sysadmin
san_password = sysadmin
naviseccli_path = /opt/Navisphere/bin/naviseccli
volume_driver=cinder.volume.drivers.emc.emc_cli_fc.EMCCLIFCDriver
initiator_auto_registration=True

```

2.1.6.3.2. Multi-backend configuration

Here is a sample of a multi-backend configuration. See following sections for the detail of each option. Replace **EMCCLIFCDriver** to **EMCCLIISCSIDriver** if your are using the iSCSI driver.

```

[DEFAULT]
enabled_backends=backendA, backendB

[backendA]
storage_vnx_pool_names = Pool_01_SAS, Pool_02_FLASH
san_ip = 10.10.72.41
storage_vnx_security_file_dir = /etc/secfile/array1
naviseccli_path = /opt/Navisphere/bin/naviseccli
volume_driver=cinder.volume.drivers.emc.emc_cli_fc.EMCCLIFCDriver
initiator_auto_registration=True

[backendB]
storage_vnx_pool_names = Pool_02_SAS
san_ip = 10.10.26.101
san_login = username
san_password = password
naviseccli_path = /opt/Navisphere/bin/naviseccli
volume_driver=cinder.volume.drivers.emc.emc_cli_fc.EMCCLIFCDriver
initiator_auto_registration=True

```

For more details on multi-backends, see [OpenStack Cloud Administration Guide](#)

2.1.6.3.3. Required configurations

2.1.6.3.3.1. IP of the VNX Storage Processors

Specify the SP A and SP B IP to connect.

```

san_ip = <IP of VNX Storage Processor A>
san_secondary_ip = <IP of VNX Storage Processor B>

```

2.1.6.3.3.2. VNX login credentials

There are two ways to specify the credentials.

- Use plain text username and password.

Supply for plain username and password as below.

■

```
san_login = <VNX account with administrator role>
san_password = <password for VNX account>
storage_vnx_authentication_type = global
```

Valid values for **storage_vnx_authentication_type** are: **global** (default), **local**, **ldap**

- Use Security file

This approach avoids the plain text password in your cinder configuration file. Supply a security file as below:

```
storage_vnx_security_file_dir=<path to security file>
```

Check the Unisphere CLI user guide or [Section 2.1.6.8.1, “Authenticate by security file”](#) for how to create a security file.

2.1.6.3.3.3. Path to your Unisphere CLI

Specify the absolute path to your naviseccli.

```
naviseccli_path = /opt/Navisphere/bin/naviseccli
```

2.1.6.3.3.4. Driver name

- For the FC Driver, add the following option:

```
volume_driver=cinder.volume.drivers.emc.emc_cli_fc.EMCCLIFCDriver
```

- For iSCSI Driver, add following option:

```
volume_driver=cinder.volume.drivers.emc.emc_cli_iscsi.EMCCLIISCSIDriver
```

2.1.6.3.4. Optional configurations

2.1.6.3.4.1. VNX pool names

Specify the list of pools to be managed, separated by ','. They should already exist in VNX.

```
storage_vnx_pool_names = pool 1, pool 2
```

If this value is not specified, all pools of the array will be used.

2.1.6.3.4.2. Initiator auto registration

When **initiator_auto_registration=True**, the driver will automatically register initiators to all working target ports of the VNX array during volume attaching (The driver will skip those initiators that have already been registered) if the option **io_port_list** is not specified in **cinder.conf**.

If the user wants to register the initiators with some specific ports but not register with the other ports, this functionality should be disabled.

When a comma-separated list is given to `io_port_list`, the driver will only register the initiator to the ports specified in the list and only return target port(s) which belong to the target ports in the `io_port_list` instead of all target ports.

- Example for FC ports:

```
io_port_list=a-1,B-3
```

`a` or `B` is *Storage Processor*, number `1` and `3` are *Port ID*.

- Example for iSCSI ports:

```
io_port_list=a-1-0,B-3-0
```

`a` or `B` is *Storage Processor*, the first numbers `1` and `3` are *Port ID* and the second number `0` is *Virtual Port ID*



NOTE

- Rather than de-registered, the registered ports will be simply bypassed whatever they are in 'io_port_list' or not.
- The driver will raise an exception if ports in `io_port_list` are not existed in VNX during startup.

2.1.6.3.4.3. Force delete volumes in storage group

Some **available** volumes may remain in storage group on the VNX array due to some OpenStack timeout issue. But the VNX array do not allow the user to delete the volumes which are in storage group. Option `force_delete_lun_in_storagegroup` is introduced to allow the user to delete the **available** volumes in this tricky situation.

When `force_delete_lun_in_storagegroup=True` in the back-end section, the driver will move the volumes out of storage groups and then delete them if the user tries to delete the volumes that remain in storage group on the VNX array.

The default value of `force_delete_lun_in_storagegroup` is `False`.

2.1.6.3.4.4. Over subscription in thin provisioning

Over subscription allows that the sum of all volumes' capacity (provisioned capacity) to be larger than the pool's total capacity.

`max_over_subscription_ratio` in the back-end section is the ratio of provisioned capacity over total capacity.

The default value of `max_over_subscription_ratio` is 20.0, which means the provisioned capacity can not exceed the total capacity. If the value of this ratio is set larger than 1.0, the provisioned capacity can exceed the total capacity.

2.1.6.3.4.5. Storage group automatic deletion

For volume attaching, the driver has a storage group on VNX for each compute node hosting the vm instances which are going to consume VNX Block Storage (using compute node's hostname as storage

group's name). All the volumes attached to the VM instances in a compute node will be put into the storage group. If `destroy_empty_storage_group=True`, the driver will remove the empty storage group after its last volume is detached. For data safety, it does not suggest to set `destroy_empty_storage_group=True` unless the VNX is exclusively managed by one Block Storage node because consistent lock_path is required for operation synchronization for this behavior.

2.1.6.3.4.6. Initiator auto deregistration

Enabling storage group automatic deletion is the precondition of this function. If `initiator_auto_deregistration=True` is set, the driver will deregister all the initiators of the host after its storage group is deleted.

2.1.6.3.4.7. FC SAN auto zoning

The EMC VNX FC driver supports FC SAN auto zoning when ZoneManager is configured. Set `zoning_mode` to `fabric` in `DEFAULT` section to enable this feature. For ZoneManager configuration, refer to Block Storage official guide.

2.1.6.3.4.8. Volume number threshold

In VNX, there is a limitation on the number of pool volumes that can be created in the system. When the limitation is reached, no more pool volumes can be created even if there is remaining capacity in the storage pool. In other words, if the scheduler dispatches a volume creation request to a back end that has free capacity but reaches the volume limitation, the creation fails.

The default value of `check_max_pool_luns_threshold` is `False`. When `check_max_pool_luns_threshold=True`, the pool-based back end will check the limit and will report 0 free capacity to the scheduler if the limit is reached. So the scheduler will be able to skip this kind of pool-based back end that runs out of the pool volume number.

2.1.6.3.4.9. iSCSI initiators

`iscsi_initiators` is a dictionary of IP addresses of the iSCSI initiator ports on OpenStack Nova/Cinder nodes which want to connect to VNX via iSCSI. If this option is configured, the driver will leverage this information to find an accessible iSCSI target portal for the initiator when attaching volume. Otherwise, the iSCSI target portal will be chosen in a relative random way.

This option is only valid for iSCSI driver.

Here is an example. VNX will connect `host1` with `10.0.0.1` and `10.0.0.2`. And it will connect `host2` with `10.0.0.3`.

The key name (like `host1` in the example) should be the output of command `hostname`.

```
iscsi_initiators = {"host1":["10.0.0.1", "10.0.0.2"],"host2":["10.0.0.3"]}
```

2.1.6.3.4.10. Default timeout

Specify the timeout(minutes) for operations like LUN migration, LUN creation, etc. For example, LUN migration is a typical long running operation, which depends on the LUN size and the load of the array. An upper bound in the specific deployment can be set to avoid unnecessary long wait.

The default value for this option is infinite.

Example:

```
default_timeout = 10
```

2.1.6.3.4.11. Max LUNs per storage group

`max_luns_per_storage_group` specify the max number of LUNs in a storage group. Default value is 255. It is also the max value supported by VNX.

2.1.6.3.4.12. Ignore pool full threshold

if `ignore_pool_full_threshold` is set to `True`, driver will force LUN creation even if the full threshold of pool is reached. Default to `False`

2.1.6.4. Extra spec options

Extra specs are used in volume types created in cinder as the preferred property of the volume.

The Block storage scheduler will use extra specs to find the suitable back end for the volume and the Block storage driver will create the volume based on the properties specified by the extra spec.

Use following command to create a volume type:

```
$ cinder type-create "demoVolumeType"
```

Use following command to update the extra spec of a volume type:

```
$ cinder type-key "demoVolumeType" set provisioning:type=thin
```

Volume types can also be configured in OpenStack Horizon.

In VNX Driver, we defined several extra specs. They are introduced below:

2.1.6.4.1. Provisioning type

- **Key: `provisioning:type`**
- **Possible Values:**
 - `thick`

Volume is fully provisioned.

Example 2.5. creating a thick volume type:

```
$ cinder type-create "ThickVolumeType"
$ cinder type-key "ThickVolumeType" set provisioning:type=thick
thick_provisioning_support='<is> True'
```

- `thin`

Volume is virtually provisioned

Example 2.6. creating a thin volume type:

```
$ cinder type-create "ThinVolumeType"
$ cinder type-key "ThinVolumeType" set provisioning:type=thin
thin_provisioning_support='<is> True'
```

- **deduplicated**

Volume is **thin** and deduplication is enabled. The administrator shall go to VNX to configure the system level deduplication settings. To create a deduplicated volume, the VNX Deduplication license must be activated on VNX, and specify **deduplication_support=True** to let Block Storage scheduler find the proper volume back end.

Example 2.7. creating a deduplicated volume type:

```
$ cinder type-create "DeduplicatedVolumeType"
$ cinder type-key "DeduplicatedVolumeType" set
provisioning:type=deduplicated deduplication_support='<is> True'
```

- **compressed**

Volume is **thin** and compression is enabled. The administrator shall go to the VNX to configure the system level compression settings. To create a compressed volume, the VNX Compression license must be activated on VNX , and use **compression_support=True** to let Block Storage scheduler find a volume back end. VNX does not support creating snapshots on a compressed volume.

Example 2.8. creating a compressed volume type:

```
$ cinder type-create "CompressedVolumeType"
$ cinder type-key "CompressedVolumeType" set
provisioning:type=compressed compression_support='<is> True'
```

- **Default: thick**

**NOTE**

provisioning:type replaces the old spec key **storage_type:provisioning**. The latter one will be obsoleted in the next release. If both **provisioning:type** and **storage_type:provisioning** are set in the volume type, the value of **provisioning:type** will be used.

2.1.6.4.2. Storage tiering support

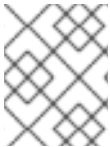
- **Key: storage_type:tiering**
- **Possible Values:**

- **StartHighThenAuto**
- **Auto**
- **HighestAvailable**
- **LowestAvailable**
- **NoMovement**
- **Default: StartHighThenAuto**

VNX supports fully automated storage tiering which requires the FAST license activated on the VNX. The OpenStack administrator can use the extra spec key **storagetype:tiering** to set the tiering policy of a volume and use the key **fast_support='<is> True'** to let Block Storage scheduler find a volume back end which manages a VNX with FAST license activated. Here are the five supported values for the extra spec key **storagetype:tiering**:

Example 2.9. creating a volume types with tiering policy:

```
$ cinder type-create "ThinVolumeOnLowestAvaibleTier"
$ cinder type-key "CompressedVolumeOnLowestAvaibleTier" set
provisioning:type=thin storagetype:tiering=Auto fast_support='<is> True'
```



NOTE

Tiering policy can not be applied to a deduplicated volume. Tiering policy of the deduplicated LUN align with the settings of the pool.

2.1.6.4.3. FAST cache support

- **Key: fast_cache_enabled**
- **Possible Values:**
 - **True**
 - **False**
- **Default: False**

VNX has FAST Cache feature which requires the FAST Cache license activated on the VNX. Volume will be created on the backend with FAST cache enabled when **True** is specified.

2.1.6.4.4. Snap-copy

- **Key: copytype:snap**
- **Possible Values:**
 - **True**
 - **False**

- **Default: False**

The VNX driver supports snap-copy, which extremely accelerates the process for creating a copied volume.

By default, the driver will do full data copy when creating a volume from a snapshot or cloning a volume, which is time-consuming especially for large volumes. When the snap-copy is used, the driver will simply create a snapshot and mount it as a volume for the 2 kinds of operations which will be instant even for large volumes.

To enable this functionality, the source volume should have **copytype:snap=True** in the extra specs of its volume type. Then the new volume cloned from the source or copied from the snapshot for the source, will be in fact a snap-copy instead of a full copy. If a full copy is needed, retype/migration can be used to convert the snap-copy volume to a full-copy volume which may be time-consuming.

```
$ cinder type-create "SnapCopy"
$ cinder type-key "SnapCopy" set copytype:snap=True
```

User can determine whether the volume is a snap-copy volume or not by showing its metadata. If the 'lun_type' in metadata is 'smp', the volume is a snap-copy volume. Otherwise, it is a full-copy volume.

```
$ cinder metadata-show <volume>
```

Constraints:

- **copytype:snap=True** is not allowed in the volume type of a consistency group.
- Clone and snapshot creation are not allowed on a copied volume created through the snap-copy before it is converted to a full copy.
- The number of snap-copy volume created from a source volume is limited to 255 at one point in time.
- The source volume which has snap-copy volume can not be deleted.

2.1.6.4.5. Pool name

- **Key: pool_name**
- **Possible Values:** name of the storage pool managed by cinder
- **Default:** None

If the user wants to create a volume on a certain storage pool in a backend that manages multiple pools, a volume type with a extra spec specified storage pool should be created first, then the user can use this volume type to create the volume.

Example 2.10. Creating the volume type:

```
$ cinder type-create "HighPerf"
$ cinder type-key "HighPerf" set pool_name=Pool_02_SASFLASH
volume_backend_name=vnx_41
```

2.1.6.4.6. Obsoleted extra specs in Mitaka

Avoid using following extra spec keys.

- `storagetype:provisioning`
- `storagetype:pool`

2.1.6.5. Advanced features

2.1.6.5.1. Read-only volumes

OpenStack supports read-only volumes. The following command can be used to set a volume as read-only.

```
$ cinder readonly-mode-update <volume> True
```

After a volume is marked as read-only, the driver will forward the information when a hypervisor is attaching the volume and the hypervisor will make sure the volume is read-only.

2.1.6.5.2. Efficient non-disruptive volume backup

The default implementation in Cinder for non-disruptive volume backup is not efficient since a cloned volume will be created during backup.

The approach of efficient backup is to create a snapshot for the volume and connect this snapshot (a mount point in VNX) to the Cinder host for volume backup. This eliminates migration time involved in volume clone.

Constraints:

- Backup creation for a snap-copy volume is not allowed if the volume status is `in-use` since snapshot cannot be taken from this volume.

2.1.6.6. Best practice

2.1.6.6.1. Multipath setup

Enabling multipath volume access is recommended for robust data access. The major configuration includes:

- Install `multipath-tools`, `sysfsutils` and `sg3-utils` on nodes hosting Nova-Compute and Cinder-Volume services (Check the operating system manual for the system distribution for specific installation steps. For Red Hat based distributions, they should be `device-mapper-multipath`, `sysfsutils` and `sg3_utils`).
- Specify `use_multipath_for_image_xfer=true` in `cinder.conf` for each FC/iSCSI back end.
- Specify `iscsi_use_multipath=True` in `libvirt` section of `nova.conf`. This option is valid for both iSCSI and FC driver.

For multipath-tools, here is an EMC recommended sample of `/etc/multipath.conf`.

`user_friendly_names` is not specified in the configuration and thus it will take the default value `no`. It is NOT recommended to set it to `yes` because it may fail operations such as VM live migration.

```
blacklist {
    # Skip the files under /dev that are definitely not FC/iSCSI devices
    # Different system may need different customization
    devnode "(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
    devnode "^hd[a-z][0-9]*"
    devnode "^cciss!c[0-9]d[0-9]*[p[0-9]*]"

    # Skip LUNZ device from VNX
    device {
        vendor "DGC"
        product "LUNZ"
    }
}

defaults {
    user_friendly_names no
    flush_on_last_del yes
}

devices {
    # Device attributed for EMC CLARiiON and VNX series ALUA
    device {
        vendor "DGC"
        product ".*"
        product_blacklist "LUNZ"
        path_grouping_policy group_by_prio
        path_selector "round-robin 0"
        path_checker emc_clariion
        features "1 queue_if_no_path"
        hardware_handler "1 alua"
        prio alua
        failback immediate
    }
}
```



NOTE

When multipath is used in OpenStack, multipath faulty devices may come out in Nova-Compute nodes due to different issues ([Bug 1336683](#) is a typical example).

A solution to completely avoid faulty devices has not been found yet. `faulty_device_cleanup.py` mitigates this issue when VNX iSCSI storage is used. Cloud administrators can deploy the script in all Nova-Compute nodes and use a CRON job to run the script on each Nova-Compute node periodically so that faulty devices will not stay too long. See [VNX faulty device cleanup](#) for detailed usage and the script.

2.1.6.7. Restrictions and limitations

2.1.6.7.1. iSCSI port cache

EMC VNX iSCSI driver caches the iSCSI ports information, so that the user should restart the cinder-

volume service or wait for seconds (which is configured by `periodic_interval` in `cinder.conf`) before any volume attachment operation after changing the iSCSI port configurations. Otherwise the attachment may fail because the old iSCSI port configurations were used.

2.1.6.7.2. No extending for volume with snapshots

VNX does not support extending the thick volume which has a snapshot. If the user tries to extend a volume which has a snapshot, the status of the volume would change to `error_extending`.

2.1.6.7.3. Limitations for deploying cinder on computer node

It is not recommended to deploy the driver on a compute node if `cinder upload-to-image --force True` is used against an in-use volume. Otherwise, `cinder upload-to-image --force True` will terminate the data access of the vm instance to the volume.

2.1.6.7.4. Storage group with host names in VNX

When the driver notices that there is no existing storage group that has the host name as the storage group name, it will create the storage group and also add the compute node's or Block Storage nodes' registered initiators into the storage group.

If the driver notices that the storage group already exists, it will assume that the registered initiators have also been put into it and skip the operations above for better performance.

It is recommended that the storage administrator does not create the storage group manually and instead relies on the driver for the preparation. If the storage administrator needs to create the storage group manually for some special requirements, the correct registered initiators should be put into the storage group as well (otherwise the following volume attaching operations will fail).

2.1.6.7.5. EMC storage-assisted volume migration

EMC VNX driver supports storage-assisted volume migration, when the user starts migrating with `cinder migrate --force-host-copy False <volume_id> <host>` or `cinder migrate <volume_id> <host>`, cinder will try to leverage the VNX's native volume migration functionality.

In following scenarios, VNX storage-assisted volume migration will not be triggered:

1. Volume migration between back ends with different storage protocol, ex, FC and iSCSI.
2. Volume is to be migrated across arrays.

2.1.6.8. Appendix

2.1.6.8.1. Authenticate by security file

VNX credentials are necessary when the driver connects to the VNX system. Credentials in global, local and ldap scopes are supported. There are two approaches to provide the credentials:

The recommended one is using the Navisphere CLI security file to provide the credentials which can get rid of providing the plain text credentials in the configuration file. Following is the instruction on how to do this.

1. Find out the Linux user id of the `cinder-volume` processes. Assuming the service `cinder-volume` is running by the account `cinder`.

2. Run `su` as root user.
3. In `/etc/passwd`, change `cinder:x:113:120::/var/lib/cinder:/bin/false` to `cinder:x:113:120::/var/lib/cinder:/bin/bash` (This temporary change is to make step 4 work.)
4. Save the credentials on behave of `cinder` user to a security file (assuming the array credentials are `admin/admin` in `global` scope). In the command below, the `'-secfilepath'` switch is used to specify the location to save the security file.

```
# su -l cinder -c '/opt/Navisphere/bin/naviseccli -AddUserSecurity -
user admin -password admin -scope 0 -secfilepath <location>'
```

5. Change `cinder:x:113:120::/var/lib/cinder:/bin/bash` back to `cinder:x:113:120::/var/lib/cinder:/bin/false` in `/etc/passwd`
6. Remove the credentials options `san_login`, `san_password` and `storage_vnx_authentication_type` from `cinder.conf`. (normally it is `/etc/cinder/cinder.conf`). Add option `storage_vnx_security_file_dir` and set its value to the directory path of your security file generated in step 4. Omit this option if `-secfilepath` is not used in step 4.
7. Restart the `cinder-volume` service to validate the change.

2.1.6.8.2. Register FC port with VNX

This configuration is only required when `initiator_auto_registration=False`.

To access VNX storage, the compute nodes should be registered on VNX first if initiator auto registration is not enabled.

To perform "Copy Image to Volume" and "Copy Volume to Image" operations, the nodes running the `cinder-volume` service (Block Storage nodes) must be registered with the VNX as well.

The steps mentioned below are for the compute nodes. Follow the same steps for the Block Storage nodes also (The steps can be skipped if initiator auto registration is enabled).

1. Assume `20:00:00:24:FF:48:BA:C2:21:00:00:24:FF:48:BA:C2` is the WWN of a FC initiator port name of the compute node whose hostname and IP are `myhost1` and `10.10.61.1`. Register `20:00:00:24:FF:48:BA:C2:21:00:00:24:FF:48:BA:C2` in Unisphere:
 - a. Login to Unisphere, go to `FNMO0000000000->Hosts->Initiators`.
 - b. Refresh and wait until the initiator `20:00:00:24:FF:48:BA:C2:21:00:00:24:FF:48:BA:C2` with SP Port `A-1` appears.
 - c. Click the **Register** button, select **CLARiiON/VNX** and enter the hostname (which is the output of the linux command `hostname`) and IP address:
 - Hostname: `myhost1`
 - IP: `10.10.61.1`
 - Click **Register**

d. Then host **10.10.61.1** will appear under **Hosts->Host List** as well.

2. Register the wwn with more ports if needed.

2.1.6.8.3. Register iSCSI port with VNX

This configuration is only required when `initiator_auto_registration=False`.

To access VNX storage, the compute nodes should be registered on VNX first if initiator auto registration is not enabled.

To perform "Copy Image to Volume" and "Copy Volume to Image" operations, the nodes running the cinder-volume service (Block Storage nodes) must be registered with the VNX as well.

The steps mentioned below are for the compute nodes. Follow the same steps for the Block Storage nodes also (The steps can be skipped if initiator auto registration is enabled).

1. On the compute node with IP address **10.10.61.1** and hostname **myhost1**, execute the following commands (assuming **10.10.61.35** is the iSCSI target):

a. Start the iSCSI initiator service on the node

```
# /etc/init.d/open-iscsi start
```

b. Discover the iSCSI target portals on VNX

```
# iscsiadm -m discovery -t st -p 10.10.61.35
```

c. Enter **/etc/iscsi**

```
# cd /etc/iscsi
```

d. Find out the iqn of the node

```
# more initiatorname.iscsi
```

2. Login to VNX from the compute node using the target corresponding to the SPA port:

```
# iscsiadm -m node -T iqn.1992-04.com.emc:cx.apm01234567890.a0 -p 10.10.61.35 -l
```

3. Assume **iqn.1993-08.org.debian:01:1a2b3c4d5f6g** is the initiator name of the compute node. Register **iqn.1993-08.org.debian:01:1a2b3c4d5f6g** in Unisphere:

a. Login to Unisphere, go to **FN0000000000->Hosts->Initiators** .

b. Refresh and wait until the initiator **iqn.1993-08.org.debian:01:1a2b3c4d5f6g** with **SP Port A-8v0** appears.

c. Click the **Register** button, select **CLARiiON/VNX** and enter the hostname (which is the output of the linux command `hostname`) and IP address:

- **Hostname:myhost1**

- IP: **10.10.61.1**
 - Click **Register**
- d. Then host **10.10.61.1** will appear under **Hosts->Host List** as well.

4. Logout iSCSI on the node:

```
# iscsiadm -m node -u
```

5. Login to VNX from the compute node using the target corresponding to the SPB port:

```
# iscsiadm -m node -T iqn.1992-04.com.emc:cx.apm01234567890.b8 -p 10.10.61.36 -l
```

6. In Unisphere register the initiator with the SPB port.

7. Logout iSCSI on the node:

```
# iscsiadm -m node -u
```

8. Register the iqn with more ports if needed.

2.1.7. EMC XtremIO Block Storage driver configuration

The high performance XtremIO All Flash Array (AFA) offers Block Storage services to OpenStack. Using the driver, OpenStack Block Storage hosts can connect to an XtremIO Storage cluster.

This section explains how to configure and connect an OpenStack block storage host to an XtremIO storage cluster.

2.1.7.1. Support matrix

- Xtremapp: Version 3.0 and 4.0

2.1.7.2. Supported operations

- Create, delete, clone, attach, and detach volumes
- Create and delete volume snapshots
- Create a volume from a snapshot
- Copy an image to a volume
- Copy a volume to an image
- Extend a volume
- Manage and unmanage a volume
- Get volume statistics

2.1.7.3. XtremIO Block Storage driver configuration

Edit the `cinder.conf` file by adding the configuration below under the **[DEFAULT]** section of the file in case of a single back end or under a separate section in case of multiple back ends (for example **[XTREMIO]**). The configuration file is usually located under the following path `/etc/cinder/cinder.conf`.

For a configuration example, refer to the configuration [example](#).

2.1.7.3.1. XtremIO driver name

Configure the driver name by adding the following parameter:

- For iSCSI `volume_driver = cinder.volume.drivers.emc.xtremio.XtremIOIscsiDriver`
- For Fibre Channel `volume_driver = cinder.volume.drivers.emc.xtremio.XtremIOFibreChannelDriver`

2.1.7.3.2. XtremIO management server (XMS) IP

To retrieve the management IP, use the `show-xms` CLI command.

Configure the management IP by adding the following parameter: `san_ip = XMS Management IP`

2.1.7.3.3. XtremIO cluster name

In XtremIO version 4.0, a single XMS can manage multiple cluster back ends. In such setups, the administrator is required to specify the cluster name (in addition to the XMS IP). Each cluster must be defined as a separate back end.

To retrieve the Cluster Name, run the `show-clusters` CLI command.

Configure the cluster name by adding the `xtremio_cluster_name = Cluster-Name`



NOTE

When a single cluster is managed in XtremIO version 4.0, the cluster name is not required.

2.1.7.3.4. XtremIO user credentials

OpenStack Block Storage requires an XtremIO XMS user with administrative privileges. XtremIO recommends creating a dedicated OpenStack user account that holds an administrative user role.

Refer to the *XtremIO User Guide* for details on user account management

Create an XMS account using either the XMS GUI or the `add-user-account` CLI command.

Configure the user credentials by adding the following parameters:

`san_login = XMS username`

`san_password = XMS username password`

2.1.7.4. Multiple back ends

Configuring multiple storage back ends enables you to create several back-end storage solutions that serve the same OpenStack Compute resources.

When a volume is created, the scheduler selects the appropriate back end to handle the request, according to the specified volume type.

2.1.7.5. Setting thin provisioning and multipathing parameters

To support thin provisioning and multipathing in the XtremIO Array, the following parameters from the Nova and Cinder configuration files should be modified as follows:

- Thin Provisioning

All XtremIO volumes are thin provisioned. The default value of 20 should be maintained for the `max_over_subscription_ratio` parameter.

The `use_cow_images` parameter in the `nova.conf` file should be set to `False` as follows:

```
use_cow_images = false
```

- Multipathing

The `use_multipath_for_image_xfer` parameter in the `cinder.conf` file should be set to `True` as follows:

```
use_multipath_for_image_xfer = true
```

2.1.7.6. Restarting OpenStack Block Storage

Save the `cinder.conf` file and restart cinder by running the following command:

```
$ openstack-service restart cinder-volume
```

2.1.7.7. Configuring CHAP

The XtremIO Block Storage driver supports CHAP initiator authentication. If CHAP initiator authentication is required, set the CHAP Authentication mode to initiator.

To set the CHAP initiator mode using CLI, run the following CLI command:

```
$ modify-chap chap-authentication-mode=initiator
```

The CHAP initiator mode can also be set via the XMS GUI

Refer to *XtremIO User Guide* for details on CHAP configuration via GUI and CLI.

The CHAP initiator authentication credentials (username and password) are generated automatically by the Block Storage driver. Therefore, there is no need to configure the initial CHAP credentials manually in XMS.

2.1.7.8. Configuration example cinder.conf example file

You can update the `cinder.conf` file by editing the necessary parameters as follows:

```
[Default]
enabled_backends = XtremIO

[XtremIO]
volume_driver =
cinder.volume.drivers.emc.xtremio.XtremIOFibreChannelDriver
san_ip = XMS_IP
xtremio_cluster_name = Cluster01
san_login = XMS_USER
san_password = XMS_PASSWD
volume_backend_name = XtremIOAFA
```

2.1.8. Fujitsu ETERNUS DX driver

The Fujitsu ETERNUS DX driver provides FC and iSCSI support for ETERNUS DX S3 series.

The driver performs volume operations by communicating with ETERNUS DX. It uses a CIM client in Python called PyWBEM to perform CIM operations over HTTP. You can specify RAID Group and Thin Provisioning Pool (TPP) in ETERNUS DX as a storage pool.

System requirements

- Firmware version V10L30 or later is required.
- An Advanced Copy Feature license is required to create a snapshot and a clone.
- The pywbem should be installed on the Controller node.



NOTE

The multipath environment with ETERNUS Multipath Driver is unsupported.

Supported operations

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.
- Create a volume from a snapshot.
- Copy an image to a volume.
- Copy a volume to an image.
- Clone a volume.
- Extend a volume. ^[1]
- Get volume statistics.

2.1.8.1. Configure the Fujitsu ETERNUS device

Before you can define the Fujitsu ETERNUS device as a Block Storage back end, you need to configure storage pools and ports on the device first. Consult your device documentation for details on each step:

1. Set up a LAN connection between the Controller nodes (where the Block Storage service is hosted) and MNT ports of the ETERNUS device.
2. Set up a SAN connection between the Compute nodes and CA ports of the ETERNUS device.
3. Log in to the ETERNUS device using an account with the **Admin** role.
4. Enable the SMI-S of ETERNUS DX.
5. Register an **Advanced Copy Feature** license and configure the copy table size.
6. Create a storage pool for volumes. This pool will be used later in the **EternusPool** setting in [Section 2.1.8.2, “Configuring the Back End”](#).

**NOTE**

If you want to create volume snapshots on a different storage pool, create a storage pool for that as well. This pool will be used in the **EternusSnapPool** setting in [Section 2.1.8.2, “Configuring the Back End”](#).

7. Create a *Snap Data Pool Volume (SDPV)* to enable Snap Data Pool (SDP) for the **create a snapshot** function.
8. Configure *storage ports* to be used by the Block Storage service. Then:
 - a. Set those ports to **CA** mode.
 - b. Enable the **host-affinity** settings of those storage ports. To enable **host-affinity**, run the following from the ETERNUS CLI for each port:

```
CLI> set PROTO-parameters -host-affinity enable -port CM# CA#
PORT
```

Where: * *PROTO* defines which storage protocol is in use, as in **fc** (Fibre Channel) or **iscsi**. * *CM# CA#* refer to the controller enclosure where the port is located. * *PORT* is the port number.

2.1.8.2. Configuring the Back End

Fujitsu Eternus back ends use either of the following drivers:

- `cinder.volume.drivers.fujitsu.eternus_dx_fc.FJDXFCDriver` (fibre channel)
- `cinder.volume.drivers.fujitsu.eternus_dx_iscsi.FJDXISCSIDriver` (iSCSI)

The settings for Fujitsu Eternus back ends are defined in a separate XML file. To define a back end, set **volume_driver** to the corresponding driver and **cinder_eternus_config_file** to point to the back end's XML configuration file. For example, if your fibre channel back end settings are defined in `/etc/cinder/eternus-dx.xml`, use:

```
volume_driver = cinder.volume.drivers.fujitsu.eternus_dx_fc.FJDXFCDriver
cinder_eternus_config_file = /etc/cinder/eternus_dx.xml
```

If you set the driver without defining **cinder_eternus_config_file**, then the driver will use **cinder_eternus_config_file = etc/cinder/cinder_fujitsu_eternus_dx.xml** by default.

The XML configuration file should contain the following settings:

EternusIP

IP address of the SMI-S connection of the ETERNUS device. Specifically, use the IP address of the MNT port of the device.

EternusPort

port number for the SMI-S connection port of the ETERNUS device.

EternusUser

User name to be used for the SMI-S connection (**EternusIP**).

EternusPassword

Corresponding password of **EternusUser** on **EternusIP**.

EternusPool

Name of the storage pool created for volumes (from [Section 2.1.8.1, “Configure the Fujitsu ETERNUS device”](#)). Specifically, use the pool’s RAID Group name or TPP name in the ETERNUS device.

EternusSnapPool

Name of the storage pool created for volume snapshots (from [Section 2.1.8.1, “Configure the Fujitsu ETERNUS device”](#)). Specifically, use the pool’s RAID Group name in the ETERNUS device. If you did not create a different pool for snapshots, use the same value as **EternusPool**.

EternusISCSIIP

(iSCSI only) IP address for iSCSI connections to the ETERNUS device. You can specify multiple IPs by creating an entry for each one.

For example, with a fibre-channel back end:

```
<?xml version='1.0' encoding='UTF-8'?>
<FUJITSU>
<EternusIP>0.0.0.0</EternusIP>
<EternusPort>5988</EternusPort>
<EternusUser>smisuser</EternusUser>
<EternusPassword>smispassword</EternusPassword>
<EternusPool>raid5_0001</EternusPool>
<EternusSnapPool>raid5_0001</EternusSnapPool>
</FUJITSU>
```

With an iSCSI back end:

```
<?xml version='1.0' encoding='UTF-8'?>
<FUJITSU>
<EternusIP>0.0.0.0</EternusIP>
<EternusPort>5988</EternusPort>
<EternusUser>smisuser</EternusUser>
<EternusPassword>smispassword</EternusPassword>
<EternusPool>raid5_0001</EternusPool>
<EternusSnapPool>raid5_0001</EternusSnapPool>
<EternusISCSIIP>1.1.1.1</EternusISCSIIP>
<EternusISCSIIP>1.1.1.2</EternusISCSIIP>
<EternusISCSIIP>1.1.1.3</EternusISCSIIP>
<EternusISCSIIP>1.1.1.4</EternusISCSIIP>
</FUJITSU>
```

2.1.9. HDS HNAS iSCSI and NFS driver

This OpenStack Block Storage volume driver provides iSCSI and NFS support for [Hitachi NAS Platform Models 3080, 3090, 4040, 4060, 4080 and 4100](#).

2.1.9.1. Supported operations

The NFS and iSCSI drivers support these operations:

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.
- Create a volume from a snapshot.
- Copy an image to a volume.
- Copy a volume to an image.
- Clone a volume.
- Extend a volume.
- Get volume statistics.
- Manage and unmanage a volume.

2.1.9.2. HNAS storage requirements

Before using iSCSI and NFS services, use the HNAS configuration and management GUI (SMU) or SSC CLI to create storage pool(s), file system(s), and assign an EVS. Make sure that the file system used is not created as a `replication target`. Additionally:

For NFS:

Create NFS exports, choose a path for them (it must be different from `/`) and set the `Show snapshots` option to `hide and disable access`.

Also, in the "Access Configuration" set the option `norootsquash`, e.g. `"* (rw, norootsquash)"`, so HNAS cinder driver can change the permissions of its volumes.

In order to use the hardware accelerated features of NFS HNAS, we recommend setting `max-nfs-version` to 3. Refer to HNAS command line reference to see how to configure this option.

For iSCSI:

You need to set an iSCSI domain.

2.1.9.3. Block storage host requirements

The Block storage host requires the `nfs-utils` package.

If you are not using SSH, you need the HDS SSC to communicate with an HNAS array using the `SSC` commands. This utility package is available in the RPM package distributed with the hardware through physical media or it can be manually copied from the SMU to the Block Storage host.

2.1.9.4. Package installation

If you are installing the driver from a RPM or DEB package, follow the steps bellow:

1. Install the dependencies:

```
# yum install nfs-utils nfs-utils-lib
```

2. Configure the driver as described in the [Section 2.1.9.5, “Driver configuration”](#) section.
3. Restart all cinder services (volume, scheduler and backup).

2.1.9.5. Driver configuration

The HDS driver supports the concept of differentiated services (also referred as quality of service) by mapping volume types to services provided through HNAS.

HNAS supports a variety of storage options and file system capabilities, which are selected through the definition of volume types and the use of multiple back ends. The driver maps up to four volume types into separated exports or file systems, and can support any number if using multiple back ends.

The configuration for the driver is read from an XML-formatted file (one per back end), which you need to create and set its path in the `cinder.conf` configuration file. Below are the configuration needed in the `cinder.conf` configuration file [2]:

```
[DEFAULT]
enabled_backends = hnas_iscsi1, hnas_nfs1
```

For HNAS iSCSI driver create this section:

```
[hnas_iscsi1]
volume_driver = cinder.volume.drivers.hitachi.hnas_iscsi.HDSISCSIDriver
hds_hnas_iscsi_config_file = /path/to/config/hnas_config_file.xml
volume_backend_name = HNAS-ISCASI
```

For HNAS NFS driver create this section:

```
[hnas_nfs1]
volume_driver = cinder.volume.drivers.hitachi.hnas_nfs.HDSNFSDriver
hds_hnas_nfs_config_file = /path/to/config/hnas_config_file.xml
volume_backend_name = HNAS-NFS
```

The XML file has the following format:

```
<?xml version = "1.0" encoding = "UTF-8" ?>
<config>
  <mgmt_ip0>172.24.44.15</mgmt_ip0>
  <hnas_cmd>ssc</hnas_cmd>
  <chap_enabled>False</chap_enabled>
  <ssh_enabled>False</ssh_enabled>
  <cluster_admin_ip0>10.1.1.1</cluster_admin_ip0>
  <username>supervisor</username>
  <password>supervisor</password>
  <svc_0>
```

```

    <volume_type>default</volume_type>
    <iscsi_ip>172.24.44.20</iscsi_ip>
    <hdp>fs01-husvm</hdp>
  </svc_0>
  <svc_1>
    <volume_type>platinum</volume_type>
    <iscsi_ip>172.24.44.20</iscsi_ip>
    <hdp>fs01-platinum</hdp>
  </svc_1>
</config>

```

2.1.9.6. HNAS volume driver XML configuration options

An OpenStack Block Storage node using HNAS drivers can have up to four services. Each service is defined by a `svc_n` tag (`svc_0`, `svc_1`, `svc_2`, or `svc_3` ^[3], for example). These are the configuration options available for each service label:

Table 2.6. Configuration options for service labels

Option	Type	Default	Description
volume_type	Required	default	When a create_volume call with a certain volume type happens, the volume type will try to be matched up with this tag. In each configuration file you must define the default volume type in the service labels and, if no volume type is specified, the default is used. Other labels are case sensitive and should match exactly. If no configured volume types match the incoming requested type, an error occurs in the volume creation.
iscsi_ip	Required only for iSCSI		An iSCSI IP address dedicated to the service.
hdp	Required		<p>For iSCSI driver: virtual file system label associated with the service.</p> <p>For NFS driver: path to the volume (<code><ip_address>:/<path></code>) associated with the service.</p> <p>Additionally, this entry must be added in the file used to list available NFS shares. This file is located, by default, in <code>/etc/cinder/nfs_shares</code> or you can specify the location in the nfs_shares_config option in the <code>cinder.conf</code> configuration file.</p>

These are the configuration options available to the **config** section of the XML config file:

Table 2.7. Configuration options

Option	Type	Default	Description
mgmt_ip0	Required		Management Port 0 IP address. Should be the IP address of the "Admin" EVS.
hnas_cmd	Optional	ssc	Command to communicate to HNAS array.
chap_enabled	Optional (iSCSI only)	True	Boolean tag used to enable CHAP authentication protocol.
username	Required	supervisor	It's always required on HNAS.
password	Required	supervisor	Password is always required on HNAS.
svc_0, svc_1, svc_2, svc_3	Optional	(at least one label has to be defined)	Service labels: these four predefined names help four different sets of configuration options. Each can specify HDP and a unique volume type.
cluster_admin_ip0	Optional if ssh_enabled is True		The address of HNAS cluster admin.
ssh_enabled	Optional	False	Enables SSH authentication between Block Storage host and the SMU.
ssh_private_key	Required if ssh_enabled is True	False	Path to the SSH private key used to authenticate in HNAS SMU. The public key must be uploaded to HNAS SMU using ssh-register-public-key (this is an SSH subcommand). Note that copying the public key HNAS using ssh-copy-id doesn't work properly as the SMU periodically wipe out those keys.

2.1.9.7. Service labels

HNAS driver supports differentiated types of service using the service labels. It is possible to create up to four types of them, as gold, platinum, silver and ssd, for example.

After creating the services in the XML configuration file, you must configure one **volume_type** per service. Each **volume_type** must have the metadata **service_label** with the same name configured in the **<volume_type>** section of that service. If this is not set, OpenStack Block Storage will schedule the volume creation to the pool with largest available free space or other criteria configured in volume filters.

```
$ cinder type-create default
$ cinder type-key default set service_label=default
$ cinder type-create platinum-tier
```

```
$ cinder type-key platinum set service_label=platinum
```

2.1.9.8. Multi-back-end configuration

If you use multiple back ends and intend to enable the creation of a volume in a specific back end, you must configure volume types to set the `volume_backend_name` option to the appropriate back end. Then, create `volume_type` configurations with the same `volume_backend_name`.

```
$ cinder type-create 'iscsi'
$ cinder type-key 'iscsi' set volume_backend_name = 'HNAS-ISCASI'
$ cinder type-create 'nfs'
$ cinder type-key 'nfs' set volume_backend_name = 'HNAS-NFS'
```

You can deploy multiple OpenStack HNAS drivers instances that each control a separate HNAS array. Each service (`svc_0`, `svc_1`, `svc_2`, `svc_3`) on the instances need to have a `volume_type` and `service_label` metadata associated with it. If no metadata is associated with a pool, OpenStack Block Storage filtering algorithm selects the pool with the largest available free space.

2.1.9.9. SSH configuration

Instead of using **SSC** on the Block Storage host and store its credential on the XML configuration file, HNAS driver supports **SSH** authentication. To configure that:

1. If you don't have a pair of public keys already generated, create it in the Block Storage host (leave the pass-phrase empty):

```
$ mkdir -p /opt/hds/ssh
$ ssh-keygen -f /opt/hds/ssh/hnaskey
```

2. Change the owner of the key to `cinder` (or the user the volume service will be run):

```
# chown -R cinder.cinder /opt/hds/ssh
```

3. Create the directory "ssh_keys" in the SMU server:

```
$ ssh [manager|supervisor]@<smu-ip> 'mkdir -p /var/opt/mercury-main/home/[manager|supervisor]/ssh_keys/'
```

4. Copy the public key to the "ssh_keys" directory:

```
$ scp /opt/hds/ssh/hnaskey.pub [manager|supervisor]@<smu-ip>:/var/opt/mercury-main/home/[manager|supervisor]/ssh_keys/
```

5. Access the SMU server:

```
$ ssh [manager|supervisor]@<smu-ip>
```

6. Run the command to register the SSH keys:

```
$ ssh-register-public-key -u [manager|supervisor] -f
ssh_keys/hnaskey.pub
```

7. Check the communication with HNAS in the Block Storage host:

```
$ ssh -i /opt/hds/ssh/hnaskey [manager|supervisor]@<smu-ip> 'ssc
<cluster_admin_ip0> df -a'
```

`<cluster_admin_ip0>` is "localhost" for single node deployments. This should return a list of available file systems on HNAS.

2.1.9.10. Editing the XML config file:

1. Set the "username".
2. Enable SSH adding the line "`<ssh_enabled> True</ssh_enabled>`" under "`<config>`" section.
3. Set the private key path: "`<ssh_private_key> /opt/hds/ssh/hnaskey</ssh_private_key>`" under "`<config>`" section.
4. If the HNAS is in a multi-cluster configuration set "`<cluster_admin_ip0>`" to the cluster node admin IP. In a single node HNAS, leave it empty.
5. Restart cinder services.



WARNING

Note that copying the public key HNAS using `ssh-copy-id` doesn't work properly as the SMU periodically wipe out those keys.

2.1.9.11. Manage and unmanage

The `manage` and `unmanage` are two new API extensions that add some new features to the driver. The `manage` action on an existing volume is very similar to a volume creation. It creates a volume entry on OpenStack Block Storage DB, but instead of creating a new volume in the back end, it only adds a 'link' to an existing volume. Volume name, description, `volume_type`, `metadata` and `availability_zone` are supported as in a normal volume creation.

The `unmanage` action on an existing volume removes the volume from the OpenStack Block Storage DB, but keeps the actual volume in the back-end. From an OpenStack Block Storage perspective the volume would be deleted, but it would still exist for outside use.



HOW TO MANAGE:

On the Dashboard:

For NFS:

1. Under the tab System -> Volumes choose the option [+ Manage Volume]
2. Fill the fields Identifier, Host and Volume Type with volume information to be managed:

- **Identifier:** ip:/type/volume_name Example: 172.24.44.34:/silver/volume-test
- **Host:** host@backend-name#pool_name Example: myhost@hnas-nfs#test_silver
- **Volume Name:** volume_name Example: volume-test
- **Volume Type:** choose a type of volume Example: silver

For iSCSI:

1. Under the tab System -> Volumes choose the option [+ Manage Volume]
2. Fill the fields Identifier, Host, Volume Name and Volume Type with volume information to be managed:
 - **Identifier:** filesystem-name/volume-name Example: filesystem-test/volume-test
 - **Host:** host@backend-name#pool_name Example: myhost@hnas-iscsi#test_silver
 - **Volume Name:** volume_name Example: volume-test
 - **Volume Type:** choose a type of volume Example: silver

By CLI:

```
$ cinder --os-volume-api-version 2 manage [--source-name
<source-name>][--id-type <id-type>] [--name <name>][--
description <description>][--volume-type <volume-type>] [--
availability-zone <availability-zone>][--metadata [<key=value>
[<key=value> ...]]][--bootable] <host> [<key=value>
[<key=value> ...]]
```

Example:

For NFS:

```
$ cinder --os-volume-api-version 2 manage --name <volume-test>
--volume-type <silver> --source-name
<172.24.44.34:/silver/volume-test> <myhost@hnas-
nfs#test_silver>
```

For iSCSI:

```
$ cinder --os-volume-api-version 2 manage --name <volume-test>
--volume-type <silver> --source-name <filesystem-test/volume-
test> <myhost@hnas-iscsi#test_silver>
```



HOW TO UNMANAGE:

On Dashboard:

1. Under the tab [System -> Volumes] choose a volume
2. On the volume options, choose [**+Unmanage Volume**]
3. Check the data and confirm.

By CLI:

```
$ cinder --os-volume-api-version 2 unmanage <volume>
```

Example:

```
$ cinder --os-volume-api-version 2 unmanage <voltest>
```

2.1.9.12. Additional notes

- The `get_volume_stats()` function always provides the available capacity based on the combined sum of all the HDPs that are used in these services labels.
- After changing the configuration on the storage, the OpenStack Block Storage driver must be restarted.
- On Red Hat, if the system is configured to use SELinux, you need to set "`virt_use_nfs = on`" for NFS driver work properly.

```
# setsebool -P virt_use_nfs on
```

- It is not possible to manage a volume if there is a slash ('/') or a colon (':') on the volume name.

2.1.10. Hitachi storage volume driver

Hitachi storage volume driver provides iSCSI and Fibre Channel support for Hitachi storages.

2.1.10.1. System requirements

Supported storages:

- Hitachi Virtual Storage Platform G1000 (VSP G1000)
- Hitachi Virtual Storage Platform (VSP)
- Hitachi Unified Storage VM (HUS VM)
- Hitachi Unified Storage 100 Family (HUS 100 Family)

Required software:

- RAID Manager Ver 01-32-03/01 or later for VSP G1000/VSP/HUS VM
- Hitachi Storage Navigator Modular 2 (HSNM2) Ver 27.50 or later for HUS 100 Family

**NOTE**

HSNM2 needs to be installed under `/usr/stonavm`.

Required licenses:

- Hitachi In-System Replication Software for VSP G1000/VSP/HUS VM
- (Mandatory) ShadowImage in-system replication for HUS 100 Family
- (Optional) Copy-on-Write Snapshot for HUS 100 Family

Additionally, the **pexpect** package is required.

2.1.10.2. Supported operations

- Create, delete, attach and detach volumes.
- Create, list and delete volume snapshots.
- Create a volume from a snapshot.
- Copy a volume to an image.
- Copy an image to a volume.
- Clone a volume.
- Extend a volume.
- Get volume statistics.

2.1.10.3. Configuration

Set up Hitachi storage

You need to specify settings as described below. For details about each step, see the user's guide of the storage device. Use a storage administrative software such as Storage Navigator to set up the storage device so that LDEVs and host groups can be created and deleted, and LDEVs can be connected to the server and can be asynchronously copied.

1. Create a Dynamic Provisioning pool.
2. Connect the ports at the storage to the Controller node and Compute nodes.
3. For VSP G1000/VSP/HUS VM, set "port security" to "enable" for the ports at the storage.
4. For HUS 100 Family, set "Host Group security"/"iSCSI target security" to "ON" for the ports at the storage.
5. For the ports at the storage, create host groups (iSCSI targets) whose names begin with HBSD- for the Controller node and each Compute node. Then register a WWN (initiator IQN) for each of the Controller node and Compute nodes.
6. For VSP G1000/VSP/HUS VM, perform the following:

- Create a storage device account belonging to the Administrator User Group. (To use multiple storage devices, create the same account name for all the target storage devices, and specify the same resource group and permissions.)
- Create a command device (In-Band), and set user authentication to ON.
- Register the created command device to the host group for the Controller node.
- To use the Thin Image function, create a pool for Thin Image.

7. For HUS 100 Family, perform the following:

- Use the command **auunitaddauto** to register the unit name and controller of the storage device to HSNM2.
- When connecting via iSCSI, if you are using CHAP certification, specify the same user and password as that used for the storage port.

Set up Hitachi Gigabit Fibre Channel adaptor

Change a parameter of the hfcldd driver and update the initram file if Hitachi Gigabit Fibre Channel adaptor is used.

```
# /opt/hitachi/drivers/hba/hfcmgr -E hfc_rport_lu_scan 1
# dracut -f initramfs-KERNEL_VERSION.img KERNEL_VERSION
# reboot
```

Set up Hitachi storage volume driver

1. Create directory.

```
# mkdir /var/lock/hbsd
# chown cinder:cinder /var/lock/hbsd
```

2. Create "volume type" and "volume key".

This example shows that HUS100_SAMPLE is created as "volume type" and hus100_backend is registered as "volume key".

```
$ cinder type-create HUS100_SAMPLE
$ cinder type-key HUS100_SAMPLE set
volume_backend_name=hus100_backend
```

Specify any identical "volume type" name and "volume key".

To confirm the created "volume type", execute the following command:

```
$ cinder extra-specs-list
```

3. Edit **/etc/cinder/cinder.conf** as follows.

If you use Fibre Channel:

```
volume_driver = cinder.volume.drivers.hitachi.hbsd_fc.HBSDFCDriver
```

-

If you use iSCSI:

```
volume_driver =
cinder.volume.drivers.hitachi.hbsd_iscsi.HBSDISCSIDriver
```

Also, set **volume_backend_name** created by **cinder type-key**

```
volume_backend_name = hus100_backend
```

This table shows configuration options for Hitachi storage volume driver.

Table 2.8. Description of Hitachi storage volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
hitachi_add_chap_user = <i>False</i>	(BoolOpt) Add CHAP user
hitachi_async_copy_check_interval = <i>10</i>	(IntOpt) Interval to check copy asynchronously
hitachi_auth_method = <i>None</i>	(StrOpt) iSCSI authentication method
hitachi_auth_password = <i>HBSD-CHAP-password</i>	(StrOpt) iSCSI authentication password
hitachi_auth_user = <i>HBSD-CHAP-user</i>	(StrOpt) iSCSI authentication username
hitachi_copy_check_interval = <i>3</i>	(IntOpt) Interval to check copy
hitachi_copy_speed = <i>3</i>	(IntOpt) Copy speed of storage system
hitachi_default_copy_method = <i>FULL</i>	(StrOpt) Default copy method of storage system
hitachi_group_range = <i>None</i>	(StrOpt) Range of group number
hitachi_group_request = <i>False</i>	(BoolOpt) Request for creating HostGroup or iSCSI Target
hitachi_horcm_add_conf = <i>True</i>	(BoolOpt) Add to HORCM configuration
hitachi_horcm_numbers = <i>200,201</i>	(StrOpt) Instance numbers for HORCM
hitachi_horcm_password = <i>None</i>	(StrOpt) Password of storage system for HORCM

Configuration option = Default value	Description
hitachi_horcm_resource_lock_time out = 600	(IntOpt) Timeout until a resource lock is released, in seconds. The value must be between 0 and 7200.
hitachi_horcm_user = None	(StrOpt) Username of storage system for HORCM
hitachi_ldev_range = None	(StrOpt) Range of logical device of storage system
hitachi_pool_id = None	(IntOpt) Pool ID of storage system
hitachi_serial_number = None	(StrOpt) Serial number of storage system
hitachi_target_ports = None	(StrOpt) Control port names for HostGroup or iSCSI Target
hitachi_thin_pool_id = None	(IntOpt) Thin pool ID of storage system
hitachi_unit_name = None	(StrOpt) Name of an array unit
hitachi_zoning_request = False	(BoolOpt) Request for FC Zone creating HostGroup

4. Restart Block Storage service.

When the startup is done, "MSGID0003-I: The storage backend can be used." is output into `/var/log/cinder/volume.log` as follows.

```
2014-09-01 10:34:14.169 28734 WARNING cinder.volume.drivers.hitachi.
hbsd_common [req-a0bb70b5-7c3f-422a-a29e-6a55d6508135 None None]
MSGID0003-I: The storage backend can be used. (config_group:
hus100_backend)
```

2.1.11. HPE 3PAR Fibre Channel and iSCSI drivers

The `HPE3PARFCDriver` and `HPE3PARISCSIDriver` drivers, which are based on the Block Storage service (Cinder) plug-in architecture, run volume operations by communicating with the HPE 3PAR storage system over HTTP, HTTPS, and SSH connections. The HTTP and HTTPS communications use `hp3parclient`, which is part of the Python standard library.

For information about how to manage HPE 3PAR storage systems, see the HPE 3PAR user documentation.

2.1.11.1. System requirements

To use the HPE 3PAR drivers, install the following software and components on the HPE 3PAR storage system:

- HPE 3PAR Operating System software version 3.1.3 MU1 or higher.
 - Deduplication provisioning requires SSD disks and HPE 3PAR Operating System software version 3.2.1 MU1 or higher.
 - Enabling Flash Cache Policy requires the following:
 - Array must contain SSD disks.
 - HPE 3PAR Operating System software version 3.2.1 MU2 or higher.
 - python-3parclient version 4.2.0 or newer.
 - Array must have the Adaptive Flash Cache license installed.
 - Flash Cache must be enabled on the array with the CLI command `createflashcache SIZE`, where *SIZE* must be in 16 GB increments. For example, `createflashcache 128g` will create 128 GB of Flash Cache for each node pair in the array.
 - The Dynamic Optimization license is required to support any feature that results in a volume changing provisioning type or CPG. This may apply to the volume `migrate`, `retype`, and `manage` commands.
 - The Virtual Copy License is required to support any feature that involves volume snapshots. This applies to the volume `snapshot - *` commands.
- HPE 3PAR drivers will now check the licenses installed on the array and disable driver capabilities based on available licenses. This will apply to thin provisioning, QoS support and volume replication.
- HPE 3PAR Web Services API Server must be enabled and running
- One Common Provisioning Group (CPG)
- Additionally, you must install the python-3parclient version 4.2.0 or newer from the Python standard library on the system with the enabled Block Storage service volume drivers.

2.1.11.2. Supported operations

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.
- Create a volume from a snapshot.
- Copy an image to a volume.
- Copy a volume to an image.
- Clone a volume.
- Extend a volume.
- Migrate a volume with back-end assistance.
- Retype a volume.

- Manage and unmanage a volume.
- Create, delete, update, snapshot, and clone consistency groups.
- Create and delete consistency group snapshots.
- Create a consistency group from a consistency group snapshot or another group.

Volume type support for both HPE 3PAR drivers includes the ability to set the following capabilities in the OpenStack Block Storage API `cinder.api.contrib.types_extra_specs` volume type extra specs extension module:

- `hpe3par:snap_cpg`
- `hpe3par:provisioning`
- `hpe3par:persona`
- `hpe3par:vvs`
- `hpe3par:flash_cache`

To work with the default filter scheduler, the key values are case sensitive and scoped with `hpe3par:`. For information about how to set the key-value pairs and associate them with a volume type, run the following command:

```
$ cinder help type-key
```



NOTE

Volumes that are cloned only support extra specs keys `cpg`, `snap_cpg`, `provisioning` and `vvs`. The others are ignored. In addition the comments section of the cloned volume in the HPE 3PAR StoreServ storage array is not populated.

If volume types are not used or a particular key is not set for a volume type, the following defaults are used:

- `hpe3par:cpg` - Defaults to the `hpe3par_cpg` setting in the `cinder.conf` file.
- `hpe3par:snap_cpg` - Defaults to the `hpe3par_snap` setting in the `cinder.conf` file. If `hpe3par_snap` is not set, it defaults to the `hpe3par_cpg` setting.
- `hpe3par:provisioning` - Defaults to thin provisioning, the valid values are `thin`, `full`, and `dedup`.
- `hpe3par:persona` - Defaults to the `2 - Generic-ALUA` persona. The valid values are, `1 - Generic`, `2 - Generic-ALUA`, `3 - Generic-legacy`, `4 - HPUX-legacy`, `5 - AIX-legacy`, `6 - EGENERA`, `7 - ONTAP-legacy`, `8 - VMware`, `9 - OpenVMS`, `10 - HPUX`, and `11 - WindowsServer`.
- `hpe3par:flash_cache` - Defaults to `false`, the valid values are `true` and `false`.

QoS support for both HPE 3PAR drivers includes the ability to set the following capabilities in the OpenStack Block Storage API `cinder.api.contrib.qos_specs_manage` qos specs extension module:

- **minBWS**
- **maxBWS**
- **minIOPS**
- **maxIOPS**
- **latency**
- **priority**

The qos keys above no longer require to be scoped but must be created and associated to a volume type. For information about how to set the key-value pairs and associate them with a volume type, run the following commands:

```
$ cinder help qos-create
```

```
$ cinder help qos-key
```

```
$ cinder help qos-associate
```

The following keys require that the HPE 3PAR StoreServ storage array has a Priority Optimization license installed.

- **hpe3par : vvs** - The virtual volume set name that has been predefined by the Administrator with Quality of Service (QoS) rules associated to it. If you specify extra_specs **hpe3par : vvs**, the qos_specs **minIOPS**, **maxIOPS**, **minBWS**, and **maxBWS** settings are ignored.
- **minBWS** - The QoS I/O issue bandwidth minimum goal in MBs. If not set, the I/O issue bandwidth rate has no minimum goal.
- **maxBWS** - The QoS I/O issue bandwidth rate limit in MBs. If not set, the I/O issue bandwidth rate has no limit.
- **minIOPS** - The QoS I/O issue count minimum goal. If not set, the I/O issue count has no minimum goal.
- **maxIOPS** - The QoS I/O issue count rate limit. If not set, the I/O issue count rate has no limit.
- **latency** - The latency goal in milliseconds.
- **priority** - The priority of the QoS rule over other rules. If not set, the priority is normal, valid values are low, normal and high.



NOTE

Since the Icehouse release, minIOPS and maxIOPS must be used together to set I/O limits. Similarly, minBWS and maxBWS must be used together. If only one is set the other will be set to the same value.

The following keys require that the HPE 3PAR StoreServ storage array has an Adaptive Flash Cache license installed.

- **hpe3par:flash_cache** - The flash-cache policy, which can be turned on and off by setting the value to **true** or **false**.

2.1.11.3. Enable the HPE 3PAR Fibre Channel and iSCSI drivers

The **HP3PARFCDriver** and **HP3PARISCSIDriver** are installed with the OpenStack software.

1. Install the **hp3parclient** Python package on the OpenStack Block Storage system.

```
# pip install 'python-3parclient>=4.0,<5.0'
```

2. Verify that the HPE 3PAR Web Services API server is enabled and running on the HPE 3PAR storage system.

- a. Log onto the HP 3PAR storage system with administrator access.

```
$ ssh 3paradm@<HP 3PAR IP Address>
```

- b. View the current state of the Web Services API Server.

```
# showwsapi
-Service- -State- -HTTP_State- HTTP_Port -HTTPS_State- HTTPS_Port
-Version-
Enabled   Active Enabled           8008           Enabled      8080
1.1
```

- c. If the Web Services API Server is disabled, start it.

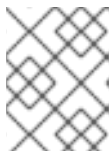
```
# startwsapi
```

3. If the HTTP or HTTPS state is disabled, enable one of them.

```
# setwsapi -http enable
```

or

```
# setwsapi -https enable
```



NOTE

To stop the Web Services API Server, use the **stopwsapi** command. For other options run the **setwsapi -h** command.

4. If you are not using an existing CPG, create a CPG on the HPE 3PAR storage system to be used as the default location for creating volumes.
5. Make the following changes in the **/etc/cinder/cinder.conf** file.

```
## REQUIRED SETTINGS# 3PAR WS API Server URL
hpe3par_api_url=https://10.10.0.141:8080/api/v1

# 3PAR username with the 'edit' role
```

```
hpe3par_username=edit3par

# 3PAR password for the user specified in hpe3par_username
hpe3par_password=3parpass

# 3PAR CPG to use for volume creation
hpe3par_cpg=OpenStackCPG_RAID5_NL

# IP address of SAN controller for SSH access to the array
san_ip=10.10.22.241

# Username for SAN controller for SSH access to the array
san_login=3paradm

# Password for SAN controller for SSH access to the array
san_password=3parpass

# FIBRE CHANNEL(uncomment the next line to enable the FC driver)
#
volume_driver=cinder.volume.drivers.hpe.hpe_3par_fc.HPE3PARFCDriver

# iSCSI (uncomment the next line to enable the iSCSI driver and
# hpe3par_iscsi_ips or iscsi_ip_address)
#volume_driver=cinder.volume.drivers.hpe.hpe_3par_iscsi.HPE3PARISCSI
Driver

# iSCSI multiple port configuration
# hpe3par_iscsi_ips=10.10.220.253:3261,10.10.222.234

# Still available for single port iSCSI configuration
#iscsi_ip_address=10.10.220.253

# Enable HTTP debugging to 3PAR
hpe3par_debug=False

# Enable CHAP authentication for iSCSI connections.
hpe3par_iscsi_chap_enabled=false

# The CPG to use for Snapshots for volumes. If empty hpe3par_cpg
will be
# used.
hpe3par_snap_cpg=OpenStackSNAP_CPG

# Time in hours to retain a snapshot. You can't delete it before
this
# expires.
hpe3par_snapshot_retention=48

# Time in hours when a snapshot expires and is deleted. This must be
# larger than retention.
hpe3par_snapshot_expiration=72

# The ratio of oversubscription when thin provisioned volumes are
# involved. Default ratio is 20.0, this means that a provisioned
# capacity can be 20 times of the total physical capacity.
```



```
max_over_subscription_ratio=20.0
```

```
# This flag represents the percentage of reserved back-end capacity.
reserved_percentage=15
```



NOTE

You can enable only one driver on each cinder instance unless you enable multiple back-end support.



NOTE

You can configure one or more iSCSI addresses by using the **hpe3par_iscsi_ips** option. When you configure multiple addresses, the driver selects the iSCSI port with the fewest active volumes at attach time. The IP address might include an IP port by using a colon (:) to separate the address from port. If you do not define an IP port, the default port 3260 is used. Separate IP addresses with a comma (,). The **iscsi_ip_address/iscsi_port** options might be used as an alternative to **hpe3par_iscsi_ips** for single port iSCSI configuration.

6. Save the changes to the **cinder.conf** file and restart the **cinder-volume** service.

The HPE 3PAR Fibre Channel and iSCSI drivers are now enabled on your OpenStack system. If you experience problems, review the Block Storage service log files for errors.

2.1.12. Huawei storage driver

The Huawei driver supports the iSCSI and Fibre Channel connections and enables OceanStor T series V200R002, OceanStor 18000 series V100R001 and OceanStor V3 series V300R002 storage to provide block storage services for OpenStack.

Supported operations

- Create, delete, expand, attach, and detach volumes.
- Create and delete a snapshot.
- Copy an image to a volume.
- Copy a volume to an image.
- Create a volume from a snapshot.
- Clone a volume.

Configure block storage nodes

1. Modify the **cinder.conf** configuration file and add **volume_driver** and **cinder_huawei_conf_file** items.
 - Example for configuring a storage system:

```

volume_driver = cinder.volume.drivers.huawei.HuaweiVolumeDriver
cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf.xml

```

- Example for configuring multiple storage systems:

```

enabled_backends = t_iscsi, 18000_iscsi
[t_iscsi]
volume_driver = cinder.volume.drivers.huawei.HuaweiVolumeDriver
cinder_huawei_conf_file =
/etc/cinder/cinder_huawei_conf_t_iscsi.xml
volume_backend_name = HuaweiITISCSIDriver

[18000_iscsi]
volume_driver = cinder.volume.drivers.huawei.HuaweiVolumeDriver
cinder_huawei_conf_file =
/etc/cinder/cinder_huawei_conf_18000_iscsi.xml
volume_backend_name = Huawei18000ISCSIDriver

```

2. In `/etc/cinder`, create a driver configuration file. The driver configuration file name must be the same as the `cinder_huawei_conf_file` item in the `cinder_conf` configuration file.

3. Configure product and protocol.

Product and Protocol indicate the storage system type and link type respectively. For the OceanStor 18000 series V100R001 storage systems, the driver configuration file is as follows:

```

<?xml version='1.0' encoding='UTF-8'?>
<config>
  <Storage>
    <Product>18000</Product>
    <Protocol>iSCSI</Protocol>
    <RestURL>https://x.x.x.x/deviceManager/rest/</RestURL>
    <UserName>xxxxxxxx</UserName>
    <UserPassword>xxxxxxxx</UserPassword>
  </Storage>
  <LUN>
    <LUNType>Thick</LUNType>
    <WriteType>1</WriteType>
    <MirrorSwitch>0</MirrorSwitch>
    <LUNcopyWaitInterval>5</LUNcopyWaitInterval>
    <Timeout>432000</Timeout>
    <StoragePool>xxxxxxxx</StoragePool>
  </LUN>
  <iSCSI>
    <DefaultTargetIP>x.x.x.x</DefaultTargetIP>
    <Initiator Name="xxxxxxxx" TargetIP="x.x.x.x"/>
    <Initiator Name="xxxxxxxx" TargetIP="x.x.x.x"/>
  </iSCSI>
  <Host OSType="Linux" HostIP="x.x.x.x, x.x.x.x"/>
</config>

```

NOTE**Note for fibre channel driver configuration**

- In the configuration files of OceanStor T series V200R002 and OceanStor V3 V300R002, parameter configurations are the same with the exception of the RestURL parameter. The following describes how to configure the RestURL parameter:

```
<RestURL>https://x.x.x.x:8088/deviceManager/rest/</RestURL>
```

- For a Fibre Channel driver, you do not need to configure an iSCSI target IP address. Delete the iSCSI configuration from the preceding examples.

```
<iSCSI>
  <DefaultTargetIP>x.x.x.x</DefaultTargetIP>
  <Initiator Name="xxxxxxxxx"
TargetIP="x.x.x.x"/>
  <Initiator Name="xxxxxxxxx"
TargetIP="x.x.x.x"/>
</iSCSI>
```

This table describes the Huawei storage driver configuration options:

Table 2.9. Huawei storage driver configuration options

Property	Type	Default	Description
Product	Mandatory	-	Type of a storage product. Valid values are T , TV3 , or 18000 .
Protocol	Mandatory	-	Type of a protocol. Valid values are iSCSI or FC .
RestURL	Mandatory	-	Access address of the Rest port (required only for the 18000)
UserName	Mandatory	-	User name of an administrator
UserPassword	Mandatory	-	Password of an administrator

Property	Type	Default	Description
LUNType	Optional	Thin	Type of a created LUN. Valid values are Thick or Thin .
StripUnitSize	Optional	64	Stripe depth of a created LUN. The value is expressed in KB. This flag is not valid for a thin LUN.
WriteType	Optional	1	Cache write method. The method can be write back, write through, or Required write back. The default value is 1 , indicating write back.
MirrorSwitch	Optional	1	Cache mirroring policy. The default value is 1 , indicating that a mirroring policy is used.
Prefetch Type	Optional	3	Cache prefetch strategy. The strategy can be constant prefetch, variable prefetch, or intelligent prefetch. Default value is 3 , which indicates intelligent prefetch and is not required for the OceanStor 18000 series.
Prefetch Value	Optional	0	Cache prefetch value.

Property	Type	Default	Description
LUNcopyWaitInterval	Optional	5	After LUN copy is enabled, the plug-in frequently queries the copy progress. You can set a value to specify the query interval.
Timeout	Optional	432,000	Timeout period for waiting LUN copy of an array to complete.
StoragePool	Mandatory	-	Name of a storage pool that you want to use.
DefaultTargetIP	Optional	-	Default IP address of the iSCSI port provided for compute nodes.
InitiatorName	Optional	-	Name of a compute node initiator.
InitiatorTargetIP	Optional	-	IP address of the iSCSI port provided for compute nodes.
OSType	Optional	Linux	The OS type for a compute node.
HostIP	Optional	-	The IPs for compute nodes.



NOTE FOR THE CONFIGURATION

1. You can configure one iSCSI target port for each or all compute nodes. The driver checks whether a target port IP address is configured for the current compute node. If not, select **DefaultTargetIP**.
2. Only one storage pool can be configured.
3. For details about LUN configuration information, see the **show lun general** command in the command-line interface (CLI) documentation or run the **help -c show lun general** on the storage system CLI.
4. After the driver is loaded, the storage system obtains any modification of the driver configuration file in real time and you do not need to restart the **cinder-volume** service.

4. Restart the Cinder service.

2.1.13. IBM Storwize family and SVC volume driver

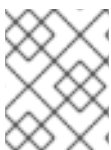
The volume management driver for Storwize family and SAN Volume Controller (SVC) provides OpenStack Compute instances with access to IBM Storwize family or SVC storage systems.

2.1.13.1. Configure the Storwize family and SVC system

Network configuration

The Storwize family or SVC system must be configured for iSCSI, Fibre Channel, or both.

If using iSCSI, each Storwize family or SVC node should have at least one iSCSI IP address. The IBM Storwize/SVC driver uses an iSCSI IP address associated with the volume's preferred node (if available) to attach the volume to the instance, otherwise it uses the first available iSCSI IP address of the system. The driver obtains the iSCSI IP address directly from the storage system; you do not need to provide these iSCSI IP addresses directly to the driver.



NOTE

If using iSCSI, ensure that the compute nodes have iSCSI network access to the Storwize family or SVC system.



NOTE

OpenStack Nova's Grizzly version supports iSCSI multipath. Once this is configured on the Nova host (outside the scope of this documentation), multipath is enabled.

If using Fibre Channel (FC), each Storwize family or SVC node should have at least one WWPN port configured. If the **storwize_svc_multipath_enabled** flag is set to True in the Cinder configuration file, the driver uses all available WWPNs to attach the volume to the instance (details about the configuration flags appear in the [next section](#)). If the flag is not set, the driver uses the WWPN associated with the volume's preferred node (if available), otherwise it uses the first available WWPN of the system. The driver obtains the WWPNs directly from the storage system; you do not need to provide these WWPNs directly to the driver.

**NOTE**

If using FC, ensure that the compute nodes have FC connectivity to the Storwize family or SVC system.

iSCSI CHAP authentication

If using iSCSI for data access and the `storwize_svc_iscsi_chap_enabled` is set to `True`, the driver will associate randomly-generated CHAP secrets with all hosts on the Storwize family system. OpenStack compute nodes use these secrets when creating iSCSI connections.

**NOTE**

CHAP secrets are added to existing hosts as well as newly-created ones. If the CHAP option is enabled, hosts will not be able to access the storage without the generated secrets.

**NOTE**

Not all OpenStack Compute drivers support CHAP authentication. Check compatibility before using.

**NOTE**

CHAP secrets are passed from OpenStack Block Storage to Compute in clear text. This communication should be secured to ensure that CHAP secrets are not discovered.

Configure storage pools

Each instance of the IBM Storwize/SVC driver allocates all volumes in a single pool. The pool should be created in advance and be provided to the driver using the `storwize_svc_volpool_name` configuration flag. Details about the configuration flags and how to provide the flags to the driver appear in the [next section](#).

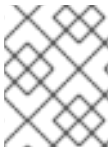
Configure user authentication for the driver

The driver requires access to the Storwize family or SVC system management interface. The driver communicates with the management using SSH. The driver should be provided with the Storwize family or SVC management IP using the `san_ip` flag, and the management port should be provided by the `san_ssh_port` flag. By default, the port value is configured to be port 22 (SSH).

**NOTE**

Make sure the compute node running the `cinder-volume` management driver has SSH network access to the storage system.

To allow the driver to communicate with the Storwize family or SVC system, you must provide the driver with a user on the storage system. The driver has two authentication methods: password-based authentication and SSH key pair authentication. The user should have an Administrator role. It is suggested to create a new user for the management driver. Consult your storage and security administrator regarding the preferred authentication method and how passwords or SSH keys should be stored in a secure manner.

**NOTE**

When creating a new user on the Storwize or SVC system, make sure the user belongs to the Administrator group or to another group that has an Administrator role.

If using password authentication, assign a password to the user on the Storwize or SVC system. The driver configuration flags for the user and password are `san_login` and `san_password`, respectively.

If you are using the SSH key pair authentication, create SSH private and public keys using the instructions below or by any other method. Associate the public key with the user by uploading the public key: select the "choose file" option in the Storwize family or SVC management GUI under "SSH public key". Alternatively, you may associate the SSH public key using the command line interface; details can be found in the Storwize and SVC documentation. The private key should be provided to the driver using the `san_private_key` configuration flag.

Create a SSH key pair with OpenSSH

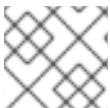
You can create an SSH key pair using OpenSSH, by running:

```
$ ssh-keygen -t rsa
```

The command prompts for a file to save the key pair. For example, if you select 'key' as the filename, two files are created: `key` and `key.pub`. The `key` file holds the private SSH key and `key.pub` holds the public SSH key.

The command also prompts for a pass phrase, which should be empty.

The private key file should be provided to the driver using the `san_private_key` configuration flag. The public key should be uploaded to the Storwize family or SVC system using the storage management GUI or command line interface.

**NOTE**

Ensure that Cinder has read permissions on the private key file.

2.1.13.2. Configure the Storwize family and SVC driver**Enable the Storwize family and SVC driver**

Set the volume driver to the Storwize family and SVC driver by setting the `volume_driver` option in `cinder.conf` as follows:

```
volume_driver = cinder.volume.drivers.ibm.storwize_svc.StorwizeSVCDriver
```

Storwize family and SVC driver options in cinder.conf

The following options specify default values for all volumes. Some can be over-ridden using volume types, which are described below.

Table 2.10. List of configuration flags for Storwize storage and SVC driver

Flag name	Type	Default	Description
san_ip	Required		Management IP or host name
san_ssh_port	Optional	22	Management port
san_login	Required		Management login username
san_password	Required ^[a]		Management login password
san_private_key	Required ^[a]		Management login SSH private key
storwize_svc_volpool_name	Required		Default pool name for volumes
storwize_svc_vol_rsize	Optional	2	Initial physical allocation (percentage) ^[b]
storwize_svc_vol_warning	Optional	0 (disabled)	Space allocation warning threshold (percentage) ^[b]
storwize_svc_vol_autoexpand	Optional	True	Enable or disable volume auto expand ^[c]
storwize_svc_vol_grainsize	Optional	256	Volume grain size ^[b] in KB
storwize_svc_vol_compression	Optional	False	Enable or disable Real-time Compression ^[d]
storwize_svc_vol_easytier	Optional	True	Enable or disable Easy Tier ^[e]
storwize_svc_vol_iogrp	Optional	0	The I/O group in which to allocate vdisks
storwize_svc_flashcopy_timeout	Optional	120	FlashCopy timeout threshold ^[f] (seconds)
storwize_svc_connection_protocol	Optional	iSCSI	Connection protocol to use (currently supports 'iSCSI' or 'FC')
storwize_svc_iscsi_chap_enabled	Optional	True	Configure CHAP authentication for iSCSI connections

Flag name	Type	Default	Description
storwize_svc_multipath_enabled	Optional	False	Enable multipath for FC connections [g]
storwize_svc_multihost_enabled	Optional	True	Enable mapping vdisks to multiple hosts [h]
storwize_svc_vol_nofmtdisk	Optional	False	Enable or disable fast format [i]

[a] The authentication requires either a password (**san_password**) or SSH private key (**san_private_key**). One must be specified. If both are specified, the driver uses only the SSH private key.

[b] The driver creates thin-provisioned volumes by default. The **storwize_svc_vol_rsize** flag defines the initial physical allocation percentage for thin-provisioned volumes, or if set to **-1**, the driver creates full allocated volumes. More details about the available options are available in the Storwize family and SVC documentation.

[c] Defines whether thin-provisioned volumes can be auto expanded by the storage system, a value of **True** means that auto expansion is enabled, a value of **False** disables auto expansion. Details about this option can be found in the **autoexpand** flag of the Storwize family and SVC command line interface **mkvdisk** command.

[d] Defines whether Real-time Compression is used for the volumes created with OpenStack. Details on Real-time Compression can be found in the Storwize family and SVC documentation. The Storwize or SVC system must have compression enabled for this feature to work.

[e] Defines whether Easy Tier is used for the volumes created with OpenStack. Details on EasyTier can be found in the Storwize family and SVC documentation. The Storwize or SVC system must have Easy Tier enabled for this feature to work.

[f] The driver wait timeout threshold when creating an OpenStack snapshot. This is actually the maximum amount of time that the driver waits for the Storwize family or SVC system to prepare a new FlashCopy mapping. The driver accepts a maximum wait time of 600 seconds (10 minutes).

[g] Multipath for iSCSI connections requires no storage-side configuration and is enabled if the compute host has multipath configured.

[h] This option allows the driver to map a vdisk to more than one host at a time. This scenario occurs during migration of a virtual machine with an attached volume; the volume is simultaneously mapped to both the source and destination compute hosts. If your deployment does not require attaching vdisks to multiple hosts, setting this flag to False will provide added safety.

[i] Defines whether or not the fast formatting of thick-provisioned volumes is disabled at creation. The default value is **False** and a value of **True** means that fast format is disabled. Details about this option can be found in the **nofmtdisk** flag of the Storwize family and SVC command line interface **mkvdisk** command.

Table 2.11. Description of IBM Storwize driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
storwize_svc_allow_tenant_qos = <i>False</i>	(BoolOpt) Allow tenants to specify QOS on create
storwize_svc_connection_protocol = <i>iSCSI</i>	(StrOpt) Connection protocol (iSCSI/FC)

Configuration option = Default value	Description
storwize_svc_flashcopy_timeout = 120	(IntOpt) Maximum number of seconds to wait for FlashCopy to be prepared.
storwize_svc_iscsi_chap_enabled = <i>True</i>	(BoolOpt) Configure CHAP authentication for iSCSI connections (Default: Enabled)
storwize_svc_multihostmap_enabled = <i>True</i>	(BoolOpt) Allows vdisk to multi host mapping
storwize_svc_multipath_enabled = <i>False</i>	(BoolOpt) Connect with multipath (FC only; iSCSI multipath is controlled by Nova)
storwize_svc_npiv_compatibility_mode = <i>True</i>	(BoolOpt) Indicate whether svc driver is compatible for NPIV setup. If it is compatible, it will allow no wwpns being returned on get_conn_fc_wwpns during initialize_connection. It should always be set to True. It will be deprecated and removed in M release.
storwize_svc_stretched_cluster_partner = <i>None</i>	(StrOpt) If operating in stretched cluster mode, specify the name of the pool in which mirrored copies are stored.Example: "pool2"
storwize_svc_vol_autoexpand = <i>True</i>	(BoolOpt) Storage system autoexpand parameter for volumes (True/False)
storwize_svc_vol_compression = <i>False</i>	(BoolOpt) Storage system compression option for volumes
storwize_svc_vol_easytier = <i>True</i>	(BoolOpt) Enable Easy Tier for volumes
storwize_svc_vol_grainsize = 256	(IntOpt) Storage system grain size parameter for volumes (32/64/128/256)
storwize_svc_vol_iogrp = 0	(IntOpt) The I/O group in which to allocate volumes
storwize_svc_vol_rsize = 2	(IntOpt) Storage system space-efficiency parameter for volumes (percentage)
storwize_svc_vol_warning = 0	(IntOpt) Storage system threshold for volume capacity warnings (percentage)
storwize_svc_volpool_name = <i>volpool</i>	(StrOpt) Storage system storage pool for volumes

Placement with volume types

The IBM Storwize/SVC driver exposes capabilities that can be added to the **extra_specs** of volume types, and used by the filter scheduler to determine placement of new volumes. Make sure to prefix

these keys with **capabilities:** to indicate that the scheduler should use them. The following **extra specs** are supported:

- **capabilities:volume_back-end_name** - Specify a specific back-end where the volume should be created. The back-end name is a concatenation of the name of the IBM Storwize/SVC storage system as shown in **lssystem**, an underscore, and the name of the pool (mdisk group). For example:

```
capabilities:volume_back-end_name=myV7000_openstackpool
```

- **capabilities:compression_support** - Specify a back-end according to compression support. A value of **True** should be used to request a back-end that supports compression, and a value of **False** will request a back-end that does not support compression. If you do not have constraints on compression support, do not set this key. Note that specifying **True** does not enable compression; it only requests that the volume be placed on a back-end that supports compression. Example syntax:

```
capabilities:compression_support='<is> True'
```

- **capabilities:easytier_support** - Similar semantics as the **compression_support** key, but for specifying according to support of the Easy Tier feature. Example syntax:

```
capabilities:easytier_support='<is> True'
```

- **capabilities:storage_protocol** - Specifies the connection protocol used to attach volumes of this type to instances. Legal values are **iSCSI** and **FC**. This **extra specs** value is used for both placement and setting the protocol used for this volume. In the example syntax, note **<in>** is used as opposed to **<is>** used in the previous examples.

```
capabilities:storage_protocol='<in> FC'
```

Configure per-volume creation options

Volume types can also be used to pass options to the IBM Storwize/SVC driver, which over-ride the default values set in the configuration file. Contrary to the previous examples where the "capabilities" scope was used to pass parameters to the Cinder scheduler, options can be passed to the IBM Storwize/SVC driver with the "drivers" scope.

The following **extra specs** keys are supported by the IBM Storwize/SVC driver:

- **rsiz**
- **warning**
- **autoexpand**
- **grainsize**
- **compression**
- **easytier**
- **multipath**

- `iogrp`

These keys have the same semantics as their counterparts in the configuration file. They are set similarly; for example, `rsize=2` or `compression=False`.

Example: Volume types

In the following example, we create a volume type to specify a controller that supports iSCSI and compression, to use iSCSI when attaching the volume, and to enable compression:

```
$ cinder type-create compressed
$ cinder type-key compressed set capabilities:storage_protocol='<in>
iSCSI' capabilities:compression_support='<is> True'
drivers:compression=True
```

We can then create a 50GB volume using this type:

```
$ cinder create --display-name "compressed volume" --volume-type
compressed 50
```

Volume types can be used, for example, to provide users with different

- performance levels (such as, allocating entirely on an HDD tier, using Easy Tier for an HDD-SDD mix, or allocating entirely on an SSD tier)
- resiliency levels (such as, allocating volumes in pools with different RAID levels)
- features (such as, enabling/disabling Real-time Compression)

QOS

The Storwize driver provides QOS support for storage volumes by controlling the I/O amount. QOS is enabled by editing the `etc/cinder/cinder.conf` file and setting the `storwize_svc_allow_tenant_qos` to `True`.

There are three ways to set the Storwize `IOThrottling` parameter for storage volumes:

- Add the `qos:IOThrottling` key into a QOS specification and associate it with a volume type.
- Add the `qos:IOThrottling` key into an extra specification with a volume type.
- Add the `qos:IOThrottling` key to the storage volume metadata.



NOTE

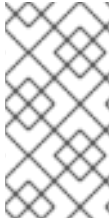
If you are changing a volume type with QOS to a new volume type without QOS, the QOS configuration settings will be removed.

2.1.13.3. Operational notes for the Storwize family and SVC driver

Migrate volumes

In the context of OpenStack Block Storage's volume migration feature, the IBM Storwize/SVC driver enables the storage's virtualization technology. When migrating a volume from one pool to another, the

volume will appear in the destination pool almost immediately, while the storage moves the data in the background.



NOTE

To enable this feature, both pools involved in a given volume migration must have the same values for `extent_size`. If the pools have different values for `extent_size`, the data will still be moved directly between the pools (not host-side copy), but the operation will be synchronous.

Extend volumes

The IBM Storwize/SVC driver allows for extending a volume's size, but only for volumes without snapshots.

Snapshots and clones

Snapshots are implemented using FlashCopy with no background copy (space-efficient). Volume clones (volumes created from existing volumes) are implemented with FlashCopy, but with background copy enabled. This means that volume clones are independent, full copies. While this background copy is taking place, attempting to delete or extend the source volume will result in that operation waiting for the copy to complete.

Volume retype

The IBM Storwize/SVC driver enables you to modify volume types. When you modify volume types, you can also change these extra specs properties:

- `rsiz`
- `warning`
- `autoexpand`
- `grainsize`
- `compression`
- `easytier`
- `iogrp`
- `nofmtdisk`



NOTE

When you change the `rsiz`, `grainsize` or `compression` properties, volume copies are asynchronously synchronized on the array.



NOTE

To change the `iogrp` property, IBM Storwize/SVC firmware version 6.4.0 or later is required.

2.1.14. IBM XIV and DS8000 volume driver

The IBM Storage Driver for OpenStack is a Block Storage driver that supports IBM XIV and IBM DS8000 storage systems over Fiber channel and iSCSI.

Set the following in your `cinder.conf`, and use the following options to configure it.

```
volume_driver = cinder.volume.drivers.xiv_ds8k.XIVDS8KDriver
```

Table 2.12. Description of IBM XIV and DS8000 volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>san_clustername =</code>	(StrOpt) Cluster name to use for creating volumes
<code>san_ip =</code>	(StrOpt) IP address of SAN controller
<code>san_login = admin</code>	(StrOpt) Username for SAN controller
<code>san_password =</code>	(StrOpt) Password for SAN controller
<code>xiv_chap = disabled</code>	(StrOpt) CHAP authentication mode, effective only for iscsi (disabled enabled)
<code>xiv_ds8k_connection_type = iscsi</code>	(StrOpt) Connection type to the IBM Storage Array
<code>xiv_ds8k_proxy =</code> <code>xiv_ds8k_openstack.nova_proxy.XIVDS8KNovaProxy</code>	(StrOpt) Proxy driver that connects to the IBM Storage Array



NOTE

To use the IBM Storage Driver for OpenStack you must download and install the package available at: http://www.ibm.com/support/fixcentral/swg/selectFixes?parent=Enterprise%2BStorage%2BServers&product=ibm/Storage_Disk/XIV+Storage+Sy

For full documentation refer to IBM's online documentation available at <http://pic.dhe.ibm.com/infocenter/strhosts/ic/topic/com.ibm.help.strghosts.doc/nova-homepage.html>.

2.1.15. LVM

The default volume back-end uses local volumes managed by LVM.

This driver supports different transport protocols to attach volumes, currently iSCSI and iSER.

**NOTE**

The Block Storage iSCSI LVM driver has significant performance issues. In production environments, with high I/O activity, there are many potential issues which could affect performance or data integrity.

Red Hat strongly recommends using a certified Block Storage plug-in provider for storage in a production environment. The software iSCSI LVM driver should be used and is only supported for single node evaluations and proof of concept environments.

Set the following in your `cinder.conf` configuration file, and use the following options to configure for iSCSI transport:

```
volume_driver = cinder.volume.drivers.lvm.LVMVolumeDriver
iscsi_protocol = iscsi
```

Use the following options to configure for the iSER transport:

```
volume_driver = cinder.volume.drivers.lvm.LVMVolumeDriver
iscsi_protocol = iser
```

Table 2.13. Description of LVM configuration options

Configuration option = Default value	Description
[DEFAULT]	
lvm_conf_file = <i>/etc/cinder/lvm.conf</i>	(StrOpt) LVM conf file to use for the LVM driver in Cinder; this setting is ignored if the specified file does not exist (You can also specify 'None' to not use a conf file even if one exists).
lvm_mirrors = 0	(IntOpt) If >0, create LVs with multiple mirrors. Note that this requires lvm_mirrors + 2 PVs with available space
lvm_type = <i>default</i>	(StrOpt) Type of LVM volumes to deploy; (default, thin, or auto). Auto defaults to thin if thin is supported.
volume_group = <i>cinder-volumes</i>	(StrOpt) Name for the VG that will contain exported volumes

2.1.16. NetApp unified driver

The NetApp unified driver is a block storage driver that supports multiple storage families and protocols. A storage family corresponds to storage systems built on different NetApp technologies such as clustered Data ONTAP, Data ONTAP operating in 7-Mode, and E-Series. The storage protocol refers to the protocol used to initiate data storage and access operations on those storage systems like iSCSI and NFS. The NetApp unified driver can be configured to provision and manage OpenStack

volumes on a given storage family using a specified storage protocol. The OpenStack volumes can then be used for accessing and storing data using the storage protocol on the storage family system. The NetApp unified driver is an extensible interface that can support new storage families and protocols.



NOTE

With the Juno release of OpenStack, OpenStack Block Storage has introduced the concept of "storage pools", in which a single OpenStack Block Storage back end may present one or more logical storage resource pools from which OpenStack Block Storage will select as a storage location when provisioning volumes.

In releases prior to Juno, the NetApp unified driver contained some "scheduling" logic that determined which NetApp storage container (namely, a FlexVol volume for Data ONTAP, or a dynamic disk pool for E-Series) that a new OpenStack Block Storage volume would be placed into.

With the introduction of pools, all scheduling logic is performed completely within the OpenStack Block Storage scheduler, as each NetApp storage container is directly exposed to the OpenStack Block Storage scheduler as a storage pool; whereas previously, the NetApp unified driver presented an aggregated view to the scheduler and made a final placement decision as to which NetApp storage container the OpenStack Block Storage volume would be provisioned into.

2.1.16.1. NetApp clustered Data ONTAP storage family

The NetApp clustered Data ONTAP storage family represents a configuration group which provides OpenStack compute instances access to clustered Data ONTAP storage systems. At present it can be configured in OpenStack Block Storage to work with iSCSI and NFS storage protocols.

2.1.16.1.1. NetApp iSCSI configuration for clustered Data ONTAP

The NetApp iSCSI configuration for clustered Data ONTAP is an interface from OpenStack to clustered Data ONTAP storage systems for provisioning and managing the SAN block storage entity; that is, a NetApp LUN which can be accessed using the iSCSI protocol.

The iSCSI configuration for clustered Data ONTAP is a direct interface from OpenStack Block Storage to the clustered Data ONTAP instance and as such does not require additional management software to achieve the desired functionality. It uses NetApp APIs to interact with the clustered Data ONTAP instance.

Configuration options for clustered Data ONTAP family with iSCSI protocol

Configure the volume driver, storage family and storage protocol to the NetApp unified driver, clustered Data ONTAP, and iSCSI respectively by setting the `volume_driver`, `netapp_storage_family` and `netapp_storage_protocol` options in `cinder.conf` as follows:

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_cluster
netapp_storage_protocol = iscsi
netapp_vserver = openstack-vserver
netapp_server_hostname = myhostname
netapp_server_port = port
netapp_login = username
netapp_password = password
```

**NOTE**

To use the iSCSI protocol, you must override the default value of `netapp_storage_protocol` with `iscsi`.

Table 2.14. Description of NetApp cDOT iSCSI driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>netapp_login = None</code>	(StrOpt) Administrative user account name used to access the storage system or proxy server.
<code>netapp_lun_ostype = None</code>	(StrOpt) This option defines the type of operating system that will access a LUN exported from Data ONTAP; it is assigned to the LUN at the time it is created.
<code>netapp_lun_space_reservation = enabled</code>	(StrOpt) This option determines if storage space is reserved for LUN allocation. If enabled, LUNs are thick provisioned. If space reservation is disabled, storage space is allocated on demand.
<code>netapp_partner_backend_name = None</code>	(StrOpt) The name of the config.conf stanza for a Data ONTAP (7-mode) HA partner. This option is only used by the driver when connecting to an instance with a storage family of Data ONTAP operating in 7-Mode, and it is required if the storage protocol selected is FC.
<code>netapp_password = None</code>	(StrOpt) Password for the administrative user account specified in the <code>netapp_login</code> option.
<code>netapp_pool_name_search_pattern = (.+)</code>	(StrOpt) This option is used to restrict provisioning to the specified pools. Specify the value of this option to be a regular expression which will be applied to the names of objects from the storage backend which represent pools in Cinder. This option is only utilized when the storage protocol is configured to use iSCSI or FC.
<code>netapp_server_hostname = None</code>	(StrOpt) The hostname (or IP address) for the storage system or proxy server.
<code>netapp_server_port = None</code>	(IntOpt) The TCP port to use for communication with the storage system or proxy server. If not specified, Data ONTAP drivers will use 80 for HTTP and 443 for HTTPS; E-Series will use 8080 for HTTP and 8443 for HTTPS.

Configuration option = Default value	Description
netapp_size_multiplier = 1.2	(FloatOpt) The quantity to be multiplied by the requested volume size to ensure enough space is available on the virtual storage server (Vserver) to fulfill the volume creation request. Note: this option is deprecated and will be removed in favor of "reserved_percentage" in the Mitaka release.
netapp_storage_family = <i>ontap_cluster</i>	(StrOpt) The storage family type used on the storage system; valid values are <i>ontap_7mode</i> for using Data ONTAP operating in 7-Mode, <i>ontap_cluster</i> for using clustered Data ONTAP, or <i>eseries</i> for using E-Series.
netapp_storage_protocol = <i>None</i>	(StrOpt) The storage protocol to be used on the data path with the storage system.
netapp_transport_type = <i>http</i>	(StrOpt) The transport protocol used when communicating with the storage system or proxy server.
netapp_vserver = <i>None</i>	(StrOpt) This option specifies the virtual storage server (Vserver) name on the storage cluster on which provisioning of block storage volumes should occur.

**NOTE**

If you specify an account in the **netapp_login** that only has virtual storage server (Vserver) administration privileges (rather than cluster-wide administration privileges), some advanced features of the NetApp unified driver will not work and you may see warnings in the OpenStack Block Storage logs.

TIP

For more information on these options and other deployment and operational scenarios, visit the [NetApp OpenStack Deployment and Operations Guide](#).

2.1.16.1.2. NetApp NFS configuration for clustered Data ONTAP

The NetApp NFS configuration for clustered Data ONTAP is an interface from OpenStack to a clustered Data ONTAP system for provisioning and managing OpenStack volumes on NFS exports provided by the clustered Data ONTAP system that are accessed using the NFS protocol.

The NFS configuration for clustered Data ONTAP is a direct interface from OpenStack Block Storage to the clustered Data ONTAP instance and as such does not require any additional management software to achieve the desired functionality. It uses NetApp APIs to interact with the clustered Data ONTAP instance.

Configuration options for the clustered Data ONTAP family with NFS protocol

Configure the volume driver, storage family, and storage protocol to NetApp unified driver, clustered Data ONTAP, and NFS respectively by setting the `volume_driver`, `netapp_storage_family` and `netapp_storage_protocol` options in `cinder.conf` as follows:

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_cluster
netapp_storage_protocol = nfs
netapp_vserver = openstack-vserver
netapp_server_hostname = myhostname
netapp_server_port = port
netapp_login = username
netapp_password = password
nfs_shares_config = /etc/cinder/nfs_shares
```

Table 2.15. Description of NetApp cDOT NFS driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
expiry_thres_minutes = 720	(IntOpt) This option specifies the threshold for last access time for images in the NFS image cache. When a cache cleaning cycle begins, images in the cache that have not been accessed in the last M minutes, where M is the value of this parameter, will be deleted from the cache to create free space on the NFS share.
netapp_copyoffload_tool_path = None	(StrOpt) This option specifies the path of the NetApp copy offload tool binary. Ensure that the binary has execute permissions set which allow the effective user of the cinder-volume process to execute the file.
netapp_host_type = None	(StrOpt) This option defines the type of operating system for all initiators that can access a LUN. This information is used when mapping LUNs to individual hosts or groups of hosts.
netapp_host_type = None	(StrOpt) This option defines the type of operating system for all initiators that can access a LUN. This information is used when mapping LUNs to individual hosts or groups of hosts.
netapp_login = None	(StrOpt) Administrative user account name used to access the storage system or proxy server.
netapp_lun_ostype = None	(StrOpt) This option defines the type of operating system that will access a LUN exported from Data ONTAP; it is assigned to the LUN at the time it is created.

Configuration option = Default value	Description
netapp_partner_backend_name = <i>None</i>	(StrOpt) The name of the config.conf stanza for a Data ONTAP (7-mode) HA partner. This option is only used by the driver when connecting to an instance with a storage family of Data ONTAP operating in 7-Mode, and it is required if the storage protocol selected is FC.
netapp_password = <i>None</i>	(StrOpt) Password for the administrative user account specified in the netapp_login option.
netapp_pool_name_search_pattern = <i>(.+)</i>	(StrOpt) This option is used to restrict provisioning to the specified pools. Specify the value of this option to be a regular expression which will be applied to the names of objects from the storage backend which represent pools in Cinder. This option is only utilized when the storage protocol is configured to use iSCSI or FC.
netapp_server_hostname = <i>None</i>	(StrOpt) The hostname (or IP address) for the storage system or proxy server.
netapp_server_port = <i>None</i>	(IntOpt) The TCP port to use for communication with the storage system or proxy server. If not specified, Data ONTAP drivers will use 80 for HTTP and 443 for HTTPS; E-Series will use 8080 for HTTP and 8443 for HTTPS.
netapp_storage_family = <i>ontap_cluster</i>	(StrOpt) The storage family type used on the storage system; valid values are <i>ontap_7mode</i> for using Data ONTAP operating in 7-Mode, <i>ontap_cluster</i> for using clustered Data ONTAP, or <i>eseries</i> for using E-Series.
netapp_storage_protocol = <i>None</i>	(StrOpt) The storage protocol to be used on the data path with the storage system.
netapp_transport_type = <i>http</i>	(StrOpt) The transport protocol used when communicating with the storage system or proxy server.
netapp_vserver = <i>None</i>	(StrOpt) This option specifies the virtual storage server (Vserver) name on the storage cluster on which provisioning of block storage volumes should occur.
thres_avl_size_perc_start = <i>20</i>	(IntOpt) If the percentage of available space for an NFS share has dropped below the value specified by this option, the NFS image cache will be cleaned.

Configuration option = Default value	Description
thres_avl_size_perc_stop = 60	(IntOpt) When the percentage of available space on an NFS share has reached the percentage specified by this option, the driver will stop clearing files from the NFS image cache that have not been accessed in the last M minutes, where M is the value of the expiry_thres_minutes configuration option.

**NOTE**

Additional NetApp NFS configuration options are shared with the generic NFS driver. These options can be found here: [Table 2.20, “Description of NFS storage configuration options”](#).

**NOTE**

If you specify an account in the `netapp_login` that only has virtual storage server (Vserver) administration privileges (rather than cluster-wide administration privileges), some advanced features of the NetApp unified driver will not work and you may see warnings in the OpenStack Block Storage logs.

NetApp NFS Copy Offload client

A feature was added in the Icehouse release of the NetApp unified driver that enables Image Service images to be efficiently copied to a destination Block Storage volume. When the Block Storage and Image Service are configured to use the NetApp NFS Copy Offload client, a controller-side copy will be attempted before reverting to downloading the image from the Image Service. This improves image provisioning times while reducing the consumption of bandwidth and CPU cycles on the host(s) running the Image and Block Storage services. This is due to the copy operation being performed completely within the storage cluster.

The NetApp NFS Copy Offload client can be used in either of the following scenarios:

- The Image Service is configured to store images in an NFS share that is exported from a NetApp FlexVol volume *and* the destination for the new Block Storage volume will be on an NFS share exported from a different FlexVol volume than the one used by the Image Service. Both FlexVols must be located within the same cluster.
- The source image from the Image Service has already been cached in an NFS image cache within a Block Storage backend. The cached image resides on a different FlexVol volume than the destination for the new Block Storage volume. Both FlexVols must be located within the same cluster.

To use this feature, you must configure the Image Service, as follows:

- Set the `default_store` configuration option to `file`.
- Set the `filesystem_store_datadir` configuration option to the path to the Image Service NFS export.
- Set the `show_image_direct_url` configuration option to `True`.
- Set the `show_multiple_locations` configuration option to `True`.



IMPORTANT

If configured without the proper policy settings, a non-admin user of the Image Service can replace active image data (that is, switch out a current image without other users knowing). See the OSSN announcement (recommended actions) for configuration information:

<https://wiki.openstack.org/wiki/OSSN/OSSN-0065>

- Set the `filesystem_store_metadata_file` configuration option to a metadata file. The metadata file should contain a JSON object that contains the correct information about the NFS export used by the Image Service, similar to:

```
{
  "share_location": "nfs://192.168.0.1/myGlanceExport",
  "mount_point": "/var/lib/glance/images",
  "type": "nfs"
}
```

To use this feature, you must configure the Block Storage service, as follows:

- Set the `netapp_copyoffload_tool_path` configuration option to the path to the NetApp Copy Offload binary.
- Set the `glance_api_version` configuration option to 2.



IMPORTANT

This feature requires that:

- The storage system must have Data ONTAP v8.2 or greater installed.
- The vStorage feature must be enabled on each storage virtual machine (SVM, also known as a Vserver) that is permitted to interact with the copy offload client.
- To configure the copy offload workflow, enable NFS v4.0 or greater and export it from the SVM.

TIP

To download the NetApp copy offload binary to be utilized in conjunction with the `netapp_copyoffload_tool_path` configuration option, visit the Utility Toolchest page at the [NetApp Support portal](#) (login is required).

TIP

For more information on these options and other deployment and operational scenarios, visit the [NetApp OpenStack Deployment and Operations Guide](#).

2.1.16.1.3. NetApp-supported extra specs for clustered Data ONTAP

Extra specs enable vendors to specify extra filter criteria that the Block Storage scheduler uses when it determines which volume node should fulfill a volume provisioning request. When you use the NetApp unified driver with a clustered Data ONTAP storage system, you can leverage extra specs with

OpenStack Block Storage volume types to ensure that OpenStack Block Storage volumes are created on storage back ends that have certain properties. For example, when you configure QoS, mirroring, or compression for a storage back end.

Extra specs are associated with OpenStack Block Storage volume types, so that when users request volumes of a particular volume type, the volumes are created on storage back ends that meet the list of requirements. For example, the back ends have the available space or extra specs. You can use the specs in the following table when you define OpenStack Block Storage volume types by using the `cinder type-key` command.

Table 2.16. Description of extra specs options for NetApp Unified Driver with Clustered Data ONTAP

Extra spec	Type	Description
<code>netapp_raid_type</code>	String	Limit the candidate volume list based on one of the following raid types: <code>raid4</code> , <code>raid_dp</code> .
<code>netapp_disk_type</code>	String	Limit the candidate volume list based on one of the following disk types: <code>ATA</code> , <code>BSAS</code> , <code>EATA</code> , <code>FCAL</code> , <code>FSAS</code> , <code>LUN</code> , <code>MSATA</code> , <code>SAS</code> , <code>SATA</code> , <code>SCSI</code> , <code>XATA</code> , <code>XSAS</code> , or <code>SSD</code> .
<code>netapp:qos_policy_group^[a]</code>	String	Specify the name of a QoS policy group, which defines measurable Service Level Objectives, that should be applied to the OpenStack Block Storage volume at the time of volume creation. Ensure that the QoS policy group object within Data ONTAP should be defined before an OpenStack Block Storage volume is created, and that the QoS policy group is not associated with the destination FlexVol volume.
<code>netapp_mirrored</code>	Boolean	Limit the candidate volume list to only the ones that are mirrored on the storage controller.
<code>netapp_unmirrored^[b]</code>	Boolean	Limit the candidate volume list to only the ones that are not mirrored on the storage controller.
<code>netapp_dedup</code>	Boolean	Limit the candidate volume list to only the ones that have deduplication enabled on the storage controller.
<code>netapp_nodedup^[b]</code>	Boolean	Limit the candidate volume list to only the ones that have deduplication disabled on the storage controller.
<code>netapp_compression</code>	Boolean	Limit the candidate volume list to only the ones that have compression enabled on the storage controller.
<code>netapp_nocompression^[b]</code>	Boolean	Limit the candidate volume list to only the ones that have compression disabled on the storage controller.

Extra spec	Type	Description
netapp_thin_provisioned	Boolean	Limit the candidate volume list to only the ones that support thin provisioning on the storage controller.
netapp_thick_provisioned ^[b]	Boolean	Limit the candidate volume list to only the ones that support thick provisioning on the storage controller.

[a] Note that this extra spec has a colon (:) in its name because it is used by the driver to assign the QoS policy group to the OpenStack Block Storage volume after it has been provisioned.

[b] In the Juno release, these negative-assertion extra specs are formally deprecated by the NetApp unified driver. Instead of using the deprecated negative-assertion extra specs (for example, **netapp_unmirrored**) with a value of **true**, use the corresponding positive-assertion extra spec (for example, **netapp_mirrored**) with a value of **false**.

2.1.16.2. NetApp Data ONTAP operating in 7-Mode storage family

The NetApp Data ONTAP operating in 7-Mode storage family represents a configuration group which provides OpenStack compute instances access to 7-Mode storage systems. At present it can be configured in OpenStack Block Storage to work with iSCSI and NFS storage protocols.

2.1.16.2.1. NetApp iSCSI configuration for Data ONTAP operating in 7-Mode

The NetApp iSCSI configuration for Data ONTAP operating in 7-Mode is an interface from OpenStack to Data ONTAP operating in 7-Mode storage systems for provisioning and managing the SAN block storage entity, that is, a LUN which can be accessed using iSCSI protocol.

The iSCSI configuration for Data ONTAP operating in 7-Mode is a direct interface from OpenStack to Data ONTAP operating in 7-Mode storage system and it does not require additional management software to achieve the desired functionality. It uses NetApp ONTAPI to interact with the Data ONTAP operating in 7-Mode storage system.

Configuration options for the Data ONTAP operating in 7-Mode storage family with iSCSI protocol

Configure the volume driver, storage family and storage protocol to the NetApp unified driver, Data ONTAP operating in 7-Mode, and iSCSI respectively by setting the **volume_driver**, **netapp_storage_family** and **netapp_storage_protocol** options in **cinder.conf** as follows:

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_7mode
netapp_storage_protocol = iscsi
netapp_server_hostname = myhostname
netapp_server_port = 80
netapp_login = username
netapp_password = password
```



NOTE

To use the iSCSI protocol, you must override the default value of **netapp_storage_protocol** with **iscsi**.

Table 2.17. Description of NetApp 7-Mode iSCSI driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
netapp_login = <i>None</i>	(StrOpt) Administrative user account name used to access the storage system or proxy server.
netapp_partner_backend_name = <i>None</i>	(StrOpt) The name of the config.conf stanza for a Data ONTAP (7-mode) HA partner. This option is only used by the driver when connecting to an instance with a storage family of Data ONTAP operating in 7-Mode, and it is required if the storage protocol selected is FC.
netapp_password = <i>None</i>	(StrOpt) Password for the administrative user account specified in the netapp_login option.
netapp_pool_name_search_pattern = <i>(.+)</i>	(StrOpt) This option is used to restrict provisioning to the specified pools. Specify the value of this option to be a regular expression which will be applied to the names of objects from the storage backend which represent pools in Cinder. This option is only utilized when the storage protocol is configured to use iSCSI or FC.
netapp_server_hostname = <i>None</i>	(StrOpt) The hostname (or IP address) for the storage system or proxy server.
netapp_server_port = <i>None</i>	(IntOpt) The TCP port to use for communication with the storage system or proxy server. If not specified, Data ONTAP drivers will use 80 for HTTP and 443 for HTTPS; E-Series will use 8080 for HTTP and 8443 for HTTPS.
netapp_size_multiplier = <i>1.2</i>	(FloatOpt) The quantity to be multiplied by the requested volume size to ensure enough space is available on the virtual storage server (Vserver) to fulfill the volume creation request. Note: this option is deprecated and will be removed in favor of "reserved_percentage" in the Mitaka release.
netapp_storage_family = <i>ontap_cluster</i>	(StrOpt) The storage family type used on the storage system; valid values are <i>ontap_7mode</i> for using Data ONTAP operating in 7-Mode, <i>ontap_cluster</i> for using clustered Data ONTAP, or <i>eseries</i> for using E-Series.
netapp_storage_protocol = <i>None</i>	(StrOpt) The storage protocol to be used on the data path with the storage system.

Configuration option = Default value	Description
netapp_transport_type = <i>http</i>	(StrOpt) The transport protocol used when communicating with the storage system or proxy server.
netapp_vfiler = <i>None</i>	(StrOpt) The vFiler unit on which provisioning of block storage volumes will be done. This option is only used by the driver when connecting to an instance with a storage family of Data ONTAP operating in 7-Mode. Only use this option when utilizing the MultiStore feature on the NetApp storage system.

TIP

For more information on these options and other deployment and operational scenarios, visit the [NetApp OpenStack Deployment and Operations Guide](#).

2.1.16.2.2. NetApp NFS configuration for Data ONTAP operating in 7-Mode

The NetApp NFS configuration for Data ONTAP operating in 7-Mode is an interface from OpenStack to Data ONTAP operating in 7-Mode storage system for provisioning and managing OpenStack volumes on NFS exports provided by the Data ONTAP operating in 7-Mode storage system which can then be accessed using NFS protocol.

The NFS configuration for Data ONTAP operating in 7-Mode is a direct interface from OpenStack Block Storage to the Data ONTAP operating in 7-Mode instance and as such does not require any additional management software to achieve the desired functionality. It uses NetApp ONTAPI to interact with the Data ONTAP operating in 7-Mode storage system.

Configuration options for the Data ONTAP operating in 7-Mode family with NFS protocol

Configure the volume driver, storage family, and storage protocol to the NetApp unified driver, Data ONTAP operating in 7-Mode, and NFS respectively by setting the **volume_driver**, **netapp_storage_family** and **netapp_storage_protocol** options in **cinder.conf** as follows:

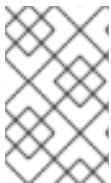
```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_7mode
netapp_storage_protocol = nfs
netapp_server_hostname = myhostname
netapp_server_port = 80
netapp_login = username
netapp_password = password
nfs_shares_config = /etc/cinder/nfs_shares
```

Table 2.18. Description of NetApp 7-Mode NFS driver configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
expiry_thres_minutes = 720	(IntOpt) This option specifies the threshold for last access time for images in the NFS image cache. When a cache cleaning cycle begins, images in the cache that have not been accessed in the last M minutes, where M is the value of this parameter, will be deleted from the cache to create free space on the NFS share.
netapp_login = <i>None</i>	(StrOpt) Administrative user account name used to access the storage system or proxy server.
netapp_partner_backend_name = <i>None</i>	(StrOpt) The name of the config.conf stanza for a Data ONTAP (7-mode) HA partner. This option is only used by the driver when connecting to an instance with a storage family of Data ONTAP operating in 7-Mode, and it is required if the storage protocol selected is FC.
netapp_password = <i>None</i>	(StrOpt) Password for the administrative user account specified in the netapp_login option.
netapp_pool_name_search_pattern = (.+)	(StrOpt) This option is used to restrict provisioning to the specified pools. Specify the value of this option to be a regular expression which will be applied to the names of objects from the storage backend which represent pools in Cinder. This option is only utilized when the storage protocol is configured to use iSCSI or FC.
netapp_server_hostname = <i>None</i>	(StrOpt) The hostname (or IP address) for the storage system or proxy server.
netapp_server_port = <i>None</i>	(IntOpt) The TCP port to use for communication with the storage system or proxy server. If not specified, Data ONTAP drivers will use 80 for HTTP and 443 for HTTPS; E-Series will use 8080 for HTTP and 8443 for HTTPS.
netapp_storage_family = <i>ontap_cluster</i>	(StrOpt) The storage family type used on the storage system; valid values are <i>ontap_7mode</i> for using Data ONTAP operating in 7-Mode, <i>ontap_cluster</i> for using clustered Data ONTAP, or <i>eseries</i> for using E-Series.
netapp_storage_protocol = <i>None</i>	(StrOpt) The storage protocol to be used on the data path with the storage system.

Configuration option = Default value	Description
<code>netapp_transport_type = http</code>	(StrOpt) The transport protocol used when communicating with the storage system or proxy server.
<code>netapp_vfiler = None</code>	(StrOpt) The vFiler unit on which provisioning of block storage volumes will be done. This option is only used by the driver when connecting to an instance with a storage family of Data ONTAP operating in 7-Mode. Only use this option when utilizing the MultiStore feature on the NetApp storage system.
<code>thres_avl_size_perc_start = 20</code>	(IntOpt) If the percentage of available space for an NFS share has dropped below the value specified by this option, the NFS image cache will be cleaned.
<code>thres_avl_size_perc_stop = 60</code>	(IntOpt) When the percentage of available space on an NFS share has reached the percentage specified by this option, the driver will stop clearing files from the NFS image cache that have not been accessed in the last M minutes, where M is the value of the <code>expiry_thres_minutes</code> configuration option.

**NOTE**

Additional NetApp NFS configuration options are shared with the generic NFS driver. For a description of these, see [Table 2.20, “Description of NFS storage configuration options”](#).

TIP

For more information on these options and other deployment and operational scenarios, visit the [NetApp OpenStack Deployment and Operations Guide](#).

2.1.16.3. NetApp E-Series storage family

The NetApp E-Series storage family represents a configuration group which provides OpenStack compute instances access to E-Series storage systems. At present it can be configured in OpenStack Block Storage to work with the iSCSI storage protocol.

2.1.16.3.1. NetApp iSCSI configuration for E-Series

The NetApp iSCSI configuration for E-Series is an interface from OpenStack to E-Series storage systems for provisioning and managing the SAN block storage entity; that is, a NetApp LUN which can be accessed using the iSCSI protocol.

The iSCSI configuration for E-Series is an interface from OpenStack Block Storage to the E-Series proxy instance and as such requires the deployment of the proxy instance in order to achieve the desired functionality. The driver uses REST APIs to interact with the E-Series proxy instance, which in turn interacts directly with the E-Series controllers.

The use of multipath and DM-MP are required when using the OpenStack Block Storage driver for E-Series. In order for OpenStack Block Storage and OpenStack Compute to take advantage of multiple paths, the following configuration options must be correctly configured:

- The `use_multipath_for_image_xfer` option should be set to `True` in the `cinder.conf` file within the driver-specific stanza (for example, `[myDriver]`).
- The `iscsi_use_multipath` option should be set to `True` in the `nova.conf` file within the `[libvirt]` stanza.

Configuration options for E-Series storage family with iSCSI protocol

Configure the volume driver, storage family, and storage protocol to the NetApp unified driver, E-Series, and iSCSI respectively by setting the `volume_driver`, `netapp_storage_family` and `netapp_storage_protocol` options in `cinder.conf` as follows:

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = eseries
netapp_storage_protocol = iscsi
netapp_server_hostname = myhostname
netapp_server_port = 80
netapp_login = username
netapp_password = password
netapp_controller_ips = 1.2.3.4, 5.6.7.8
netapp_sa_password = arrayPassword
netapp_storage_pools = pool1, pool2
use_multipath_for_image_xfer = True
```



NOTE

To use the E-Series driver, you must override the default value of `netapp_storage_family` with `eseries`.



NOTE

To use the iSCSI protocol, you must override the default value of `netapp_storage_protocol` with `iscsi`.

Table 2.19. Description of NetApp E-Series driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>netapp_controller_ips = None</code>	(StrOpt) This option is only utilized when the storage family is configured to <code>eseries</code> . This option is used to restrict provisioning to the specified controllers. Specify the value of this option to be a comma separated list of controller hostnames or IP addresses to be used for provisioning.

Configuration option = Default value	Description
netapp_enable_multiattach = <i>False</i>	(BoolOpt) This option specifies whether the driver should allow operations that require multiple attachments to a volume. An example would be live migration of servers that have volumes attached. When enabled, this backend is limited to 256 total volumes in order to guarantee volumes can be accessed by more than one host.
netapp_host_type = <i>None</i>	(StrOpt) This option defines the type of operating system for all initiators that can access a LUN. This information is used when mapping LUNs to individual hosts or groups of hosts.
netapp_login = <i>None</i>	(StrOpt) Administrative user account name used to access the storage system or proxy server.
netapp_partner_backend_name = <i>None</i>	(StrOpt) The name of the config.conf stanza for a Data ONTAP (7-mode) HA partner. This option is only used by the driver when connecting to an instance with a storage family of Data ONTAP operating in 7-Mode, and it is required if the storage protocol selected is FC.
netapp_password = <i>None</i>	(StrOpt) Password for the administrative user account specified in the netapp_login option.
netapp_pool_name_search_pattern = <i>(.+)</i>	(StrOpt) This option is used to restrict provisioning to the specified pools. Specify the value of this option to be a regular expression which will be applied to the names of objects from the storage backend which represent pools in Cinder. This option is only utilized when the storage protocol is configured to use iSCSI or FC.
netapp_sa_password = <i>None</i>	(StrOpt) Password for the NetApp E-Series storage array.
netapp_server_hostname = <i>None</i>	(StrOpt) The hostname (or IP address) for the storage system or proxy server.
netapp_server_port = <i>None</i>	(IntOpt) The TCP port to use for communication with the storage system or proxy server. If not specified, Data ONTAP drivers will use 80 for HTTP and 443 for HTTPS; E-Series will use 8080 for HTTP and 8443 for HTTPS.

Configuration option = Default value	Description
<code>netapp_storage_family = ontap_cluster</code>	(StrOpt) The storage family type used on the storage system; valid values are <code>ontap_7mode</code> for using Data ONTAP operating in 7-Mode, <code>ontap_cluster</code> for using clustered Data ONTAP, or <code>eseries</code> for using E-Series.
<code>netapp_transport_type = http</code>	(StrOpt) The transport protocol used when communicating with the storage system or proxy server.
<code>netapp_webservice_path = /devmgr/v2</code>	(StrOpt) This option is used to specify the path to the E-Series proxy application on a proxy server. The value is combined with the value of the <code>netapp_transport_type</code> , <code>netapp_server_hostname</code> , and <code>netapp_server_port</code> options to create the URL used by the driver to connect to the proxy application.

TIP

For more information on these options and other deployment and operational scenarios, visit the [NetApp OpenStack Deployment and Operations Guide](#).

2.1.16.4. Upgrading prior NetApp drivers to the NetApp unified driver

NetApp introduced a new unified block storage driver in Havana for configuring different storage families and storage protocols. This requires defining upgrade path for NetApp drivers which existed in releases prior to Havana. This section covers the upgrade configuration for NetApp drivers to the new unified configuration and a list of deprecated NetApp drivers.

2.1.16.4.1. Upgraded NetApp drivers

This section describes how to update OpenStack Block Storage configuration from a pre-Havana release to the unified driver format.

Driver upgrade configuration

1. NetApp iSCSI direct driver for Clustered Data ONTAP in Grizzly (or earlier).

```
volume_driver =
cinder.volume.drivers.netapp.iscsi.NetAppDirectCmodeISCSIDriver
```

NetApp unified driver configuration.

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_cluster
netapp_storage_protocol = iscsi
```

2. NetApp NFS direct driver for Clustered Data ONTAP in Grizzly (or earlier).

■


```
volume_driver =
cinder.volume.drivers.netapp.nfs.NetAppDirectCmodeNfsDriver
```

NetApp unified driver configuration.

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_cluster
netapp_storage_protocol = nfs
```

3. NetApp iSCSI direct driver for Data ONTAP operating in 7-Mode storage controller in Grizzly (or earlier)

```
volume_driver =
cinder.volume.drivers.netapp.iscsi.NetAppDirect7modeISCSIDriver
```

NetApp unified driver configuration

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_7mode
netapp_storage_protocol = iscsi
```

4. NetApp NFS direct driver for Data ONTAP operating in 7-Mode storage controller in Grizzly (or earlier)

```
volume_driver =
cinder.volume.drivers.netapp.nfs.NetAppDirect7modeNfsDriver
```

NetApp unified driver configuration

```
volume_driver = cinder.volume.drivers.netapp.common.NetAppDriver
netapp_storage_family = ontap_7mode
netapp_storage_protocol = nfs
```

2.1.16.4.2. Deprecated NetApp drivers

This section lists the NetApp drivers in earlier releases that are deprecated in Havana.

1. NetApp iSCSI driver for clustered Data ONTAP.

```
volume_driver =
cinder.volume.drivers.netapp.iscsi.NetAppCmodeISCSIDriver
```

2. NetApp NFS driver for clustered Data ONTAP.

```
volume_driver =
cinder.volume.drivers.netapp.nfs.NetAppCmodeNfsDriver
```

3. NetApp iSCSI driver for Data ONTAP operating in 7-Mode storage controller.

```
volume_driver = cinder.volume.drivers.netapp.iscsi.NetAppISCSIDriver
```

4. NetApp NFS driver for Data ONTAP operating in 7-Mode storage controller.

```
volume_driver = cinder.volume.drivers.netapp.nfs.NetAppNFSDriver
```

**NOTE**

For support information on deprecated NetApp drivers in the Havana release, visit the [NetApp OpenStack Deployment and Operations Guide](#).

2.1.17. NFS driver

The Network File System (NFS) is a distributed file system protocol originally developed by Sun Microsystems in 1984. An NFS server *exports* one or more of its file systems, known as *shares*. An NFS client can mount these exported shares on its own file system. You can perform file actions on this mounted remote file system as if the file system were local.

2.1.17.1. How the NFS driver works

The NFS driver, and other drivers based on it, work quite differently than a traditional block storage driver.

The NFS driver does not actually allow an instance to access a storage device at the block level. Instead, files are created on an NFS share and mapped to instances, which emulates a block device. This works in a similar way to QEMU, which stores instances in the `/var/lib/nova/instances` directory.

2.1.17.2. Enable the NFS driver and related options

To use Cinder with the NFS driver, first set the `volume_driver` in `cinder.conf`:

```
volume_driver=cinder.volume.drivers.nfs.NfsDriver
```

The following table contains the options supported by the NFS driver.

Table 2.20. Description of NFS storage configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>nfs_mount_attempts = 3</code>	(IntOpt) The number of attempts to mount nfs shares before raising an error. At least one attempt will be made to mount an nfs share, regardless of the value specified.
<code>nfs_mount_options = None</code>	(StrOpt) Mount options passed to the nfs client. See section of the nfs man page for details.
<code>nfs_mount_point_base = \$state_path/mnt</code>	(StrOpt) Base dir containing mount points for nfs shares.

Configuration option = Default value	Description
<code>nfs_oversub_ratio = 1.0</code>	(FloatOpt) This will compare the allocated to available space on the volume destination. If the ratio exceeds this number, the destination will no longer be valid. Note that this option is deprecated in favor of "max_oversubscription_ratio" and will be removed in the Mitaka release.
<code>nfs_shares_config = /etc/cinder/nfs_shares</code>	(StrOpt) File with the list of available nfs shares
<code>nfs_sparsed_volumes = True</code>	(BoolOpt) Create volumes as sparsed files which take no space. If set to False volume is created as regular file. In such case volume creation takes a lot of time.
<code>nfs_used_ratio = 0.95</code>	(FloatOpt) Percent of ACTUAL usage of the underlying volume before no new volumes can be allocated to the volume destination. Note that this option is deprecated in favor of "reserved_percentage" and will be removed in the Mitaka release.



NOTE

As of the Icehouse release, the NFS driver (and other drivers based off it) will attempt to mount shares using version 4.1 of the NFS protocol (including pNFS). If the mount attempt is unsuccessful due to a lack of client or server support, a subsequent mount attempt that requests the default behavior of the `mount.nfs` command will be performed. On most distributions, the default behavior is to attempt mounting first with NFS v4.0, then silently fall back to NFS v3.0 if necessary. If the `nfs_mount_options` configuration option contains a request for a specific version of NFS to be used, or if specific options are specified in the shares configuration file specified by the `nfs_shares_config` configuration option, the mount will be attempted as requested with no subsequent attempts.

2.1.17.3. How to use the NFS driver

1. Access to one or more NFS servers. Creating an NFS server is outside the scope of this document. This example assumes access to the following NFS servers and mount points:

- `192.168.1.200:/storage`
- `192.168.1.201:/storage`
- `192.168.1.202:/storage`

This example demonstrates the use of with this driver with multiple NFS servers. Multiple servers are not required. One is usually enough.

2. Add your list of NFS servers to the file you specified with the `nfs_shares_config` option. For example, if the value of this option was set to `/etc/cinder/shares.txt`, then:

—

```
# cat /etc/cinder/shares.txt
192.168.1.200:/storage 192.168.1.201:/storage 192.168.1.202:/storage
```

Comments are allowed in this file. They begin with a #.

3. Configure the `nfs_mount_point_base` option. This is a directory where `cinder-volume` mounts all NFS shares stored in `shares.txt`. For this example, `/var/lib/cinder/nfs` is used. You can, of course, use the default value of `$state_path/mnt`.
4. Start the `cinder-volume` service. `/var/lib/cinder/nfs` should now contain a directory for each NFS share specified in `shares.txt`. The name of each directory is a hashed name:

```
# ls /var/lib/cinder/nfs/
... 46c5db75dc3a3a50a10bfd1a456a9f3f ...
```

5. You can now create volumes as you normally would:

```
$ nova volume-create --display-name myvol 5
# ls /var/lib/cinder/nfs/46c5db75dc3a3a50a10bfd1a456a9f3f
volume-a8862558-e6d6-4648-b5df-bb84f31c8935
```

This volume can also be attached and deleted just like other volumes. However, snapshotting is *not* supported.

NFS driver notes

- `cinder-volume` manages the mounting of the NFS shares as well as volume creation on the shares. Keep this in mind when planning your OpenStack architecture. If you have one master NFS server, it might make sense to only have one `cinder-volume` service to handle all requests to that NFS server. However, if that single server is unable to handle all requests, more than one `cinder-volume` service is needed as well as potentially more than one NFS server.
- Because data is stored in a file and not actually on a block storage device, you might not see the same IO performance as you would with a traditional block storage driver. Test accordingly.
- Despite possible IO performance loss, having volume data stored in a file might be beneficial. For example, backing up volumes can be as easy as copying the volume files.



NOTE

Regular IO flushing and syncing still stands.

2.1.18. SolidFire

The SolidFire Cluster is a high performance all SSD iSCSI storage device that provides massive scale out capability and extreme fault tolerance. A key feature of the SolidFire cluster is the ability to set and modify during operation specific QoS levels on a volume for volume basis. The SolidFire cluster offers this along with de-duplication, compression, and an architecture that takes full advantage of SSDs.

To configure the use of a SolidFire cluster with Block Storage, modify your `cinder.conf` file as follows:

```

volume_driver = cinder.volume.drivers.solidfire.SolidFireDriver
san_ip = 172.17.1.182           # the address of your MVIP
san_login = sfadmin             # your cluster admin login
san_password = sfpasword       # your cluster admin password
sf_account_prefix = ''         # prefix for tenant account creation on
solidfire cluster

```



WARNING

Older versions of the SolidFire driver (prior to Icehouse) created a unique account prefixed with `$cinder-volume-service-hostname-$tenant-id` on the SolidFire cluster for each tenant. Unfortunately, this account formation resulted in issues for High Availability (HA) installations and installations where the `cinder-volume` service can move to a new node. The current default implementation does not experience this issue as no prefix is used. For installations created on a prior release, the OLD default behavior can be configured by using the keyword `"hostname"` in `sf_account_prefix`.

Table 2.21. Description of SolidFire driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>sf_account_prefix</code> = <i>None</i>	(StrOpt) Create SolidFire accounts with this prefix. Any string can be used here, but the string <code>"hostname"</code> is special and will create a prefix using the cinder node hostname (previous default behavior). The default is NO prefix.
<code>sf_allow_template_caching</code> = <i>True</i>	(BoolOpt) Create an internal cache of copy of images when a bootable volume is created to eliminate fetch from glance and qemu-conversion on subsequent calls.
<code>sf_allow_tenant_qos</code> = <i>False</i>	(BoolOpt) Allow tenants to specify QOS on create
<code>sf_api_port</code> = <i>443</i>	(IntOpt) SolidFire API port. Useful if the device api is behind a proxy on a different port.
<code>sf_emulate_512</code> = <i>True</i>	(BoolOpt) Set 512 byte emulation on volume creation;
<code>sf_enable_volume_mapping</code> = <i>True</i>	(BoolOpt) Create an internal mapping of volume IDs and account. Optimizes lookups and performance at the expense of memory, very large deployments may want to consider setting to False.

Configuration option = Default value	Description
sf_svip = <i>None</i>	(StrOpt) Overrides default cluster SVIP with the one specified. This is required for deployments that have implemented the use of VLANs for iSCSI networks in their cloud.
sf_template_account_name = <i>openstack-vtemplate</i>	(StrOpt) Account name on the SolidFire Cluster to use as owner of template/cache volumes (created if does not exist).

2.1.19. Tintri

Tintri VMstore is a smart storage that sees, learns and adapts for cloud and virtualization. The Tintri Cinder driver will interact with configured VMstore running Tintri OS 4.0 and above. It supports various operations using Tintri REST APIs and NFS protocol.

To configure the use of a Tintri VMstore with Block Storage, perform the following actions:

1. Edit the `etc/cinder/cinder.conf` file and set the `cinder.volume.drivers.tintri` options:

```
volume_driver=cinder.volume.drivers.tintri.TintriDriver
# Mount options passed to the nfs client. See section of the
# nfs man page for details. (string value)
nfs_mount_options=vers=3,lookupcache=pos

#
# Options defined in cinder.volume.drivers.tintri
#

# The hostname (or IP address) for the storage system (string
# value)
tintri_server_hostname={Tintri VMstore Management IP}

# User name for the storage system (string value)
tintri_server_username={username}

# Password for the storage system (string value)
tintri_server_password={password}

# API version for the storage system (string value)
#tintri_api_version=v310

# Following options needed for NFS configuration
# File with the list of available nfs shares (string value)
#nfs_shares_config=/etc/cinder/nfs_shares
```

2. Edit the `etc/nova/nova.conf` file, and set the `nfs_mount_options`:

```
nfs_mount_options=vers=3
```

3. Edit the `/etc/cinder/nfs_shares` file, and add the Tintri VMstore mount points associated with the configured VMstore management IP in the `cinder.conf` file:

```
{vmstore_data_ip}:/tintri/{submount1}
{vmstore_data_ip}:/tintri/{submount2}
```

Table 2.22. Description of Tintri volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
tintri_api_version = <i>v310</i>	(StrOpt) API version for the storage system
tintri_server_hostname = <i>None</i>	(StrOpt) The hostname (or IP address) for the storage system
tintri_server_password = <i>None</i>	(StrOpt) Password for the storage system
tintri_server_username = <i>None</i>	(StrOpt) User name for the storage system

2.1.20. Violin Memory 7000 Series FSP volume driver

The OpenStack V7000 driver package from Violin Memory adds Block Storage service support for Violin 7300 Flash Storage Platforms (FSPs) and 7700 FSP controllers.

The driver package release can be used with any OpenStack Liberty deployment for all 7300 FSPs and 7700 FSP controllers running Concerto 7.5.3 and later using Fibre Channel HBAs.

2.1.20.1. System requirements

To use the Violin driver, the following are required:

- Violin 7300/7700 series FSP with:
 - Concerto OS version 7.5.3 or later
 - Fibre channel host interfaces
- The Violin block storage driver: This driver implements the block storage API calls. The driver is included with the OpenStack Liberty release.
- The `vmemclient` library: This is the Violin Array Communications library to the Flash Storage Platform through a REST-like interface. The client can be installed using the python `pip` installer tool. Further information on `vmemclient` can be found on [PyPI](#).

```
pip install vmemclient
```

2.1.20.2. Supported operations

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.

- Create a volume from a snapshot.
- Copy an image to a volume.
- Copy a volume to an image.
- Clone a volume.
- Extend a volume.

note

Listed operations are supported for thick, thin, and dedup luns, with the exception of cloning. Cloning operations are supported only on thick luns.

2.1.20.3. Driver configuration

Once the array is configured as per the installation guide, it is simply a matter of editing the cinder configuration file to add or modify the parameters. The driver currently only supports fibre channel configuration.

2.1.20.3.1. Fibre channel configuration

Set the following in your `cinder.conf` configuration file, replacing the variables using the guide in the following section:

```
volume_driver = cinder.volume.drivers.violin.v7000_fcp.V7000FCPDriver
volume_backend_name = vmem_violinfsp
extra_capabilities = VMEM_CAPABILITIES
san_ip = VMEM_MGMT_IP
san_login = VMEM_USER_NAME
san_password = VMEM_PASSWORD
use_multipath_for_image_xfer = true
```

2.1.20.3.2. Configuration parameters

Description of configuration value placeholders:

VMEM_CAPABILITIES

User defined capabilities, a JSON formatted string specifying key-value pairs (string value). The ones particularly supported are **dedup** and **thin**. Only these two capabilities are listed here in `cinder.conf` file, indicating this backend be selected for creating luns which have a volume type associated with them that have **dedup** or **thin** `extra_specs` specified. For example, if the FSP is configured to support dedup luns, set the associated driver capabilities to: `{"dedup":"True","thin":"True"}`.

VMEM_MGMT_IP

External IP address or host name of the Violin 7300 Memory Gateway. This can be an IP address or host name.

VMEM_USER_NAME

Log-in user name for the Violin 7300 Memory Gateway or 7700 FSP controller. This user must have administrative rights on the array or controller.

VMEM_PASSWORD

Log-in user's password.

2.2. BACKUP DRIVERS

This section describes how to configure the **cinder-backup** service and its drivers.

To set a backup driver, use the **backup_driver** flag. By default there is no backup driver enabled.

2.2.1. Ceph backup driver

The Ceph backup driver backs up volumes of any type to a Ceph back-end store. The driver can also detect whether the volume to be backed up is a Ceph RBD volume, and if so, it tries to perform incremental and differential backups.

For source Ceph RBD volumes, you can perform backups within the same Ceph pool (not recommended). You can also perform backups between different Ceph pools and between different Ceph clusters.

At the time of writing, differential backup support in Ceph/librbd was quite new. This driver attempts a differential backup in the first instance. If the differential backup fails, the driver falls back to full backup/copy.

If incremental backups are used, multiple backups of the same volume are stored as snapshots so that minimal space is consumed in the backup store. It takes far less time to restore a volume than to take a full copy.

**NOTE**

Block Storage enables you to:

- Restore to a new volume, which is the default and recommended action.
- Restore to the original volume from which the backup was taken. The restore action takes a full copy because this is the safest action.

To enable the Ceph backup driver, include the following option in the **cinder.conf** file:

```
backup_driver = cinder.backup.drivers.ceph
```

The following configuration options are available for the Ceph backup driver.

Table 2.23. Description of Ceph backup driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
backup_ceph_chunk_size = 134217728	(IntOpt) The chunk size, in bytes, that a backup is broken into before transfer to the Ceph object store.

Configuration option = Default value	Description
backup_ceph_conf = <code>/etc/ceph/ceph.conf</code>	(StrOpt) Ceph configuration file to use.
backup_ceph_pool = <code>backups</code>	(StrOpt) The Ceph pool where volume backups are stored.
backup_ceph_stripe_count = <code>0</code>	(IntOpt) RBD stripe count to use when creating a backup image.
backup_ceph_stripe_unit = <code>0</code>	(IntOpt) RBD stripe unit to use when creating a backup image.
backup_ceph_user = <code>cinder</code>	(StrOpt) The Ceph user to connect with. Default here is to use the same user as for Cinder volumes. If not using cephx this should be set to None.
restore_discard_excess_bytes = <code>True</code>	(BoolOpt) If True, always discard excess bytes when restoring volumes i.e. pad with zeroes.

This example shows the default options for the Ceph backup driver.

```

backup_ceph_conf=/etc/ceph/ceph.conf
backup_ceph_user = cinder
backup_ceph_chunk_size = 134217728
backup_ceph_pool = backups
backup_ceph_stripe_unit = 0
backup_ceph_stripe_count = 0

```

2.2.2. IBM Tivoli Storage Manager backup driver

The IBM Tivoli Storage Manager (TSM) backup driver enables performing volume backups to a TSM server.

The TSM client should be installed and configured on the machine running the **cinder-backup** service. See the *IBM Tivoli Storage Manager Backup-Archive Client Installation and User's Guide* for details on installing the TSM client.

To enable the IBM TSM backup driver, include the following option in **cinder.conf**:

```

backup_driver = cinder.backup.drivers.tsm

```

The following configuration options are available for the TSM backup driver.

Table 2.24. Description of IBM Tivoli Storage Manager backup driver configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
backup_tsm_compression = <i>True</i>	(BoolOpt) Enable or Disable compression for backups
backup_tsm_password = <i>password</i>	(StrOpt) TSM password for the running username
backup_tsm_volume_prefix = <i>backup</i>	(StrOpt) Volume prefix for the backup id when backing up to TSM

This example shows the default options for the TSM backup driver.

```
backup_tsm_volume_prefix = backup
backup_tsm_password = password
backup_tsm_compression = True
```

2.2.3. Swift backup driver

The backup driver for the swift back end performs a volume backup to an object storage system.

To enable the swift backup driver, include the following option in the `cinder.conf` file:

```
backup_driver = cinder.backup.drivers.swift
```

The following configuration options are available for the Swift back-end backup driver.

Table 2.25. Description of Swift backup driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
backup_swift_auth = <i>per_user</i>	(StrOpt) Swift authentication mechanism
backup_swift_auth_version = <i>1</i>	(StrOpt) Swift authentication version. Specify "1" for auth 1.0, or "2" for auth 2.0
backup_swift_block_size = <i>32768</i>	(IntOpt) The size in bytes that changes are tracked for incremental backups. <code>backup_swift_object_size</code> has to be multiple of <code>backup_swift_block_size</code> .
backup_swift_ca_cert_file = <i>None</i>	(StrOpt) Location of the CA certificate file to use for swift client requests.
backup_swift_container = <i>volumebackups</i>	(StrOpt) The default Swift container to use

Configuration option = Default value	Description
backup_swift_enable_progress_timer = <i>True</i>	(BoolOpt) Enable or Disable the timer to send the periodic progress notifications to Ceilometer when backing up the volume to the Swift backend storage. The default value is True to enable the timer.
backup_swift_key = <i>None</i>	(StrOpt) Swift key for authentication
backup_swift_object_size = <i>52428800</i>	(IntOpt) The size in bytes of Swift backup objects
backup_swift_retry_attempts = <i>3</i>	(IntOpt) The number of retries to make for Swift operations
backup_swift_retry_backoff = <i>2</i>	(IntOpt) The backoff time in seconds between Swift retries
backup_swift_tenant = <i>None</i>	(StrOpt) Swift tenant/account name. Required when connecting to an auth 2.0 system
backup_swift_url = <i>None</i>	(StrOpt) The URL of the Swift endpoint
backup_swift_user = <i>None</i>	(StrOpt) Swift user name
swift_catalog_info = <i>object-store:swift:publicURL</i>	(StrOpt) Info to match when looking for swift in the service catalog. Format is: separated values of the form: <service_type>:<service_name>:<endpoint_type> - Only used if backup_swift_url is unset

To enable the swift backup driver for 1.0 or 2.0 authentication version, specify **1** or **2** correspondingly. For example:

```
backup_swift_auth_version = 2
```

In addition, the 2.0 authentication system requires **backup_swift_tenant** setting:

```
backup_swift_tenant = <None>
```

This example shows the default options for the Swift back-end backup driver.

```
backup_swift_url = http://localhost:8080/v1/AUTH_
backup_swift_auth = per_user
backup_swift_auth_version = 1
backup_swift_user = <None>
backup_swift_key = <None>
backup_swift_container = volumebackups
backup_swift_object_size = 52428800
```

```

backup_swift_retry_attempts = 3
backup_swift_retry_backoff = 2
backup_compression_algorithm = zlib

```

2.2.4. NFS backup driver

The backup driver for the NFS back end backs up volumes of any type to an NFS exported backup repository.

To enable the NFS backup driver, include the following option in the **[DEFAULT]** section of the `cinder.conf` file:

```

backup_driver = cinder.backup.drivers.nfs

```

The following configuration options are available for the NFS back-end backup driver.

Table 2.26. Description of NFS backup driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
backup_container = <i>None</i>	(StrOpt) Custom directory to use for backups.
backup_enable_progress_timer = <i>True</i>	(BoolOpt) Enable or Disable the timer to send the periodic progress notifications to Ceilometer when backing up the volume to the backend storage. The default value is True to enable the timer.
backup_file_size = <i>1999994880</i>	(IntOpt) The maximum size in bytes of the files used to hold backups. If the volume being backed up exceeds this size, then it will be backed up into multiple files. <code>backup_file_size</code> must be a multiple of <code>backup_sha_block_size_bytes</code> .
backup_mount_options = <i>None</i>	(StrOpt) Mount options passed to the NFS client. See NFS man page for details.
backup_mount_point_base = <i>\$state_path/backup_mount</i>	(StrOpt) Base dir containing mount point for NFS share.
backup_sha_block_size_bytes = <i>32768</i>	(IntOpt) The size in bytes that changes are tracked for incremental backups. <code>backup_file_size</code> has to be multiple of <code>backup_sha_block_size_bytes</code> .
backup_share = <i>None</i>	(StrOpt) NFS share in hostname:path, ipv4addr:path, or "[ipv6addr]:path" format.

2.3. BLOCK STORAGE SAMPLE CONFIGURATION FILES

All the files in this section can be found in `/etc/cinder`.

2.3.1. cinder.conf

The `cinder.conf` file is installed in `/etc/cinder` by default. When you manually install the Block Storage service, the options in the `cinder.conf` file are set to default values.

The `cinder.conf` file contains most of the options to configure the Block Storage service.

```
[DEFAULT]

#
# Options defined in oslo.messaging
#

# ZeroMQ bind address. Should be a wildcard (*), an ethernet
# interface, or IP. The "host" option should point or resolve
# to this address. (string value)
#rpc_zmq_bind_address=*

# MatchMaker driver. (string value)
#rpc_zmq_matchmaker=local

# ZeroMQ receiver listening port. (integer value)
#rpc_zmq_port=9501

# Number of ZeroMQ contexts, defaults to 1. (integer value)
#rpc_zmq_contexts=1

# Maximum number of ingress messages to locally buffer per
# topic. Default is unlimited. (integer value)
#rpc_zmq_topic_backlog=<None>

# Directory for holding IPC sockets. (string value)
#rpc_zmq_ipc_dir=/var/run/openstack

# Name of this node. Must be a valid hostname, FQDN, or IP
# address. Must match "host" option, if running Nova. (string
# value)
#rpc_zmq_host=cinder

# Seconds to wait before a cast expires (TTL). Only supported
# by impl_zmq. (integer value)
#rpc_cast_timeout=30

# Heartbeat frequency. (integer value)
#matchmaker_heartbeat_freq=300

# Heartbeat time-to-live. (integer value)
#matchmaker_heartbeat_ttl=600

# Size of RPC thread pool. (integer value)
#rpc_thread_pool_size=64

# Driver or drivers to handle sending notifications. (multi
# valued)
#notification_driver=
```

```

# AMQP topic used for OpenStack notifications. (list value)
# Deprecated group/name - [rpc_notifier2]/topics
#notification_topics=notifications

# Seconds to wait for a response from a call. (integer value)
#rpc_response_timeout=60

# A URL representing the messaging driver to use and its full
# configuration. If not set, we fall back to the rpc_backend
# option and driver specific configuration. (string value)
#transport_url=<None>

# The messaging driver to use, defaults to rabbit. Other
# drivers include qpid and zmq. (string value)
#rpc_backend=rabbit

# The default exchange under which topics are scoped. May be
# overridden by an exchange name specified in the
# transport_url option. (string value)
#control_exchange=openstack

#
# Options defined in cinder.exception
#

# Make exception message format errors fatal. (boolean value)
#fatal_exception_format_errors=false

#
# Options defined in cinder.quota
#

# Number of volumes allowed per project (integer value)
#quota_volumes=10

# Number of volume snapshots allowed per project (integer
# value)
#quota_snapshots=10

# Number of consistencygroups allowed per project (integer
# value)
#quota_consistencygroups=10

# Total amount of storage, in gigabytes, allowed for volumes
# and snapshots per project (integer value)
#quota_gigabytes=1000

# Number of volume backups allowed per project (integer value)
#quota_backups=10

# Total amount of storage, in gigabytes, allowed for backups
# per project (integer value)
#quota_backup_gigabytes=1000

```

```
# Number of seconds until a reservation expires (integer
# value)
#reservation_expire=86400

# Count of reservations until usage is refreshed (integer
# value)
#until_refresh=0

# Number of seconds between subsequent usage refreshes
# (integer value)
#max_age=0

# Default driver to use for quota checks (string value)
#quota_driver=cinder.quota.DbQuotaDriver

# Enables or disables use of default quota class with default
# quota. (boolean value)
#use_default_quota_class=true

#
# Options defined in cinder.service
#

# Interval, in seconds, between nodes reporting state to
# datastore (integer value)
#report_interval=10

# Interval, in seconds, between running periodic tasks
# (integer value)
#periodic_interval=60

# Range, in seconds, to randomly delay when starting the
# periodic task scheduler to reduce stampeding. (Disable by
# setting to 0) (integer value)
#periodic_fuzzy_delay=60

# IP address on which OpenStack Volume API listens (string
# value)
#osapi_volume_listen=0.0.0.0

# Port on which OpenStack Volume API listens (integer value)
#osapi_volume_listen_port=8776

# Number of workers for OpenStack Volume API service. The
# default is equal to the number of CPUs available. (integer
# value)
#osapi_volume_workers=<None>

#
# Options defined in cinder.ssh_utils
#

# Option to enable strict host key checking. When set to
# "True" Cinder will only connect to systems with a host key
```



```

# present in the configured "ssh_hosts_key_file". When set to
# "False" the host key will be saved upon first connection and
# used for subsequent connections. Default=False (boolean
# value)
#strict_ssh_host_key_policy=false

# File containing SSH host keys for the systems with which
# Cinder needs to communicate. OPTIONAL:
# Default=$state_path/ssh_known_hosts (string value)
#ssh_hosts_key_file=$state_path/ssh_known_hosts

#
# Options defined in cinder.test
#

# File name of clean sqlite db (string value)
#sqlite_clean_db=clean.sqlite

#
# Options defined in cinder.wsgi
#

# Maximum line size of message headers to be accepted.
# max_header_line may need to be increased when using large
# tokens (typically those generated by the Keystone v3 API
# with big service catalogs). (integer value)
#max_header_line=16384

# Timeout for client connections' socket operations. If an
# incoming connection is idle for this number of seconds it
# will be closed. A value of '0' means wait forever. (integer
# value)
#client_socket_timeout=900

# If False, closes the client socket connection explicitly.
# Setting it to True to maintain backward compatibility.
# Recommended setting is set it to False. (boolean value)
#wsgi_keep_alive=true

# Sets the value of TCP_KEEPALIVE (True/False) for each server
# socket. (boolean value)
#tcp_keepalive=true

# Sets the value of TCP_KEEPIDLE in seconds for each server
# socket. Not supported on OS X. (integer value)
#tcp_keepidle=600

# Sets the value of TCP_KEEPINTVL in seconds for each server
# socket. Not supported on OS X. (integer value)
#tcp_keepalive_interval=<None>

# Sets the value of TCP_KEEPCNT for each server socket. Not
# supported on OS X. (integer value)
#tcp_keepalive_count=<None>

```

```
# CA certificate file to use to verify connecting clients
# (string value)
#ssl_ca_file=<None>

# Certificate file to use when starting the server securely
# (string value)
#ssl_cert_file=<None>

# Private key file to use when starting the server securely
# (string value)
#ssl_key_file=<None>

#
# Options defined in cinder.api.common
#

# The maximum number of items that a collection resource
# returns in a single response (integer value)
#osapi_max_limit=1000

# Base URL that will be presented to users in links to the
# OpenStack Volume API (string value)
# Deprecated group/name - [DEFAULT]/osapi_compute_link_prefix
#osapi_volume_base_URL=<None>

#
# Options defined in cinder.api.middleware.auth
#

# Treat X-Forwarded-For as the canonical remote address. Only
# enable this if you have a sanitizing proxy. (boolean value)
#use_forwarded_for=false

#
# Options defined in cinder.api.middleware.sizelimit
#

# Max size for body of a request (integer value)
#osapi_max_request_body_size=114688

#
# Options defined in cinder.api.views.versions
#

# Public url to use for versions endpoint. The default is
# None, which will use the request's host_url attribute to
# populate the URL base. If Cinder is operating behind a
# proxy, you will want to change this to represent the proxy's
# URL. (string value)
#public_endpoint=<None>
```

```

#
# Options defined in cinder.backup.chunkeddriver
#

# Compression algorithm (None to disable) (string value)
#backup_compression_algorithm=zlib


#
# Options defined in cinder.backup.driver
#

# Backup metadata version to be used when backing up volume
# metadata. If this number is bumped, make sure the service
# doing the restore supports the new version. (integer value)
#backup_metadata_version=2

# The number of chunks or objects, for which one Ceilometer
# notification will be sent (integer value)
#backup_object_number_per_notification=10

# Interval, in seconds, between two progress notifications
# reporting the backup status (integer value)
#backup_timer_interval=120


#
# Options defined in cinder.backup.drivers.ceph
#

# Ceph configuration file to use. (string value)
#backup_ceph_conf=/etc/ceph/ceph.conf

# The Ceph user to connect with. Default here is to use the
# same user as for Cinder volumes. If not using cephx this
# should be set to None. (string value)
#backup_ceph_user=cinder

# The chunk size, in bytes, that a backup is broken into
# before transfer to the Ceph object store. (integer value)
#backup_ceph_chunk_size=134217728

# The Ceph pool where volume backups are stored. (string
# value)
#backup_ceph_pool=backups

# RBD stripe unit to use when creating a backup image.
# (integer value)
#backup_ceph_stripe_unit=0

# RBD stripe count to use when creating a backup image.
# (integer value)
#backup_ceph_stripe_count=0

# If True, always discard excess bytes when restoring volumes

```

```
# i.e. pad with zeroes. (boolean value)
#restore_discard_excess_bytes=true

#
# Options defined in cinder.backup.drivers.nfs
#

# The maximum size in bytes of the files used to hold backups.
# If the volume being backed up exceeds this size, then it
# will be backed up into multiple files. (integer value)
#backup_file_size=1999994880

# The size in bytes that changes are tracked for incremental
# backups. backup_swift_object_size has to be multiple of
# backup_swift_block_size. (integer value)
#backup_sha_block_size_bytes=32768

# Enable or Disable the timer to send the periodic progress
# notifications to Ceilometer when backing up the volume to
# the backend storage. The default value is True to enable the
# timer. (boolean value)
#backup_enable_progress_timer=true

# Base dir containing mount point for NFS share. (string
# value)
#backup_mount_point_base=$state_path/backup_mount

# NFS share in fqdn:path, ipv4addr:path, or "[ipv6addr]:path"
# format. (string value)
#backup_share=<None>

# Mount options passed to the NFS client. See NFS man page for
# details. (string value)
#backup_mount_options=<None>

# Custom container to use for backups. (string value)
#backup_container=<None>

#
# Options defined in cinder.backup.drivers.swift
#

# The URL of the Swift endpoint (string value)
#backup_swift_url=<None>

# Info to match when looking for swift in the service catalog.
# Format is: separated values of the form:
# <service_type>:<service_name>:<endpoint_type> - Only used if
# backup_swift_url is unset (string value)
#swift_catalog_info=object-store:swift:publicURL

# Swift authentication mechanism (string value)
#backup_swift_auth=per_user
```

```

# Swift authentication version. Specify "1" for auth 1.0, or
# "2" for auth 2.0 (string value)
#backup_swift_auth_version=1

# Swift tenant/account name. Required when connecting to an
# auth 2.0 system (string value)
#backup_swift_tenant=<None>

# Swift user name (string value)
#backup_swift_user=<None>

# Swift key for authentication (string value)
#backup_swift_key=<None>

# The default Swift container to use (string value)
#backup_swift_container=volumebackups

# The size in bytes of Swift backup objects (integer value)
#backup_swift_object_size=52428800

# The size in bytes that changes are tracked for incremental
# backups. backup_swift_object_size has to be multiple of
# backup_swift_block_size. (integer value)
#backup_swift_block_size=32768

# The number of retries to make for Swift operations (integer
# value)
#backup_swift_retry_attempts=3

# The backoff time in seconds between Swift retries (integer
# value)
#backup_swift_retry_backoff=2

# Enable or Disable the timer to send the periodic progress
# notifications to Ceilometer when backing up the volume to
# the Swift backend storage. The default value is True to
# enable the timer. (boolean value)
#backup_swift_enable_progress_timer=true

#
# Options defined in cinder.backup.drivers.tsm
#

# Volume prefix for the backup id when backing up to TSM
# (string value)
#backup_tsm_volume_prefix=backup

# TSM password for the running username (string value)
#backup_tsm_password=password

# Enable or Disable compression for backups (boolean value)
#backup_tsm_compression=true

#

```

```
# Options defined in cinder.backup.manager
#

# Driver to use for backups. (string value)
# Deprecated group/name - [DEFAULT]/backup_service
#backup_driver=cinder.backup.drivers.swift


#
# Options defined in cinder.cmd.volume
#

# Backend override of host value. (string value)
# Deprecated group/name - [DEFAULT]/host
#backend_host=<None>


#
# Options defined in cinder.cmd.volume_usage_audit
#

# If this option is specified then the start time specified is
# used instead of the start time of the last completed audit
# period. (string value)
#start_time=<None>

# If this option is specified then the end time specified is
# used instead of the end time of the last completed audit
# period. (string value)
#end_time=<None>

# Send the volume and snapshot create and delete notifications
# generated in the specified period. (boolean value)
#send_actions=false


#
# Options defined in cinder.common.config
#

# File name for the paste.deploy config for cinder-api (string
# value)
#api_paste_config=api-paste.ini

# Top-level directory for maintaining cinder's state (string
# value)
# Deprecated group/name - [DEFAULT]/pybasedir
#state_path=/var/lib/cinder

# IP address of this host (string value)
#my_ip=10.0.0.1

# Default glance host name or IP (string value)
#glance_host=$my_ip

# Default glance port (integer value)
```

```

#glance_port=9292

# A list of the glance API servers available to cinder
# ([hostname|ip]:port) (list value)
#glance_api_servers=$glance_host:$glance_port

# Version of the glance API to use (integer value)
#glance_api_version=1

# Number retries when downloading an image from glance
# (integer value)
#glance_num_retries=0

# Allow to perform insecure SSL (https) requests to glance
# (boolean value)
#glance_api_insecure=false

# Enables or disables negotiation of SSL layer compression. In
# some cases disabling compression can improve data
# throughput, such as when high network bandwidth is available
# and you use compressed image formats like qcow2. (boolean
# value)
#glance_api_ssl_compression=false

# Location of ca certificates file to use for glance client
# requests. (string value)
#glance_ca_certificates_file=<None>

# http/https timeout value for glance operations. If no value
# (None) is supplied here, the glanceclient default value is
# used. (integer value)
#glance_request_timeout=<None>

# The topic that scheduler nodes listen on (string value)
#scheduler_topic=cinder-scheduler

# The topic that volume nodes listen on (string value)
#volume_topic=cinder-volume

# The topic that volume backup nodes listen on (string value)
#backup_topic=cinder-backup

# DEPRECATED: Deploy v1 of the Cinder API. (boolean value)
#enable_v1_api=true

# Deploy v2 of the Cinder API. (boolean value)
#enable_v2_api=true

# Enables or disables rate limit of the API. (boolean value)
#api_rate_limit=true

# Specify list of extensions to load when using
# osapi_volume_extension option with
# cinder.api.contrib.select_extensions (list value)
#osapi_volume_ext_list=

```

```
# osapi volume extension to load (multi valued)
#osapi_volume_extension=cinder.api.contrib.standard_extensions

# Full class name for the Manager for volume (string value)
#volume_manager=cinder.volume.manager.VolumeManager

# Full class name for the Manager for volume backup (string
# value)
#backup_manager=cinder.backup.manager.BackupManager

# Full class name for the Manager for scheduler (string value)
#scheduler_manager=cinder.scheduler.manager.SchedulerManager

# Name of this node. This can be an opaque identifier. It is
# not necessarily a host name, FQDN, or IP address. (string
# value)
#host=cinder

# Availability zone of this node (string value)
#storage_availability_zone=nova

# Default availability zone for new volumes. If not set, the
# storage_availability_zone option value is used as the
# default for new volumes. (string value)
#default_availability_zone=<None>

# Default volume type to use (string value)
#default_volume_type=<None>

# Time period for which to generate volume usages. The options
# are hour, day, month, or year. (string value)
#volume_usage_audit_period=month

# Path to the rootwrap configuration file to use for running
# commands as root (string value)
#rootwrap_config=/etc/cinder/rootwrap.conf

# Enable monkey patching (boolean value)
#monkey_patch=false

# List of modules/decorators to monkey patch (list value)
#monkey_patch_modules=

# Maximum time since last check-in for a service to be
# considered up (integer value)
#service_down_time=60

# The full class name of the volume API class to use (string
# value)
#volume_api_class=cinder.volume.api.API

# The full class name of the volume backup API class (string
# value)
#backup_api_class=cinder.backup.api.API

# The strategy to use for auth. Supports noauth, keystone, and
```



```

# deprecated. (string value)
#auth_strategy=noauth

# A list of backend names to use. These backend names should
# be backed by a unique [CONFIG] group with its options (list
# value)
#enabled_backends=<None>

# Whether snapshots count against gigabyte quota (boolean
# value)
#no_snapshot_gb_quota=false

# The full class name of the volume transfer API class (string
# value)
#transfer_api_class=cinder.transfer.api.API

# The full class name of the volume replication API class
# (string value)
#replication_api_class=cinder.replication.api.API

# The full class name of the consistencygroup API class
# (string value)
#consistencygroup_api_class=cinder.consistencygroup.api.API

# OpenStack privileged account username. Used for requests to
# other services (such as Nova) that require an account with
# special rights. (string value)
#os_privileged_user_name=<None>

# Password associated with the OpenStack privileged account.
# (string value)
#os_privileged_user_password=<None>

# Tenant name associated with the OpenStack privileged
# account. (string value)
#os_privileged_user_tenant=<None>

#
# Options defined in cinder.compute
#

# The full class name of the compute API class to use (string
# value)
#compute_api_class=cinder.compute.nova.API

#
# Options defined in cinder.compute.nova
#

# Match this value when searching for nova in the service
# catalog. Format is: separated values of the form:
# <service_type>:<service_name>:<endpoint_type> (string value)
#nova_catalog_info=compute:Compute Service:publicURL

```

```
# Same as nova_catalog_info, but for admin endpoint. (string
# value)
#nova_catalog_admin_info=compute:Compute Service:adminURL

# Override service catalog lookup with template for nova
# endpoint e.g. http://localhost:8774/v2/%(project_id)s
# (string value)
#nova_endpoint_template=<None>

# Same as nova_endpoint_template, but for admin endpoint.
# (string value)
#nova_endpoint_admin_template=<None>

# Region name of this node (string value)
#os_region_name=<None>

# Location of ca certificates file to use for nova client
# requests. (string value)
#nova_ca_certificates_file=<None>

# Allow to perform insecure SSL requests to nova (boolean
# value)
#nova_api_insecure=false

#
# Options defined in cinder.db.api
#

# Services to be added to the available pool on create
# (boolean value)
#enable_new_services=true

# Template string to be used to generate volume names (string
# value)
#volume_name_template=volume-%s

# Template string to be used to generate snapshot names
# (string value)
#snapshot_name_template=snapshot-%s

# Template string to be used to generate backup names (string
# value)
#backup_name_template=backup-%s

#
# Options defined in cinder.db.base
#

# Driver to use for database access (string value)
#db_driver=cinder.db

#
# Options defined in cinder.image.glance
```

```

#

# Default core properties of image (list value)
#glance_core_properties=checksum,container_format,disk_format,image_name,i
mage_id,min_disk,min_ram,name,size

# A list of url schemes that can be downloaded directly via
# the direct_url. Currently supported schemes: [file]. (list
# value)
#allowed_direct_url_schemes=

#

# Options defined in cinder.image.image_utils
#

# Directory used for temporary storage during image conversion
# (string value)
#image_conversion_dir=$state_path/conversion

#

# Options defined in cinder.openstack.common.eventlet_backdoor
#

# Enable eventlet backdoor. Acceptable values are 0, <port>,
# and <start>:<end>, where 0 results in listening on a random
# tcp port number; <port> results in listening on the
# specified port number (and not enabling backdoor if that
# port is in use); and <start>:<end> results in listening on
# the smallest unused port number within the specified range
# of port numbers. The chosen port is displayed in the
# service's log file. (string value)
#backdoor_port=<None>

#

# Options defined in cinder.openstack.common.periodic_task
#

# Some periodic tasks can be run in a separate process. Should
# we run them here? (boolean value)
#run_external_periodic_tasks=true

#

# Options defined in cinder.openstack.common.policy
#

# The JSON file that defines policies. (string value)
#policy_file=policy.json

# Default rule. Enforced when a requested rule is not found.
# (string value)
#policy_default_rule=default

```

```
# Directories where policy configuration files are stored.
# They can be relative to any directory in the search path
# defined by the config_dir option, or absolute paths. The
# file defined by policy_file must exist for these directories
# to be searched. Missing or empty directories are ignored.
# (multi valued)
#policy_dirs=policy.d

#
# Options defined in cinder.openstack.common.versionutils
#

# Enables or disables fatal status of deprecations. (boolean
# value)
#fatal_deprecations=false

#
# Options defined in cinder.scheduler.driver
#

# The scheduler host manager class to use (string value)
#scheduler_host_manager=cinder.scheduler.host_manager.HostManager

# Maximum number of attempts to schedule an volume (integer
# value)
#scheduler_max_attempts=3

#
# Options defined in cinder.scheduler.host_manager
#

# Which filter class names to use for filtering hosts when not
# specified in the request. (list value)
#scheduler_default_filters=AvailabilityZoneFilter,CapacityFilter,Capabilit
iesFilter

# Which weigher class names to use for weighing hosts. (list
# value)
#scheduler_default_weighers=CapacityWeigher

#
# Options defined in cinder.scheduler.manager
#

# Default scheduler driver to use (string value)
#scheduler_driver=cinder.scheduler.filter_scheduler.FilterScheduler

#
# Options defined in cinder.scheduler.scheduler_options
#
```

```

# Absolute path to scheduler configuration JSON file. (string
# value)
#scheduler_json_config_location=

#
# Options defined in cinder.scheduler.simple
#

# This configure option has been deprecated along with the
# SimpleScheduler. New scheduler is able to gather capacity
# information for each host, thus setting the maximum number
# of volume gigabytes for host is no longer needed. It's safe
# to remove this configure from cinder.conf. (integer value)
#max_gigabytes=10000

#
# Options defined in cinder.scheduler.weights.capacity
#

# Multiplier used for weighing volume capacity. Negative
# numbers mean to stack vs spread. (floating point value)
#capacity_weight_multiplier=1.0

# Multiplier used for weighing volume capacity. Negative
# numbers mean to stack vs spread. (floating point value)
#allocated_capacity_weight_multiplier=-1.0

#
# Options defined in cinder.scheduler.weights.volume_number
#

# Multiplier used for weighing volume number. Negative numbers
# mean to spread vs stack. (floating point value)
#volume_number_multiplier=-1.0

#
# Options defined in cinder.transfer.api
#

# The number of characters in the salt. (integer value)
#volume_transfer_salt_length=8

# The number of characters in the autogenerated auth key.
# (integer value)
#volume_transfer_key_length=16

#
# Options defined in cinder.volume.api
#

# Cache volume availability zones in memory for the provided

```

```
# duration in seconds (integer value)
#az_cache_duration=3600

# Create volume from snapshot at the host where snapshot
# resides (boolean value)
#snapshot_same_host=true

# Ensure that the new volumes are the same AZ as snapshot or
# source volume (boolean value)
#cloned_volume_same_az=true

#
# Options defined in cinder.volume.driver
#

# The maximum number of times to rescan iSER targetto find
# volume (integer value)
#num_iser_scan_tries=3

# This option is deprecated and unused. It will be removed in
# the Liberty release. (integer value)
#iser_num_targets=<None>

# Prefix for iSER volumes (string value)
#iser_target_prefix=iqn.2010-10.org.openstack:

# The IP address that the iSER daemon is listening on (string
# value)
#iser_ip_address=$my_ip

# The port that the iSER daemon is listening on (integer
# value)
#iser_port=3260

# The name of the iSER target user-land tool to use (string
# value)
#iser_helper=tgtadm

# Number of times to attempt to run flakey shell commands
# (integer value)
#num_shell_tries=3

# The percentage of backend capacity is reserved (integer
# value)
#reserved_percentage=0

# This option is deprecated and unused. It will be removed in
# the Liberty release. (integer value)
#iscsi_num_targets=<None>

# Prefix for iSCSI volumes (string value)
#iscsi_target_prefix=iqn.2010-10.org.openstack:

# The IP address that the iSCSI daemon is listening on (string
# value)
```

```

#iscsi_ip_address=$my_ip

# The list of secondary IP addresses of the iSCSI daemon (list
# value)
#iscsi_secondary_ip_addresses=

# The port that the iSCSI daemon is listening on (integer
# value)
#iscsi_port=3260

# The maximum number of times to rescan targets to find volume
# (integer value)
# Deprecated group/name - [DEFAULT]/num_iscsi_scan_tries
#num_volume_device_scan_tries=3

# The backend name for a given driver implementation (string
# value)
#volume_backend_name=<None>

# Do we attach/detach volumes in cinder using multipath for
# volume to image and image to volume transfers? (boolean
# value)
#use_multipath_for_image_xfer=false

# If this is set to True, attachment of volumes for image
# transfer will be aborted when multipathd is not running.
# Otherwise, it will fallback to single path. (boolean value)
#enforce_multipath_for_image_xfer=false

# Method used to wipe old volumes (string value)
#volume_clear=zero

# Size in MiB to wipe at start of old volumes. 0 => all
# (integer value)
#volume_clear_size=0

# The flag to pass to ionice to alter the i/o priority of the
# process used to zero a volume after deletion, for example
# "-c3" for idle only priority. (string value)
#volume_clear_ionice=<None>

# iSCSI target user-land tool to use. tgtadm is default, use
# lioadm for LIO iSCSI support, scstadmin for SCST target
# support, iseradm for the ISER protocol, ietadm for iSCSI
# Enterprise Target, iscsictl for Chelsio iSCSI Target or fake
# for testing. (string value)
#iscsi_helper=tgtadm

# Volume configuration file storage directory (string value)
#volumes_dir=$state_path/volumes

# IET configuration file (string value)
#iet_conf=/etc/iet/ietd.conf

# Chiscsi (CXT) global defaults configuration file (string
# value)

```

```
#chiscsi_conf=/etc/chelsio-iscsi/chiscsi.conf

# This option is deprecated and unused. It will be removed in
# the next release. (string value)
#lio_initiator_iqns=

# Sets the behavior of the iSCSI target to either perform
# blockio or fileio optionally, auto can be set and Cinder
# will autodetect type of backing device (string value)
#iscsi_iotype=fileio

# The default block size used when copying/clearing volumes
# (string value)
#volume_dd_blocksize=1M

# The blkio cgroup name to be used to limit bandwidth of
# volume copy (string value)
#volume_copy_blkio_cgroup_name=cinder-volume-copy

# The upper limit of bandwidth of volume copy. 0 => unlimited
# (integer value)
#volume_copy_bps_limit=0

# Sets the behavior of the iSCSI target to either perform
# write-back(on) or write-through(off). This parameter is
# valid if iscsi_helper is set to tgtadm or iseradm. (string
# value)
#iscsi_write_cache=on

# Determines the iSCSI protocol for new iSCSI volumes, created
# with tgtadm or lioadm target helpers. In order to enable
# RDMA, this parameter should be set with the value "iser".
# The supported iSCSI protocol values are "iscsi" and "iser".
# (string value)
#iscsi_protocol=iscsi

# The path to the client certificate key for verification, if
# the driver supports it. (string value)
#driver_client_cert_key=<None>

# The path to the client certificate for verification, if the
# driver supports it. (string value)
#driver_client_cert=<None>

# Tell driver to use SSL for connection to backend storage if
# the driver supports it. (boolean value)
#driver_use_ssl=false

# Float representation of the over subscription ratio when
# thin provisioning is involved. Default ratio is 20.0,
# meaning provisioned capacity can be 20 times of the total
# physical capacity. If the ratio is 10.5, it means
# provisioned capacity can be 10.5 times of the total physical
# capacity. A ratio of 1.0 means provisioned capacity cannot
# exceed the total physical capacity. A ratio lower than 1.0
# will be ignored and the default value will be used instead.
```



```

# (floating point value)
#max_over_subscription_ratio=20.0

# Certain iSCSI targets have predefined target names, SCST
# target driver uses this name. (string value)
#scst_target_iqn_name=<None>

# SCST target implementation can choose from multiple SCST
# target drivers. (string value)
#scst_target_driver=iscsi

# Option to enable/disable CHAP authentication for targets.
# (boolean value)
# Deprecated group/name - [DEFAULT]/eqlx_use_chap
#use_chap_auth=false

# CHAP user name. (string value)
# Deprecated group/name - [DEFAULT]/eqlx_chap_login
#chap_username=

# Password for specified CHAP account name. (string value)
# Deprecated group/name - [DEFAULT]/eqlx_chap_password
#chap_password=

# Namespace for driver private data values to be saved in.
# (string value)
#driver_data_namespace=<None>

# String representation for an equation that will be used to
# filter hosts. Only used when the driver filter is set to be
# used by the Cinder scheduler. (string value)
#filter_function=<None>

# String representation for an equation that will be used to
# determine the goodness of a host. Only used when using the
# goodness weigher is set to be used by the Cinder scheduler.
# (string value)
#goodness_function=<None>

#
# Options defined in cinder.volume.drivers.block_device
#

# List of all available devices (list value)
#available_devices=

#
# Options defined in cinder.volume.drivers.cloudbyte.options
#

# These values will be used for CloudByte storage's addQos API
# call. (dict value)
#cb_add_qosgroup=latency:15,iops:10,graceallowed:false,iopscontrol:true,me
mlimit:0,throughput:0,tpcontrol:false,networkspeed:0

```

```

# Driver will use this API key to authenticate against the
# CloudByte storage's management interface. (string value)
#cb_apikey=None

# CloudByte storage specific account name. This maps to a
# project name in OpenStack. (string value)
#cb_account_name=None

# This corresponds to the name of Tenant Storage Machine (TSM)
# in CloudByte storage. A volume will be created in this TSM.
# (string value)
#cb_tsm_name=None

# A retry value in seconds. Will be used by the driver to
# check if volume creation was successful in CloudByte
# storage. (integer value)
#cb_confirm_volume_create_retry_interval=5

# Will confirm a successful volume creation in CloudByte
# storage by making this many number of attempts. (integer
# value)
#cb_confirm_volume_create_retries=3

# These values will be used for CloudByte storage's
# createVolume API call. (dict value)
#cb_create_volume=compression:off,deduplication:off,blocklength:512B,sync:
always,protocoltype:ISCSI,recordsize:16k

#
# Options defined in cinder.volume.drivers.datera
#

# DEPRECATED: This will be removed in the Liberty release. Use
# san_login and san_password instead. This directly sets the
# Datera API token. (string value)
#datera_api_token=<None>

# Datera API port. (string value)
#datera_api_port=7717

# Datera API version. (string value)
#datera_api_version=1

# Number of replicas to create of an inode. (string value)
#datera_num_replicas=3

#
# Options defined in cinder.volume.drivers.dell.dell_storagecenter_common
#

# Storage Center System Serial Number (integer value)
#dell_sc_ssn=64702

```

```

# Dell API port (integer value)
#dell_sc_api_port=3033

# Name of the server folder to use on the Storage Center
# (string value)
#dell_sc_server_folder=openstack

# Name of the volume folder to use on the Storage Center
# (string value)
#dell_sc_volume_folder=openstack

#
# Options defined in cinder.volume.drivers.emc.emc_vmax_common
#

# use this file for cinder emc plugin config data (string
# value)
#cinder_emc_config_file=/etc/cinder/cinder_emc_config.xml

#
# Options defined in cinder.volume.drivers.emc.emc_vnx_cli
#

# VNX authentication scope type. (string value)
#storage_vnx_authentication_type=global

# Directory path that contains the VNX security file. Make
# sure the security file is generated first. (string value)
#storage_vnx_security_file_dir=<None>

# Naviseccli Path. (string value)
#naviseccli_path=

# Storage pool name. (string value)
#storage_vnx_pool_name=<None>

# VNX secondary SP IP Address. (string value)
#san_secondary_ip=<None>

# Default timeout for CLI operations in minutes. For example,
# LUN migration is a typical long running operation, which
# depends on the LUN size and the load of the array. An upper
# bound in the specific deployment can be set to avoid
# unnecessary long wait. By default, it is 365 days long.
# (integer value)
#default_timeout=525600

# Default max number of LUNs in a storage group. By default,
# the value is 255. (integer value)
#max_luns_per_storage_group=255

# To destroy storage group when the last LUN is removed from
# it. By default, the value is False. (boolean value)
#destroy_empty_storage_group=false

```

```
# Mapping between hostname and its iSCSI initiator IP
# addresses. (string value)
#iscsi_initiators=

# Automatically register initiators. By default, the value is
# False. (boolean value)
#initiator_auto_registration=false

# Automatically deregister initiators after the related
# storage group is destroyed. By default, the value is False.
# (boolean value)
#initiator_auto_deregistration=false

# Report free_capacity_gb as 0 when the limit to maximum
# number of pool LUNs is reached. By default, the value is
# False. (boolean value)
#check_max_pool_luns_threshold=false

# Delete a LUN even if it is in Storage Groups. (boolean
# value)
#force_delete_lun_in_storagegroup=false

#
# Options defined in cinder.volume.drivers.emc.xtremio
#

# XMS cluster id in multi-cluster environment (string value)
#xtremio_cluster_name=

#
# Options defined in cinder.volume.drivers.eqlx
#

# Group name to use for creating volumes. Defaults to
# "group-0". (string value)
#eqlx_group_name=group-0

# Timeout for the Group Manager cli command execution. Default
# is 30. (integer value)
#eqlx_cli_timeout=30

# Maximum retry count for reconnection. Default is 5. (integer
# value)
#eqlx_cli_max_retries=5

# Use CHAP authentication for targets. Note that this option
# is deprecated in favour of "use_chap_auth" as specified in
# cinder/volume/driver.py and will be removed in next release.
# (boolean value)
#eqlx_use_chap=false

# Existing CHAP account name. Note that this option is
# deprecated in favour of "chap_username" as specified in
```

```

# cinder/volume/driver.py and will be removed in next release.
# (string value)
#eqlx_chap_login=admin

# Password for specified CHAP account name. Note that this
# option is deprecated in favour of "chap_password" as
# specified in cinder/volume/driver.py and will be removed in
# the next release (string value)
#eqlx_chap_password=password

# Pool in which volumes will be created. Defaults to
# "default". (string value)
#eqlx_pool=default

#
# Options defined in cinder.volume.drivers.glusterfs
#

# File with the list of available gluster shares (string
# value)
#glusterfs_shares_config=/etc/cinder/glusterfs_shares

# Create volumes as sparsed files which take no space.If set
# to False volume is created as regular file.In such case
# volume creation takes a lot of time. (boolean value)
#glusterfs_sparsed_volumes=true

# Create volumes as QCOW2 files rather than raw files.
# (boolean value)
#glusterfs_qcow2_volumes=false

# Base dir containing mount points for gluster shares. (string
# value)
#glusterfs_mount_point_base=$state_path/mnt

#
# Options defined in cinder.volume.drivers.hds.hds
#

# The configuration file for the Cinder HDS driver for HUS
# (string value)
#hds_cinder_config_file=/opt/hds/hus/cinder_hus_conf.xml

#
# Options defined in cinder.volume.drivers.hds.iscsi
#

# Configuration file for HDS iSCSI cinder plugin (string
# value)
#hds_hnas_iscsi_config_file=/opt/hds/hnas/cinder_iscsi_conf.xml

#

```

```
# Options defined in cinder.volume.drivers.hds.nfs
#

# Configuration file for HDS NFS cinder plugin (string value)
#hds_hnas_nfs_config_file=/opt/hds/hnas/cinder_nfs_conf.xml

#
# Options defined in cinder.volume.drivers.hitachi.hbsd_common
#

# Serial number of storage system (string value)
#hitachi_serial_number=<None>

# Name of an array unit (string value)
#hitachi_unit_name=<None>

# Pool ID of storage system (integer value)
#hitachi_pool_id=<None>

# Thin pool ID of storage system (integer value)
#hitachi_thin_pool_id=<None>

# Range of logical device of storage system (string value)
#hitachi_ldev_range=<None>

# Default copy method of storage system (string value)
#hitachi_default_copy_method=FULL

# Copy speed of storage system (integer value)
#hitachi_copy_speed=3

# Interval to check copy (integer value)
#hitachi_copy_check_interval=3

# Interval to check copy asynchronously (integer value)
#hitachi_async_copy_check_interval=10

# Control port names for HostGroup or iSCSI Target (string
# value)
#hitachi_target_ports=<None>

# Range of group number (string value)
#hitachi_group_range=<None>

# Request for creating HostGroup or iSCSI Target (boolean
# value)
#hitachi_group_request=false

#
# Options defined in cinder.volume.drivers.hitachi.hbsd_fc
#

# Request for FC Zone creating HostGroup (boolean value)
#hitachi_zoning_request=false
```

```

#
# Options defined in cinder.volume.drivers.hitachi.hbsd_horcm
#

# Instance numbers for HORCM (string value)
#hitachi_horcm_numbers=200,201

# Username of storage system for HORCM (string value)
#hitachi_horcm_user=<None>

# Password of storage system for HORCM (string value)
#hitachi_horcm_password=<None>

# Add to HORCM configuration (boolean value)
#hitachi_horcm_add_conf=true


#
# Options defined in cinder.volume.drivers.hitachi.hbsd_iscsi
#

# Add CHAP user (boolean value)
#hitachi_add_chap_user=false

# iSCSI authentication method (string value)
#hitachi_auth_method=<None>

# iSCSI authentication username (string value)
#hitachi_auth_user=HBSD-CHAP-user

# iSCSI authentication password (string value)
#hitachi_auth_password=HBSD-CHAP-password


#
# Options defined in cinder.volume.drivers.huawei
#

# The configuration file for the Cinder Huawei driver (string
# value)
#cinder_huawei_conf_file=/etc/cinder/cinder_huawei_conf.xml


#
# Options defined in cinder.volume.drivers.ibm.flashsystem
#

# Connection protocol should be FC. (string value)
#flashsystem_connection_protocol=FC

# Connect with multipath (FC only). (boolean value)
#flashsystem_multipath_enabled=false

# Allows vdisk to multi host mapping. (boolean value)

```

```
#flashsystem_multihostmap_enabled=true

#
# Options defined in cinder.volume.drivers.ibm.gpfs
#

# Specifies the path of the GPFS directory where Block Storage
# volume and snapshot files are stored. (string value)
#gpfs_mount_point_base=<None>

# Specifies the path of the Image service repository in GPFS.
# Leave undefined if not storing images in GPFS. (string
# value)
#gpfs_images_dir=<None>

# Specifies the type of image copy to be used. Set this when
# the Image service repository also uses GPFS so that image
# files can be transferred efficiently from the Image service
# to the Block Storage service. There are two valid values:
# "copy" specifies that a full copy of the image is made;
# "copy_on_write" specifies that copy-on-write optimization
# strategy is used and unmodified blocks of the image file are
# shared efficiently. (string value)
#gpfs_images_share_mode=<None>

# Specifies an upper limit on the number of indirections
# required to reach a specific block due to snapshots or
# clones. A lengthy chain of copy-on-write snapshots or
# clones can have a negative impact on performance, but
# improves space utilization. 0 indicates unlimited clone
# depth. (integer value)
#gpfs_max_clone_depth=0

# Specifies that volumes are created as sparse files which
# initially consume no space. If set to False, the volume is
# created as a fully allocated file, in which case, creation
# may take a significantly longer time. (boolean value)
#gpfs_sparse_volumes=true

# Specifies the storage pool that volumes are assigned to. By
# default, the system storage pool is used. (string value)
#gpfs_storage_pool=system

#
# Options defined in cinder.volume.drivers.ibm.ibmnas
#

# IBMNAS platform type to be used as backend storage; valid
# values are - v7ku : for using IBM Storwize V7000 Unified,
# sonas : for using IBM Scale Out NAS, gpfs-nas : for using
# NFS based IBM GPFS deployments. (string value)
#ibmnas_platform_type=v7ku
```



```

#
# Options defined in cinder.volume.drivers.ibm.storwize_svc
#

# Storage system storage pool for volumes (string value)
#storwize_svc_volpool_name=volpool

# Storage system space-efficiency parameter for volumes
# (percentage) (integer value)
#storwize_svc_vol_rsize=2

# Storage system threshold for volume capacity warnings
# (percentage) (integer value)
#storwize_svc_vol_warning=0

# Storage system autoexpand parameter for volumes (True/False)
# (boolean value)
#storwize_svc_vol_autoexpand=true

# Storage system grain size parameter for volumes
# (32/64/128/256) (integer value)
#storwize_svc_vol_grainsize=256

# Storage system compression option for volumes (boolean
# value)
#storwize_svc_vol_compression=false

# Enable Easy Tier for volumes (boolean value)
#storwize_svc_vol_easytier=true

# The I/O group in which to allocate volumes (integer value)
#storwize_svc_vol_iogrp=0

# Maximum number of seconds to wait for FlashCopy to be
# prepared. Maximum value is 600 seconds (10 minutes) (integer
# value)
#storwize_svc_flashcopy_timeout=120

# Connection protocol (iSCSI/FC) (string value)
#storwize_svc_connection_protocol=iSCSI

# Configure CHAP authentication for iSCSI connections
# (Default: Enabled) (boolean value)
#storwize_svc_iscsi_chap_enabled=true

# Connect with multipath (FC only; iSCSI multipath is
# controlled by Nova) (boolean value)
#storwize_svc_multipath_enabled=false

# Allows vdisk to multi host mapping (boolean value)
#storwize_svc_multihostmap_enabled=true

# Indicate whether svc driver is compatible for NPIV setup. If
# it is compatible, it will allow no wwpns being returned on
# get_conn_fc_wwpns during initialize_connection (boolean
# value)

```

```

#storwize_svc_npiv_compatibility_mode=false

# Allow tenants to specify QOS on create (boolean value)
#storwize_svc_allow_tenant_qos=false

# If operating in stretched cluster mode, specify the name of
# the pool in which mirrored copies are stored.Example:
# "pool2" (string value)
#storwize_svc_stretched_cluster_partner=<None>

#
# Options defined in cinder.volume.drivers.ibm.xiv_ds8k
#

# Proxy driver that connects to the IBM Storage Array (string
# value)
#xiv_ds8k_proxy=xiv_ds8k_openstack.nova_proxy.XIVDS8KNovaProxy

# Connection type to the IBM Storage Array (string value)
#xiv_ds8k_connection_type=iscsi

# CHAP authentication mode, effective only for iscsi
# (disabled|enabled) (string value)
#xiv_chap=disabled

#
# Options defined in cinder.volume.drivers.lvm
#

# Name for the VG that will contain exported volumes (string
# value)
#volume_group=cinder-volumes

# If >0, create LVs with multiple mirrors. Note that this
# requires lvm_mirrors + 2 PVs with available space (integer
# value)
#lvm_mirrors=0

# Type of LVM volumes to deploy (string value)
#lvm_type=default

# LVM conf file to use for the LVM driver in Cinder; this
# setting is ignored if the specified file does not exist (You
# can also specify 'None' to not use a conf file even if one
# exists). (string value)
#lvm_conf_file=/etc/cinder/lvm.conf

#
# Options defined in cinder.volume.drivers.netapp.options
#

# The vFiler unit on which provisioning of block storage
# volumes will be done. This option is only used by the driver

```

```

# when connecting to an instance with a storage family of Data
# ONTAP operating in 7-Mode. Only use this option when
# utilizing the MultiStore feature on the NetApp storage
# system. (string value)
#netapp_vfiler=<None>

# The name of the config.conf stanza for a Data ONTAP (7-mode)
# HA partner. This option is only used by the driver when
# connecting to an instance with a storage family of Data
# ONTAP operating in 7-Mode, and it is required if the storage
# protocol selected is FC. (string value)
#netapp_partner_backend_name=<None>

# Administrative user account name used to access the storage
# system or proxy server. (string value)
#netapp_login=<None>

# Password for the administrative user account specified in
# the netapp_login option. (string value)
#netapp_password=<None>

# This option specifies the virtual storage server (Vserver)
# name on the storage cluster on which provisioning of block
# storage volumes should occur. (string value)
#netapp_vserver=<None>

# The hostname (or IP address) for the storage system or proxy
# server. (string value)
#netapp_server_hostname=<None>

# The TCP port to use for communication with the storage
# system or proxy server. If not specified, Data ONTAP drivers
# will use 80 for HTTP and 443 for HTTPS; E-Series will use
# 8080 for HTTP and 8443 for HTTPS. (integer value)
#netapp_server_port=<None>

# This option is used to specify the path to the E-Series
# proxy application on a proxy server. The value is combined
# with the value of the netapp_transport_type,
# netapp_server_hostname, and netapp_server_port options to
# create the URL used by the driver to connect to the proxy
# application. (string value)
#netapp_webservice_path=/devmgr/v2

# This option is only utilized when the storage family is
# configured to eseries. This option is used to restrict
# provisioning to the specified controllers. Specify the value
# of this option to be a comma separated list of controller
# hostnames or IP addresses to be used for provisioning.
# (string value)
#netapp_controller_ips=<None>

# Password for the NetApp E-Series storage array. (string
# value)
#netapp_sa_password=<None>

```

```
# This option is used to restrict provisioning to the
# specified storage pools. Only dynamic disk pools are
# currently supported. Specify the value of this option to be
# a comma separated list of disk pool names to be used for
# provisioning. (string value)
#netapp_storage_pools=<None>

# This option is used to define how the controllers in the
# E-Series storage array will work with the particular
# operating system on the hosts that are connected to it.
# (string value)
#netapp_eseries_host_type=linux_dm_mp

# If the percentage of available space for an NFS share has
# dropped below the value specified by this option, the NFS
# image cache will be cleaned. (integer value)
#thres_avl_size_perc_start=20

# When the percentage of available space on an NFS share has
# reached the percentage specified by this option, the driver
# will stop clearing files from the NFS image cache that have
# not been accessed in the last M minutes, where M is the
# value of the expiry_thres_minutes configuration option.
# (integer value)
#thres_avl_size_perc_stop=60

# This option specifies the threshold for last access time for
# images in the NFS image cache. When a cache cleaning cycle
# begins, images in the cache that have not been accessed in
# the last M minutes, where M is the value of this parameter,
# will be deleted from the cache to create free space on the
# NFS share. (integer value)
#expiry_thres_minutes=720

# This option specifies the path of the NetApp copy offload
# tool binary. Ensure that the binary has execute permissions
# set which allow the effective user of the cinder-volume
# process to execute the file. (string value)
#netapp_copyoffload_tool_path=<None>

# The quantity to be multiplied by the requested volume size
# to ensure enough space is available on the virtual storage
# server (Vserver) to fulfill the volume creation request.
# (floating point value)
#netapp_size_multiplier=1.2

# This option is only utilized when the storage protocol is
# configured to use iSCSI or FC. This option is used to
# restrict provisioning to the specified controller volumes.
# Specify the value of this option to be a comma separated
# list of NetApp controller volume names to be used for
# provisioning. (string value)
#netapp_volume_list=<None>

# The storage family type used on the storage system; valid
# values are ontap_7mode for using Data ONTAP operating in
```

```

# 7-Mode, ontap_cluster for using clustered Data ONTAP, or
# eseries for using E-Series. (string value)
#netapp_storage_family=ontap_cluster

# The storage protocol to be used on the data path with the
# storage system. (string value)
#netapp_storage_protocol=<None>

# The transport protocol used when communicating with the
# storage system or proxy server. (string value)
#netapp_transport_type=http

#
# Options defined in cinder.volume.drivers.nfs
#

# File with the list of available nfs shares (string value)
#nfs_shares_config=/etc/cinder/nfs_shares

# Create volumes as sparsed files which take no space.If set
# to False volume is created as regular file.In such case
# volume creation takes a lot of time. (boolean value)
#nfs_sparsed_volumes=true

# Percent of ACTUAL usage of the underlying volume before no
# new volumes can be allocated to the volume destination.
# (floating point value)
#nfs_used_ratio=0.95

# This will compare the allocated to available space on the
# volume destination. If the ratio exceeds this number, the
# destination will no longer be valid. (floating point value)
#nfs_oversub_ratio=1.0

# Base dir containing mount points for nfs shares. (string
# value)
#nfs_mount_point_base=$state_path/mnt

# Mount options passed to the nfs client. See section of the
# nfs man page for details. (string value)
#nfs_mount_options=<None>

# The number of attempts to mount nfs shares before raising an
# error. At least one attempt will be made to mount an nfs
# share, regardless of the value specified. (integer value)
#nfs_mount_attempts=3

#
# Options defined in cinder.volume.drivers.nimble
#

# Nimble Controller pool name (string value)
#nimble_pool_name=default

```

```
# Nimble Subnet Label (string value)
#nimble_subnet_label=*

#
# Options defined in cinder.volume.drivers.openvstorage
#

# Vpool to use for volumes - backend is defined by vpool not
# by us. (string value)
#vpool_name=

#
# Options defined in cinder.volume.drivers.prophetstor.options
#

# DPL pool uuid in which DPL volumes are stored. (string
# value)
#dpl_pool=

# DPL port number. (integer value)
#dpl_port=8357

#
# Options defined in cinder.volume.drivers.pure
#

# REST API authorization token. (string value)
#pure_api_token=<None>

#
# Options defined in cinder.volume.drivers.quobyte
#

# URL to the Quobyte volume e.g., quobyte://<DIR host>/<volume
# name> (string value)
#quobyte_volume_url=<None>

# Path to a Quobyte Client configuration file. (string value)
#quobyte_client_cfg=<None>

# Create volumes as sparse files which take no space. If set
# to False, volume is created as regular file. In such case
# volume creation takes a lot of time. (boolean value)
#quobyte_sparsed_volumes=true

# Create volumes as QCOW2 files rather than raw files.
# (boolean value)
#quobyte_qcow2_volumes=true

# Base dir containing the mount point for the Quobyte volume.
# (string value)
#quobyte_mount_point_base=$state_path/mnt
```

```

#
# Options defined in cinder.volume.drivers.rbd
#

# The RADOS pool where rbd volumes are stored (string value)
#rbd_pool=rbd

# The RADOS client name for accessing rbd volumes - only set
# when using cephx authentication (string value)
#rbd_user=<None>

# Path to the ceph configuration file (string value)
#rbd_ceph_conf=

# Flatten volumes created from snapshots to remove dependency
# from volume to snapshot (boolean value)
#rbd_flatten_volume_from_snapshot=false

# The libvirt uuid of the secret for the rbd_user volumes
# (string value)
#rbd_secret_uuid=<None>

# Directory where temporary image files are stored when the
# volume driver does not write them directly to the volume.
# Warning: this option is now deprecated, please use
# image_conversion_dir instead. (string value)
#volume_tmp_dir=<None>

# Maximum number of nested volume clones that are taken before
# a flatten occurs. Set to 0 to disable cloning. (integer
# value)
#rbd_max_clone_depth=5

# Volumes will be chunked into objects of this size (in
# megabytes). (integer value)
#rbd_store_chunk_size=4

# Timeout value (in seconds) used when connecting to ceph
# cluster. If value < 0, no timeout is set and default
# librados value is used. (integer value)
#rados_connect_timeout=-1


#
# Options defined in cinder.volume.drivers.remotefs
#

# IP address or Hostname of NAS system. (string value)
#nas_ip=

# User name to connect to NAS system. (string value)
#nas_login=admin

# Password to connect to NAS system. (string value)

```

```

#nas_password=

# SSH port to use to connect to NAS system. (integer value)
#nas_ssh_port=22

# Filename of private key to use for SSH authentication.
# (string value)
#nas_private_key=

# Allow network-attached storage systems to operate in a
# secure environment where root level access is not permitted.
# If set to False, access is as the root user and insecure. If
# set to True, access is not as root. If set to auto, a check
# is done to determine if this is a new installation: True is
# used if so, otherwise False. Default is auto. (string value)
#nas_secure_file_operations=auto

# Set more secure file permissions on network-attached storage
# volume files to restrict broad other/world access. If set to
# False, volumes are created with open permissions. If set to
# True, volumes are created with permissions for the cinder
# user and group (660). If set to auto, a check is done to
# determine if this is a new installation: True is used if so,
# otherwise False. Default is auto. (string value)
#nas_secure_file_permissions=auto

# Path to the share to use for storing Cinder volumes. For
# example: "/srv/export1" for an NFS server export available
# at 10.0.5.10:/srv/export1 . (string value)
#nas_share_path=

# Options used to mount the storage backend file system where
# Cinder volumes are stored. (string value)
#nas_mount_options=<None>

#
# Options defined in cinder.volume.drivers.san.hp.hp_3par_common
#

# 3PAR WSAPI Server Url like https://<3par ip>:8080/api/v1
# (string value)
#hp3par_api_url=

# 3PAR Super user username (string value)
#hp3par_username=

# 3PAR Super user password (string value)
#hp3par_password=

# List of the CPG(s) to use for volume creation (list value)
#hp3par_cpg=OpenStack

# The CPG to use for Snapshots for volumes. If empty the
# userCPG will be used. (string value)
#hp3par_cpg_snap=

```



```

# The time in hours to retain a snapshot.  You can't delete it
# before this expires. (string value)
#hp3par_snapshot_retention=

# The time in hours when a snapshot expires  and is deleted.
# This must be larger than expiration (string value)
#hp3par_snapshot_expiration=

# Enable HTTP debugging to 3PAR (boolean value)
#hp3par_debug=false

# List of target iSCSI addresses to use. (list value)
#hp3par_iscsi_ips=

# Enable CHAP authentication for iSCSI connections. (boolean
# value)
#hp3par_iscsi_chap_enabled=false


#
# Options defined in cinder.volume.drivers.san.hp.hp_leftthand_rest_proxy
#

# HP LeftHand WSAPI Server Url like https://<LeftHand
# ip>:8081/lhos (string value)
#hplefthand_api_url=<None>

# HP LeftHand Super user username (string value)
#hplefthand_username=<None>

# HP LeftHand Super user password (string value)
#hplefthand_password=<None>

# HP LeftHand cluster name (string value)
#hplefthand_clustername=<None>

# Configure CHAP authentication for iSCSI connections
# (Default: Disabled) (boolean value)
#hplefthand_iscsi_chap_enabled=false

# Enable HTTP debugging to LeftHand (boolean value)
#hplefthand_debug=false


#
# Options defined in cinder.volume.drivers.san.san
#

# Use thin provisioning for SAN volumes? (boolean value)
#san_thin_provision=true

# IP address of SAN controller (string value)
#san_ip=

# Username for SAN controller (string value)

```

```
#san_login=admin

# Password for SAN controller (string value)
#san_password=

# Filename of private key to use for SSH authentication
# (string value)
#san_private_key=

# Cluster name to use for creating volumes (string value)
#san_clustername=

# SSH port to use with SAN (integer value)
#san_ssh_port=22

# Execute commands locally instead of over SSH; use if the
# volume service is running on the SAN device (boolean value)
#san_is_local=false

# SSH connection timeout in seconds (integer value)
#ssh_conn_timeout=30

# Minimum ssh connections in the pool (integer value)
#ssh_min_pool_conn=1

# Maximum ssh connections in the pool (integer value)
#ssh_max_pool_conn=5

#
# Options defined in cinder.volume.drivers.scality
#

# Path or URL to Scality SOFS configuration file (string
# value)
#scality_sofs_config=<None>

# Base dir where Scality SOFS shall be mounted (string value)
#scality_sofs_mount_point=$state_path/scality

# Path from Scality SOFS root to volume dir (string value)
#scality_sofs_volume_dir=cinder/volumes

#
# Options defined in cinder.volume.drivers.smbfs
#

# File with the list of available smbfs shares. (string value)
#smbfs_shares_config=/etc/cinder/smbfs_shares

# Default format that will be used when creating volumes if no
# volume format is specified. (string value)
#smbfs_default_volume_format=qcow2

# Create volumes as sparsed files which take no space rather
```

```

# than regular files when using raw format, in which case
# volume creation takes lot of time. (boolean value)
#smbfs_sparsed_volumes=true

# Percent of ACTUAL usage of the underlying volume before no
# new volumes can be allocated to the volume destination.
# (floating point value)
#smbfs_used_ratio=0.95

# This will compare the allocated to available space on the
# volume destination. If the ratio exceeds this number, the
# destination will no longer be valid. (floating point value)
#smbfs_oversub_ratio=1.0

# Base dir containing mount points for smbfs shares. (string
# value)
#smbfs_mount_point_base=$state_path/mnt

# Mount options passed to the smbfs client. See mount.cifs man
# page for details. (string value)
#smbfs_mount_options=noperm,file_mode=0775,dir_mode=0775

#
# Options defined in cinder.volume.drivers.solidfire
#

# Set 512 byte emulation on volume creation; (boolean value)
#sf_emulate_512=true

# Allow tenants to specify QOS on create (boolean value)
#sf_allow_tenant_qos=false

# Create SolidFire accounts with this prefix. Any string can
# be used here, but the string "hostname" is special and will
# create a prefix using the cinder node hostname (previous
# default behavior). The default is NO prefix. (string value)
#sf_account_prefix=<None>

# Account name on the SolidFire Cluster to use as owner of
# template/cache volumes (created if does not exist). (string
# value)
#sf_template_account_name=openstack-vtemplate

# Create an internal cache of copy of images when a bootable
# volume is created to eliminate fetch from glance and qemu-
# conversion on subsequent calls. (boolean value)
#sf_allow_template_caching=true

# SolidFire API port. Useful if the device api is behind a
# proxy on a different port. (integer value)
#sf_api_port=443

#
# Options defined in cinder.volume.drivers.srb

```

```
#

# Comma-separated list of REST servers IP to connect to. (eg
# http://IP1/,http://IP2:81/path (string value)
#srb_base_urls=<None>

#
# Options defined in cinder.volume.drivers.violin.v6000_common
#

# IP address or hostname of mg-a (string value)
#gateway_mga=<None>

# IP address or hostname of mg-b (string value)
#gateway_mgb=<None>

# Use igroups to manage targets and initiators (boolean value)
#use_igroups=false

# Global backend request timeout, in seconds (integer value)
#request_timeout=300

#
# Options defined in cinder.volume.drivers.vmware.vmdk
#

# IP address for connecting to VMware ESX/VC server. (string
# value)
#vmware_host_ip=<None>

# Username for authenticating with VMware ESX/VC server.
# (string value)
#vmware_host_username=<None>

# Password for authenticating with VMware ESX/VC server.
# (string value)
#vmware_host_password=<None>

# Optional VIM service WSDL Location e.g
# http://<server>/vimService.wsdl. Optional over-ride to
# default location for bug work-arounds. (string value)
#vmware_wsdl_location=<None>

# Number of times VMware ESX/VC server API must be retried
# upon connection related issues. (integer value)
#vmware_api_retry_count=10

# The interval (in seconds) for polling remote tasks invoked
# on VMware ESX/VC server. (floating point value)
#vmware_task_poll_interval=0.5

# Name for the folder in the VC datacenter that will contain
# cinder volumes. (string value)
#vmware_volume_folder=cinder-volumes
```

```

# Timeout in seconds for VMDK volume transfer between Cinder
# and Glance. (integer value)
#vmware_image_transfer_timeout_secs=7200

# Max number of objects to be retrieved per batch. Query
# results will be obtained in batches from the server and not
# in one shot. Server may still limit the count to something
# less than the configured value. (integer value)
#vmware_max_objects_retrieval=100

# Optional string specifying the VMware VC server version. The
# driver attempts to retrieve the version from VMware VC
# server. Set this configuration only if you want to override
# the VC server version. (string value)
#vmware_host_version=<None>

# Directory where virtual disks are stored during volume
# backup and restore. (string value)
#vmware_tmp_dir=/tmp

#
# Options defined in cinder.volume.drivers.windows.windows
#

# Path to store VHD backed volumes (string value)
#windows_iscsi_lun_path=C:\iSCSIVirtualDisks

#
# Options defined in cinder.volume.drivers.xio
#

# Default storage pool for volumes. (integer value)
#ise_storage_pool=1

# Raid level for ISE volumes. (integer value)
#ise_raid=1

# Number of retries (per port) when establishing connection to
# ISE management port. (integer value)
#ise_connection_retries=5

# Interval (secs) between retries. (integer value)
#ise_retry_interval=1

# Number on retries to get completion status after issuing a
# command to ISE. (integer value)
#ise_completion_retries=30

#
# Options defined in cinder.volume.drivers.zfssa.zfssanfs
#

```

```
# Data path IP address (string value)
#zfssa_data_ip=<None>

# HTTPS port number (string value)
#zfssa_https_port=443

# Options to be passed while mounting share over nfs (string
# value)
#zfssa_nfs_mount_options=

# Storage pool name. (string value)
#zfssa_nfs_pool=

# Project name. (string value)
#zfssa_nfs_project=NFSPProject

# Share name. (string value)
#zfssa_nfs_share=nfs_share

# Data compression. (string value)
#zfssa_nfs_share_compression=off

# Synchronous write bias-latency, throughput. (string value)
#zfssa_nfs_share_logbias=latency

# REST connection timeout. (seconds) (integer value)
#zfssa_rest_timeout=<None>

#
# Options defined in cinder.volume.manager
#

# Driver to use for volume creation (string value)
#volume_driver=cinder.volume.drivers.lvm.LVMISCSIDriver

# Timeout for creating the volume to migrate to when
# performing volume migration (seconds) (integer value)
#migration_create_volume_timeout_secs=300

# Offload pending volume delete during volume service startup
# (boolean value)
#volume_service_inithost_offload=false

# FC Zoning mode configured (string value)
#zoning_mode=none

# User defined capabilities, a JSON formatted string
# specifying key/value pairs. The key/value pairs can be used
# by the CapabilitiesFilter to select between backends when
# requests specify volume types. For example, specifying a
# service level or the geographical location of a backend,
# then creating a volume type to allow the user to select by
# these different properties. (string value)
#extra_capabilities={}
```

[BRCD_FABRIC_EXAMPLE]

```
#
# Options defined in cinder.zonemanager.drivers.brocade.brcd_fabric_opts
#

# Management IP of fabric (string value)
#fc_fabric_address=

# Fabric user ID (string value)
#fc_fabric_user=

# Password for user (string value)
#fc_fabric_password=

# Connecting port (integer value)
#fc_fabric_port=22

# overridden zoning policy (string value)
#zoning_policy=initiator-target

# overridden zoning activation state (boolean value)
#zone_activate=true

# overridden zone name prefix (string value)
#zone_name_prefix=<None>

# Principal switch WWN of the fabric (string value)
#principal_switch_wwn=<None>
```

[CISCO_FABRIC_EXAMPLE]

```
#
# Options defined in cinder.zonemanager.drivers.cisco.cisco_fabric_opts
#

# Management IP of fabric (string value)
#cisco_fc_fabric_address=

# Fabric user ID (string value)
#cisco_fc_fabric_user=

# Password for user (string value)
#cisco_fc_fabric_password=

# Connecting port (integer value)
#cisco_fc_fabric_port=22

# overridden zoning policy (string value)
#cisco_zoning_policy=initiator-target

# overridden zoning activation state (boolean value)
#cisco_zone_activate=true
```

```

# overridden zone name prefix (string value)
#cisco_zone_name_prefix=<None>

# VSAN of the Fabric (string value)
#cisco_zoning_vsan=<None>

[database]

#
# Options defined in oslo.db.concurrency
#

# Enable the experimental use of thread pooling for all DB API
# calls (boolean value)
# Deprecated group/name - [DEFAULT]/dbapi_use_tpool
#use_tpool=false

[fc-zone-manager]

#
# Options defined in
#cinder.zonemanager.drivers.brocade.brcd_fc_zone_driver
#

# Southbound connector for zoning operation (string value)
#brcd_sb_connector=cinder.zonemanager.drivers.brocade.brcd_fc_zone_client_
cli.BrcdFCZoneClientCLI

#
# Options defined in cinder.zonemanager.drivers.cisco.cisco_fc_zone_driver
#

# Southbound connector for zoning operation (string value)
#cisco_sb_connector=cinder.zonemanager.drivers.cisco.cisco_fc_zone_client_
cli.CiscoFCZoneClientCLI

#
# Options defined in cinder.zonemanager.fc_zone_manager
#

# FC Zone Driver responsible for zone management (string
# value)
#zone_driver=cinder.zonemanager.drivers.brocade.brcd_fc_zone_driver.BrcdFC
ZoneDriver

# Zoning policy configured by user; valid values include
# "initiator-target" or "initiator" (string value)
#zoning_policy=initiator-target

# Comma separated list of Fibre Channel fabric names. This
# list of names is used to retrieve other SAN credentials for
# connecting to each SAN fabric (string value)

```



```

#fc_fabric_names=<None>

# FC SAN Lookup Service (string value)
#fc_san_lookup_service=cinder.zonemanager.drivers.brocade.brcd_fc_san_look
up_service.BrcdFCSanLookupService

[keymgr]

#
# Options defined in cinder.keymgr
#

# The full class name of the key manager API class (string
# value)
#api_class=cinder.keymgr.conf_key_mgr.ConfKeyManager

#
# Options defined in cinder.keymgr.conf_key_mgr
#

# Fixed key returned by key manager, specified in hex (string
# value)
#fixed_key=<None>

#
# Options defined in cinder.keymgr.key_mgr
#

# Authentication url for encryption service. (string value)
#encryption_auth_url=http://localhost:5000/v3

# Url for encryption service. (string value)
#encryption_api_url=http://localhost:9311/v1

[keystone_authtoken]

#
# Options defined in keystonemiddleware.auth_token
#

# Complete public Identity API endpoint. (string value)
#auth_uri=<None>

# API version of the admin Identity API endpoint. (string
# value)
#auth_version=<None>

# Do not handle authorization requests within the middleware,
# but delegate the authorization decision to downstream WSGI
# components. (boolean value)
#delay_auth_decision=false

```

```
# Request timeout value for communicating with Identity API
# server. (integer value)
#http_connect_timeout=<None>

# How many times are we trying to reconnect when communicating
# with Identity API Server. (integer value)
#http_request_max_retries=3

# Env key for the swift cache. (string value)
#cache=<None>

# Required if identity server requires client certificate
# (string value)
#certfile=<None>

# Required if identity server requires client certificate
# (string value)
#keyfile=<None>

# A PEM encoded Certificate Authority to use when verifying
# HTTPS connections. Defaults to system CAs. (string value)
#cafile=<None>

# Verify HTTPS connections. (boolean value)
#insecure=false

# Directory used to cache files related to PKI tokens. (string
# value)
#signing_dir=<None>

# Optionally specify a list of memcached server(s) to use for
# caching. If left undefined, tokens will instead be cached
# in-process. (list value)
# Deprecated group/name - [DEFAULT]/memcache_servers
#memcached_servers=<None>

# In order to prevent excessive effort spent validating
# tokens, the middleware caches previously-seen tokens for a
# configurable duration (in seconds). Set to -1 to disable
# caching completely. (integer value)
#token_cache_time=300

# Determines the frequency at which the list of revoked tokens
# is retrieved from the Identity service (in seconds). A high
# number of revocation events combined with a low cache
# duration may significantly reduce performance. (integer
# value)
#revocation_cache_time=10

# (Optional) If defined, indicate whether token data should be
# authenticated or authenticated and encrypted. Acceptable
# values are MAC or ENCRYPT. If MAC, token data is
# authenticated (with HMAC) in the cache. If ENCRYPT, token
# data is encrypted and authenticated in the cache. If the
# value is not one of these options or empty, auth_token will
# raise an exception on initialization. (string value)
```

```

#memcache_security_strategy=<None>

# (Optional, mandatory if memcache_security_strategy is
# defined) This string is used for key derivation. (string
# value)
#memcache_secret_key=<None>

# (Optional) Number of seconds memcached server is considered
# dead before it is tried again. (integer value)
#memcache_pool_dead_retry=300

# (Optional) Maximum total number of open connections to every
# memcached server. (integer value)
#memcache_pool_maxsize=10

# (Optional) Socket timeout in seconds for communicating with
# a memcache server. (integer value)
#memcache_pool_socket_timeout=3

# (Optional) Number of seconds a connection to memcached is
# held unused in the pool before it is closed. (integer value)
#memcache_pool_unused_timeout=60

# (Optional) Number of seconds that an operation will wait to
# get a memcache client connection from the pool. (integer
# value)
#memcache_pool_conn_get_timeout=10

# (Optional) Use the advanced (eventlet safe) memcache client
# pool. The advanced pool will only work under python 2.x.
# (boolean value)
#memcache_use_advanced_pool=false

# (Optional) Indicate whether to set the X-Service-Catalog
# header. If False, middleware will not ask for service
# catalog on token validation and will not set the X-Service-
# Catalog header. (boolean value)
#include_service_catalog=true

# Used to control the use and type of token binding. Can be
# set to: "disabled" to not check token binding. "permissive"
# (default) to validate binding information if the bind type
# is of a form known to the server and ignore it if not.
# "strict" like "permissive" but if the bind type is unknown
# the token will be rejected. "required" any form of token
# binding is needed to be allowed. Finally the name of a
# binding method that must be present in tokens. (string
# value)
#enforce_token_bind=permissive

# If true, the revocation list will be checked for cached
# tokens. This requires that PKI tokens are configured on the
# identity server. (boolean value)
#check_revocations_for_cached=false

# Hash algorithms to use for hashing PKI tokens. This may be a

```

```
# single algorithm or multiple. The algorithms are those
# supported by Python standard hashlib.new(). The hashes will
# be tried in the order given, so put the preferred one first
# for performance. The result of the first hash will be stored
# in the cache. This will typically be set to multiple values
# only while migrating from a less secure algorithm to a more
# secure one. Once all the old tokens are expired this option
# should be set to a single value for better performance.
# (list value)
#hash_algorithms=md5

[matchmaker_redis]

#
# Options defined in oslo.messaging
#

# Host to locate redis. (string value)
#host=127.0.0.1

# Use this port to connect to redis host. (integer value)
#port=6379

# Password for Redis server (optional). (string value)
#password=<None>

[matchmaker_ring]

#
# Options defined in oslo.messaging
#

# Matchmaker ring file (JSON). (string value)
# Deprecated group/name - [DEFAULT]/matchmaker_ringfile
#ringfile=/etc/oslo/matchmaker_ring.json

[oslo_messaging_amqp]

#
# Options defined in oslo.messaging
#

# address prefix used when sending to a specific server
# (string value)
#server_request_prefix=exclusive

# address prefix used when broadcasting to all servers (string
# value)
#broadcast_prefix=broadcast

# address prefix when sending to any server in group (string
# value)
#group_request_prefix=unicast
```

```

# Name for the AMQP container (string value)
#container_name=<None>

# Timeout for inactive connections (in seconds) (integer
# value)
#idle_timeout=0

# Debug: dump AMQP frames to stdout (boolean value)
#trace=false

# CA certificate PEM file for verifying server certificate
# (string value)
#ssl_ca_file=

# Identifying certificate PEM file to present to clients
# (string value)
#ssl_cert_file=

# Private key PEM file used to sign cert_file certificate
# (string value)
#ssl_key_file=

# Password for decrypting ssl_key_file (if encrypted) (string
# value)
#ssl_key_password=<None>

# Accept clients using either SSL or plain TCP (boolean value)
#allow_insecure_clients=false

[oslo_messaging_qpid]

#
# Options defined in oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
#amqp_durable_queues=false

# Auto-delete queues in AMQP. (boolean value)
#amqp_auto_delete=false

# Size of RPC connection pool. (integer value)
#rpc_conn_pool_size=30

# Qpid broker hostname. (string value)
#qpid_hostname=localhost

# Qpid broker port. (integer value)
#qpid_port=5672

# Qpid HA cluster host:port pairs. (list value)
#qpid_hosts=$qpid_hostname:$qpid_port

```

```

# Username for Qpid connection. (string value)
#qpid_username=

# Password for Qpid connection. (string value)
#qpid_password=

# Space separated list of SASL mechanisms to use for auth.
# (string value)
#qpid_sasl_mechanisms=

# Seconds between connection keepalive heartbeats. (integer
# value)
#qpid_heartbeat=60

# Transport to use, either 'tcp' or 'ssl'. (string value)
#qpid_protocol=tcp

# Whether to disable the Nagle algorithm. (boolean value)
#qpid_tcp_nodelay=true

# The number of prefetched messages held by receiver. (integer
# value)
#qpid_receiver_capacity=1

# The qpid topology version to use. Version 1 is what was
# originally used by impl_qpid. Version 2 includes some
# backwards-incompatible changes that allow broker federation
# to work. Users should update to version 2 when they are
# able to take everything down, as it requires a clean break.
# (integer value)
#qpid_topology_version=1

[oslo_messaging_rabbit]

#
# Options defined in oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
#amqp_durable_queues=false

# Auto-delete queues in AMQP. (boolean value)
#amqp_auto_delete=false

# Size of RPC connection pool. (integer value)
#rpc_conn_pool_size=30

# SSL version to use (valid only if SSL enabled). Valid values
# are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may
# be available on some distributions. (string value)
#kombu_ssl_version=

# SSL key file (valid only if SSL enabled). (string value)
#kombu_ssl_keyfile=

```

```

# SSL cert file (valid only if SSL enabled). (string value)
#kombu_ssl_certfile=

# SSL certification authority file (valid only if SSL
# enabled). (string value)
#kombu_ssl_ca_certs=

# How long to wait before reconnecting in response to an AMQP
# consumer cancel notification. (floating point value)
#kombu_reconnect_delay=1.0

# The RabbitMQ broker address where a single node is used.
# (string value)
#rabbit_host=localhost

# The RabbitMQ broker port where a single node is used.
# (integer value)
#rabbit_port=5672

# RabbitMQ HA cluster host:port pairs. (list value)
#rabbit_hosts=$rabbit_host:$rabbit_port

# Connect over SSL for RabbitMQ. (boolean value)
#rabbit_use_ssl=false

# The RabbitMQ userid. (string value)
#rabbit_userid=guest

# The RabbitMQ password. (string value)
#rabbit_password=guest

# The RabbitMQ login method. (string value)
#rabbit_login_method=AMQPLAIN

# The RabbitMQ virtual host. (string value)
#rabbit_virtual_host=/

# How frequently to retry connecting with RabbitMQ. (integer
# value)
#rabbit_retry_interval=1

# How long to backoff for between retries when connecting to
# RabbitMQ. (integer value)
#rabbit_retry_backoff=2

# Maximum number of RabbitMQ connection retries. Default is 0
# (infinite retry count). (integer value)
#rabbit_max_retries=0

# Use HA queues in RabbitMQ (x-ha-policy: all). If you change
# this option, you must wipe the RabbitMQ database. (boolean
# value)
#rabbit_ha_queues=false

# Number of seconds after which the Rabbit broker is

```

```

# considered down if heartbeat's keep-alive fails (0 disables
# the heartbeat, >0 enables it. Enabling heartbeats requires
# kombu>=3.0.7 and amqp>=1.4.0). EXPERIMENTAL (integer value)
#heartbeat_timeout_threshold=0

# How often times during the heartbeat_timeout_threshold we
# check the heartbeat. (integer value)
#heartbeat_rate=2

# Deprecated, use rpc_backend=kombu+memory or rpc_backend=fake
# (boolean value)
#fake_rabbit=false

[profiler]

#
# Options defined in cinder.service
#

# If False fully disable profiling feature. (boolean value)
#profiler_enabled=false

# If False doesn't trace SQL requests. (boolean value)
#trace_sqlalchemy=false

[DEFAULT]

[keystone_auth_token]

#
# From keystonemiddleware.auth_token
#

# Complete public Identity API endpoint. (string value)
#auth_uri = <None>

# API version of the admin Identity API endpoint. (string value)
#auth_version = <None>

# Do not handle authorization requests within the middleware, but
# delegate the authorization decision to downstream WSGI components.
# (boolean value)
#delay_auth_decision = false

# Request timeout value for communicating with Identity API server.
# (integer value)
#http_connect_timeout = <None>

# How many times are we trying to reconnect when communicating with
# Identity API Server. (integer value)
#http_request_max_retries = 3

# Env key for the swift cache. (string value)

```



```

#cache = <None>

# Required if identity server requires client certificate (string
# value)
#certfile = <None>

# Required if identity server requires client certificate (string
# value)
#keyfile = <None>

# A PEM encoded Certificate Authority to use when verifying HTTPS
# connections. Defaults to system CAs. (string value)
#cafile = <None>

# Verify HTTPS connections. (boolean value)
#insecure = false

# Directory used to cache files related to PKI tokens. (string value)
#signing_dir = <None>

# Optionally specify a list of memcached server(s) to use for caching.
# If left undefined, tokens will instead be cached in-process. (list
# value)
# Deprecated group/name - [DEFAULT]/memcache_servers
#memcached_servers = <None>

# In order to prevent excessive effort spent validating tokens, the
# middleware caches previously-seen tokens for a configurable duration
# (in seconds). Set to -1 to disable caching completely. (integer
# value)
#token_cache_time = 300

# Determines the frequency at which the list of revoked tokens is
# retrieved from the Identity service (in seconds). A high number of
# revocation events combined with a low cache duration may
# significantly reduce performance. (integer value)
#revocation_cache_time = 10

# (Optional) If defined, indicate whether token data should be
# authenticated or authenticated and encrypted. Acceptable values are
# MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in
# the cache. If ENCRYPT, token data is encrypted and authenticated in
# the cache. If the value is not one of these options or empty,
# auth_token will raise an exception on initialization. (string value)
#memcache_security_strategy = <None>

# (Optional, mandatory if memcache_security_strategy is defined) This
# string is used for key derivation. (string value)
#memcache_secret_key = <None>

# (Optional) Number of seconds memcached server is considered dead
# before it is tried again. (integer value)
#memcache_pool_dead_retry = 300

# (Optional) Maximum total number of open connections to every
# memcached server. (integer value)

```

```
#memcache_pool_maxsize = 10

# (Optional) Socket timeout in seconds for communicating with a
# memcache server. (integer value)
#memcache_pool_socket_timeout = 3

# (Optional) Number of seconds a connection to memcached is held
# unused in the pool before it is closed. (integer value)
#memcache_pool_unused_timeout = 60

# (Optional) Number of seconds that an operation will wait to get a
# memcache client connection from the pool. (integer value)
#memcache_pool_conn_get_timeout = 10

# (Optional) Use the advanced (eventlet safe) memcache client pool.
# The advanced pool will only work under python 2.x. (boolean value)
#memcache_use_advanced_pool = false

# (Optional) Indicate whether to set the X-Service-Catalog header. If
# False, middleware will not ask for service catalog on token
# validation and will not set the X-Service-Catalog header. (boolean
# value)
#include_service_catalog = true

# Used to control the use and type of token binding. Can be set to:
# "disabled" to not check token binding. "permissive" (default) to
# validate binding information if the bind type is of a form known to
# the server and ignore it if not. "strict" like "permissive" but if
# the bind type is unknown the token will be rejected. "required" any
# form of token binding is needed to be allowed. Finally the name of a
# binding method that must be present in tokens. (string value)
#enforce_token_bind = permissive

# If true, the revocation list will be checked for cached tokens. This
# requires that PKI tokens are configured on the identity server.
# (boolean value)
#check_revocations_for_cached = false

# Hash algorithms to use for hashing PKI tokens. This may be a single
# algorithm or multiple. The algorithms are those supported by Python
# standard hashlib.new(). The hashes will be tried in the order given,
# so put the preferred one first for performance. The result of the
# first hash will be stored in the cache. This will typically be set
# to multiple values only while migrating from a less secure algorithm
# to a more secure one. Once all the old tokens are expired this
# option should be set to a single value for better performance. (list
# value)
#hash_algorithms = md5

# Prefix to prepend at the beginning of the path. Deprecated, use
# identity_uri. (string value)
#auth_admin_prefix =

# Host providing the admin Identity API endpoint. Deprecated, use
# identity_uri. (string value)
#auth_host = 127.0.0.1
```

```

# Port of the admin Identity API endpoint. Deprecated, use
# identity_uri. (integer value)
#auth_port = 35357

# Protocol of the admin Identity API endpoint (http or https).
# Deprecated, use identity_uri. (string value)
#auth_protocol = https

# Complete admin Identity API endpoint. This should specify the
# unversioned root endpoint e.g. https://localhost:35357/ (string
# value)
#identity_uri = <None>

# This option is deprecated and may be removed in a future release.
# Single shared secret with the Keystone configuration used for
# bootstrapping a Keystone installation, or otherwise bypassing the
# normal authentication process. This option should not be used, use
# `admin_user` and `admin_password` instead. (string value)
#admin_token = <None>

# Service username. (string value)
#admin_user = <None>

# Service user password. (string value)
#admin_password = <None>

# Service tenant name. (string value)
#admin_tenant_name = admin

```

2.3.2. api-paste.ini

Use the `api-paste.ini` file to configure the Block Storage API service.

```

#####
# OpenStack #
#####

[composite:osapi_volume]
use = call:cinder.api:root_app_factory
/: apiversions
/v1: openstack_volume_api_v1
/v2: openstack_volume_api_v2

[composite:openstack_volume_api_v1]
use = call:cinder.api.middleware.auth:pipeline_factory
noauth = request_id faultwrap sizelimit osprofiler noauth apiv1
keystone = request_id faultwrap sizelimit osprofiler authtoken
keystonecontext apiv1
keystone_nolimit = request_id faultwrap sizelimit osprofiler authtoken
keystonecontext apiv1

```

```

[composite:openstack_volume_api_v2]
use = call:cinder.api.middleware.auth:pipeline_factory
noauth = request_id faultwrap sizelimit osprofiler noauth apiv2
keystone = request_id faultwrap sizelimit osprofiler authtoken
keystonecontext apiv2
keystone_nolimit = request_id faultwrap sizelimit osprofiler authtoken
keystonecontext apiv2

[filter:request_id]
paste.filter_factory = oslo_middleware.request_id:RequestId.factory

[filter:faultwrap]
paste.filter_factory = cinder.api.middleware.fault:FaultWrapper.factory

[filter:osprofiler]
paste.filter_factory = osprofiler.web:WsgiMiddleware.factory
hmac_keys = SECRET_KEY
enabled = yes

[filter:noauth]
paste.filter_factory = cinder.api.middleware.auth:NoAuthMiddleware.factory

[filter:sizelimit]
paste.filter_factory =
cinder.api.middleware.sizelimit:RequestBodySizeLimiter.factory

[app:apiv1]
paste.app_factory = cinder.api.v1.router:APIRouter.factory

[app:apiv2]
paste.app_factory = cinder.api.v2.router:APIRouter.factory

[pipeline:apiversions]
pipeline = faultwrap osvolumeverSIONapp

[app:osvolumeverSIONapp]
paste.app_factory = cinder.api.versions:Versions.factory

#####
# Shared #
#####

[filter:keystonecontext]
paste.filter_factory =
cinder.api.middleware.auth:CinderKeystoneContext.factory

[filter:authtoken]
paste.filter_factory = keystonemiddleware.auth_token:filter_factory

```

2.3.3. policy.json

The `policy.json` file defines additional access controls that apply to the Block Storage service.

-

```

{
  "context_is_admin": "role:admin",
  "admin_or_owner": "is_admin:True or project_id:%(project_id)s",
  "default": "rule:admin_or_owner",

  "admin_api": "is_admin:True",

  "volume:create": "",
  "volume:delete": "",
  "volume:get": "",
  "volume:get_all": "",
  "volume:get_volume_metadata": "",
  "volume:get_volume_admin_metadata": "rule:admin_api",
  "volume:delete_volume_admin_metadata": "rule:admin_api",
  "volume:update_volume_admin_metadata": "rule:admin_api",
  "volume:get_snapshot": "",
  "volume:get_all_snapshots": "",
  "volume:extend": "",
  "volume:update_readonly_flag": "",
  "volume:retype": "",

  "volume_extension:types_manage": "rule:admin_api",
  "volume_extension:types_extra_specs": "rule:admin_api",
  "volume_extension:volume_type_access": "",
  "volume_extension:volume_type_access:addProjectAccess":
"rule:admin_api",
  "volume_extension:volume_type_access:removeProjectAccess":
"rule:admin_api",
  "volume_extension:volume_type_encryption": "rule:admin_api",
  "volume_extension:volume_encryption_metadata": "rule:admin_or_owner",
  "volume_extension:extended_snapshot_attributes": "",
  "volume_extension:volume_image_metadata": "",

  "volume_extension:quotas:show": "",
  "volume_extension:quotas:update": "rule:admin_api",
  "volume_extension:quota_classes": "",

  "volume_extension:volume_admin_actions:reset_status":
"rule:admin_api",
  "volume_extension:snapshot_admin_actions:reset_status":
"rule:admin_api",
  "volume_extension:backup_admin_actions:reset_status":
"rule:admin_api",
  "volume_extension:volume_admin_actions:force_delete":
"rule:admin_api",
  "volume_extension:volume_admin_actions:force_detach":
"rule:admin_api",
  "volume_extension:snapshot_admin_actions:force_delete":
"rule:admin_api",
  "volume_extension:volume_admin_actions:migrate_volume":
"rule:admin_api",
  "volume_extension:volume_admin_actions:migrate_volume_completion":
"rule:admin_api",

  "volume_extension:volume_host_attribute": "rule:admin_api",
  "volume_extension:volume_tenant_attribute": "rule:admin_or_owner",

```

```

"volume_extension:volume_mig_status_attribute": "rule:admin_api",
"volume_extension:hosts": "rule:admin_api",
"volume_extension:services": "rule:admin_api",

"volume_extension:volume_manage": "rule:admin_api",
"volume_extension:volume_unmanage": "rule:admin_api",

"volume:services": "rule:admin_api",

"volume:create_transfer": "",
"volume:accept_transfer": "",
"volume:delete_transfer": "",
"volume:get_all_transfers": "",

"volume_extension:replication:promote": "rule:admin_api",
"volume_extension:replication:reenable": "rule:admin_api",

"backup:create" : "",
"backup:delete": "",
"backup:get": "",
"backup:get_all": "",
"backup:restore": "",
"backup:backup-import": "rule:admin_api",
"backup:backup-export": "rule:admin_api",

"snapshot_extension:snapshot_actions:update_snapshot_status": "",

"consistencygroup:create" : "group:nobody",
"consistencygroup:delete": "group:nobody",
"consistencygroup:update": "group:nobody",
"consistencygroup:get": "group:nobody",
"consistencygroup:get_all": "group:nobody",

"consistencygroup:create_cgsnapshot" : "group:nobody",
"consistencygroup:delete_cgsnapshot": "group:nobody",
"consistencygroup:get_cgsnapshot": "group:nobody",
"consistencygroup:get_all_cgsnapshots": "group:nobody",

"scheduler_extension:scheduler_stats:get_pools" : "rule:admin_api"
}

```

2.3.4. rootwrap.conf

The **rootwrap.conf** file defines configuration values used by the **rootwrap** script when the Block Storage service must escalate its privileges to those of the root user.

```

# Configuration for cinder-rootwrap
# This file should be owned by (and only-writeable by) the root user

[DEFAULT]
# List of directories to load filter definitions from (separated by ',').
# These directories MUST all be only writeable by root !
filters_path=/etc/cinder/rootwrap.d,/usr/share/cinder/rootwrap

# List of directories to search executables in, in case filters do not

```

```
# explicitly specify a full path (separated by ',')
# If not specified, defaults to system PATH environment variable.
# These directories MUST all be only writeable by root !
exec_dirs=/sbin,/usr/sbin,/bin,/usr/bin,/usr/local/bin,/usr/local/sbin

# Enable logging to syslog
# Default value is False
use_syslog=False

# Which syslog facility to use.
# Valid values include auth, authpriv, syslog, local0, local1...
# Default value is 'syslog'
syslog_log_facility=syslog

# Which messages to log.
# INFO means log all usage
# ERROR means only log unsuccessful attempts
syslog_log_level=ERROR
```

2.4. LOG FILES USED BY BLOCK STORAGE

The corresponding log file of each Block Storage service is stored in the `/var/log/cinder/` directory of the host on which each service runs.

Table 2.27. Log files used by Block Storage services

Log file	Service/interface
api.log	openstack-cinder-api
cinder-manage.log	cinder-manage
scheduler.log	openstack-cinder-scheduler
volume.log	openstack-cinder-volume

2.5. FIBRE CHANNEL ZONE MANAGER

The Fibre Channel Zone Manager allows FC SAN Zone/Access control management in conjunction with Fibre Channel block storage. The configuration of Fibre Channel Zone Manager and various zone drivers are described in this section.

2.5.1. Configure Block Storage to use Fibre Channel Zone Manager

If Block Storage is configured to use a Fibre Channel volume driver that supports Zone Manager, update `cinder.conf` to add the following configuration options to enable Fibre Channel Zone Manager.

Make the following changes in the `/etc/cinder/cinder.conf` file.

Table 2.28. Description of zoning configuration options

Configuration option = Default value	Description
[DEFAULT]	
zoning_mode = <i>none</i>	(StrOpt) FC Zoning mode configured
[fc-zone-manager]	
fc_fabric_names = <i>None</i>	(StrOpt) Comma separated list of Fibre Channel fabric names. This list of names is used to retrieve other SAN credentials for connecting to each SAN fabric
fc_san_lookup_service = <i>cinder.zonemanager.drivers.brocade.brcd_fc_san_lookup_service.BrcdFCSanLookupService</i>	(StrOpt) FC SAN Lookup Service
zone_driver = <i>cinder.zonemanager.drivers.brocade.brcd_fc_zone_driver.BrcdFCZoneDriver</i>	(StrOpt) FC Zone Driver responsible for zone management
zoning_policy = <i>initiator-target</i>	(StrOpt) Zoning policy configured by user; valid values include "initiator-target" or "initiator"

To use different Fibre Channel Zone Drivers, use the parameters described in this section.



NOTE

When multi backend configuration is used, provide the **zoning_mode** configuration option as part of the volume driver configuration where **volume_driver** option is specified.



NOTE

Default value of **zoning_mode** is **None** and this needs to be changed to **fabric** to allow fabric zoning.



NOTE

zoning_policy can be configured as **initiator-target** or **initiator**

2.5.2. Brocade Fibre Channel Zone Driver

Brocade Fibre Channel Zone Driver performs zoning operations through SSH. Configure Brocade Zone Driver and lookup service by specifying the following parameters:

Table 2.29. Description of zoning manager configuration options

Configuration option = Default value	Description
[fc-zone-manager]	
brcd_sb_connector = <i>cinder.zonemanager.drivers.brocade.brcd_fc_zone_client_cli.BrcdFCZoneClientCLI</i>	(StrOpt) Southbound connector for zoning operation

Configure SAN fabric parameters in the form of fabric groups as described in the example below:

Table 2.30. Description of zoning fabrics configuration options

Configuration option = Default value	Description
[BRCD_FABRIC_EXAMPLE]	
fc_fabric_address =	(StrOpt) Management IP of fabric
fc_fabric_password =	(StrOpt) Password for user
fc_fabric_port = 22	(IntOpt) Connecting port
fc_fabric_user =	(StrOpt) Fabric user ID
principal_switch_wwn = None	(StrOpt) Principal switch WWN of the fabric
zone_activate = True	(BoolOpt) overridden zoning activation state
zone_name_prefix = None	(StrOpt) overridden zone name prefix
zoning_policy = initiator-target	(StrOpt) overridden zoning policy



NOTE

Define a fabric group for each fabric using the fabric names used in **fc_fabric_names** configuration option as group name.

2.5.2.1. System requirements

Brocade Fibre Channel Zone Driver requires firmware version FOS v6.4 or higher.

As a best practice for zone management, use a user account with **zoneadmin** role. Users with **admin** role (including the default **admin** user account) are limited to a maximum of two concurrent SSH sessions.

For information about how to manage Brocade Fibre Channel switches, see the Brocade Fabric OS user documentation.

2.5.3. Cisco Fibre Channel Zone Driver

Cisco Fibre Channel Zone Driver automates the zoning operations through SSH. Configure Cisco Zone Driver, Cisco Southbound connector, FC SAN lookup service and Fabric name.

Set the following options in the `cinder.conf` configuration file.

```
[fc-zone-manager]
zone_driver =
cinder.zonemanager.drivers.cisco.cisco_fc_zone_driver.CiscoFCZoneDriver
fc_san_lookup_service =
cinder.zonemanager.drivers.cisco.cisco_fc_san_lookup_service.CiscoFCSanLookupService
fc_fabric_names = CISCO_FABRIC_EXAMPLE
cisco_sb_connector =
cinder.zonemanager.drivers.cisco.cisco_fc_zone_client_cli.CiscoFCZoneClientCLI
```

Table 2.31. Description of cisco zoning manager configuration options

Configuration option = Default value	Description
[fc-zone-manager]	
cisco_sb_connector = <i>cinder.zonemanager.drivers.cisco.cisco_fc_zone_client_cli.CiscoFCZoneClientCLI</i>	(StrOpt) Southbound connector for zoning operation

Configure SAN fabric parameters in the form of fabric groups as described in the example below:

Table 2.32. Description of cisco zoning fabrics configuration options

Configuration option = Default value	Description
[CISCO_FABRIC_EXAMPLE]	
cisco_fc_fabric_address =	(StrOpt) Management IP of fabric
cisco_fc_fabric_password =	(StrOpt) Password for user
cisco_fc_fabric_port = 22	(IntOpt) Connecting port
cisco_fc_fabric_user =	(StrOpt) Fabric user ID
cisco_zone_activate = True	(BoolOpt) overridden zoning activation state
cisco_zone_name_prefix = None	(StrOpt) overridden zone name prefix
cisco_zoning_policy = initiator-target	(StrOpt) overridden zoning policy
cisco_zoning_vsan = None	(StrOpt) VSAN of the Fabric

**NOTE**

Define a fabric group for each fabric using the fabric names used in `fc_fabric_names` configuration option as group name.

The Cisco Fibre Channel Zone Driver supports basic and enhanced zoning modes. The zoning VSAN must exist with an active zone set name which is same as the `fc_fabric_names` option.

2.5.3.1. System requirements

Cisco MDS 9000 Family Switches.

Cisco MDS NX-OS Release 6.2(9) or later.

For information about how to manage Cisco Fibre Channel switches, see the Cisco MDS 9000 user documentation.

2.6. ADDITIONAL OPTIONS

These options can also be set in the `cinder.conf` file.

Table 2.33. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
api_paste_config = <i>api-paste.ini</i>	(StrOpt) File name for the paste.deploy config for cinder-api
api_rate_limit = <i>True</i>	(BoolOpt) Enables or disables rate limit of the API.
az_cache_duration = <i>3600</i>	(IntOpt) Cache volume availability zones in memory for the provided duration in seconds
backend_host = <i>None</i>	(StrOpt) Backend override of host value.
default_timeout = <i>525600</i>	(IntOpt) Default timeout for CLI operations in minutes. For example, LUN migration is a typical long running operation, which depends on the LUN size and the load of the array. An upper bound in the specific deployment can be set to avoid unnecessary long wait. By default, it is 365 days long.
enable_v1_api = <i>True</i>	(BoolOpt) DEPRECATED: Deploy v1 of the Cinder API.
enable_v2_api = <i>True</i>	(BoolOpt) Deploy v2 of the Cinder API.

Configuration option = Default value	Description
extra_capabilities = {}	(StrOpt) User defined capabilities, a JSON formatted string specifying key/value pairs. The key/value pairs can be used by the CapabilitiesFilter to select between backends when requests specify volume types. For example, specifying a service level or the geographical location of a backend, then creating a volume type to allow the user to select by these different properties.
ignore_pool_full_threshold = <i>False</i>	(BoolOpt) Force LUN creation even if the full threshold of pool is reached.
management_ips =	(StrOpt) List of Management IP addresses (separated by commas)
max_header_line = <i>16384</i>	(IntOpt) Maximum line size of message headers to be accepted. max_header_line may need to be increased when using large tokens (typically those generated by the Keystone v3 API with big service catalogs).
osapi_max_limit = <i>1000</i>	(IntOpt) The maximum number of items that a collection resource returns in a single response
osapi_max_request_body_size = <i>114688</i>	(IntOpt) Max size for body of a request
osapi_volume_base_URL = <i>None</i>	(StrOpt) Base URL that will be presented to users in links to the OpenStack Volume API
osapi_volume_ext_list =	(ListOpt) Specify list of extensions to load when using osapi_volume_extension option with cinder.api.contrib.select_extensions
osapi_volume_extension = <i>['cinder.api.contrib.standard_extensions']</i>	(MultiStrOpt) osapi volume extension to load
osapi_volume_listen = <i>0.0.0.0</i>	(StrOpt) IP address on which OpenStack Volume API listens
osapi_volume_listen_port = <i>8776</i>	(IntOpt) Port on which OpenStack Volume API listens
osapi_volume_workers = <i>None</i>	(IntOpt) Number of workers for OpenStack Volume API service. The default is equal to the number of CPUs available.
password =	(StrOpt) Password for Redis server (optional).

Configuration option = Default value	Description
per_volume_size_limit = -1	(IntOpt) Max size allowed per volume, in gigabytes
port = 6379	(IntOpt) Use this port to connect to redis host.
public_endpoint = None	(StrOpt) Public url to use for versions endpoint. The default is None, which will use the request's host_url attribute to populate the URL base. If Cinder is operating behind a proxy, you will want to change this to represent the proxy's URL.
query_volume_filters = name, status, metadata, availability_zone	(ListOpt) Volume filter options which non-admin user could use to query volumes. Default values are: ['name', 'status', 'metadata', 'availability_zone']
transfer_api_class = cinder.transfer.api.API	(StrOpt) The full class name of the volume transfer API class
volume_api_class = cinder.volume.api.API	(StrOpt) The full class name of the volume API class to use
volume_name_template = volume-%s	(StrOpt) Template string to be used to generate volume names
volume_number_multiplier = -1.0	(FloatOpt) Multiplier used for weighing volume number. Negative numbers mean to spread vs stack.
volume_transfer_key_length = 16	(IntOpt) The number of characters in the autogenerated auth key.
volume_transfer_salt_length = 8	(IntOpt) The number of characters in the salt.
[oslo_middleware]	
max_request_body_size = 114688	(IntOpt) The maximum body size for each request, in bytes.
secure_proxy_ssl_header = X-Forwarded-Proto	(StrOpt) The HTTP Header that will be used to determine what the original request protocol scheme was, even if it was hidden by an SSL termination proxy.
[oslo_policy]	
policy_default_rule = default	(StrOpt) Default rule. Enforced when a requested rule is not found.

Configuration option = Default value	Description
policy_dirs = <i>['policy.d']</i>	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the <code>config_dir</code> option, or absolute paths. The file defined by <code>policy_file</code> must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(StrOpt) The JSON file that defines policies.
[oslo_versionedobjects]	
fatal_exception_format_errors = <i>False</i>	(BoolOpt) Make exception message format errors fatal

Table 2.34. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>openstack</i>	(StrOpt) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
notification_driver = <i>[]</i>	(MultiStrOpt) The Drivers(s) to handle sending notifications. Possible values are messaging, messagingv2, routing, log, test, noop
notification_topics = <i>notifications</i>	(ListOpt) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(StrOpt) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the <code>rpc_backend</code> option and driver specific configuration.

Table 2.35. Description of authorization configuration options

Configuration option = Default value	Description
[DEFAULT]	
auth_strategy = <i>keystone</i>	(StrOpt) The strategy to use for auth. Supports noauth, keystone, and deprecated.

Table 2.36. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(StrOpt) Service user password.
admin_tenant_name = <i>admin</i>	(StrOpt) Service tenant name.
admin_token = <i>None</i>	(StrOpt) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(StrOpt) Service username.
auth_admin_prefix =	(StrOpt) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
auth_host = <i>127.0.0.1</i>	(StrOpt) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_plugin = <i>None</i>	(StrOpt) Name of the plugin to load
auth_port = <i>35357</i>	(IntOpt) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_protocol = <i>https</i>	(StrOpt) Protocol of the admin Identity API endpoint (http or https). Deprecated, use <code>identity_uri</code> .
auth_section = <i>None</i>	(StrOpt) Config Section from which to load plugin specific options
auth_uri = <i>None</i>	(StrOpt) Complete public Identity API endpoint.
auth_version = <i>None</i>	(StrOpt) API version of the admin Identity API endpoint.
cache = <i>None</i>	(StrOpt) Env key for the swift cache.
cafile = <i>None</i>	(StrOpt) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate

Configuration option = Default value	Description
check_revocations_for_cached = <i>False</i>	(BoolOpt) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.
delay_auth_decision = <i>False</i>	(BoolOpt) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(StrOpt) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(ListOpt) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(IntOpt) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(IntOpt) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(StrOpt) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/
include_service_catalog = <i>True</i>	(BoolOpt) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.

Configuration option = Default value	Description
keyfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate
memcache_pool_conn_get_timeout = 10	(IntOpt) (Optional) Number of seconds that an operation will wait to get a memcached client connection from the pool.
memcache_pool_dead_retry = 300	(IntOpt) (Optional) Number of seconds memcached server is considered dead before it is tried again.
memcache_pool_maxsize = 10	(IntOpt) (Optional) Maximum total number of open connections to every memcached server.
memcache_pool_socket_timeout = 3	(IntOpt) (Optional) Socket timeout in seconds for communicating with a memcached server.
memcache_pool_unused_timeout = 60	(IntOpt) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
memcache_secret_key = <i>None</i>	(StrOpt) (Optional, mandatory if memcache_security_strategy is defined) This string is used for key derivation.
memcache_security_strategy = <i>None</i>	(StrOpt) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. Acceptable values are MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, auth_token will raise an exception on initialization.
memcache_use_advanced_pool = <i>False</i>	(BoolOpt) (Optional) Use the advanced (eventlet safe) memcached client pool. The advanced pool will only work under python 2.x.
region_name = <i>None</i>	(StrOpt) The region in which the identity server can be found.
revocation_cache_time = 10	(IntOpt) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
signing_dir = <i>None</i>	(StrOpt) Directory used to cache files related to PKI tokens.

Configuration option = Default value	Description
token_cache_time = 300	(IntOpt) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 2.37. Description of backups configuration options

Configuration option = Default value	Description
[DEFAULT]	
backup_api_class = <i>cinder.backup.api.API</i>	(StrOpt) The full class name of the volume backup API class
backup_compression_algorithm = <i>zlib</i>	(StrOpt) Compression algorithm (None to disable)
backup_driver = <i>cinder.backup.drivers.swift</i>	(StrOpt) Driver to use for backups.
backup_manager = <i>cinder.backup.manager.BackupManager</i>	(StrOpt) Full class name for the Manager for volume backup
backup_metadata_version = 2	(IntOpt) Backup metadata version to be used when backing up volume metadata. If this number is bumped, make sure the service doing the restore supports the new version.
backup_name_template = <i>backup-%s</i>	(StrOpt) Template string to be used to generate backup names
backup_object_number_per_notification = 10	(IntOpt) The number of chunks or objects, for which one Ceilometer notification will be sent
backup_posix_path = <i>\$state_path/backup</i>	(StrOpt) Path specifying where to store backups.
backup_service_inithost_offload = <i>False</i>	(BoolOpt) Offload pending backup delete during backup service startup.
backup_timer_interval = 120	(IntOpt) Interval, in seconds, between two progress notifications reporting the backup status
backup_topic = <i>cinder-backup</i>	(StrOpt) The topic that volume backup nodes listen on
snapshot_name_template = <i>snapshot-%s</i>	(StrOpt) Template string to be used to generate snapshot names

Configuration option = Default value	Description
snapshot_same_host = <i>True</i>	(BoolOpt) Create volume from snapshot at the host where snapshot resides

Table 2.38. Description of block device configuration options

Configuration option = Default value	Description
[DEFAULT]	
available_devices =	(ListOpt) List of all available devices

Table 2.39. Description of CA and SSL configuration options

Configuration option = Default value	Description
[DEFAULT]	
ssl_ca_file = <i>None</i>	(StrOpt) CA certificate file to use to verify connecting clients
ssl_cert_file = <i>None</i>	(StrOpt) Certificate file to use when starting the server securely
ssl_key_file = <i>None</i>	(StrOpt) Private key file to use when starting the server securely

Table 2.40. Description of CloudByte volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
cb_account_name = <i>None</i>	(StrOpt) CloudByte storage specific account name. This maps to a project name in OpenStack.
cb_add_qosgroup = {'latency': '15', 'iops': '10', 'graceallowed': 'false', 'iopscontrol': 'true', 'memlimit': '0', 'throughput': '0', 'tpcontrol': 'false', 'networkspeed': '0'}	(DictOpt) These values will be used for CloudByte storage's addQos API call.
cb_apikey = <i>None</i>	(StrOpt) Driver will use this API key to authenticate against the CloudByte storage's management interface.

Configuration option = Default value	Description
cb_auth_group = <i>None</i>	(StrOpt) This corresponds to the discovery authentication group in CloudByte storage. Chap users are added to this group. Driver uses the first user found for this group. Default value is None.
cb_confirm_volume_create_retries = 3	(IntOpt) Will confirm a successful volume creation in CloudByte storage by making this many number of attempts.
cb_confirm_volume_create_retry_interval = 5	(IntOpt) A retry value in seconds. Will be used by the driver to check if volume creation was successful in CloudByte storage.
cb_confirm_volume_delete_retries = 3	(IntOpt) Will confirm a successful volume deletion in CloudByte storage by making this many number of attempts.
cb_confirm_volume_delete_retry_interval = 5	(IntOpt) A retry value in seconds. Will be used by the driver to check if volume deletion was successful in CloudByte storage.
cb_create_volume = {'compression': 'off', 'deduplication': 'off', 'blocklength': '512B', 'sync': 'always', 'protocoltype': 'ISCSI', 'recordsize': '16k'}	(DictOpt) These values will be used for CloudByte storage's createVolume API call.
cb_tsm_name = <i>None</i>	(StrOpt) This corresponds to the name of Tenant Storage Machine (TSM) in CloudByte storage. A volume will be created in this TSM.

Table 2.41. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
allow_availability_zone_fallback = <i>False</i>	(BoolOpt) If the requested Cinder availability zone is unavailable, fall back to the value of <code>default_availability_zone</code> , then <code>storage_availability_zone</code> , instead of failing.
chap_password =	(StrOpt) Password for specified CHAP account name.
chap_username =	(StrOpt) CHAP user name.
chiscsi_conf = <i>/etc/chelsio-iscsi/chiscsi.conf</i>	(StrOpt) Chiscsi (CXT) global defaults configuration file

Configuration option = Default value	Description
cinder_internal_tenant_project_id = <i>None</i>	(StrOpt) ID of the project which will be used as the Cinder internal tenant.
cinder_internal_tenant_user_id = <i>None</i>	(StrOpt) ID of the user to be used in volume operations as the Cinder internal tenant.
client_socket_timeout = <i>900</i>	(IntOpt) Timeout for client connections' socket operations. If an incoming connection is idle for this number of seconds it will be closed. A value of '0' means wait forever.
compute_api_class = <i>cinder.compute.nova.API</i>	(StrOpt) The full class name of the compute API class to use
consistencygroup_api_class = <i>cinder.consistencygroup.api.API</i>	(StrOpt) The full class name of the consistencygroup API class
default_availability_zone = <i>None</i>	(StrOpt) Default availability zone for new volumes. If not set, the <code>storage_availability_zone</code> option value is used as the default for new volumes.
default_volume_type = <i>None</i>	(StrOpt) Default volume type to use
driver_data_namespace = <i>None</i>	(StrOpt) Namespace for driver private data values to be saved in.
driver_ssl_cert_verify = <i>False</i>	(BoolOpt) If set to True the http client will validate the SSL certificate of the backend endpoint.
enable_force_upload = <i>False</i>	(BoolOpt) Enables the Force option on <code>upload_to_image</code> . This enables running <code>upload_volume</code> on in-use volumes for backends that support it.
enable_new_services = <i>True</i>	(BoolOpt) Services to be added to the available pool on create
end_time = <i>None</i>	(StrOpt) If this option is specified then the end time specified is used instead of the end time of the last completed audit period.
enforce_multipath_for_image_xfer = <i>False</i>	(BoolOpt) If this is set to True, attachment of volumes for image transfer will be aborted when <code>multipathd</code> is not running. Otherwise, it will fallback to single path.
executor_thread_pool_size = <i>64</i>	(IntOpt) Size of executor thread pool.

Configuration option = Default value	Description
host = <i>localhost</i>	(StrOpt) Name of this node. This can be an opaque identifier. It is not necessarily a host name, FQDN, or IP address.
iet_conf = <i>/etc/iet/ietd.conf</i>	(StrOpt) IET configuration file
iscsi_secondary_ip_addresses =	(ListOpt) The list of secondary IP addresses of the iSCSI daemon
managed_replication_target = <i>True</i>	(BoolOpt) There are two types of target configurations managed (replicate to another configured backend) or unmanaged (replicate to a device not managed by Cinder).
max_over_subscription_ratio = <i>20.0</i>	(FloatOpt) Float representation of the over subscription ratio when thin provisioning is involved. Default ratio is 20.0, meaning provisioned capacity can be 20 times of the total physical capacity. If the ratio is 10.5, it means provisioned capacity can be 10.5 times of the total physical capacity. A ratio of 1.0 means provisioned capacity cannot exceed the total physical capacity. A ratio lower than 1.0 will be ignored and the default value will be used instead.
memcached_servers = <i>None</i>	(ListOpt) Memcached servers or None for in process cache.
monkey_patch = <i>False</i>	(BoolOpt) Enable monkey patching
monkey_patch_modules =	(ListOpt) List of modules/decorators to monkey patch
my_ip = <i>10.0.0.1</i>	(StrOpt) IP address of this host
no_snapshot_gb_quota = <i>False</i>	(BoolOpt) Whether snapshots count against gigabyte quota
num_shell_tries = <i>3</i>	(IntOpt) Number of times to attempt to run flakey shell commands
os_privileged_user_auth_url = <i>None</i>	(StrOpt) Auth URL associated with the OpenStack privileged account.
os_privileged_user_name = <i>None</i>	(StrOpt) OpenStack privileged account username. Used for requests to other services (such as Nova) that require an account with special rights.

Configuration option = Default value	Description
os_privileged_user_password = <i>None</i>	(StrOpt) Password associated with the OpenStack privileged account.
os_privileged_user_tenant = <i>None</i>	(StrOpt) Tenant name associated with the OpenStack privileged account.
periodic_fuzzy_delay = <i>60</i>	(IntOpt) Range, in seconds, to randomly delay when starting the periodic task scheduler to reduce stampeding. (Disable by setting to 0)
periodic_interval = <i>60</i>	(IntOpt) Interval, in seconds, between running periodic tasks
replication_api_class = <i>cinder.replication.api.API</i>	(StrOpt) The full class name of the volume replication API class
replication_devices = <i>None</i>	(ListOpt) List of k/v pairs representing a replication target for this backend device. For unmanaged the format is: {'key-1'='val1' 'key-2'='val2'...},{...} and for managed devices its simply a list of valid configured backend_names that the driver supports replicating to: backend-a,bakcend-b...
report_interval = <i>10</i>	(IntOpt) Interval, in seconds, between nodes reporting state to datastore
request_timeout = <i>300</i>	(IntOpt) Global backend request timeout, in seconds
reserved_percentage = <i>0</i>	(IntOpt) The percentage of backend capacity is reserved
rootwrap_config = <i>/etc/cinder/rootwrap.conf</i>	(StrOpt) Path to the rootwrap configuration file to use for running commands as root
send_actions = <i>False</i>	(BoolOpt) Send the volume and snapshot create and delete notifications generated in the specified period.
service_down_time = <i>60</i>	(IntOpt) Maximum time since last check-in for a service to be considered up
sqlite_clean_db = <i>clean.sqlite</i>	(StrOpt) File name of clean sqlite db
ssh_hosts_key_file = <i>\$state_path/ssh_known_hosts</i>	(StrOpt) File containing SSH host keys for the systems with which Cinder needs to communicate. OPTIONAL: Default=\$state_path/ssh_known_hosts

Configuration option = Default value	Description
start_time = <i>None</i>	(StrOpt) If this option is specified then the start time specified is used instead of the start time of the last completed audit period.
state_path = <i>/var/lib/cinder</i>	(StrOpt) Top-level directory for maintaining cinder's state
storage_availability_zone = <i>nova</i>	(StrOpt) Availability zone of this node
strict_ssh_host_key_policy = <i>False</i>	(BoolOpt) Option to enable strict host key checking. When set to "True" Cinder will only connect to systems with a host key present in the configured "ssh_hosts_key_file". When set to "False" the host key will be saved upon first connection and used for subsequent connections. Default=False
tcp_keepalive = <i>True</i>	(BoolOpt) Sets the value of TCP_KEEPALIVE (True/False) for each server socket.
tcp_keepalive_count = <i>None</i>	(IntOpt) Sets the value of TCP_KEEPCNT for each server socket. Not supported on OS X.
tcp_keepalive_interval = <i>None</i>	(IntOpt) Sets the value of TCP_KEEPINTVL in seconds for each server socket. Not supported on OS X.
tcp_keepidle = <i>600</i>	(IntOpt) Sets the value of TCP_KEEPIDLE in seconds for each server socket. Not supported on OS X.
until_refresh = <i>0</i>	(IntOpt) Count of reservations until usage is refreshed
use_chap_auth = <i>False</i>	(BoolOpt) Option to enable/disable CHAP authentication for targets.
use_forwarded_for = <i>False</i>	(BoolOpt) Treat X-Forwarded-For as the canonical remote address. Only enable this if you have a sanitizing proxy.
watch_log_file = <i>False</i>	(BoolOpt) (Optional) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log-file option is specified and Linux platform is used. This option is ignored if log_config_append is set.

Configuration option = Default value	Description
<code>wsgi_keep_alive = True</code>	(BoolOpt) If False, closes the client socket connection explicitly. Setting it to True to maintain backward compatibility. Recommended setting is set it to False.
[keystone_authtoken]	
<code>memcached_servers = None</code>	(ListOpt) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.

Table 2.42. Description of Compute configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>nova_api_insecure = False</code>	(BoolOpt) Allow to perform insecure SSL requests to nova
<code>nova_ca_certificates_file = None</code>	(StrOpt) Location of ca certificates file to use for nova client requests.
<code>nova_catalog_admin_info = compute:Compute Service:adminURL</code>	(StrOpt) Same as nova_catalog_info, but for admin endpoint.
<code>nova_catalog_info = compute:Compute Service:publicURL</code>	(StrOpt) Match this value when searching for nova in the service catalog. Format is: separated values of the form: <service_type>:<service_name>:<endpoint_type>
<code>nova_endpoint_admin_template = None</code>	(StrOpt) Same as nova_endpoint_template, but for admin endpoint.
<code>nova_endpoint_template = None</code>	(StrOpt) Override service catalog lookup with template for nova endpoint e.g. <code>http://localhost:8774/v2/%(project_id)s</code>
<code>os_region_name = None</code>	(StrOpt) Region name of this node

Table 2.43. Description of database configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
db_driver = <i>cinder.db</i>	(StrOpt) Driver to use for database access
[database]	
backend = <i>sqlalchemy</i>	(StrOpt) The back end to use for the database.
connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(BoolOpt) Add Python stack traces to SQL as comment strings.
db_inc_retry_interval = <i>True</i>	(BoolOpt) If True, increases the interval between retries of a database operation up to db_max_retry_interval .
db_max_retries = <i>20</i>	(IntOpt) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = <i>10</i>	(IntOpt) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.
db_retry_interval = <i>1</i>	(IntOpt) Seconds between retries of a database transaction.
idle_timeout = <i>3600</i>	(IntOpt) Timeout before idle SQL connections are reaped.
max_overflow = <i>None</i>	(IntOpt) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = <i>None</i>	(IntOpt) Maximum number of SQL connections to keep open in a pool.
max_retries = <i>10</i>	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = <i>1</i>	(IntOpt) Minimum number of SQL connections to keep open in a pool.

Configuration option = Default value	Description
mysql_sql_mode = <i>TRADITIONAL</i>	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: <code>mysql_sql_mode=</code>
pool_timeout = <i>None</i>	(IntOpt) If set, use this value for <code>pool_timeout</code> with SQLAlchemy.
retry_interval = <i>10</i>	(IntOpt) Interval between retries of opening a SQL connection.
slave_connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = <i>oslo.sqlite</i>	(StrOpt) The file name to use with SQLite.
sqlite_synchronous = <i>True</i>	(BoolOpt) If True, SQLite uses synchronous mode.
use_db_reconnect = <i>False</i>	(BoolOpt) Enable the experimental use of database reconnect on connection lost.
use_tpool = <i>False</i>	(BoolOpt) Enable the experimental use of thread pooling for all DB API calls

Table 2.44. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
trace_flags = <i>None</i>	(ListOpt) List of options that control which trace info is written to the DEBUG log level to assist developers. Valid values are <code>method</code> and <code>api</code> .

Table 2.45. Description of EMC configuration options

Configuration option = Default value	Description
[DEFAULT]	
check_max_pool_luns_threshold = <i>False</i>	(BoolOpt) Report <code>free_capacity_gb</code> as 0 when the limit to maximum number of pool LUNs is reached. By default, the value is False.

Configuration option = Default value	Description
cinder_emc_config_file = <i>/etc/cinder/cinder_emc_config.xml</i>	(StrOpt) use this file for cinder emc plugin config data
destroy_empty_storage_group = <i>False</i>	(BoolOpt) To destroy storage group when the last LUN is removed from it. By default, the value is False.
force_delete_lun_in_storagegroup = <i>False</i>	(BoolOpt) Delete a LUN even if it is in Storage Groups.
initiator_auto_deregistration = <i>False</i>	(BoolOpt) Automatically deregister initiators after the related storage group is destroyed. By default, the value is False.
initiator_auto_registration = <i>False</i>	(BoolOpt) Automatically register initiators. By default, the value is False.
io_port_list = *	(StrOpt) Comma separated iSCSI or FC ports to be used in Nova or Cinder.
iscsi_initiators =	(StrOpt) Mapping between hostname and its iSCSI initiator IP addresses.
max_luns_per_storage_group = <i>255</i>	(IntOpt) Default max number of LUNs in a storage group. By default, the value is 255.
naviseccli_path =	(StrOpt) Naviseccli Path.
storage_vnx_authentication_type = <i>global</i>	(StrOpt) VNX authentication scope type.
storage_vnx_pool_names = <i>None</i>	(StrOpt) Comma-separated list of storage pool names to be used.
storage_vnx_security_file_dir = <i>None</i>	(StrOpt) Directory path that contains the VNX security file. Make sure the security file is generated first.
xtremio_array_busy_retry_count = <i>5</i>	(IntOpt) Number of retries in case array is busy
xtremio_array_busy_retry_interval = <i>5</i>	(IntOpt) Interval between retries in case array is busy
xtremio_cluster_name =	(StrOpt) XMS cluster id in multi-cluster environment

Table 2.46. Description of IBM FlashSystem volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
flashsystem_connection_protocol = <i>FC</i>	(StrOpt) Connection protocol should be FC. (Default is FC.)
flashsystem_iscsi_portid = <i>0</i>	(IntOpt) Default iSCSI Port ID of FlashSystem. (Default port is 0.)
flashsystem_multihostmap_enabled = <i>True</i>	(BoolOpt) Allows vdisk to multi host mapping. (Default is True)
flashsystem_multipath_enabled = <i>False</i>	(BoolOpt) Connect with multipath (FC only).(Default is false.)

Table 2.47. Description of IBM SONAS and Storwize V7000 volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
ibmnas_platform_type = <i>v7ku</i>	(StrOpt) IBMNAS platform type to be used as backend storage; valid values are - v7ku : for using IBM Storwize V7000 Unified, sonas : for using IBM Scale Out NAS, gpfs-nas : for using NFS based IBM GPFS deployments.
nas_ip =	(StrOpt) IP address or Hostname of NAS system.
nas_login = <i>admin</i>	(StrOpt) User name to connect to NAS system.
nas_mount_options = <i>None</i>	(StrOpt) Options used to mount the storage backend file system where Cinder volumes are stored.
nas_password =	(StrOpt) Password to connect to NAS system.
nas_private_key =	(StrOpt) Filename of private key to use for SSH authentication.
nas_secure_file_operations = <i>auto</i>	(StrOpt) Allow network-attached storage systems to operate in a secure environment where root level access is not permitted. If set to False, access is as the root user and insecure. If set to True, access is not as root. If set to auto, a check is done to determine if this is a new installation: True is used if so, otherwise False. Default is auto.

Configuration option = Default value	Description
nas_secure_file_permissions = <i>auto</i>	(StrOpt) Set more secure file permissions on network-attached storage volume files to restrict broad other/world access. If set to False, volumes are created with open permissions. If set to True, volumes are created with permissions for the cinder user and group (660). If set to auto, a check is done to determine if this is a new installation: True is used if so, otherwise False. Default is auto.
nas_share_path =	(StrOpt) Path to the share to use for storing Cinder volumes. For example: "/srv/export1" for an NFS server export available at 10.0.5.10:/srv/export1 .
nas_ssh_port = 22	(IntOpt) SSH port to use to connect to NAS system.

Table 2.48. Description of images configuration options

Configuration option = Default value	Description
[DEFAULT]	
allowed_direct_url_schemes =	(ListOpt) A list of url schemes that can be downloaded directly via the direct_url. Currently supported schemes: [file].
glance_api_insecure = <i>False</i>	(BoolOpt) Allow to perform insecure SSL (https) requests to glance
glance_api_servers = <i>\$glance_host:\$glance_port</i>	(ListOpt) A list of the glance API servers available to cinder ([hostname ip]:port)
glance_api_ssl_compression = <i>False</i>	(BoolOpt) Enables or disables negotiation of SSL layer compression. In some cases disabling compression can improve data throughput, such as when high network bandwidth is available and you use compressed image formats like qcow2.
glance_api_version = 1	(IntOpt) Version of the glance API to use
glance_ca_certificates_file = <i>None</i>	(StrOpt) Location of ca certificates file to use for glance client requests.
glance_core_properties = <i>checksum, container_format, disk_format, image_name, image_id, min_disk, min_ram, name, size</i>	(ListOpt) Default core properties of image
glance_host = <i>\$my_ip</i>	(StrOpt) Default glance host name or IP

Configuration option = Default value	Description
glance_num_retries = 0	(IntOpt) Number retries when downloading an image from glance
glance_port = 9292	(IntOpt) Default glance port
glance_request_timeout = None	(IntOpt) http/https timeout value for glance operations. If no value (None) is supplied here, the glanceclient default value is used.
image_conversion_dir = <i>\$state_path/conversion</i>	(StrOpt) Directory used for temporary storage during image conversion
image_upload_use_cinder_backend = False	(BoolOpt) If set to True, upload-to-image in raw format will create a cloned volume and register its location to the image service, instead of uploading the volume content. The cinder backend and locations support must be enabled in the image service, and glance_api_version must be set to 2.
image_upload_use_internal_tenant = False	(BoolOpt) If set to True, the image volume created by upload-to-image will be placed in the internal tenant. Otherwise, the image volume is created in the current context's tenant.
image_volume_cache_enabled = False	(BoolOpt) Enable the image volume cache for this backend.
image_volume_cache_max_count = 0	(IntOpt) Max number of entries allowed in the image volume cache. 0 => unlimited.
image_volume_cache_max_size_gb = 0	(IntOpt) Max size of the image volume cache for this backend in GB. 0 => unlimited.
use_multipath_for_image_xfer = False	(BoolOpt) Do we attach/detach volumes in cinder using multipath for volume to image and image to volume transfers?

Table 2.49. Description of key manager configuration options

Configuration option = Default value	Description
[keymgr]	
api_class = <i>cinder.keymgr.conf_key_mgr.ConfKeyManager</i>	(StrOpt) The full class name of the key manager API class
encryption_api_url = <i>http://localhost:9311/v1</i>	(StrOpt) Url for encryption service.

Configuration option = Default value	Description
encryption_auth_url = <i>http://localhost:5000/v3</i>	(StrOpt) Authentication url for encryption service.
fixed_key = <i>None</i>	(StrOpt) Fixed key returned by key manager, specified in hex

Table 2.50. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(BoolOpt) Print debugging output (set logging level to DEBUG instead of default INFO level).
default_log_levels = <i>amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystonemiddleware=WARN, routes.middleware=WARN, stevedore=WARN, taskflow=WARN</i>	(ListOpt) List of logger=LEVEL pairs. This option is ignored if log_config_append is set.
fatal_deprecations = <i>False</i>	(BoolOpt) Enables or disables fatal status of deprecations.
fatal_exception_format_errors = <i>False</i>	(BoolOpt) Make exception message format errors fatal.
instance_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, log_format).

Configuration option = Default value	Description
log_date_format = %Y-%m-%d %H:%M:%S	(StrOpt) Format string for %(asctime)s in log records. Default: %(default)s . This option is ignored if log_config_append is set.
log_dir = None	(StrOpt) (Optional) The base directory used for relative --log-file paths. This option is ignored if log_config_append is set.
log_file = None	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout. This option is ignored if log_config_append is set.
log_format = None	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecate, use logging_context_format_string and logging_default_format_string instead. This option is ignored if log_config_append is set.
logging_context_format_string = %(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%(message)s	(StrOpt) Format string to use for log messages with context.
logging_debug_format_suffix = %(funcName)s %(pathname)s:%(lineno)d	(StrOpt) Data to append to log format when level is DEBUG.
logging_default_format_string = %(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [-] %(instance)s%(message)s	(StrOpt) Format string to use for log messages without context.
logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d ERROR %(name)s %(instance)s	(StrOpt) Prefix each line of exception output with this format.
publish_errors = False	(BoolOpt) Enables or disables publication of error events.
syslog_log_facility = LOG_USER	(StrOpt) Syslog facility to receive log lines. This option is ignored if log_config_append is set.
use_stderr = True	(BoolOpt) Log output to standard error. This option is ignored if log_config_append is set.
use_syslog = False	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.

Configuration option = Default value	Description
use_syslog_rfc_format = <i>True</i>	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in Kilo, and will be removed in Mitaka, along with this option. This option is ignored if log_config_append is set.
verbose = <i>True</i>	(BoolOpt) If set to false, will disable INFO logging level, making WARNING the default.

Table 2.51. Description of NAS configuration options

Configuration option = Default value	Description
[DEFAULT]	
nas_ip =	(StrOpt) IP address or Hostname of NAS system.
nas_login = <i>admin</i>	(StrOpt) User name to connect to NAS system.
nas_mount_options = <i>None</i>	(StrOpt) Options used to mount the storage backend file system where Cinder volumes are stored.
nas_password =	(StrOpt) Password to connect to NAS system.
nas_private_key =	(StrOpt) Filename of private key to use for SSH authentication.
nas_secure_file_operations = <i>auto</i>	(StrOpt) Allow network-attached storage systems to operate in a secure environment where root level access is not permitted. If set to False, access is as the root user and insecure. If set to True, access is not as root. If set to auto, a check is done to determine if this is a new installation: True is used if so, otherwise False. Default is auto.
nas_secure_file_permissions = <i>auto</i>	(StrOpt) Set more secure file permissions on network-attached storage volume files to restrict broad other/world access. If set to False, volumes are created with open permissions. If set to True, volumes are created with permissions for the cinder user and group (660). If set to auto, a check is done to determine if this is a new installation: True is used if so, otherwise False. Default is auto.

Configuration option = Default value	Description
nas_share_path =	(StrOpt) Path to the share to use for storing Cinder volumes. For example: <code>"/srv/export1"</code> for an NFS server export available at <code>10.0.5.10:/srv/export1</code> .
nas_ssh_port = 22	(IntOpt) SSH port to use to connect to NAS system.

Table 2.52. Description of Open vStorage driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
vpool_name =	(StrOpt) Vpool to use for volumes - backend is defined by vpool not by us.

Table 2.53. Description of oslo_middleware configuration options

Configuration option = Default value	Description
[oslo_middleware]	
max_request_body_size = 114688	(IntOpt) The maximum body size for each request, in bytes.

Table 2.54. Description of profiler configuration options

Configuration option = Default value	Description
[profiler]	
profiler_enabled = False	(BoolOpt) If False fully disable profiling feature.
trace_sqlalchemy = False	(BoolOpt) If False doesn't trace SQL requests.

Table 2.55. Description of Pure Storage driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
pure_api_token = None	(StrOpt) REST API authorization token.

Table 2.56. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
qpid_heartbeat = <i>60</i>	(IntOpt) Seconds between connection keepalive heartbeats.
qpid_hostname = <i>localhost</i>	(StrOpt) Qpid broker hostname.
qpid_hosts = <i>\$qpid_hostname:\$qpid_port</i>	(ListOpt) Qpid HA cluster host:port pairs.
qpid_password =	(StrOpt) Password for Qpid connection.
qpid_port = <i>5672</i>	(IntOpt) Qpid broker port.
qpid_protocol = <i>tcp</i>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
qpid_receiver_capacity = <i>1</i>	(IntOpt) The number of prefetched messages held by receiver.
qpid_sasl_mechanisms =	(StrOpt) Space separated list of SASL mechanisms to use for auth.
qpid_tcp_nodelay = <i>True</i>	(BoolOpt) Whether to disable the Nagle algorithm.
qpid_topology_version = <i>1</i>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
qpid_username =	(StrOpt) Username for Qpid connection.
send_single_reply = <i>False</i>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

Table 2.57. Description of quota configuration options

Configuration option = Default value	Description
[DEFAULT]	
max_age = 0	(IntOpt) Number of seconds between subsequent usage refreshes
quota_backup_gigabytes = 1000	(IntOpt) Total amount of storage, in gigabytes, allowed for backups per project
quota_backups = 10	(IntOpt) Number of volume backups allowed per project
quota_consistencygroups = 10	(IntOpt) Number of consistencygroups allowed per project
quota_driver = <i>cinder.quota.DbQuotaDriver</i>	(StrOpt) Default driver to use for quota checks
quota_gigabytes = 1000	(IntOpt) Total amount of storage, in gigabytes, allowed for volumes and snapshots per project
quota_snapshots = 10	(IntOpt) Number of volume snapshots allowed per project
quota_volumes = 10	(IntOpt) Number of volumes allowed per project
reservation_expire = 86400	(IntOpt) Number of seconds until a reservation expires
use_default_quota_class = <i>True</i>	(BoolOpt) Enables or disables use of default quota class with default quota.

Table 2.58. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
fake_rabbit = <i>False</i>	(BoolOpt) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>

Configuration option = Default value	Description
heartbeat_rate = 2	(IntOpt) How often times during the heartbeat_timeout_threshold we check the heartbeat.
heartbeat_timeout_threshold = 60	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disable the heartbeat). EXPERIMENTAL
kombu_reconnect_delay = 1.0	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_reconnect_timeout = 60	(IntOpt) How long to wait before considering a reconnect attempt to have failed. This value should not be longer than rpc_response_timeout.
kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
rabbit_ha_queues = <i>False</i>	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
rabbit_host = <i>localhost</i>	(StrOpt) The RabbitMQ broker address where a single node is used.
rabbit_hosts = <i>\$rabbit_host:\$rabbit_port</i>	(ListOpt) RabbitMQ HA cluster host:port pairs.
rabbit_login_method = <i>AMQPLAIN</i>	(StrOpt) The RabbitMQ login method.
rabbit_max_retries = 0	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = <i>guest</i>	(StrOpt) The RabbitMQ password.
rabbit_port = 5672	(IntOpt) The RabbitMQ broker port where a single node is used.

Configuration option = Default value	Description
<code>rabbit_retry_backoff = 2</code>	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
<code>rabbit_retry_interval = 1</code>	(IntOpt) How frequently to retry connecting with RabbitMQ.
<code>rabbit_use_ssl = False</code>	(BoolOpt) Connect over SSL for RabbitMQ.
<code>rabbit_userid = guest</code>	(StrOpt) The RabbitMQ userid.
<code>rabbit_virtual_host = /</code>	(StrOpt) The RabbitMQ virtual host.
<code>send_single_reply = False</code>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

Table 2.59. Description of Redis configuration options

Configuration option = Default value	Description
[matchmaker_redis]	
<code>host = 127.0.0.1</code>	(StrOpt) Host to locate redis.
<code>password =</code>	(StrOpt) Password for Redis server (optional).
<code>port = 6379</code>	(IntOpt) Use this port to connect to redis host.

Table 2.60. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>rpc_backend = rabbit</code>	(StrOpt) The messaging driver to use, defaults to rabbit. Other drivers include qpid and zmq.

Configuration option = Default value	Description
rpc_cast_timeout = 30	(IntOpt) Seconds to wait before a cast expires (TTL). Only supported by impl_zmq.
rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
rpc_poll_timeout = 1	(IntOpt) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
rpc_response_timeout = 60	(IntOpt) Seconds to wait for a response from a call.
volume_topic = <i>cinder-volume</i>	(StrOpt) The topic that volume nodes listen on
[oslo_concurrency]	
disable_process_locking = <i>False</i>	(BoolOpt) Enables or disables inter-process locks.
lock_path = <i>None</i>	(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(BoolOpt) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>	(StrOpt) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(StrOpt) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(StrOpt) address prefix when sending to any server in group
idle_timeout = 0	(IntOpt) Timeout for inactive connections (in seconds)
password =	(StrOpt) Password for message broker authentication
sasl_config_dir =	(StrOpt) Path to directory that contains the SASL configuration

Configuration option = Default value	Description
sasl_config_name =	(StrOpt) Name of configuration file (without .conf suffix)
sasl_mechanisms =	(StrOpt) Space separated list of acceptable SASL mechanisms
server_request_prefix = <i>exclusive</i>	(StrOpt) address prefix used when sending to a specific server
ssl_ca_file =	(StrOpt) CA certificate PEM file to verify server certificate
ssl_cert_file =	(StrOpt) Identifying certificate PEM file to present to clients
ssl_key_file =	(StrOpt) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(StrOpt) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>	(BoolOpt) Debug: dump AMQP frames to stdout
username =	(StrOpt) User name for message broker authentication

Table 2.61. Description of SAN configuration options

Configuration option = Default value	Description
[DEFAULT]	
san_clustername =	(StrOpt) Cluster name to use for creating volumes
san_ip =	(StrOpt) IP address of SAN controller
san_is_local = <i>False</i>	(BoolOpt) Execute commands locally instead of over SSH; use if the volume service is running on the SAN device
san_login = <i>admin</i>	(StrOpt) Username for SAN controller
san_password =	(StrOpt) Password for SAN controller
san_private_key =	(StrOpt) Filename of private key to use for SSH authentication

Configuration option = Default value	Description
san_secondary_ip = <i>None</i>	(StrOpt) VNX secondary SP IP Address.
san_ssh_port = 22	(IntOpt) SSH port to use with SAN
san_thin_provision = <i>True</i>	(BoolOpt) Use thin provisioning for SAN volumes?
ssh_conn_timeout = 30	(IntOpt) SSH connection timeout in seconds
ssh_max_pool_conn = 5	(IntOpt) Maximum ssh connections in the pool
ssh_min_pool_conn = 1	(IntOpt) Minimum ssh connections in the pool

Table 2.62. Description of scheduler configuration options

Configuration option = Default value	Description
[DEFAULT]	
filter_function = <i>None</i>	(StrOpt) String representation for an equation that will be used to filter hosts. Only used when the driver filter is set to be used by the Cinder scheduler.
goodness_function = <i>None</i>	(StrOpt) String representation for an equation that will be used to determine the goodness of a host. Only used when using the goodness weigher is set to be used by the Cinder scheduler.
scheduler_default_filters = <i>AvailabilityZoneFilter, CapacityFilter, CapabilitiesFilter</i>	(ListOpt) Which filter class names to use for filtering hosts when not specified in the request.
scheduler_default_weighers = <i>CapacityWeigher</i>	(ListOpt) Which weigher class names to use for weighing hosts.
scheduler_driver = <i>cinder.scheduler.filter_scheduler.FilterScheduler</i>	(StrOpt) Default scheduler driver to use
scheduler_host_manager = <i>cinder.scheduler.host_manager.HostManager</i>	(StrOpt) The scheduler host manager class to use
scheduler_json_config_location =	(StrOpt) Absolute path to scheduler configuration JSON file.
scheduler_manager = <i>cinder.scheduler.manager.SchedulerManager</i>	(StrOpt) Full class name for the Manager for scheduler

Configuration option = Default value	Description
scheduler_max_attempts = 3	(IntOpt) Maximum number of attempts to schedule an volume
scheduler_topic = <i>cinder-scheduler</i>	(StrOpt) The topic that scheduler nodes listen on

Table 2.63. Description of SCST volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
scst_target_driver = <i>iscsi</i>	(StrOpt) SCST target implementation can choose from multiple SCST target drivers.
scst_target_iqn_name = <i>None</i>	(StrOpt) Certain ISCSI targets have predefined target names, SCST target driver uses this name.

Table 2.64. Description of Scality REST Block storage driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
srb_base_urls = <i>None</i>	(StrOpt) Comma-separated list of REST servers IP to connect to. (eg http://IP1/,http://IP2:81/path

Table 2.65. Description of storage configuration options

Configuration option = Default value	Description
[DEFAULT]	
allocated_capacity_weight_multiplier = <i>-1.0</i>	(FloatOpt) Multiplier used for weighing volume capacity. Negative numbers mean to stack vs spread.
capacity_weight_multiplier = <i>1.0</i>	(FloatOpt) Multiplier used for weighing volume capacity. Negative numbers mean to stack vs spread.
enabled_backends = <i>None</i>	(ListOpt) A list of backend names to use. These backend names should be backed by a unique [CONFIG] group with its options

Configuration option = Default value	Description
iscsi_helper = <i>tgtadm</i>	(StrOpt) iSCSI target user-land tool to use. <i>tgtadm</i> is default, use <i>lioadm</i> for LIO iSCSI support, <i>scstadmin</i> for SCST target support, <i>iseradm</i> for the iSER protocol, <i>ietadm</i> for iSCSI Enterprise Target, <i>iscsictl</i> for Chelsio iSCSI Target or <i>fake</i> for testing.
iscsi_iotype = <i>fileio</i>	(StrOpt) Sets the behavior of the iSCSI target to either perform blockio or fileio optionally, auto can be set and Cinder will autodetect type of backing device
iscsi_ip_address = <i>\$my_ip</i>	(StrOpt) The IP address that the iSCSI daemon is listening on
iscsi_port = <i>3260</i>	(IntOpt) The port that the iSCSI daemon is listening on
iscsi_protocol = <i>iscsi</i>	(StrOpt) Determines the iSCSI protocol for new iSCSI volumes, created with <i>tgtadm</i> or <i>lioadm</i> target helpers. In order to enable RDMA, this parameter should be set with the value "iser". The supported iSCSI protocol values are "iscsi" and "iser".
iscsi_target_flags =	(StrOpt) Sets the target-specific flags for the iSCSI target. Only used for <i>tgtadm</i> to specify backing device flags using <i>bsoflags</i> option. The specified string is passed as is to the underlying tool.
iscsi_target_prefix = <i>iqn.2010-10.org.openstack:</i>	(StrOpt) Prefix for iSCSI volumes
iscsi_write_cache = <i>on</i>	(StrOpt) Sets the behavior of the iSCSI target to either perform write-back(on) or write-through(off). This parameter is valid if <i>iscsi_helper</i> is set to <i>tgtadm</i> or <i>iseradm</i> .
iser_helper = <i>tgtadm</i>	(StrOpt) The name of the iSER target user-land tool to use
iser_ip_address = <i>\$my_ip</i>	(StrOpt) The IP address that the iSER daemon is listening on
iser_port = <i>3260</i>	(IntOpt) The port that the iSER daemon is listening on
iser_target_prefix = <i>iqn.2010-10.org.openstack:</i>	(StrOpt) Prefix for iSER volumes

Configuration option = Default value	Description
migration_create_volume_timeout_secs = 300	(IntOpt) Timeout for creating the volume to migrate to when performing volume migration (seconds)
num_iser_scan_tries = 3	(IntOpt) The maximum number of times to rescan iSER target to find volume
num_volume_device_scan_tries = 3	(IntOpt) The maximum number of times to rescan targets to find volume
volume_backend_name = <i>None</i>	(StrOpt) The backend name for a given driver implementation
volume_clear = <i>zero</i>	(StrOpt) Method used to wipe old volumes
volume_clear_ionice = <i>None</i>	(StrOpt) The flag to pass to ionice to alter the i/o priority of the process used to zero a volume after deletion, for example "-c3" for idle only priority.
volume_clear_size = 0	(IntOpt) Size in MiB to wipe at start of old volumes. 0 => all
volume_copy_blkio_cgroup_name = <i>cinder-volume-copy</i>	(StrOpt) The blkio cgroup name to be used to limit bandwidth of volume copy
volume_copy_bps_limit = 0	(IntOpt) The upper limit of bandwidth of volume copy. 0 => unlimited
volume_dd_blocksize = <i>1M</i>	(StrOpt) The default block size used when copying/clearing volumes
volume_driver = <i>cinder.volume.drivers.lvm.LVMVolumeDriver</i>	(StrOpt) Driver to use for volume creation
volume_manager = <i>cinder.volume.manager.VolumeManager</i>	(StrOpt) Full class name for the Manager for volume
volume_service_inithost_offload = <i>False</i>	(BoolOpt) Offload pending volume delete during volume service startup
volume_usage_audit_period = <i>month</i>	(StrOpt) Time period for which to generate volume usages. The options are hour, day, month, or year.
volumes_dir = <i>\$state_path/volumes</i>	(StrOpt) Volume configuration file storage directory

Table 2.66. Description of Violin volume driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
gateway_mga = <i>None</i>	(StrOpt) IP address or hostname of mg-a
gateway_mgb = <i>None</i>	(StrOpt) IP address or hostname of mg-b
use_igroups = <i>False</i>	(BoolOpt) Use igroups to manage targets and initiators
violin_request_timeout = <i>300</i>	(IntOpt) Global backend request timeout, in seconds.

Table 2.67. Description of zones configuration options

Configuration option = Default value	Description
[DEFAULT]	
cloned_volume_same_az = <i>True</i>	(BoolOpt) Ensure that the new volumes are the same AZ as snapshot or source volume

2.7. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR OPENSTACK BLOCK STORAGE

Table 2.68. New options

Configuration option = Default value	Description
[DEFAULT] backup_gcs_block_size = 32768	(IntOpt) The size in bytes that changes are tracked for incremental backups. <code>backup_gcs_object_size</code> has to be multiple of <code>backup_gcs_block_size</code> .
[DEFAULT] backup_gcs_bucket = None	(StrOpt) The GCS bucket to use.
[DEFAULT] backup_gcs_bucket_location = US	(StrOpt) Location of GCS bucket.
[DEFAULT] backup_gcs_credential_file = None	(StrOpt) Absolute path of GCS service account credential file.
[DEFAULT] backup_gcs_enable_progress_timer = True	(BoolOpt) Enable or Disable the timer to send the periodic progress notifications to Ceilometer when backing up the volume to the GCS backend storage. The default value is True to enable the timer.
[DEFAULT] backup_gcs_num_retries = 3	(IntOpt) Number of times to retry.

Configuration option = Default value	Description
[DEFAULT] backup_gcs_object_size = 52428800	(IntOpt) The size in bytes of GCS backup objects.
[DEFAULT] backup_gcs_project_id = None	(StrOpt) Owner project id for GCS bucket.
[DEFAULT] backup_gcs_reader_chunk_size = 2097152	(IntOpt) GCS object will be downloaded in chunks of bytes.
[DEFAULT] backup_gcs_retry_error_codes = 429	(ListOpt) List of GCS error codes.
[DEFAULT] backup_gcs_storage_class = NEARLINE	(StrOpt) Storage class of GCS bucket.
[DEFAULT] backup_gcs_user_agent = gcscinder	(StrOpt) Http user-agent string for gcs api.
[DEFAULT] backup_gcs_writer_chunk_size = 2097152	(IntOpt) GCS object will be uploaded in chunks of bytes. Pass in a value of -1 if the file is to be uploaded as a single chunk.
[DEFAULT] backup_swift_auth_insecure = False	(BoolOpt) Bypass verification of server certificate when making SSL connection to Swift.
[DEFAULT] backup_swift_auth_url = None	(StrOpt) The URL of the Keystone endpoint
[DEFAULT] backup_use_same_host = False	(BoolOpt) Backup services use same backend.
[DEFAULT] cb_update_file_system = compression, sync, noofcopies, readonly	(ListOpt) These values will be used for CloudByte storage's updateFileSystem API call.
[DEFAULT] cb_update_qos_group = iops, latency, graceallowed	(ListOpt) These values will be used for CloudByte storage's updateQosGroup API call.
[DEFAULT] cinder_eternus_config_file = /etc/cinder/cinder_fujitsu_eternus_dx.xml	(StrOpt) config file for cinder eternus_dx volume driver
[DEFAULT] clone_check_timeout = 3600	(IntOpt) How long we check whether a clone is finished before we give up

Configuration option = Default value	Description
[DEFAULT] coho_rpc_port = 2049	(IntOpt) RPC port to connect to Coha Data MicroArray
[DEFAULT] disco_client = 127.0.0.1	(IPOpt) The IP of DMS client socket server
[DEFAULT] disco_client_port = 9898	(PortOpt) The port to connect DMS client socket server
[DEFAULT] disco_wsdl_path = /etc/cinder/DISCOService.wsdl	(StrOpt) Path to the wsdl file to communicate with DISCO request manager
[DEFAULT] drbdmanage_devs_on_controller = True	(BoolOpt) If set, the c-vol node will receive a useable /dev/drbdX device, even if the actual data is stored on other nodes only. This is useful for debugging, maintenance, and to be able to do the iSCSI export from the c-vol node.
[DEFAULT] drbdmanage_resize_plugin = drbdmanage.plugins.plugins.wait_for.WaitForVolumeSize	(StrOpt) Volume resize completion wait plugin.
[DEFAULT] drbdmanage_resize_policy = {"timeout": "60"}	(StrOpt) Volume resize completion wait policy.
[DEFAULT] drbdmanage_resource_plugin = drbdmanage.plugins.plugins.wait_for.WaitForResource	(StrOpt) Resource deployment completion wait plugin.
[DEFAULT] drbdmanage_resource_policy = {"ratio": "0.51", "timeout": "60"}	(StrOpt) Resource deployment completion wait policy.
[DEFAULT] drbdmanage_snapshot_plugin = drbdmanage.plugins.plugins.wait_for.WaitForSnapshot	(StrOpt) Snapshot completion wait plugin.
[DEFAULT] drbdmanage_snapshot_policy = {"count": "1", "timeout": "60"}	(StrOpt) Snapshot completion wait policy.
[DEFAULT] driver_ssl_cert_path = None	(StrOpt) Can be used to specify a non default path to a CA_BUNDLE file or directory with certificates of trusted CAs, which will be used to validate the backend
[DEFAULT] enable_v3_api = True	(BoolOpt) Deploy v3 of the Cinder API.

Configuration option = Default value	Description
[DEFAULT] glance_catalog_info = image:glance:publicURL	(StrOpt) Info to match when looking for glance in the service catalog. Format is: separated values of the form: <service_type>:<service_name>:<endpoint_type> - Only used if glance_api_servers are not provided.
[DEFAULT] hpe3par_api_url =	(StrOpt) 3PAR WSAPI Server Url like https://<3par ip>:8080/api/v1
[DEFAULT] hpe3par_cpg = OpenStack	(ListOpt) List of the CPG(s) to use for volume creation
[DEFAULT] hpe3par_cpg_snap =	(StrOpt) The CPG to use for Snapshots for volumes. If empty the userCPG will be used.
[DEFAULT] hpe3par_debug = False	(BoolOpt) Enable HTTP debugging to 3PAR
[DEFAULT] hpe3par_iscsi_chap_enabled = False	(BoolOpt) Enable CHAP authentication for iSCSI connections.
[DEFAULT] hpe3par_iscsi_ips =	(ListOpt) List of target iSCSI addresses to use.
[DEFAULT] hpe3par_password =	(StrOpt) 3PAR password for the user specified in hpe3par_username
[DEFAULT] hpe3par_snapshot_expiration =	(StrOpt) The time in hours when a snapshot expires and is deleted. This must be larger than expiration
[DEFAULT] hpe3par_snapshot_retention =	(StrOpt) The time in hours to retain a snapshot. You can't delete it before this expires.
[DEFAULT] hpe3par_username =	(StrOpt) 3PAR username with the 'edit' role
[DEFAULT] hpelefthand_api_url = None	(StrOpt) HPE LeftHand WSAPI Server Url like https://<LeftHand ip>:8081/lhos
[DEFAULT] hpelefthand_clustername = None	(StrOpt) HPE LeftHand cluster name
[DEFAULT] hpelefthand_debug = False	(BoolOpt) Enable HTTP debugging to LeftHand
[DEFAULT] hpelefthand_iscsi_chap_enabled = False	(BoolOpt) Configure CHAP authentication for iSCSI connections (Default: Disabled)
[DEFAULT] hpelefthand_password = None	(StrOpt) HPE LeftHand Super user password

Configuration option = Default value	Description
[DEFAULT] hpelefthand_ssh_port = 16022	(PortOpt) Port number of SSH service.
[DEFAULT] hpelefthand_username = None	(StrOpt) HPE LeftHand Super user username
[DEFAULT] hpexp_async_copy_check_interval = 10	(IntOpt) Interval to check copy asynchronously
[DEFAULT] hpexp_compute_target_ports = None	(ListOpt) Target port names of compute node for host group or iSCSI target
[DEFAULT] hpexp_copy_check_interval = 3	(IntOpt) Interval to check copy
[DEFAULT] hpexp_copy_speed = 3	(IntOpt) Copy speed of storage system
[DEFAULT] hpexp_default_copy_method = FULL	(StrOpt) Default copy method of storage system. There are two valid values: "FULL" specifies that a full copy; "THIN" specifies that a thin copy. Default value is "FULL"
[DEFAULT] hpexp_group_request = False	(BoolOpt) Request for creating host group or iSCSI target
[DEFAULT] hpexp_horcm_add_conf = True	(BoolOpt) Add to HORCM configuration
[DEFAULT] hpexp_horcm_name_only_discovery = False	(BoolOpt) Only discover a specific name of host group or iSCSI target
[DEFAULT] hpexp_horcm_numbers = 200, 201	(ListOpt) Instance numbers for HORCM
[DEFAULT] hpexp_horcm_resource_name = meta_resource	(StrOpt) Resource group name of storage system for HORCM
[DEFAULT] hpexp_horcm_user = None	(StrOpt) Username of storage system for HORCM
[DEFAULT] hpexp_ldev_range = None	(StrOpt) Logical device range of storage system
[DEFAULT] hpexp_pool = None	(StrOpt) Pool of storage system
[DEFAULT] hpexp_storage_cli = None	(StrOpt) Type of storage command line interface
[DEFAULT] hpexp_storage_id = None	(StrOpt) ID of storage system

Configuration option = Default value	Description
[DEFAULT] hpexp_target_ports = None	(ListOpt) Target port names for host group or iSCSI target
[DEFAULT] hpexp_thin_pool = None	(StrOpt) Thin pool of storage system
[DEFAULT] hpexp_zoning_request = False	(BoolOpt) Request for FC Zone creating host group
[DEFAULT] hypermetro_devices = None	(StrOpt) The remote device hypermetro will use.
[DEFAULT] keystone_catalog_info = identity:Identity Service:publicURL	(StrOpt) Info to match when looking for keystone in the service catalog. Format is: separated values of the form: <service_type>:<service_name>:<endpoint_type> - Only used if backup_swift_auth_url is unset
[DEFAULT] lvm_max_over_subscription_ratio = 1.0	(FloatOpt) max_over_subscription_ratio setting for the LVM driver. If set, this takes precedence over the general max_over_subscription_ratio option. If None, the general option is used.
[DEFAULT] nexenta_blocksize = 4096	(IntOpt) Block size for datasets
[DEFAULT] nexenta_chunksize = 16384	(IntOpt) NexentaEdge iSCSI LUN object chunk size
[DEFAULT] nexenta_client_address =	(StrOpt) NexentaEdge iSCSI Gateway client address for non-VIP service
[DEFAULT] nexenta_dataset_compression = on	(StrOpt) Compression value for new ZFS folders.
[DEFAULT] nexenta_dataset_dedup = off	(StrOpt) Deduplication value for new ZFS folders.
[DEFAULT] nexenta_dataset_description =	(StrOpt) Human-readable description for the folder.
[DEFAULT] nexenta_host =	(StrOpt) IP address of Nexenta SA
[DEFAULT] nexenta_iscsi_service =	(StrOpt) NexentaEdge iSCSI service name
[DEFAULT] nexenta_iscsi_target_portal_port = 3260	(IntOpt) Nexenta target portal port
[DEFAULT] nexenta_lun_container =	(StrOpt) NexentaEdge logical path of bucket for LUNs

Configuration option = Default value	Description
[DEFAULT] nexenta_mount_point_base = \$state_path/mnt	(StrOpt) Base directory that contains NFS share mount points
[DEFAULT] nexenta_nms_cache_volroot = True	(BoolOpt) If set True cache NexentaStor appliance volroot option value.
[DEFAULT] nexenta_ns5_blocksize = 32	(IntOpt) Block size for datasets
[DEFAULT] nexenta_password = nexenta	(StrOpt) Password to connect to Nexenta SA
[DEFAULT] nexenta_rest_address =	(StrOpt) IP address of NexentaEdge management REST API endpoint
[DEFAULT] nexenta_rest_password = nexenta	(StrOpt) Password to connect to NexentaEdge
[DEFAULT] nexenta_rest_port = 8080	(IntOpt) HTTP port to connect to Nexenta REST API server
[DEFAULT] nexenta_rest_protocol = auto	(StrOpt) Use http or https for REST connection (default auto)
[DEFAULT] nexenta_rest_user = admin	(StrOpt) User name to connect to NexentaEdge
[DEFAULT] nexenta_rrmgr_compression = 0	(IntOpt) Enable stream compression, level 1..9. 1 - gives best speed; 9 - gives best compression.
[DEFAULT] nexenta_rrmgr_connections = 2	(IntOpt) Number of TCP connections.
[DEFAULT] nexenta_rrmgr_tcp_buf_size = 4096	(IntOpt) TCP Buffer size in KiloBytes.
[DEFAULT] nexenta_shares_config = /etc/cinder/nfs_shares	(StrOpt) File with the list of available nfs shares
[DEFAULT] nexenta_sparse = False	(BoolOpt) Enables or disables the creation of sparse datasets
[DEFAULT] nexenta_sparsed_volumes = True	(BoolOpt) Enables or disables the creation of volumes as sparsed files that take no space. If disabled (False), volume is created as a regular file, which takes a long time.
[DEFAULT] nexenta_target_group_prefix = cinder/	(StrOpt) Prefix for iSCSI target groups on SA

Configuration option = Default value	Description
[DEFAULT] nexenta_target_prefix = iqn.1986-03.com.sun:02:cinder-	(StrOpt) IQN prefix for iSCSI targets
[DEFAULT] nexenta_user = admin	(StrOpt) User name to connect to Nexenta SA
[DEFAULT] nexenta_volume = cinder	(StrOpt) SA Pool that holds all volumes
[DEFAULT] nexenta_volume_group = iscsi	(StrOpt) Volume group for ns5
[DEFAULT] pure_automatic_max_oversubscription_ratio = True	(BoolOpt) Automatically determine an oversubscription ratio based on the current total data reduction values. If used this calculated value will override the max_over_subscription_ratio config option.
[DEFAULT] pure_eradicate_on_delete = False	(BoolOpt) When enabled, all Pure volumes, snapshots, and protection groups will be eradicated at the time of deletion in Cinder. Data will NOT be recoverable after a delete with this set to True! When disabled, volumes and snapshots will go into pending eradication state and can be recovered.
[DEFAULT] pure_replica_interval_default = 900	(IntOpt) Snapshot replication interval in seconds.
[DEFAULT] pure_replica_retention_long_term_default = 7	(IntOpt) Retain snapshots per day on target for this time (in days.)
[DEFAULT] pure_replica_retention_long_term_per_day_default = 3	(IntOpt) Retain how many snapshots for each day.
[DEFAULT] pure_replica_retention_short_term_default = 14400	(IntOpt) Retain all snapshots on target for this time (in seconds.)
[DEFAULT] replication_device = None	(MultiOpt) Multi opt of dictionaries to represent a replication target device. This option may be specified multiple times in a single config section to specify multiple replication target devices. Each entry takes the standard dict config form: replication_device = target_device_id: <required>,key1:value1,key2:value2...

Configuration option = Default value	Description
[DEFAULT] report_discard_supported = False	(BoolOpt) Report to clients of Cinder that the backend supports discard (aka. trim/unmap). This will not actually change the behavior of the backend or the client directly, it will only notify that it can be used.
[DEFAULT] restore_check_timeout = 3600	(IntOpt) How long we check whether a restore is finished before we give up
[DEFAULT] retry_interval = 1	(IntOpt) How long we wait before retrying to get an item detail
[DEFAULT] sf_enable_vag = False	(BoolOpt) Utilize volume access groups on a per-tenant basis.
[DEFAULT] sf_volume_prefix = UUID-	(StrOpt) Create SolidFire volumes with this prefix. Volume names are of the form <sf_volume_prefix><cinder-volume-id>. The default is to use a prefix of 'UUID-'.
[DEFAULT] smbfs_allocation_info_file_path = \$state_path/allocation_data	(StrOpt) The path of the automatically generated file containing information about volume disk space allocation.
[DEFAULT] snapshot_check_timeout = 3600	(IntOpt) How long we check whether a snapshot is finished before we give up
[DEFAULT] storwize_san_secondary_ip = None	(StrOpt) Specifies secondary management IP or hostname to be used if san_ip is invalid or becomes inaccessible.
[DEFAULT] storwize_svc_flashcopy_rate = 50	(IntOpt) Specifies the Storwize FlashCopy copy rate to be used when creating a full volume copy. The default is rate is 50, and the valid rates are 1-100.
[DEFAULT] storwize_svc_vol_nofmtdisk = False	(BoolOpt) Specifies that the volume not be formatted during creation.
[DEFAULT] suppress_requests_ssl_warnings = False	(BoolOpt) Suppress requests library SSL certificate warnings.
[DEFAULT] tegile_default_pool = None	(StrOpt) Create volumes in this pool
[DEFAULT] tegile_default_project = None	(StrOpt) Create volumes in this project

Configuration option = Default value	Description
[DEFAULT] tintri_image_cache_expiry_days = 30	(IntOpt) Delete unused image snapshots older than mentioned days
[DEFAULT] tintri_image_shares_config = None	(StrOpt) Path to image nfs shares file
[DEFAULT] volume_name_prefix = openstack-	(StrOpt) Prefix before volume name to differentiate DISCO volume created through openstack and the other ones
[DEFAULT] xtremio_volumes_per_glance_cache = 100	(IntOpt) Number of volumes created from each cached glance image
[DEFAULT] zfssa_manage_policy = loose	(StrOpt) Driver policy for volume manage.
[BRCD_FABRIC_EXAMPLE] fc_fabric_ssh_cert_path =	(StrOpt) Local SSH certificate Path.
[BRCD_FABRIC_EXAMPLE] fc_southbound_protocol = HTTP	(StrOpt) South bound connector for the fabric.
[BRCD_FABRIC_EXAMPLE] fc_virtual_fabric_id = None	(StrOpt) Virtual Fabric ID.
[coordination] backend_url = file://\$state_path	(StrOpt) The backend URL to use for distributed coordination.
[coordination] heartbeat = 1.0	(FloatOpt) Number of seconds between heartbeats for distributed coordination.
[coordination] initial_reconnect_backoff = 0.1	(FloatOpt) Initial number of seconds to wait after failed reconnection.
[coordination] max_reconnect_backoff = 60.0	(FloatOpt) Maximum number of seconds between sequential reconnection retries.
[hyperv] force_volumeutils_v1 = False	(BoolOpt) DEPRECATED: Force V1 volume utility class
[profiler] enabled = False	(BoolOpt) Enables the profiling for all services on this node. Default value is False (fully disable the profiling feature). Possible values: * True: Enables the feature * False: Disables the feature. The profiling cannot be started via this project operations. If the profiling is triggered by another project, this project part will be empty.

Configuration option = Default value	Description
[profiler] hmac_keys = SECRET_KEY	(StrOpt) Secret key(s) to use for encrypting context data for performance profiling. This string value should have the following format: <key1>[,<key2>,...<keyn>], where each key is some random string. A user who triggers the profiling via the REST API has to set one of these keys in the headers of the REST API call to include profiling results of this node for this particular project. Both "enabled" flag and "hmac_keys" config options should be set to enable profiling. Also, to generate correct profiling information across all services at least one key needs to be consistent between OpenStack projects. This ensures it can be used from client side to generate the trace, containing information from all possible resources.

Table 2.69. New default values

Option	New default value	New default value
[DEFAULT] datera_api_version	<i>1</i>	<i>2</i>
[DEFAULT] datera_num_replicas	<i>3</i>	<i>1</i>
[DEFAULT] glance_api_servers	<i>\$glance_host:\$glance_port</i>	<i>None</i>
[DEFAULT] query_volume_filters	<i>name, status, metadata, availability_zone</i>	<i>name, status, metadata, availability_zone, bootable</i>
[DEFAULT] zoning_mode	<i>none</i>	<i>None</i>
[BRCD_FABRIC_EXAMPLE] zone_name_prefix	<i>None</i>	<i>openstack</i>
[fc-zone-manager] brcd_sb_connector	<i>cinder.zonemanager.drivers.brocade.brcd_fc_zone_client_cli.BrcdFCZoneClientCLI</i>	<i>HTTP</i>

Table 2.70. Deprecated options

Configuration option = Default value	Description
[DEFAULT] enable_v1_api	<i>None</i>

Configuration option = Default value	Description
[DEFAULT] enable_v2_api	None
[DEFAULT] eqlx_chap_login	[DEFAULT] chap_username
[DEFAULT] eqlx_chap_password	[DEFAULT] chap_password
[DEFAULT] eqlx_use_chap	[DEFAULT] use_chap_auth
[DEFAULT] host	[DEFAULT] backend_host
[DEFAULT] hp3par_api_url	[DEFAULT] hpe3par_api_url
[DEFAULT] hp3par_cpg	[DEFAULT] hpe3par_cpg
[DEFAULT] hp3par_cpg_snap	[DEFAULT] hpe3par_cpg_snap
[DEFAULT] hp3par_debug	[DEFAULT] hpe3par_debug
[DEFAULT] hp3par_iscsi_chap_enabled	[DEFAULT] hpe3par_iscsi_chap_enabled
[DEFAULT] hp3par_iscsi_ips	[DEFAULT] hpe3par_iscsi_ips
[DEFAULT] hp3par_password	[DEFAULT] hpe3par_password
[DEFAULT] hp3par_snapshot_expiration	[DEFAULT] hpe3par_snapshot_expiration
[DEFAULT] hp3par_snapshot_retention	[DEFAULT] hpe3par_snapshot_retention
[DEFAULT] hp3par_username	[DEFAULT] hpe3par_username
[DEFAULT] hplefthand_api_url	[DEFAULT] hpelefthand_api_url
[DEFAULT] hplefthand_clustername	[DEFAULT] hpelefthand_clustername
[DEFAULT] hplefthand_debug	[DEFAULT] hpelefthand_debug
[DEFAULT] hplefthand_iscsi_chap_enabled	[DEFAULT] hpelefthand_iscsi_chap_enabled
[DEFAULT] hplefthand_password	[DEFAULT] hpelefthand_password
[DEFAULT] hplefthand_username	[DEFAULT] hpelefthand_username
[DEFAULT] hpxp_async_copy_check_interval	[DEFAULT] hpexp_async_copy_check_interval

Configuration option = Default value	Description
[DEFAULT] <code>hpxp_compute_target_ports</code>	<i>[DEFAULT] hpexp_compute_target_ports</i>
[DEFAULT] <code>hpxp_copy_check_interval</code>	<i>[DEFAULT] hpexp_copy_check_interval</i>
[DEFAULT] <code>hpxp_copy_speed</code>	<i>[DEFAULT] hpexp_copy_speed</i>
[DEFAULT] <code>hpxp_default_copy_method</code>	<i>[DEFAULT] hpexp_default_copy_method</i>
[DEFAULT] <code>hpxp_group_request</code>	<i>[DEFAULT] hpexp_group_request</i>
[DEFAULT] <code>hpxp_horcm_add_conf</code>	<i>[DEFAULT] hpexp_horcm_add_conf</i>
[DEFAULT] <code>hpxp_horcm_name_only_discovery</code>	<i>[DEFAULT] hpexp_horcm_name_only_discovery</i>
[DEFAULT] <code>hpxp_horcm_numbers</code>	<i>[DEFAULT] hpexp_horcm_numbers</i>
[DEFAULT] <code>hpxp_horcm_resource_name</code>	<i>[DEFAULT] hpexp_horcm_resource_name</i>
[DEFAULT] <code>hpxp_horcm_user</code>	<i>[DEFAULT] hpexp_horcm_user</i>
[DEFAULT] <code>hpxp_ldev_range</code>	<i>[DEFAULT] hpexp_ldev_range</i>
[DEFAULT] <code>hpxp_pool</code>	<i>[DEFAULT] hpexp_pool</i>
[DEFAULT] <code>hpxp_storage_cli</code>	<i>[DEFAULT] hpexp_storage_cli</i>
[DEFAULT] <code>hpxp_storage_id</code>	<i>[DEFAULT] hpexp_storage_id</i>
[DEFAULT] <code>hpxp_target_ports</code>	<i>[DEFAULT] hpexp_target_ports</i>
[DEFAULT] <code>hpxp_thin_pool</code>	<i>[DEFAULT] hpexp_thin_pool</i>
[DEFAULT] <code>hpxp_zoning_request</code>	<i>[DEFAULT] hpexp_zoning_request</i>
[DEFAULT] <code>osapi_max_request_body_size</code>	<i>[oslo_middleware] max_request_body_size</i>
[DEFAULT] <code>use_syslog</code>	<i>None</i>
[hyperv] <code>force_volumeutils_v1</code>	<i>None</i>
[profiler] <code>profiler_enabled</code>	<i>[profiler] enabled</i>

[1] Volume extension is executable only when you use TPP as a storage pool.

[2] The configuration file location may differ.

[3] There is no relative precedence or weight among these four labels.

CHAPTER 3. COMPUTE

The OpenStack Compute service is a cloud computing fabric controller, which is the main part of an IaaS system. You can use OpenStack Compute to host and manage cloud computing systems. This section describes the OpenStack Compute configuration options.

To configure your Compute installation, you must define configuration options in these files:

- **nova.conf**. Contains most of the Compute configuration options. Resides in the `/etc/nova/` directory.
- **api-paste.ini**. Defines Compute limits. Resides in the `/etc/nova/` directory.
- Related Image service and Identity service management configuration files.

Ephemeral Storage Discrepancy with Ceph

When using Red Hat Ceph as a back end for ephemeral storage, the Compute service does not calculate the amount of available storage correctly. Specifically, Compute simply adds up the amount of available storage without factoring in replication. This results in grossly overstated available storage, which in turn could cause unexpected storage oversubscription.

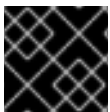
To determine the correct ephemeral storage capacity, query the Ceph service directly instead. For more information, see [BZ#1236473](#).

3.1. OVERVIEW OF NOVA.CONF

You can use a particular configuration option file by using the **option (nova.conf)** parameter when you run one of the **nova-*** services. This parameter inserts configuration option definitions from the specified configuration file name, which might be useful for debugging or performance tuning.

For a list of configuration options, see the tables in this guide.

To learn more about the **nova.conf** configuration file, review the general purpose configuration options documented in [Table 3.17, “Description of common configuration options”](#).



IMPORTANT

Do not specify quotes around Nova options.

Sections

Configuration options are grouped by section. The Compute configuration file supports the following sections:

[DEFAULT]

Contains most configuration options. If the documentation for a configuration option does not specify its section, assume that it appears in this section.

[baremetal]

Configures the baremetal hypervisor driver.

[cells]

Configures cells functionality. For details, see [Section 3.13, “Cells”](#).

[conductor]

Configures the **nova-conductor** service.

[database]

Configures the database that Compute uses.

[glance]

Configures how to access the Image service.

[image_file_url]

Configures additional filesystems to access the Image Service.

[keymgr]

Configures the key manager.

[keystone_authtoken]

Configures authorization via Identity service.

[libvirt]

Configures the hypervisor drivers using the Libvirt library: KVM, LXC, Qemu, UML, Xen.

[matchmaker_redis]

Configures a Redis server.

[matchmaker_ring]

Configures a matchmaker ring.

[metrics]

Configures weights for the metrics weighter.

[neutron]

Configures Networking specific options.

[osapi_v3]

Configures the OpenStack Compute API v3.

[rdp]

Configures RDP proxying.

[serial_console]

Configures serial console.

[spice]

Configures virtual consoles using SPICE.

[ssl]

Configures certificate authority using SSL.

[trusted_computing]

Configures the trusted computing pools functionality and how to connect to a remote attestation service.

[upgrade_levels]

Configures version locking on the RPC (message queue) communications between the various Compute services to allow live upgrading an OpenStack installation.

[vmware]

Configures the VMware hypervisor driver.

[xenserver]

Configures the XenServer hypervisor driver.

[zookeeper]

Configures the ZooKeeper ServiceGroup driver.

3.2. CONFIGURE LOGGING

You can use the `nova.conf` file to configure where Compute logs events, the level of logging, and log formats.

To customize log formats for OpenStack Compute, use the configuration option settings documented in [Table 3.35, “Description of logging configuration options”](#).

3.3. CONFIGURE AUTHENTICATION AND AUTHORIZATION

There are different methods of authentication for the OpenStack Compute project, including no authentication. The preferred system is the OpenStack Identity service, code-named Keystone.

To customize authorization settings for Compute, use the configuration options documented in [Table 3.11, “Description of authentication configuration options”](#).

To customize certificate authority settings for Compute, use the configuration options documented in [Table 3.15, “Description of CA and SSL configuration options”](#).

To customize Compute and the Identity service to use LDAP as a backend, refer to the configuration options documented in [Table 3.32, “Description of LDAP configuration options”](#).

3.4. CONFIGURE RESIZE

Resize (or Server resize) is the ability to change the flavor of a server, thus allowing it to upscale or downscale according to user needs. For this feature to work properly, you might need to configure some underlying virt layers.

3.4.1. KVM

Resize on KVM is implemented currently by transferring the images between compute nodes over ssh. For KVM you need hostnames to resolve properly and passwordless ssh access between your compute hosts. Direct access from one compute host to another is needed to copy the VM file across.

3.5. DATABASE CONFIGURATION

You can configure OpenStack Compute to use any SQLAlchemy-compatible database. The database name is **nova**. The **nova-conductor** service is the only service that writes to the database. The other Compute services access the database through the **nova-conductor** service.

To ensure that the database schema is current, run the following command:

```
# nova-manage db sync
```

If **nova-conductor** is not used, entries to the database are mostly written by the **nova-scheduler** service, although all services must be able to update entries in the database.

In either case, use the configuration option settings documented in [Table 3.22, “Description of database configuration options”](#) to configure the connection string for the nova database.

3.6. CONFIGURE THE OSLO RPC MESSAGING SYSTEM

OpenStack projects use AMQP, an open standard for messaging middleware. OpenStack services that run on multiple servers to talk to each other. OpenStack Oslo RPC supports two implementations of AMQP: **RabbitMQ** and **Qpid**.

3.6.1. Configure RabbitMQ

OpenStack Oslo RPC uses **RabbitMQ** by default. Use these options to configure the **RabbitMQ** message system. The **rpc_backend** option is not required as long as **RabbitMQ** is the default messaging system. However, if it is included the configuration, you must set it to **rabbit**.

```
rpc_backend=rabbit
```

You can use these additional options to configure the **RabbitMQ** messaging system. You can configure messaging communication for different installation scenarios, tune retries for RabbitMQ, and define the size of the RPC thread pool. To monitor notifications through RabbitMQ, you must set the **notification_driver** option to **nova.openstack.common.notifier.rpc_notifier** in the **nova.conf** file. The default for sending usage data is sixty seconds plus a random number of seconds from zero to sixty.

Table 3.1. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.

Configuration option = Default value	Description
fake_rabbit = <i>False</i>	(BoolOpt) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
heartbeat_rate = 2	(IntOpt) How often times during the <code>heartbeat_timeout_threshold</code> to check the heartbeat.
heartbeat_timeout_threshold = 0	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disables the heartbeat, >0 enables it. Enabling heartbeats requires <code>kombu>=3.0.7</code> and <code>amqp>=1.4.0</code>). EXPERIMENTAL
kombu_reconnect_delay = 1.0	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 are also available.
rabbit_ha_queues = <i>False</i>	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
rabbit_host = <i>localhost</i>	(StrOpt) The RabbitMQ broker address where a single node is used.
rabbit_hosts = <i>\$rabbit_host:\$rabbit_port</i>	(ListOpt) RabbitMQ HA cluster host:port pairs.
rabbit_login_method = <i>AMQPLAIN</i>	(StrOpt) The RabbitMQ login method.
rabbit_max_retries = 0	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = <i>guest</i>	(StrOpt) The RabbitMQ password.
rabbit_port = 5672	(IntOpt) The RabbitMQ broker port where a single node is used.

Configuration option = Default value	Description
<code>rabbit_retry_backoff = 2</code>	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
<code>rabbit_retry_interval = 1</code>	(IntOpt) How frequently to retry connecting with RabbitMQ.
<code>rabbit_use_ssl = False</code>	(BoolOpt) Connect over SSL for RabbitMQ.
<code>rabbit_userid = guest</code>	(StrOpt) The RabbitMQ userid.
<code>rabbit_virtual_host = /</code>	(StrOpt) The RabbitMQ virtual host.
<code>rpc_conn_pool_size = 30</code>	(IntOpt) Size of RPC connection pool.

3.6.2. Configure Qpid

Use these options to configure the **Qpid** messaging system for OpenStack Oslo RPC. **Qpid** is not the default messaging system, so you must enable it by setting the `rpc_backend` option in the `nova.conf` file.

```
rpc_backend=qpid
```

This critical option points the compute nodes to the **Qpid** broker (server). Set `qpid_hostname` to the host name where the broker runs in the `nova.conf` file.



NOTE

The `--qpid_hostname` parameter accepts a host name or IP address value.

```
qpid_hostname=hostname.example.com
```

If the **Qpid** broker listens on a port other than the AMQP default of **5672**, you must set the `qpid_port` option to that value:

```
qpid_port=12345
```

If you configure the **Qpid** broker to require authentication, you must add a user name and password to the configuration:

```
qpid_username=username
qpid_password=password
```

By default, TCP is used as the transport. To enable SSL, set the `qpid_protocol` option:

```
qpid_protocol=ssl
```

This table lists additional options that you use to configure the Qpid messaging driver for OpenStack Oslo RPC. These options are used infrequently.

Table 3.2. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
qpid_heartbeat = <i>60</i>	(IntOpt) Seconds between connection keepalive heartbeats.
qpid_hostname = <i>localhost</i>	(StrOpt) Qpid broker hostname.
qpid_hosts = <i>\$qpid_hostname:\$qpid_port</i>	(ListOpt) Qpid HA cluster host:port pairs.
qpid_password =	(StrOpt) Password for Qpid connection.
qpid_port = <i>5672</i>	(IntOpt) Qpid broker port.
qpid_protocol = <i>tcp</i>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
qpid_receiver_capacity = <i>1</i>	(IntOpt) The number of prefetched messages held by receiver.
qpid_sasl_mechanisms =	(StrOpt) Space separated list of SASL mechanisms to use for auth.
qpid_tcp_nodelay = <i>True</i>	(BoolOpt) Whether to disable the Nagle algorithm.
qpid_topology_version = <i>1</i>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
qpid_username =	(StrOpt) Username for Qpid connection.
rpc_conn_pool_size = <i>30</i>	(IntOpt) Size of RPC connection pool.

3.6.3. Configure messaging

Use these options to configure the **RabbitMQ** and **Qpid** messaging drivers.

Table 3.3. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>openstack</i>	(StrOpt) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
default_publisher_id = <i>None</i>	(StrOpt) Default <code>publisher_id</code> for outgoing notifications
notification_driver = <i>[]</i>	(MultiStrOpt) Driver or drivers to handle sending notifications.
notification_topics = <i>notifications</i>	(ListOpt) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(StrOpt) A URL representing the messaging driver to use and its full configuration. If not set, fall back to the <code>rpc_backend</code> option and driver specific configuration.

Table 3.4. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
matchmaker_heartbeat_freq = <i>300</i>	(IntOpt) Heartbeat frequency.
matchmaker_heartbeat_ttl = <i>600</i>	(IntOpt) Heartbeat time-to-live.
rpc_backend = <i>rabbit</i>	(StrOpt) The messaging driver to use, defaults to <code>rabbit</code> . Other drivers include <code>qpid</code> and <code>zmq</code> .
rpc_cast_timeout = <i>30</i>	(IntOpt) Seconds to wait before a cast expires (TTL). Only supported by <code>impl_zmq</code> .
rpc_response_timeout = <i>60</i>	(IntOpt) Seconds to wait for a response from a call.
rpc_thread_pool_size = <i>64</i>	(IntOpt) Size of RPC thread pool.
[cells]	
rpc_driver_queue_base = <i>cells.intercell</i>	(StrOpt) Base queue name to use when communicating between cells. Various topics by message type will be appended to this.

Configuration option = Default value		Description
[oslo_concurrency]		
disable_process_locking = <i>False</i>		(BoolOpt) Enables or disables inter-process locks.
lock_path = <i>None</i>		(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging_amqp]		
allow_insecure_clients = <i>False</i>		(BoolOpt) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>		(StrOpt) address prefix used when broadcasting to all servers
container_name = <i>None</i>		(StrOpt) Name for the AMQP container
group_request_prefix = <i>unicast</i>		(StrOpt) address prefix when sending to any server in group
idle_timeout = <i>0</i>		(IntOpt) Timeout for inactive connections (in seconds)
server_request_prefix = <i>exclusive</i>		(StrOpt) address prefix used when sending to a specific server
ssl_ca_file =		(StrOpt) CA certificate PEM file for verifying server certificate
ssl_cert_file =		(StrOpt) Identifying certificate PEM file to present to clients
ssl_key_file =		(StrOpt) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>		(StrOpt) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>		(BoolOpt) Debug: dump AMQP frames to stdout
[upgrade_levels]		

Configuration option = Default value	Description
baseapi = <i>None</i>	(StrOpt) Set a version cap for messages sent to the base api in any service

3.7. CONFIGURE THE COMPUTE API

The Compute API, run by the **nova-api** daemon, is the component of OpenStack Compute that receives and responds to user requests, whether they be direct API calls, or via the CLI tools or dashboard.

Configure Compute API password handling

The OpenStack Compute API enables users to specify an administrative password when they create or rebuild a server instance. If the user does not specify a password, a random password is generated and returned in the API response.

In practice, how the admin password is handled depends on the hypervisor in use and might require additional configuration of the instance. For example, you might have to install an agent to handle the password setting. If the hypervisor and instance configuration do not support setting a password at server create time, the password that is returned by the create API call is misleading because it was ignored.

To prevent this confusion, use the **enable_instance_password** configuration option to disable the return of the admin password for installations that do not support setting instance passwords.

Configure Compute API rate limiting

OpenStack Compute supports API rate limiting for the OpenStack API. The rate limiting allows an administrator to configure limits on the type and number of API calls that can be made in a specific time interval.

When API rate limits are exceeded, HTTP requests return an error with a status code of 403 Forbidden.

Rate limiting is not available for the EC2 API.

Define limits

To define limits, set these values:

- The **HTTP method** used in the API call, typically one of GET, PUT, POST, or DELETE.
- A **human readable URI** that is used as a friendly description of where the limit is applied.
- A **regular expression**. The limit is applied to all URIs that match the regular expression and HTTP method.
- A **limit value** that specifies the maximum count of units before the limit takes effect.
- An **interval** that specifies time frame to which the limit is applied. The interval can be SECOND, MINUTE, HOUR, or DAY.

Rate limits are applied in relative order to the HTTP method, going from least to most specific.

Default limits

Normally, you install OpenStack Compute with the following limits enabled:

Table 3.5. Default API rate limits

HTTP method	API URI	API regular expression	Limit
POST	any URI (*)	.*	120 per minute
POST	/servers	^/servers	120 per minute
PUT	any URI (*)	.*	120 per minute
GET	*changes-since*	.*changes-since.*	120 per minute
DELETE	any URI (*)	.*	120 per minute
GET	*/os-fping	^/os-fping	12 per minute

Configure and change limits

As part of the WSGI pipeline, the `/etc/nova/api-paste.ini` file defines the actual limits.

To enable limits, include the `ratelimit` filter in the API pipeline specification. If the `ratelimit` filter is removed from the pipeline, limiting is disabled. You must also define the rate limit filter. The lines appear as follows:

```
[pipeline:openstack_compute_api_v2]
pipeline = faultwrap authtoken keystonecontext ratelimit
osapi_compute_app_v2

[pipeline:openstack_volume_api_v1]
pipeline = faultwrap authtoken keystonecontext ratelimit
osapi_volume_app_v1

[filter:ratelimit]
paste.filter_factory =
nova.api.openstack.compute.limits:RateLimitingMiddleware.factory
```

To modify the limits, add a `limits` specification to the `[filter:ratelimit]` section of the file. Specify the limits in this order:

1. HTTP method
2. friendly URI
3. regex
4. limit
5. interval

The following example shows the default rate-limiting values:

```
[filter:ratelimit]
paste.filter_factory =
nova.api.openstack.compute.limits:RateLimitingMiddleware.factory
limits =(POST, "*", .*, 120, MINUTE);(POST, "*/servers", ^/servers, 120,
MINUTE);(PUT, "*", .*, 120, MINUTE);(GET, "*changes-since*", .*changes-
since.*, 120, MINUTE);(DELETE, "*", .*, 120, MINUTE);(GET, "*/os-fping",
^/os-fping, 12, MINUTE)
```

Configuration reference

The Compute API configuration options are documented in [Table 3.9, “Description of API configuration options”](#).

3.8. CONFIGURE THE EC2 API

You can set options in the `nova.conf` configuration file to control which network address and port the EC2 API listens on, the formatting of some API responses, and authentication related options.

To customize these options for OpenStack EC2 API, use the configuration option settings documented in [Table 3.24, “Description of EC2 configuration options”](#).

3.9. FIBRE CHANNEL SUPPORT IN COMPUTE

Fibre Channel support in OpenStack Compute is remote block storage attached to compute nodes for VMs.

In the Grizzly release, Fibre Channel supported only the KVM hypervisor.

Compute and Block Storage support Fibre Channel automatic zoning on Brocade and Cisco switches. On other hardware Fibre Channel arrays must be pre-zoned or directly attached to the KVM hosts.

3.9.1. KVM host requirements

You must install these packages on the KVM host:

- `sysfsutils` - Nova uses the `systool` application in this package.
- `sg3-utils` or `sg3_utils` - Nova uses the `sg_scan` and `sginfo` applications.

Installing the `multipath-tools` package is optional.

3.9.2. Install required packages

Use this command to install the system packages:

```
# yum install sysfsutils sg3_utils multipath-tools
```

3.10. ISCSI INTERFACE AND OFFLOAD SUPPORT IN COMPUTE

**NOTE**

iSCSI interface and offload support is only present since Kilo.

Compute supports open-iscsi iSCSI interfaces for offload cards. Offload hardware must be present and configured on every compute node where offload is desired. Once an open-iscsi interface is configured, the iface name (`iface.iscsi_ifacename`) should be passed to libvirt via the `iscsi_iface` parameter for use. All iscsi sessions will be bound to this iSCSI interface.

Currently supported transports (`iface.transport_name`) are `be2iscsi`, `bnx2i`, `cxgb3i`, `cxgb4i`, `qla4xxx`, `ocs`. No configuration changes are needed outside of Compute node.

iSER is currently supported via the separate iSER LibvirtISERVVolumeDriver and will be rejected if used via the `iscsi_iface` parameter.

3.10.1. iSCSI iface configuration

- Note the distinction between the transport name (`iface.transport_name`) and iface name (`iface.iscsi_ifacename`). The actual iface name must be specified via the `iscsi_iface` parameter to libvirt for offload to work.
- The default name for an iscsi iface (open-iscsi parameter `iface.iscsi_ifacename`) is in the format `transport_name.hwaddress` when generated by `iscsiadm`.
- `iscsiadm` can be used to view and generate current iface configuration. Every network interface that supports an open-iscsi transport can have one or more iscsi ifaces associated with it. If no ifaces have been configured for a network interface supported by an open-iscsi transport, this command will create a default iface configuration for that network interface. For example :

```
# iscsiadm -m iface
  default tcp,<empty>,<empty>,<empty>,<empty>
  iser iser,<empty>,<empty>,<empty>,<empty>
  bnx2i.00:05:b5:d2:a0:c2 bnx2i,00:05:b5:d2:a0:c2,5.10.10.20,
<empty>,<empty>
  cxgb4i.00:07:43:28:b2:58 cxgb4i,00:07:43:28:b2:58,102.50.50.80,
<empty>,<empty>
  qla4xxx.00:c0:dd:08:63:ea qla4xxx,00:c0:dd:08:63:ea,20.15.0.9,
<empty>,<empty>
```

The output is in the format : `iface_name`
`transport_name,hwaddress,ipaddress,net_ifacename,initiatorname`.

- Individual iface configuration can be viewed via

```
# iscsiadm -m iface -I IFACE_NAME
# BEGIN RECORD 2.0-873
iface.iscsi_ifacename = cxgb4i.00:07:43:28:b2:58
iface.net_ifacename = <empty>
iface.ipaddress = 102.50.50.80
iface.hwaddress = 00:07:43:28:b2:58
iface.transport_name = cxgb4i
iface.initiatorname = <empty>
# END RECORD
```


Configuration can be updated as desired via

```
# iscsiadm -m iface -I IFACE_NAME --op=update -n iface.SETTING -v
VALUE
```

- All iface configurations need a minimum of `iface.iface_name`, `iface.transport_name` and `iface.hwaddress` to be correctly configured to work. Some transports may require `iface.ipaddress` and `iface.net_ifacename` as well to bind correctly.

Detailed configuration instructions can be found in the [Linux* Open-iSCSI README file](#).

3.11. HYPERVISORS

Red Hat OpenStack Platform is only supported for use with the libvirt driver (using KVM as the hypervisor on Compute nodes) or the VMware vCenter hypervisor driver. See the VMware Integration Guide for more information regarding the configuration of the VMware vCenter driver.

With this release of Red Hat OpenStack Platform, Ironic is now fully supported. Ironic allows you to provision bare-metal machines using common technologies (such as PXE boot and IPMI) to cover a wide range of hardware while supporting pluggable drivers to allow the addition of vendor-specific functionality.

Red Hat does not provide support for other Compute virtualization drivers such as the deprecated VMware "direct-to-ESX" hypervisor, and non-KVM libvirt hypervisors.

3.11.1. Hypervisor configuration basics

The node where the `nova-compute` service is installed and operates on the same node that runs all of the virtual machines. This is referred to as the compute node in this guide.

By default, the selected hypervisor is KVM. To change to another hypervisor, change the `virt_type` option in the `[libvirt]` section of `nova.conf` and restart the `nova-compute` service.

Here are the general `nova.conf` options that are used to configure the compute node's hypervisor: [Table 3.28, "Description of hypervisor configuration options"](#).

Specific options for particular hypervisors can be found in the following sections.

3.11.2. KVM

KVM is configured as the default hypervisor for Compute.



NOTE

This document contains several sections about hypervisor selection. If you are reading this document linearly, you do not want to load the KVM module before you install `nova-compute`. The `nova-compute` service depends on `qemu-kvm`, which installs `/lib/udev/rules.d/45-qemu-kvm.rules`, which sets the correct permissions on the `/dev/kvm` device node.

To enable KVM explicitly, add the following configuration options to the `/etc/nova/nova.conf` file:

```
compute_driver = libvirt.LibvirtDriver
```

```
[libvirt]
virt_type = kvm
```

The KVM hypervisor supports the following virtual machine image formats:

- Raw
- QEMU Copy-on-write (qcow2)
- QED Qemu Enhanced Disk
- VMware virtual machine disk format (vmdk)

This section describes how to enable KVM on your system. For more information, see [Installing virtualization packages on an existing Red Hat Enterprise Linux system](#) from the *Red Hat Enterprise Linux Virtualization Host Configuration and Guest Installation Guide*.

3.11.2.1. Enable KVM

The following sections outline how to enable KVM based hardware virtualisation on different architectures and platforms. To perform these steps, you must be logged in as the **root** user.

3.11.2.1.1. For x86 based systems

1. To determine whether the **svm** or **vmx** CPU extensions are present, run this command:

```
# grep -E 'svm|vmx' /proc/cpuinfo
```

This command generates output if the CPU is capable of hardware virtualization. Even if output is shown, you might still need to enable virtualization in the system BIOS for full support.

If no output appears, consult your system documentation to ensure that your CPU and motherboard support hardware virtualization. Verify that any relevant hardware virtualization options are enabled in the system BIOS.

The BIOS for each manufacturer is different. If you must enable virtualization in the BIOS, look for an option containing the words **virtualization**, **VT**, **VMX**, or **SVM**.

2. To list the loaded kernel modules and verify that the **kvm** modules are loaded, run this command:

```
# lsmod | grep kvm
```

If the output includes **kvm_intel** or **kvm_amd**, the **kvm** hardware virtualization modules are loaded and your kernel meets the module requirements for OpenStack Compute.

If the output does not show that the **kvm** module is loaded, run this command to load it:

```
# modprobe -a kvm
```

Run the command for your CPU. For Intel, run this command:

```
# modprobe -a kvm-intel
```

For AMD, run this command:

```
# modprobe -a kvm-amd
```

Because a KVM installation can change user group membership, you might need to log in again for changes to take effect.

If the kernel modules do not load automatically, use the procedures listed in these subsections.

If the checks indicate that required hardware virtualization support or kernel modules are disabled or unavailable, you must either enable this support on the system or find a system with this support.



NOTE

Some systems require that you enable VT support in the system BIOS. If you believe your processor supports hardware acceleration but the previous command did not produce output, reboot your machine, enter the system BIOS, and enable the VT option.

If KVM acceleration is not supported, configure Compute to use a different hypervisor, such as [QEMU](#) or [Xen](#).

These procedures help you load the kernel modules for Intel-based and AMD-based processors if they do not load automatically during KVM installation.

3.11.2.1.1. Intel-based processors

If your compute host is Intel-based, run these commands as root to load the kernel modules:

```
# modprobe kvm
# modprobe kvm-intel
```

See [Persistent Module Loading in Red Hat Enterprise Linux 6](#) , or [Persistent Module Loading in Red Hat Enterprise Linux 7](#) respectively, for instructions on how to load the `kvm` and `kvm-amd` modules automatically.

3.11.2.1.2. AMD-based processors

If your compute host is AMD-based, run these commands as root to load the kernel modules:

```
# modprobe kvm
# modprobe kvm-amd
```

See [Persistent Module Loading in Red Hat Enterprise Linux 6](#) , or [Persistent Module Loading in Red Hat Enterprise Linux 7](#) respectively, for instructions on how to load the `kvm` and `kvm-intel` modules automatically.

3.11.2.1.2. For POWER based systems

KVM as a hypervisor is supported on POWER system's PowerNV platform.

1. To determine if your POWER platform supports KVM based virtualization run the following command:

■

```
# grep PowerNV /proc/cpuinfo
```

If the previous command generates the following output, then CPU supports KVM based virtualization

```
platform: PowerNV
```

If no output is displayed, then your POWER platform does not support KVM based hardware virtualization.

2. To list the loaded kernel modules and verify that the `kvm` modules are loaded, run the following command:

```
# lsmod | grep kvm
```

If the output includes `kvm_hv`, the `kvm` hardware virtualization modules are loaded and your kernel meets the module requirements for OpenStack Compute.

If the output does not show that the `kvm` module is loaded, run the following command to load it:

```
# modprobe -a kvm
```

For PowerNV platform, run the following command:

```
# modprobe -a kvm-hv
```

Because a KVM installation can change user group membership, you might need to log in again for changes to take effect.

3.11.2.2. Specify the CPU model of KVM guests

The Compute service enables you to control the guest CPU model that is exposed to KVM virtual machines. Use cases include:

- To maximize performance of virtual machines by exposing new host CPU features to the guest
- To ensure a consistent default CPU across all machines, removing reliance of variable QEMU defaults

In libvirt, the CPU is specified by providing a base CPU model name (which is a shorthand for a set of feature flags), a set of additional feature flags, and the topology (sockets/cores/threads). The libvirt KVM driver provides a number of standard CPU model names. These models are defined in the `/usr/share/libvirt/cpu_map.xml` file. Check this file to determine which models are supported by your local installation.

Two Compute configuration options in the `[libvirt]` group of `nova.conf` define which type of CPU model is exposed to the hypervisor when using KVM: `cpu_mode` and `cpu_model`.

The `cpu_mode` option can take one of the following values: `none`, `host-passthrough`, `host-model`, and `custom`.

Host model (default for KVM & QEMU)

If your `nova.conf` file contains `cpu_mode=host-model`, libvirt identifies the CPU model in `/usr/share/libvirt/cpu_map.xml` file that most closely matches the host, and requests additional CPU flags to complete the match. This configuration provides the maximum functionality and performance and maintains good reliability and compatibility if the guest is migrated to another host with slightly different host CPUs.

Host pass through

If your `nova.conf` file contains `cpu_mode=host-passthrough`, libvirt tells KVM to pass through the host CPU with no modifications. The difference to `host-model`, instead of only matching feature flags, every last detail of the host CPU is matched. This gives the best performance, and can be important to some apps which check low level CPU details, but it comes at a cost with respect to migration. The guest can only be migrated to a matching host CPU.

Custom

If your `nova.conf` file contains `cpu_mode=custom`, you can explicitly specify one of the supported named models using the `cpu_model` configuration option. For example, to configure the KVM guests to expose Nehalem CPUs, your `nova.conf` file should contain:

```
[libvirt]
cpu_mode = custom
cpu_model = Nehalem
```

None (default for all libvirt-driven hypervisors other than KVM & QEMU)

If your `nova.conf` file contains `cpu_mode=none`, libvirt does not specify a CPU model. Instead, the hypervisor chooses the default model.

3.11.2.3. Guest agent support

Use guest agents to enable optional access between compute nodes and guests through a socket, using the QMP protocol.

To enable this feature, you must set `hw_qemu_guest_agent=yes` as a metadata parameter on the image you want to use to create the guest-agent-capable instances from. You can explicitly disable the feature by setting `hw_qemu_guest_agent=no` in the image metadata.

3.11.2.4. KVM performance tweaks

The [VHostNet](#) kernel module improves network performance. To load the kernel module, run the following command as root:

```
# modprobe vhost_net
```

3.11.2.5. Troubleshoot KVM

Trying to launch a new virtual machine instance fails with the `ERROR` state, and the following error appears in the `/var/log/nova/nova-compute.log` file:

```
libvirtError: internal error no supported architecture for os type 'hvm'
```

This message indicates that the KVM kernel modules were not loaded.

If you cannot start VMs after installation without rebooting, the permissions might not be set correctly. This can happen if you load the KVM module before you install `nova-compute`. To check whether the group is set to `kvm`, run:

```
# ls -l /dev/kvm
```

If it is not set to `kvm`, run:

```
# udevadm trigger
```

3.11.3. QEMU

From the perspective of the Compute service, the QEMU hypervisor is very similar to the KVM hypervisor. Both are controlled through libvirt, both support the same feature set, and all virtual machine images that are compatible with KVM are also compatible with QEMU. The main difference is that QEMU does not support native virtualization. Consequently, QEMU has worse performance than KVM and is a poor choice for a production deployment.

The typical uses cases for QEMU are

- Running on older hardware that lacks virtualization support.
- Running the Compute service inside of a virtual machine for development or testing purposes, where the hypervisor does not support native virtualization for guests.

To enable QEMU, add these settings to `nova.conf`:

```
compute_driver = libvirt.LibvirtDriver

[libvirt]
virt_type = qemu
```

For some operations you may also have to install the `guestmount` utility:

```
# yum install libguestfs-tools
```

The QEMU hypervisor supports the following virtual machine image formats:

- Raw
- QEMU Copy-on-write (qcow2)
- VMware virtual machine disk format (vmdk)

3.12. SCHEDULING

Compute uses the `nova-scheduler` service to determine how to dispatch compute requests. For example, the `nova-scheduler` service determines on which host a VM should launch. In the context of filters, the term *host* means a physical node that has a `nova-compute` service running on it. You can configure the scheduler through a variety of options.

Compute is configured with the following default scheduler options in the `/etc/nova/nova.conf`

file:

```
scheduler_driver_task_period = 60
scheduler_driver = nova.scheduler.filter_scheduler.FilterScheduler
scheduler_available_filters = nova.scheduler.filters.all_filters
scheduler_default_filters = RetryFilter, AvailabilityZoneFilter,
RamFilter, ComputeFilter, ComputeCapabilitiesFilter,
ImagePropertiesFilter, ServerGroupAntiAffinityFilter,
ServerGroupAffinityFilter
```

By default, the **scheduler_driver** is configured as a filter scheduler, as described in the next section. In the default configuration, this scheduler considers hosts that meet all the following criteria:

- Have not been attempted for scheduling purposes (**RetryFilter**).
- Are in the requested availability zone (**AvailabilityZoneFilter**).
- Have sufficient RAM available (**RamFilter**).
- Can service the request (**ComputeFilter**).
- Satisfy the extra specs associated with the instance type (**ComputeCapabilitiesFilter**).
- Satisfy any architecture, hypervisor type, or virtual machine mode properties specified on the instance's image properties (**ImagePropertiesFilter**).
- Are on a different host than other instances of a group (if requested) (**ServerGroupAntiAffinityFilter**).
- Are in a set of group hosts (if requested) (**ServerGroupAffinityFilter**).

The scheduler caches its list of available hosts; use the **scheduler_driver_task_period** option to specify how often the list is updated.



NOTE

Do not configure **service_down_time** to be much smaller than **scheduler_driver_task_period**; otherwise, hosts appear to be dead while the host list is being cached.

The scheduler chooses a new host when an instance is migrated.

When evacuating instances from a host, the scheduler service honors the target host defined by the administrator on the evacuate command. If a target is not defined by the administrator, the scheduler determines the target host..

3.12.1. Filter scheduler

The filter scheduler (**nova.scheduler.filter_scheduler.FilterScheduler**) is the default scheduler for scheduling virtual machine instances. It supports filtering and weighting to make informed decisions on where a new instance should be created.

3.12.2. Filters

When the filter scheduler receives a request for a resource, it first applies filters to determine which hosts are eligible for consideration when dispatching a resource. Filters are binary: either a host is accepted by the filter, or it is rejected. Hosts that are accepted by the filter are then processed by a different algorithm to decide which hosts to use for that request, described in the [Weights](#) section.

The `scheduler_available_filters` configuration option in `nova.conf` provides the Compute service with the list of the filters that are used by the scheduler. The default setting specifies all of the filter that are included with the Compute service:

```
scheduler_available_filters = nova.scheduler.filters.all_filters
```

This configuration option can be specified multiple times. For example, if you implemented your own custom filter in Python called `myfilter.MyFilter` and you wanted to use both the built-in filters and your custom filter, your `nova.conf` file would contain:

```
scheduler_available_filters = nova.scheduler.filters.all_filters
scheduler_available_filters = myfilter.MyFilter
```

The `scheduler_default_filters` configuration option in `nova.conf` defines the list of filters that are applied by the `nova-scheduler` service. The default filters are:

```
scheduler_default_filters = RetryFilter, AvailabilityZoneFilter,
RamFilter, ComputeFilter, ComputeCapabilitiesFilter,
ImagePropertiesFilter, ServerGroupAntiAffinityFilter,
ServerGroupAffinityFilter
```

The following sections describe the available filters.

3.12.2.1. AggregateCoreFilter

Filters host by CPU core numbers with a per-aggregate `cpu_allocation_ratio` value. If the per-aggregate value is not found, the value falls back to the global setting. If the host is in more than one aggregate and more than one value is found, the minimum value will be used. For information about how to use this filter, see [Section 3.12.5, “Host aggregates and availability zones”](#). See also [Section 3.12.2.14, “CoreFilter”](#).

3.12.2.2. AggregateDiskFilter

Filters host by disk allocation with a per-aggregate `disk_allocation_ratio` value. If the per-aggregate value is not found, the value falls back to the global setting. If the host is in more than one aggregate and more than one value is found, the minimum value will be used. For information about how to use this filter, see [Section 3.12.5, “Host aggregates and availability zones”](#). See also [Section 3.12.2.17, “DiskFilter”](#).

3.12.2.3. AggregateImagePropertiesIsolation

Matches properties defined in an image's metadata against those of aggregates to determine host matches:

- If a host belongs to an aggregate and the aggregate defines one or more metadata that matches an image's properties, that host is a candidate to boot the image's instance.
- If a host does not belong to any aggregate, it can boot instances from all images.

You can configure the **AggregateImagePropertiesIsolation** filter by using the following options in the `nova.conf` file:

```
# Considers only keys matching the given namespace (string).
aggregate_image_properties_isolation_namespace = <None>

# Separator used between the namespace and keys (string).
aggregate_image_properties_isolation_separator = .
```

3.12.2.4. AggregateInstanceExtraSpecsFilter

Matches properties defined in extra specs for an instance type against admin-defined properties on a host aggregate. Works with specifications that are scoped with `aggregate_instance_extra_specs`. For backward compatibility, also works with non-scoped specifications; this action is highly discouraged because it conflicts with [ComputeCapabilitiesFilter](#) filter when you enable both filters. For information about how to use this filter, see the [host aggregates](#) section.

3.12.2.5. AggregateIoOpsFilter

Filters host by disk allocation with a per-aggregate `max_io_ops_per_host` value. If the per-aggregate value is not found, the value falls back to the global setting. If the host is in more than one aggregate and more than one value is found, the minimum value will be used. For information about how to use this filter, see [Section 3.12.5, “Host aggregates and availability zones”](#). See also [Section 3.12.2.22, “IoOpsFilter”](#).

3.12.2.6. AggregateMultiTenancyIsolation

Isolates tenants to specific [host aggregates](#). If a host is in an aggregate that has the `filter_tenant_id` metadata key, the host creates instances from only that tenant or list of tenants. A host can be in different aggregates. If a host does not belong to an aggregate with the metadata key, the host can create instances from all tenants.

3.12.2.7. AggregateNumInstancesFilter

Filters host by number of instances with a per-aggregate `max_instances_per_host` value. If the per-aggregate value is not found, the value falls back to the global setting. If the host is in more than one aggregate and thus more than one value is found, the minimum value will be used. For information about how to use this filter, see [Section 3.12.5, “Host aggregates and availability zones”](#). See also [Section 3.12.2.25, “NumInstancesFilter”](#).

3.12.2.8. AggregateRamFilter

Filters host by RAM allocation of instances with a per-aggregate `ram_allocation_ratio` value. If the per-aggregate value is not found, the value falls back to the global setting. If the host is in more than one aggregate and thus more than one value is found, the minimum value will be used. For information about how to use this filter, see [Section 3.12.5, “Host aggregates and availability zones”](#). See also [Section 3.12.2.27, “RamFilter”](#).

3.12.2.9. AggregateTypeAffinityFilter

Filters host by per-aggregate `instance_type` value. For information about how to use this filter, see [Section 3.12.5, “Host aggregates and availability zones”](#). See also [Section 3.12.2.34, “TypeAffinityFilter”](#).

3.12.2.10. AllHostsFilter

This is a no-op filter. It does not eliminate any of the available hosts.

3.12.2.11. AvailabilityZoneFilter

Filters hosts by availability zone. You must enable this filter for the scheduler to respect availability zones in requests.

3.12.2.12. ComputeCapabilitiesFilter

Matches properties defined in extra specs for an instance type against compute capabilities.

If an extra specs key contains a colon (:), anything before the colon is treated as a namespace and anything after the colon is treated as the key to be matched. If a namespace is present and is not **capabilities**, the filter ignores the namespace. For backward compatibility, also treats the extra specs key as the key to be matched if no namespace is present; this action is highly discouraged because it conflicts with [AggregateInstanceExtraSpecsFilter](#) filter when you enable both filters.

3.12.2.13. ComputeFilter

Passes all hosts that are operational and enabled.

In general, you should always enable this filter.

3.12.2.14. CoreFilter

Only schedules instances on hosts if sufficient CPU cores are available. If this filter is not set, the scheduler might over-provision a host based on cores. For example, the virtual cores running on an instance may exceed the physical cores.

You can configure this filter to enable a fixed amount of vCPU overcommitment by using the `cpu_allocation_ratio` configuration option in `nova.conf`. The default setting is:

```
cpu_allocation_ratio = 16.0
```

With this setting, if 8 vCPUs are on a node, the scheduler allows instances up to 128 vCPU to be run on that node.

To disallow vCPU overcommitment set:

```
cpu_allocation_ratio = 1.0
```



NOTE

The Compute API always returns the actual number of CPU cores available on a compute node regardless of the value of the `cpu_allocation_ratio` configuration key. As a result changes to the `cpu_allocation_ratio` are not reflected via the command line clients or the dashboard. Changes to this configuration key are only taken into account internally in the scheduler.

3.12.2.15. NUMATopologyFilter

Filters hosts based on the NUMA topology that was specified for the instance through the use of flavor `extra_specs` in combination with the image properties, as described in detail in the related nova-spec document: Filter will try to match the exact NUMA cells of the instance to those of the host. It will consider the standard over-subscription limits each cell, and provide limits to the compute host accordingly.



NOTE

If instance has no topology defined, it will be considered for any host. If instance has a topology defined, it will be considered only for NUMA capable hosts.

3.12.2.16. DifferentHostFilter

Schedules the instance on a different host from a set of instances. To take advantage of this filter, the requester must pass a scheduler hint, using `different_host` as the key and a list of instance UUIDs as the value. This filter is the opposite of the `SameHostFilter`. Using the `nova` command-line tool, use the `--hint` flag. For example:

```
$ nova boot --image cedef40a-ed67-4d10-800e-17455edce175 --flavor 1 --hint
different_host=a0cf03a5-d921-4877-bb5c-86d26cf818e1 --hint
different_host=8c19174f-4220-44f0-824a-cd1eeef10287 server-1
```

With the API, use the `os:scheduler_hints` key. For example:

```
{
  "server": {
    "name": "server-1",
    "imageRef": "cedef40a-ed67-4d10-800e-17455edce175",
    "flavorRef": "1"
  },
  "os:scheduler_hints": {
    "different_host": [
      "a0cf03a5-d921-4877-bb5c-86d26cf818e1",
      "8c19174f-4220-44f0-824a-cd1eeef10287"
    ]
  }
}
```

3.12.2.17. DiskFilter

Only schedules instances on hosts if there is sufficient disk space available for root and ephemeral storage.

You can configure this filter to enable a fixed amount of disk overcommitment by using the

disk_allocation_ratio configuration option in the `nova.conf` configuration file. The default setting disables the possibility of the overcommitment and allows launching a VM only if there is a sufficient amount of disk space available on a host:

```
disk_allocation_ratio = 1.0
```

DiskFilter always considers the value of the **disk_available_least** property and not the one of the **free_disk_gb** property of a hypervisor's statistics:

```
$ nova hypervisor-stats
+-----+-----+
| Property           | Value |
+-----+-----+
| count              | 1     |
| current_workload    | 0     |
| disk_available_least | 29    |
| free_disk_gb        | 35    |
| free_ram_mb         | 3441  |
| local_gb            | 35    |
| local_gb_used       | 0     |
| memory_mb           | 3953  |
| memory_mb_used      | 512   |
| running_vms         | 0     |
| vcpus               | 2     |
| vcpus_used          | 0     |
+-----+-----+
```

As it can be viewed from the command output above, the amount of the available disk space can be less than the amount of the free disk space. It happens because the **disk_available_least** property accounts for the virtual size rather than the actual size of images. If you use an image format that is sparse or copy on write so that each virtual instance does not require a 1:1 allocation of a virtual disk to a physical storage, it may be useful to allow the overcommitment of disk space.

To enable scheduling instances while overcommitting disk resources on the node, adjust the value of the **disk_allocation_ratio** configuration option to greater than **1.0**:

```
disk_allocation_ratio > 1.0
```



NOTE

If the value is set to **>1**, keep track of the free disk space, as the value approaching **0** may result in the incorrect functioning of instances using it at the moment.

3.12.2.18. GroupAffinityFilter



NOTE

This filter is deprecated in favor of [ServerGroupAffinityFilter](#).

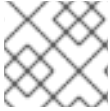
The **GroupAffinityFilter** ensures that an instance is scheduled on to a host from a set of group hosts. To take advantage of this filter, the requester must pass a scheduler hint, using **group** as the key and an arbitrary name as the value. Using the **nova** command-line tool, use the **--hint** flag. For example:

-

```
$ nova boot --image IMAGE_ID --flavor 1 --hint group=GROUP server-1
```

This filter should not be enabled at the same time as [GroupAntiAffinityFilter](#) or neither filter will work properly.

3.12.2.19. GroupAntiAffinityFilter



NOTE

This filter is deprecated in favor of [ServerGroupAntiAffinityFilter](#).

The `GroupAntiAffinityFilter` ensures that each instance in a group is on a different host. To take advantage of this filter, the requester must pass a scheduler hint, using `group` as the key and an arbitrary name as the value. Using the `nova` command-line tool, use the `--hint` flag. For example:

```
$ nova boot --image IMAGE_ID --flavor 1 --hint group=GROUP server-1
```

This filter should not be enabled at the same time as [GroupAffinityFilter](#) or neither filter will work properly.

3.12.2.20. ImagePropertiesFilter

Filters hosts based on properties defined on the instance's image. It passes hosts that can support the specified image properties contained in the instance. Properties include the architecture, hypervisor type, hypervisor version (for Xen hypervisor type only), and virtual machine mode.

For example, an instance might require a host that runs an ARM-based processor, and QEMU as the hypervisor. You can decorate an image with these properties by using:

```
$ glance image-update img-uuid --property architecture=arm --property
hypervisor_type=qemu
```

The image properties that the filter checks for are:

- **architecture:** describes the machine architecture required by the image. Examples are `i686`, `x86_64`, `arm`, and `ppc64`.
- **hypervisor_type:** describes the hypervisor required by the image. Examples are `xen`, `qemu`, and `xenapi`.



NOTE

`qemu` is used for both QEMU and KVM hypervisor types.

- **hypervisor_version_requires:** describes the hypervisor version required by the image. The property is supported for Xen hypervisor type only. It can be used to enable support for multiple hypervisor versions, and to prevent instances with newer Xen tools from being provisioned on an older version of a hypervisor. If available, the property value is compared to the hypervisor version of the compute host.

To filter the hosts by the hypervisor version, add the **hypervisor_version_requires** property on the image as metadata and pass an operator and a required hypervisor version as its value:

```
$ glance image-update img-uuid --property hypervisor_type=xen --
property hypervisor_version_requires=">=4.3"
```

- **vm_mode**: describes the hypervisor application binary interface (ABI) required by the image. Examples are **xen** for Xen 3.0 paravirtual ABI, **hvm** for native ABI, **uml** for User Mode Linux paravirtual ABI, **exe** for container virt executable ABI.

3.12.2.21. IsolatedHostsFilter

Allows the admin to define a special (isolated) set of images and a special (isolated) set of hosts, such that the isolated images can only run on the isolated hosts, and the isolated hosts can only run isolated images. The flag **restrict_isolated_hosts_to_isolated_images** can be used to force isolated hosts to only run isolated images.

The admin must specify the isolated set of images and hosts in the **nova.conf** file using the **isolated_hosts** and **isolated_images** configuration options. For example:

```
isolated_hosts = server1, server2
isolated_images = 342b492c-128f-4a42-8d3a-c5088cf27d13, ebd267a6-ca86-
4d6c-9a0e-bd132d6b7d09
```

3.12.2.22. IoOpsFilter

The **IoOpsFilter** filters hosts by concurrent I/O operations on it. Hosts with too many concurrent I/O operations will be filtered out. The **max_io_ops_per_host** option specifies the maximum number of I/O intensive instances allowed to run on a host. A host will be ignored by the scheduler if more than **max_io_ops_per_host** instances in build, resize, snapshot, migrate, rescue or unshelve task states are running on it.

3.12.2.23. JsonFilter

The **JsonFilter** allows a user to construct a custom filter by passing a scheduler hint in JSON format. The following operators are supported:

- **=**
- **<**
- **>**
- **in**
- **<=**
- **>=**
- **not**
- **or**

- and

The filter supports the following variables:

- `$free_ram_mb`
- `$free_disk_mb`
- `$total_usable_ram_mb`
- `$vcpus_total`
- `$vcpus_used`

Using the nova command-line tool, use the `--hint` flag:

```
$ nova boot --image 827d564a-e636-4fc4-a376-d36f7ebe1747 --flavor 1 --hint
query='[">=", "$free_ram_mb", 1024]' server1
```

With the API, use the `os:scheduler_hints` key:

```
{
  "server": {
    "name": "server-1",
    "imageRef": "cedef40a-ed67-4d10-800e-17455edce175",
    "flavorRef": "1"
  },
  "os:scheduler_hints": {
    "query": "[>=, $free_ram_mb, 1024]"
  }
}
```

3.12.2.24. MetricsFilter

Filters hosts based on metrics `weight_setting`. Only hosts with the available metrics are passed so that the metrics weigher will not fail due to these hosts.

3.12.2.25. NumInstancesFilter

Hosts that have more instances running than specified by the `max_instances_per_host` option are filtered out when this filter is in place.

3.12.2.26. PciPassthroughFilter

The filter schedules instances on a host if the host has devices that meet the device requests in the `extra_specs` attribute for the flavor.

3.12.2.27. RamFilter

Only schedules instances on hosts that have sufficient RAM available. If this filter is not set, the scheduler may over provision a host based on RAM (for example, the RAM allocated by virtual machine instances may exceed the physical RAM).

You can configure this filter to enable a fixed amount of RAM overcommitment by using the `ram_allocation_ratio` configuration option in `nova.conf`. The default setting is:

```
ram_allocation_ratio = 1.5
```

This setting enables 1.5 GB instances to run on any compute node with 1 GB of free RAM.



WARNING

Overcommitting is not an ideal solution for all memory issues. Rather, the recommended methods to deal with memory shortage are to allocate less memory per guest, add more physical memory to the host, or utilize swap space. If you decide to leave memory overcommitment enabled, ensure sufficient testing is performed. Contact Red Hat's support services for assistance with overcommitting.

To disable RAM overcommitment, set `ram_allocation_ratio` to `1.0`.

3.12.2.28. RetryFilter

Filters out hosts that have already been attempted for scheduling purposes. If the scheduler selects a host to respond to a service request, and the host fails to respond to the request, this filter prevents the scheduler from retrying that host for the service request.

This filter is only useful if the `scheduler_max_attempts` configuration option is set to a value greater than zero.

3.12.2.29. SameHostFilter

Schedules the instance on the same host as another instance in a set of instances. To take advantage of this filter, the requester must pass a scheduler hint, using `same_host` as the key and a list of instance UUIDs as the value. This filter is the opposite of the `DifferentHostFilter`. Using the `nova` command-line tool, use the `--hint` flag:

```
$ nova boot --image cedef40a-ed67-4d10-800e-17455edce175 --flavor 1 --hint
same_host=a0cf03a5-d921-4877-bb5c-86d26cf818e1 \ --hint
same_host=8c19174f-4220-44f0-824a-cd1eeef10287 server-1
```

With the API, use the `os:scheduler_hints` key:

```
{
  "server": {
    "name": "server-1",
    "imageRef": "cedef40a-ed67-4d10-800e-17455edce175",
    "flavorRef": "1"
  },
  "os:scheduler_hints": {
    "same_host": [
      "a0cf03a5-d921-4877-bb5c-86d26cf818e1",
```



```

    "8c19174f-4220-44f0-824a-cd1eeef10287"
  ]
}

```

3.12.2.30. ServerGroupAffinityFilter

The `ServerGroupAffinityFilter` ensures that an instance is scheduled on to a host from a set of group hosts. To take advantage of this filter, the requester must create a server group with an **affinity** policy, and pass a scheduler hint, using **group** as the key and the server group UUID as the value. Using the **nova** command-line tool, use the **--hint** flag. For example:

```

$ nova server-group-create --policy affinity group-1
$ nova boot --image IMAGE_ID --flavor 1 --hint group=SERVER_GROUP_UUID
server-1

```

3.12.2.31. ServerGroupAntiAffinityFilter

The `ServerGroupAntiAffinityFilter` ensures that each instance in a group is on a different host. To take advantage of this filter, the requester must create a server group with an **anti-affinity** policy, and pass a scheduler hint, using **group** as the key and the server group UUID as the value. Using the **nova** command-line tool, use the **--hint** flag. For example:

```

$ nova server-group-create --policy anti-affinity group-1
$ nova boot --image IMAGE_ID --flavor 1 --hint group=SERVER_GROUP_UUID
server-1

```

3.12.2.32. SimpleCIDRAffinityFilter

Schedules the instance based on host IP subnet range. To take advantage of this filter, the requester must specify a range of valid IP address in CIDR format, by passing two scheduler hints:

build_near_host_ip

The first IP address in the subnet (for example, **192.168.1.1**)

cidr

The CIDR that corresponds to the subnet (for example, **/24**)

Using the **nova** command-line tool, use the **--hint** flag. For example, to specify the IP subnet **192.168.1.1/24**

```

$ nova boot --image cedef40a-ed67-4d10-800e-17455edce175 --flavor 1 --hint
build_near_host_ip=192.168.1.1 --hint cidr=/24 server-1

```

With the API, use the **os:scheduler_hints** key:

```

{
  "server": {
    "name": "server-1",
    "imageRef": "cedef40a-ed67-4d10-800e-17455edce175",

```

```

        "flavorRef": "1"
    },
    "os:scheduler_hints": {
        "build_near_host_ip": "192.168.1.1",
        "cidr": "24"
    }
}

```

3.12.2.33. TrustedFilter

Filters hosts based on their trust. Only passes hosts that meet the trust requirements specified in the instance properties.

3.12.2.34. TypeAffinityFilter

Dynamically limits hosts to one instance type. An instance can only be launched on a host, if no instance with different instances types are running on it, or if the host has no running instances at all.

3.12.3. Weights

When resourcing instances, the filter scheduler filters and weights each host in the list of acceptable hosts. Each time the scheduler selects a host, it virtually consumes resources on it, and subsequent selections are adjusted accordingly. This process is useful when the customer asks for the same large amount of instances, because weight is computed for each requested instance.

All weights are normalized before being summed up; the host with the largest weight is given the highest priority.

If cells are used, cells are weighted by the scheduler in the same manner as hosts.

Hosts and cells are weighted based on the following options in the `/etc/nova/nova.conf` file:

Table 3.6. Host weighting options

Section	Option	Description
[DEFAULT]	ram_weight_multiplier	By default, the scheduler spreads instances across all hosts evenly. Set the ram_weight_multiplier option to a negative number if you prefer stacking instead of spreading. Use a floating-point value.
[DEFAULT]	scheduler_host_subset_size	New instances are scheduled on a host that is chosen randomly from a subset of the N best hosts. This property defines the subset size from which a host is chosen. A value of 1 chooses the first host returned by the weighting functions. This value must be at least 1. A value less than 1 is ignored, and 1 is used instead. Use an integer value.

Section	Option	Description
[DEFAULT]	scheduler_weight_classes	Defaults to nova.scheduler.weights.all_weighters , which selects the RamWeigher and MetricsWeigher. Hosts are then weighted and sorted with the largest weight winning.
[metrics]	weight_multiplier	Multiplier for weighting metrics. Use a floating-point value.
[metrics]	weight_setting	Determines how metrics are weighted. Use a comma-separated list of metricName=ratio. For example: "name1=1.0, name2=-1.0" results in: name1.value * 1.0 + name2.value * -1.0
[metrics]	required	Specifies how to treat unavailable metrics: <ul style="list-style-type: none"> • True—Raises an exception. To avoid the raised exception, you should use the scheduler filter MetricFilter to filter out hosts with unavailable metrics. • False—Treated as a negative factor in the weighting process (uses the weight_of_unavailable option).
[metrics]	weight_of_unavailable	If required is set to False, and any one of the metrics set by weight_setting is unavailable, the weight_of_unavailable value is returned to the scheduler.

For example:

```
[DEFAULT]
scheduler_host_subset_size = 1
scheduler_weight_classes = nova.scheduler.weights.all_weighters
ram_weight_multiplier = 1.0
[metrics]
weight_multiplier = 1.0
weight_setting = name1=1.0, name2=-1.0
required = false
weight_of_unavailable = -10000.0
```

Table 3.7. Cell weighting options

Section	Option	Description
[cells]	mute_weight_multiplier	Multiplier to weight mute children (hosts which have not sent capacity or capacity updates for some time). Use a negative, floating-point value.
[cells]	mute_weight_value (deprecated)	Weight value assigned to mute children. Use a positive, floating-point value with a maximum of '1.0'. This option is deprecated, use mute_weight_multiplier instead.
[cells]	offset_weight_multiplier	Multiplier to weight cells, so you can specify a preferred cell. Use a floating point value.
[cells]	ram_weight_multiplier	By default, the scheduler spreads instances across all cells evenly. Set the ram_weight_multiplier option to a negative number if you prefer stacking instead of spreading. Use a floating-point value.
[cells]	scheduler_weight_classes	Defaults to nova.cells.weights.all_weighters , which maps to all cell weighters included with Compute. Cells are then weighted and sorted with the largest weight winning.

For example:

```
[cells]
scheduler_weight_classes = nova.cells.weights.all_weighters
mute_weight_multiplier = -10.0
ram_weight_multiplier = 1.0
offset_weight_multiplier = 1.0
```

3.12.4. Chance scheduler

As an administrator, you work with the filter scheduler. However, the Compute service also uses the Chance Scheduler, **nova.scheduler.chance.ChanceScheduler**, which randomly selects from lists of filtered hosts.

3.12.5. Host aggregates and availability zones

Host aggregates are a mechanism for partitioning hosts in an OpenStack cloud, or a region of an OpenStack cloud, based on arbitrary characteristics. Examples where an administrator may want to do this include where a group of hosts have additional hardware or performance characteristics.

Host aggregates are not explicitly exposed to users. Instead administrators map flavors to host aggregates. Administrators do this by setting metadata on a host aggregate, and matching flavor extra specifications. The scheduler then endeavors to match user requests for instance of the given flavor to a host aggregate with the same key-value pair in its metadata. Compute nodes can be in more than one host aggregate.

Administrators are able to optionally expose a host aggregate as an availability zone. Availability zones are different from host aggregates in that they are explicitly exposed to the user, and hosts can only be in a single availability zone. Administrators can configure a default availability zone where instances

will be scheduled when the user fails to specify one.

Command-line interface

The nova command-line tool supports the following aggregate-related commands.

nova aggregate-list

Print a list of all aggregates.

nova aggregate-create <name> [availability-zone]

Create a new aggregate named <name>, and optionally in availability zone [availability-zone] if specified. The command returns the ID of the newly created aggregate. Hosts can be made available to multiple host aggregates. Be careful when adding a host to an additional host aggregate when the host is also in an availability zone. Pay attention when using the **aggregate-set-metadata** and **aggregate-update** commands to avoid user confusion when they boot instances in different availability zones. An error occurs if you cannot add a particular host to an aggregate zone for which it is not intended.

nova aggregate-delete <id>

Delete an aggregate with id <id>.

nova aggregate-details <id>

Show details of the aggregate with id <id>.

nova aggregate-add-host <id> <host>

Add host with name <host> to aggregate with id <id>.

nova aggregate-remove-host <id> <host>

Remove the host with name <host> from the aggregate with id <id>.

nova aggregate-set-metadata <id> <key=value> [<key=value> ...]

Add or update metadata (key-value pairs) associated with the aggregate with id <id>.

nova aggregate-update <id> <name> [<availability_zone>]

Update the name and availability zone (optional) for the aggregate.

nova host-list

List all hosts by service.

nova host-update --maintenance [enable | disable]

Put/resume host into/from maintenance.

NOTE

Only administrators can access these commands. If you try to use these commands and the user name and tenant that you use to access the Compute service do not have the **admin** role or the appropriate privileges, these errors occur:

```
ERROR: Policy does not allow compute_extension:aggregates to be
performed. (HTTP 403) (Request-ID: req-299fbff6-6729-4cef-93b2-
e7e1f96b4864)
```

```
ERROR: Policy does not allow compute_extension:hosts to be
performed. (HTTP 403) (Request-ID: req-ef2400f6-6776-4ea3-b6f1-
7704085c27d1)
```

Configure scheduler to support host aggregates

One common use case for host aggregates is when you want to support scheduling instances to a subset of compute hosts because they have a specific capability. For example, you may want to allow users to request compute hosts that have SSD drives if they need access to faster disk I/O, or access to compute hosts that have GPU cards to take advantage of GPU-accelerated code.

To configure the scheduler to support host aggregates, the **scheduler_default_filters** configuration option must contain the **AggregateInstanceExtraSpecsFilter** in addition to the other filters used by the scheduler. Add the following line to **/etc/nova/nova.conf** on the host that runs the **nova-scheduler** service to enable host aggregates filtering, as well as the other filters that are typically enabled:

```
scheduler_default_filters=AggregateInstanceExtraSpecsFilter,RetryFilter,Av
ailabilityZoneFilter,RamFilter,ComputeFilter,ComputeCapabilitiesFilter,Ima
gePropertiesFilter,ServerGroupAntiAffinityFilter,ServerGroupAffinityFilter
```

Example: Specify compute hosts with SSDs

This example configures the Compute service to enable users to request nodes that have solid-state drives (SSDs). You create a **fast-io** host aggregate in the **nova** availability zone and you add the **ssd=true** key-value pair to the aggregate. Then, you add the **node1**, and **node2** compute nodes to it.

```
$ nova aggregate-create fast-io nova
+-----+-----+-----+-----+-----+
| Id | Name   | Availability Zone | Hosts | Metadata |
+-----+-----+-----+-----+-----+
| 1  | fast-io | nova              |      |          |
+-----+-----+-----+-----+-----+

$ nova aggregate-set-metadata 1 ssd=true
+-----+-----+-----+-----+-----+
| Id | Name   | Availability Zone | Hosts | Metadata          |
+-----+-----+-----+-----+-----+
| 1  | fast-io | nova              | []    | {u'ssd': u'true'} |
+-----+-----+-----+-----+-----+

$ nova aggregate-add-host 1 node1
+-----+-----+-----+-----+-----+
```

```

| Id | Name      | Availability Zone | Hosts          | Metadata          |
+----+-----+-----+-----+-----+
| 1  | fast-io  | nova             | [u'node1']    | {u'ssd': u'true'} |
+----+-----+-----+-----+-----+

$ nova aggregate-add-host 1 node2
+----+-----+-----+-----+-----+
-----+
| Id | Name      | Availability Zone | Hosts          | Metadata          |
|
+----+-----+-----+-----+-----+
-----+
| 1  | fast-io  | nova             | [u'node1', u'node2'] | {u'ssd':
u'true'} |
+----+-----+-----+-----+-----+
-----+

```

Use the **nova flavor-create** command to create the *ssd.large* flavor called with an ID of 6, 8 GB of RAM, 80 GB root disk, and four vCPUs.

```

$ nova flavor-create ssd.large 6 8192 80 4
+----+-----+-----+-----+-----+-----+-----+-----+
-----+
| ID | Name      | Memory_MB | Disk | Ephemeral | Swap | VCPUs |
RXTX_Factor | Is_Public |
+----+-----+-----+-----+-----+-----+-----+
-----+
| 6  | ssd.large | 8192      | 80   | 0          |      | 4      | 1.0
| True      |
+----+-----+-----+-----+-----+-----+
-----+

```

Once the flavor is created, specify one or more key-value pairs that match the key-value pairs on the host aggregates with scope `aggregate_instance_extra_specs`. In this case, that is the *aggregate_instance_extra_specs:ssd=true* key-value pair. Setting a key-value pair on a flavor is done using the **nova flavor-key** command.

```
$ nova flavor-key ssd.large set aggregate_instance_extra_specs:ssd=true
```

Once it is set, you should see the `extra_specs` property of the *ssd.large* flavor populated with a key of `ssd` and a corresponding value of `true`.

```

$ nova flavor-show ssd.large
+-----+-----+
-----+
| Property          | Value          |
|
+-----+-----+
-----+
| OS-FLV-DISABLED:disabled | False          |
|
| OS-FLV-EXT-DATA:ephemeral | 0              |
|
| disk                | 80             |
|

```

```

| extra_specs                                | {u'aggregate_instance_extra_specs:ssd':
u'true'} |
| id                                          | 6
| name                                       | ssd.large
| os-flavor-access:is_public                | True
| ram                                        | 8192
| rxtx_factor                              | 1.0
| swap                                       |
| vcpus                                     | 4
+-----+-----+
-----+

```

Now, when a user requests an instance with the `ssd.large` flavor, the scheduler only considers hosts with the `ssd=true` key-value pair. In this example, these are `node1` and `node2`.



NOTE

The **key** and **value** are case sensitive strings. The Compute scheduler performs a case sensitive string match of the value.

XenServer hypervisor pools to support live migration

When using the XenAPI-based hypervisor, the Compute service uses host aggregates to manage XenServer Resource pools, which are used in supporting live migration.

3.12.6. Configuration reference

To customize the Compute scheduler, use the configuration option settings documented in [Table 3.48, “Description of scheduler configuration options”](#).

3.13. CELLS

Cells functionality enables you to scale an OpenStack Compute cloud in a more distributed fashion without having to use complicated technologies like database and message queue clustering. It supports very large deployments.

When this functionality is enabled, the hosts in an OpenStack Compute cloud are partitioned into groups called cells. Cells are configured as a tree. The top-level cell should have a host that runs a `nova-api` service, but no `nova-compute` services. Each child cell should run all of the typical `nova-*` services in a regular Compute cloud except for `nova-api`. You can think of cells as a normal Compute deployment in that each cell has its own database server and message queue broker.

The `nova-cells` service handles communication between cells and selects cells for new instances. This service is required for every cell. Communication between cells is pluggable, and currently the only option is communication through RPC.

Cells scheduling is separate from host scheduling. **nova-cells** first picks a cell. Once a cell is selected and the new build request reaches its **nova-cells** service, it is sent over to the host scheduler in that cell and the build proceeds as it would have without cells.



WARNING

Cell functionality is currently considered experimental.

3.13.1. Cell configuration options

Cells are disabled by default. All cell-related configuration options appear in the **[cells]** section in **nova.conf**. The following cell-related options are currently supported:

enable

Set to **True** to turn on cell functionality. Default is **false**.

name

Name of the current cell. Must be unique for each cell.

capabilities

List of arbitrary **key=value** pairs defining capabilities of the current cell. Values include **hypervisor=xenserver;kvm,os=linux**.

call_timeout

How long in seconds to wait for replies from calls between cells.

scheduler_filter_classes

Filter classes that the cells scheduler should use. By default, uses **"nova.cells.filters.all_filters"** to map to all cells filters included with Compute.

scheduler_weight_classes

Weight classes that the scheduler for cells uses. By default, uses **nova.cells.weights.all_weighters** to map to all cells weight algorithms included with Compute.

ram_weight_multiplier

Multiplier used to weight RAM. Negative numbers indicate that Compute should stack VMs on one host instead of spreading out new VMs to more hosts in the cell. The default value is 10.0.

3.13.2. Configure the API (top-level) cell

The cell type must be changed in the API cell so that requests can be proxied through nova-cells down to the correct cell properly. Edit the **nova.conf** file in the API cell, and specify **api** in the **cell_type** key:

■

```
[DEFAULT]
compute_api_class=nova.compute.cells_api.ComputeCellsAPI
...

[cells]
cell_type= api
```

3.13.3. Configure the child cells

Edit the `nova.conf` file in the child cells, and specify `compute` in the `cell_type` key:

```
[DEFAULT]
# Disable quota checking in child cells. Let API cell do it exclusively.
quota_driver=nova.quota.NoopQuotaDriver

[cells]
cell_type = compute
```

3.13.4. Configure the database in each cell

Before bringing the services online, the database in each cell needs to be configured with information about related cells. In particular, the API cell needs to know about its immediate children, and the child cells must know about their immediate agents. The information needed is the **RabbitMQ** server credentials for the particular cell.

Use the `nova-manage cell create` command to add this information to the database in each cell:

```
# nova-manage cell create -h
Options:
  -h, --help                show this help message and exit
  --name=<name>              Name for the new cell
  --cell_type=<parent|child> Whether the cell is a parent or child
  --username=<username>      Username for the message broker in this cell
  --password=<password>      Password for the message broker in this cell
  --hostname=<hostname>      Address of the message broker in this cell
  --port=<number>            Port number of the message broker in this cell
  --virtual_host=<virtual_host> The virtual host of the message broker in this
cell
  --woffset=<float>          (weight offset) It might be used by some cell
scheduling code in the future
  --wscale=<float>          (weight scale) It might be used by some cell
scheduling code in the future
```

As an example, assume an API cell named `api` and a child cell named `cell1`.

Within the `api` cell, specify the following RabbitMQ server information:

```
rabbit_host=10.0.0.10
rabbit_port=5672
rabbit_username=api_user
rabbit_password=api_passwd
rabbit_virtual_host=api_vhost
```

Within the **cell1** child cell, specify the following RabbitMQ server information:

```
rabbit_host=10.0.1.10
rabbit_port=5673
rabbit_username=cell1_user
rabbit_password=cell1_passwd
rabbit_virtual_host=cell1_vhost
```

You can run this in the API cell as root:

```
# nova-manage cell create --name cell1 --cell_type child --username
cell1_user --password cell1_passwd --hostname 10.0.1.10 --port 5673 --
virtual_host cell1_vhost --woffset 1.0 --wscale 1.0
```

Repeat the previous steps for all child cells.

In the child cell, run the following, as root:

```
# nova-manage cell create --name api --cell_type parent --username
api_user --password api_passwd --hostname 10.0.0.10 --port 5672 --
virtual_host api_vhost --woffset 1.0 --wscale 1.0
```

To customize the Compute cells, use the configuration option settings documented in [Table 3.16](#), “Description of cell configuration options”.

3.13.5. Cell scheduling configuration

To determine the best cell to use to launch a new instance, Compute uses a set of filters and weights defined in the `/etc/nova/nova.conf` file. The following options are available to prioritize cells for scheduling:

scheduler_filter_classes

List of filter classes. By default `nova.cells.filters.all_filters` is specified, which maps to all cells filters included with Compute (see [Section 3.12.2](#), “Filters”).

scheduler_weight_classes

List of weight classes. By default `nova.cells.weights.all_weighters` is specified, which maps to all cell weight algorithms included with Compute. The following modules are available:

- **mute_child**. Downgrades the likelihood of child cells being chosen for scheduling requests, which have not sent capacity or capability updates in a while. Options include **mute_weight_multiplier** (multiplier for mute children; value should be negative) and **mute_weight_value** (assigned to mute children; should be a positive value).

**WARNING**

The `mute_weight_value` is deprecated, use `mute_weight_multiplier` instead.

- **ram_by_instance_type.** Select cells with the most RAM capacity for the instance type being requested. Because higher weights win, Compute returns the number of available units for the instance type requested. The `ram_weight_multiplier` option defaults to 10.0 that adds to the weight by a factor of 10. Use a negative number to stack VMs on one host instead of spreading out new VMs to more hosts in the cell.
- **weight_offset.** Allows modifying the database to weight a particular cell. You can use this when you want to disable a cell (for example, '0'), or to set a default cell by making its `weight_offset` very high (for example, '999999999999999'). The highest weight will be the first cell to be scheduled for launching an instance.

Additionally, the following options are available for the cell scheduler:

scheduler_retries

Specifies how many times the scheduler tries to launch a new instance when no cells are available (default=10).

scheduler_retry_delay

Specifies the delay (in seconds) between retries (default=2).

As an admin user, you can also add a filter that directs builds to a particular cell. The `policy.json` file must have a line with `"cells_scheduler_filter:TargetCellFilter" : "is_admin:True"` to let an admin user specify a scheduler hint to direct a build to a particular cell.

3.13.6. Optional cell configuration

Cells store all inter-cell communication data, including user names and passwords, in the database. Because the cells data is not updated very frequently, use the `[cells]cells_config` option to specify a JSON file to store cells data. With this configuration, the database is no longer consulted when reloading the cells data. The file must have columns present in the Cell model (excluding common database fields and the `id` column). You must specify the queue connection information through a `transport_url` field, instead of `username`, `password`, and so on. The `transport_url` has the following form:

```
rabbit://USERNAME:PASSWORD@HOSTNAME:PORT/VIRTUAL_HOST
```

The scheme can be either `qpid` or `rabbit`, as shown previously. The following sample shows this optional configuration:

```
{
  "parent": {
    "name": "parent",
```

```

        "api_url": "http://api.example.com:8774",
        "transport_url": "rabbit://rabbit.example.com",
        "weight_offset": 0.0,
        "weight_scale": 1.0,
        "is_parent": true
    },
    "cell1": {
        "name": "cell1",
        "api_url": "http://api.example.com:8774",
        "transport_url": "rabbit://rabbit1.example.com",
        "weight_offset": 0.0,
        "weight_scale": 1.0,
        "is_parent": false
    },
    "cell2": {
        "name": "cell2",
        "api_url": "http://api.example.com:8774",
        "transport_url": "rabbit://rabbit2.example.com",
        "weight_offset": 0.0,
        "weight_scale": 1.0,
        "is_parent": false
    }
}

```

3.14. CONDUCTOR

The **nova-conductor** service enables OpenStack to function without compute nodes accessing the database. Conceptually, it implements a new layer on top of **nova-compute**. It should not be deployed on compute nodes, or else the security benefits of removing database access from **nova-compute** are negated. Just like other nova services such as **nova-api** or **nova-scheduler**, it can be scaled horizontally. You can run multiple instances of **nova-conductor** on different machines as needed for scaling purposes.

The methods exposed by **nova-conductor** are relatively simple methods used by **nova-compute** to offload its database operations. Places where **nova-compute** previously performed database access are now talking to **nova-conductor**. However, there are plans in the medium to long term to move more and more of what is currently in **nova-compute** up to the **nova-conductor** layer. The Compute service will start to look like a less intelligent slave service to **nova-conductor**. The conductor service will implement long running complex operations, ensuring forward progress and graceful error handling. This will be especially beneficial for operations that cross multiple compute nodes, such as migrations or resizes.

To customize the Conductor, use the configuration option settings documented in [Table 3.19, “Description of conductor configuration options”](#).

3.15. EXAMPLE NOVA.CONF CONFIGURATION FILES

The following sections describe the configuration options in the **nova.conf** file. You must copy the **nova.conf** file to each compute node. The sample **nova.conf** files show examples of specific configurations.

Small, private cloud

This example **nova.conf** file configures a small private cloud with cloud controller services, database

server, and messaging server on the same server. In this case, CONTROLLER_IP represents the IP address of a central server, BRIDGE_INTERFACE represents the bridge such as br100, the NETWORK_INTERFACE represents an interface to your VLAN setup, and passwords are represented as DB_PASSWORD_COMPUTE for your Compute (nova) database password, and RABBIT PASSWORD represents the password to your message queue installation.

[DEFAULT]

```
# LOGS/STATE
verbose=True
logdir=/var/log/nova
state_path=/var/lib/nova
lock_path=/var/lock/nova
rootwrap_config=/etc/nova/rootwrap.conf

# SCHEDULER
compute_scheduler_driver=nova.scheduler.filter_scheduler.FilterScheduler

# VOLUMES
# configured in cinder.conf

# COMPUTE
compute_driver=libvirt.LibvirtDriver
instance_name_template=instance-%08x
api_paste_config=/etc/nova/api-paste.ini

# COMPUTE/APIS: if you have separate configs for separate services
# this flag is required for both nova-api and nova-compute
allow_resize_to_same_host=True

# APIS
osapi_compute_extension=nova.api.openstack.compute.contrib.standard_extensions
ec2_dmz_host=192.168.206.130
s3_host=192.168.206.130

# RABBITMQ
rabbit_host=192.168.206.130

# GLANCE
image_service=nova.image.glance.GlanceImageService

# NETWORK
network_manager=nova.network.manager.FlatDHCPManager
force_dhcp_release=True
dhcpbridge_flagfile=/etc/nova/nova.conf
firewall_driver=nova.virt.libvirt.firewall.IptablesFirewallDriver
# Change my_ip to match each host
my_ip=192.168.206.130
public_interface=eth0
vlan_interface=eth0
flat_network_bridge=br100
flat_interface=eth0

# NOVNC CONSOLE
novncproxy_base_url=http://192.168.206.130:6080/vnc_auto.html
```

```
# Change vncserver_proxyclient_address and vncserver_listen to match each
compute host
vncserver_proxyclient_address=192.168.206.130
vncserver_listen=192.168.206.130

# AUTHENTICATION
auth_strategy=keystone
[keystone_authtoken]
auth_host = 127.0.0.1
auth_port = 35357
auth_protocol = http
admin_tenant_name = service
admin_user = nova
admin_password = nova
signing_dirname = /tmp/keystone-signing-nova

# GLANCE
[glance]
api_servers=192.168.206.130:9292

# DATABASE
[database]
connection=mysql://nova:yourpassword@192.168.206.130/nova

# LIBVIRT
[libvirt]
virt_type=qemu
```

KVM, Flat, MySQL, and Glance, OpenStack or EC2 API

This example `nova.conf` file, from an internal Rackspace test system, is used for demonstrations.

```
[DEFAULT]

# LOGS/STATE
verbose=True
logdir=/var/log/nova
state_path=/var/lib/nova
lock_path=/var/lock/nova
rootwrap_config=/etc/nova/rootwrap.conf

# SCHEDULER
compute_scheduler_driver=nova.scheduler.filter_scheduler.FilterScheduler

# VOLUMES
# configured in cinder.conf

# COMPUTE
compute_driver=libvirt.LibvirtDriver
instance_name_template=instance-%08x
api_paste_config=/etc/nova/api-paste.ini

# COMPUTE/APIS: if you have separate configs for separate services
# this flag is required for both nova-api and nova-compute
allow_resize_to_same_host=True
```

```
# APIS
osapi_compute_extension=nova.api.openstack.compute.contrib.standard_extensions
ec2_dmz_host=192.168.206.130
s3_host=192.168.206.130

# RABBITMQ
rabbit_host=192.168.206.130

# GLANCE
image_service=nova.image.glance.GlanceImageService

# NETWORK
network_manager=nova.network.manager.FlatDHCPManager
force_dhcp_release=True
dhcpbridge_flagfile=/etc/nova/nova.conf
firewall_driver=nova.virt.libvirt.firewall.IptablesFirewallDriver
# Change my_ip to match each host
my_ip=192.168.206.130
public_interface=eth0
vlan_interface=eth0
flat_network_bridge=br100
flat_interface=eth0

# NOVNC CONSOLE
novncproxy_base_url=http://192.168.206.130:6080/vnc_auto.html
# Change vncserver_proxyclient_address and vncserver_listen to match each
compute host
vncserver_proxyclient_address=192.168.206.130
vncserver_listen=192.168.206.130

# AUTHENTICATION
auth_strategy=keystone
[keystone_auth_token]
auth_host = 127.0.0.1
auth_port = 35357
auth_protocol = http
admin_tenant_name = service
admin_user = nova
admin_password = nova
signing_dirname = /tmp/keystone-signing-nova

# GLANCE
[glance]
api_servers=192.168.206.130:9292

# DATABASE
[database]
connection=mysql://nova:yourpassword@192.168.206.130/nova

# LIBVIRT
[libvirt]
virt_type=qemu
```


XenServer, Flat networking, MySQL, and Glance, OpenStack API

This example `nova.conf` file is from an internal Rackspace test system.

```
verbose
nodaemon
network_manager=nova.network.manager.FlatManager
image_service=nova.image.glance.GlanceImageService
flat_network_bridge=xenbr0
compute_driver=xenapi.XenAPIDriver
xenapi_connection_url=https://<XenServer IP>
xenapi_connection_username=root
xenapi_connection_password=supersecret
xenapi_image_upload_handler=nova.virt.xenapi.image.glance.GlanceStore
rescue_timeout=86400
use_ipv6=true
```

3.16. COMPUTE LOG FILES

The corresponding log file of each Compute service is stored in the `/var/log/nova/` directory of the host on which each service runs.

Table 3.8. Log files used by Compute services

Log file	Service name
<code>api.log</code>	<code>openstack-nova-api</code>
<code>cert.log</code> ^[a]	<code>openstack-nova-cert</code>
<code>compute.log</code>	<code>openstack-nova-compute</code>
<code>conductor.log</code>	<code>openstack-nova-conductor</code>
<code>consoleauth.log</code>	<code>openstack-nova-consoleauth</code>
<code>network.log</code> ^[b]	<code>openstack-nova-network</code>
<code>nova-manage.log</code>	<code>nova-manage</code>
<code>scheduler.log</code>	<code>openstack-nova-scheduler</code>

[a] The X509 certificate service (`openstack-nova-cert/nova-cert`) is only required by the EC2 API to the Compute service.

[b] The **nova** network service (`openstack-nova-network/nova-network`) only runs in deployments that are not configured to use the Networking service (**neutron**).

3.17. COMPUTE SAMPLE CONFIGURATION FILES

3.17.1. nova.conf - configuration options

For a complete list of all available configuration options for each OpenStack Compute service, run `nova-<servicename> --help`.

Table 3.9. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
api_paste_config = <i>api-paste.ini</i>	(StrOpt) File name for the paste.deploy config for nova-api
api_rate_limit = <i>False</i>	(BoolOpt) Whether to use per-user rate limiting for the api. This option is only used by v2 api. Rate limiting is removed from v3 api.
client_socket_timeout = <i>900</i>	(IntOpt) Timeout for client connections' socket operations. If an incoming connection is idle for this number of seconds it will be closed. A value of '0' means wait forever.
enable_new_services = <i>True</i>	(BoolOpt) Services to be added to the available pool on create
enabled_apis = <i>ec2, osapi_compute, metadata</i>	(ListOpt) A list of APIs to enable by default
enabled_ssl_apis =	(ListOpt) A list of APIs with enabled SSL
instance_name_template = <i>instance-%08x</i>	(StrOpt) Template string to be used to generate instance names
max_header_line = <i>16384</i>	(IntOpt) Maximum line size of message headers to be accepted. max_header_line may need to be increased when using large tokens (typically those generated by the Keystone v3 API with big service catalogs).
multi_instance_display_name_template = <i>%(name)s-%(count)d</i>	(StrOpt) When creating multiple instances with a single request using the os-multiple-create API extension, this template will be used to build the display name for each instance. The benefit is that the instances end up with different hostnames. To restore legacy behavior of every instance having the same name, set this option to "%(name)s". Valid keys for the template are: name, uuid, count.
non_inheritable_image_properties = <i>cache_in_nova, bittorrent</i>	(ListOpt) These are image properties which a snapshot should not inherit from an instance

Configuration option = Default value	Description
null_kernel = <i>nokernel</i>	(StrOpt) Kernel image that indicates not to use a kernel, but to use a raw disk image instead
osapi_compute_ext_list =	(ListOpt) Specify list of extensions to load when using <code>osapi_compute_extension</code> option with <code>nova.api.openstack.compute.contrib.select_extensions</code>
osapi_compute_extension = <i>['nova.api.openstack.compute.contrib.standard_extensions']</i>	(MultiStrOpt) osapi compute extension to load
osapi_compute_link_prefix = <i>None</i>	(StrOpt) Base URL that will be presented to users in links to the OpenStack Compute API
osapi_compute_listen = <i>0.0.0.0</i>	(StrOpt) The IP address on which the OpenStack API will listen.
osapi_compute_listen_port = <i>8774</i>	(IntOpt) The port on which the OpenStack API will listen.
osapi_compute_workers = <i>None</i>	(IntOpt) Number of workers for OpenStack API service. The default will be the number of CPUs available.
osapi_hide_server_address_states = <i>building</i>	(ListOpt) List of instance states that should hide network info
servicegroup_driver = <i>db</i>	(StrOpt) The driver for servicegroup service (valid options are: db, zk, mc)
snapshot_name_template = <i>snapshot-%s</i>	(StrOpt) Template string to be used to generate snapshot names
tcp_keepidle = <i>600</i>	(IntOpt) Sets the value of TCP_KEEPIDLE in seconds for each server socket. Not supported on OS X.
use_forwarded_for = <i>False</i>	(BoolOpt) Treat X-Forwarded-For as the canonical remote address. Only enable this if you have a sanitizing proxy.
wsgi_default_pool_size = <i>1000</i>	(IntOpt) Size of the pool of greenthreads used by wsgi
wsgi_keep_alive = <i>True</i>	(BoolOpt) If False, closes the client socket connection explicitly.

Configuration option = Default value	Description
wsgi_log_format = <i>%(client_ip)s "%(request_line)s" status: %(status_code)s len: %(body_length)s time: %(wall_seconds).7f</i>	(StrOpt) A python format string that is used as the template to generate log lines. The following values can be formatted into it: client_ip, date_time, request_line, status_code, body_length, wall_seconds.

Table 3.10. Description of API v3 configuration options

Configuration option = Default value	Description
[osapi_v3]	
enabled = <i>False</i>	(BoolOpt) Whether the V3 API is enabled or not
extensions_blacklist =	(ListOpt) A list of v3 API extensions to never load. Specify the extension aliases here.
extensions_whitelist =	(ListOpt) If the list is not empty then a v3 API extension will only be loaded if it exists in this list. Specify the extension aliases here.

Table 3.11. Description of authentication configuration options

Configuration option = Default value	Description
[DEFAULT]	
auth_strategy = <i>keystone</i>	(StrOpt) The strategy to use for auth: keystone, noauth (deprecated), or noauth2. Both noauth and noauth2 are designed for testing only, as they do no actual credential checking. noauth provides administrative credentials regardless of the passed in user, noauth2 only does if 'admin' is specified as the username.

Table 3.12. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(StrOpt) Service user password.
admin_tenant_name = <i>admin</i>	(StrOpt) Service tenant name.

Configuration option = Default value	Description
admin_token = <i>None</i>	(StrOpt) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(StrOpt) Service username.
auth_admin_prefix =	(StrOpt) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
auth_host = <i>127.0.0.1</i>	(StrOpt) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_plugin = <i>None</i>	(StrOpt) Name of the plugin to load
auth_port = <i>35357</i>	(IntOpt) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_protocol = <i>https</i>	(StrOpt) Protocol of the admin Identity API endpoint (http or https). Deprecated, use <code>identity_uri</code> .
auth_section = <i>None</i>	(StrOpt) Config Section from which to load plugin specific options
auth_uri = <i>None</i>	(StrOpt) Complete public Identity API endpoint.
auth_version = <i>None</i>	(StrOpt) API version of the admin Identity API endpoint.
cache = <i>None</i>	(StrOpt) Env key for the swift cache.
cafile = <i>None</i>	(StrOpt) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate
check_revocations_for_cached = <i>False</i>	(BoolOpt) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.

Configuration option = Default value	Description
delay_auth_decision = <i>False</i>	(BoolOpt) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(StrOpt) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(ListOpt) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(IntOpt) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(IntOpt) How many times to try to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(StrOpt) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/
include_service_catalog = <i>True</i>	(BoolOpt) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate

Configuration option = Default value	Description
<code>memcache_pool_conn_get_timeout = 10</code>	(IntOpt) (Optional) Number of seconds that an operation will wait to get a memcache client connection from the pool.
<code>memcache_pool_dead_retry = 300</code>	(IntOpt) (Optional) Number of seconds memcached server is considered dead before it is tried again.
<code>memcache_pool_maxsize = 10</code>	(IntOpt) (Optional) Maximum total number of open connections to every memcached server.
<code>memcache_pool_socket_timeout = 3</code>	(IntOpt) (Optional) Socket timeout in seconds for communicating with a memcache server.
<code>memcache_pool_unused_timeout = 60</code>	(IntOpt) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
<code>memcache_secret_key = None</code>	(StrOpt) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
<code>memcache_security_strategy = None</code>	(StrOpt) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. Acceptable values are MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.
<code>memcache_use_advanced_pool = False</code>	(BoolOpt) (Optional) Use the advanced (eventlet safe) memcache client pool. The advanced pool will only work under python 2.x.
<code>revocation_cache_time = 10</code>	(IntOpt) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
<code>signing_dir = None</code>	(StrOpt) Directory used to cache files related to PKI tokens.
<code>token_cache_time = 300</code>	(IntOpt) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 3.13. Description of availability zones configuration options

Configuration option = Default value	Description
[DEFAULT]	
default_availability_zone = <i>nova</i>	(StrOpt) Default compute node availability_zone
default_schedule_zone = <i>None</i>	(StrOpt) Availability zone to use when user does not specify one
internal_service_availability_zone = <i>internal</i>	(StrOpt) The availability_zone to show internal services under

Table 3.14. Description of Barbican configuration options

Configuration option = Default value	Description
[barbican]	
cafile = <i>None</i>	(StrOpt) PEM encoded Certificate Authority to use when verifying HTTPs connections.
catalog_info = <i>key-manager:barbican:public</i>	(StrOpt) Info to match when looking for barbican in the service catalog. Format is: separated values of the form: <service_type>:<service_name>:<endpoint_type>
certfile = <i>None</i>	(StrOpt) PEM encoded client certificate cert file
endpoint_template = <i>None</i>	(StrOpt) Override service catalog lookup with template for barbican endpoint e.g. <code>http://localhost:9311/v1/%(project_id)s</code>
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) PEM encoded client certificate key file
os_region_name = <i>None</i>	(StrOpt) Region name of this node
timeout = <i>None</i>	(IntOpt) Timeout value for http requests

Table 3.15. Description of CA and SSL configuration options

Configuration option = Default value	Description
[DEFAULT]	
ca_file = <i>cacert.pem</i>	(StrOpt) Filename of root CA

Configuration option = Default value	Description
ca_path = <i>\$state_path/CA</i>	(StrOpt) Where to keep the root CA
cert = <i>self.pem</i>	(StrOpt) SSL certificate file
cert_manager = <i>nova.cert.manager.CertManager</i>	(StrOpt) Full class name for the Manager for cert
cert_topic = <i>cert</i>	(StrOpt) The topic cert nodes listen on
crl_file = <i>crl.pem</i>	(StrOpt) Filename of root Certificate Revocation List
key_file = <i>private/cakey.pem</i>	(StrOpt) Filename of private key
keys_path = <i>\$state_path/keys</i>	(StrOpt) Where to keep the keys
project_cert_subject = <i>/C=US/ST=California/O=OpenStack/OU=NovaDev/CN=project-ca-%.16s-%s</i>	(StrOpt) Subject for certificate for projects, %s for project, timestamp
ssl_ca_file = <i>None</i>	(StrOpt) CA certificate file to use to verify connecting clients
ssl_cert_file = <i>None</i>	(StrOpt) SSL certificate of API server
ssl_key_file = <i>None</i>	(StrOpt) SSL private key of API server
use_project_ca = <i>False</i>	(BoolOpt) Should a CA be used for each project?
user_cert_subject = <i>/C=US/ST=California/O=OpenStack/OU=NovaDev/CN=%s.16s-%.16s-%s</i>	(StrOpt) Subject for certificate for users, %s for project, user, timestamp
[ssl]	
ca_file = <i>None</i>	(StrOpt) CA certificate file to use to verify connecting clients.
cert_file = <i>None</i>	(StrOpt) Certificate file to use when starting the server securely.
key_file = <i>None</i>	(StrOpt) Private key file to use when starting the server securely.

Table 3.16. Description of cell configuration options

Configuration option = Default value	Description
[cells]	
call_timeout = 60	(IntOpt) Seconds to wait for response from a call to a cell.
capabilities = <i>hypervisor=xenserver;kvm, os=linux</i>	(ListOpt) Key/Multi-value list with the capabilities of the cell
cell_type = <i>compute</i>	(StrOpt) Type of cell: api or compute
cells_config = <i>None</i>	(StrOpt) Configuration file from which to read cells configuration. If given, overrides reading cells from the database.
db_check_interval = 60	(IntOpt) Interval, in seconds, for getting fresh cell information from the database.
driver = <i>nova.cells.rpc_driver.CellsRPCDriver</i>	(StrOpt) Cells communication driver to use
enable = <i>False</i>	(BoolOpt) Enable cell functionality
instance_update_num_instances = 1	(IntOpt) Number of instances to update per periodic task run
instance_updated_at_threshold = 3600	(IntOpt) Number of seconds after an instance was updated or deleted to continue to update cells
manager = <i>nova.cells.manager.CellsManager</i>	(StrOpt) Manager for cells
max_hop_count = 10	(IntOpt) Maximum number of hops for cells routing.
mute_child_interval = 300	(IntOpt) Number of seconds after which a lack of capability and capacity updates signals the child cell is to be treated as a mute.
mute_weight_multiplier = -10.0	(FloatOpt) Multiplier used to weigh mute children. (The value should be negative.)
mute_weight_value = 1000.0	(FloatOpt) Weight value assigned to mute children. (The value should be positive.)
name = <i>nova</i>	(StrOpt) Name of this cell
offset_weight_multiplier = 1.0	(FloatOpt) Multiplier used to weigh offset weigher.
reserve_percent = 10.0	(FloatOpt) Percentage of cell capacity to hold in reserve. Affects both memory and disk utilization

Configuration option = Default value	Description
topic = <i>cells</i>	(StrOpt) The topic cells nodes listen on

Table 3.17. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
bindir = <i>/usr/local/bin</i>	(StrOpt) Directory where nova binaries are installed
compute_topic = <i>compute</i>	(StrOpt) The topic compute nodes listen on
console_topic = <i>console</i>	(StrOpt) The topic console proxy nodes listen on
consoleauth_topic = <i>consoleauth</i>	(StrOpt) The topic console auth proxy nodes listen on
host = <i>localhost</i>	(StrOpt) Name of this node. This can be an opaque identifier. It is not necessarily a hostname, FQDN, or IP address. However, the node name must be valid within an AMQP key.
memcached_servers = <i>None</i>	(ListOpt) Memcached servers or None for in process cache.
my_ip = <i>10.0.0.1</i>	(StrOpt) IP address of this host
notify_api_faults = <i>False</i>	(BoolOpt) If set, send api.fault notifications on caught exceptions in the API service.
notify_on_state_change = <i>None</i>	(StrOpt) If set, send compute.instance.update notifications on instance state changes. Valid values are None for no notifications, "vm_state" for notifications on VM state changes, or "vm_and_task_state" for notifications on VM and task state changes.
pybasedir = <i>/usr/lib/python/site-packages/nova</i>	(StrOpt) Directory where the nova python module is installed
report_interval = <i>10</i>	(IntOpt) Seconds between nodes reporting state to datastore

Configuration option = Default value	Description
rootwrap_config = <i>/etc/nova/rootwrap.conf</i>	(StrOpt) Path to the rootwrap configuration file to use for running commands as root
service_down_time = <i>60</i>	(IntOpt) Maximum time since last check-in for up service
state_path = <i>\$pybasedir</i>	(StrOpt) Top-level directory for maintaining nova's state
tempdir = <i>None</i>	(StrOpt) Explicitly specify the temporary working directory
[keystone_authtoken]	
memcached_servers = <i>None</i>	(ListOpt) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.
[workarounds]	
destroy_after_evacuate = <i>True</i>	(BoolOpt) Whether to destroy instances on startup when it is suspected that they have previously been evacuated. This can result in data loss if undesired. See https://launchpad.net/bugs/1419785
disable_libvirt_livesnapshot = <i>True</i>	(BoolOpt) When using libvirt 1.2.2 fails live snapshots intermittently under load. This config option provides mechanism to disable livesnapshot while this is resolved. See https://bugs.launchpad.net/nova/+bug/1334398
disable_rootwrap = <i>False</i>	(BoolOpt) This option allows a fallback to sudo for performance reasons. For example see https://bugs.launchpad.net/nova/+bug/1415106

Table 3.18. Description of Compute configuration options

Configuration option = Default value	Description
[DEFAULT]	
compute_available_monitors = <i>['nova.compute.monitors.all_monitors']</i>	(MultiStrOpt) Monitor classes available to the compute which may be specified more than once.

Configuration option = Default value	Description
compute_driver = <i>None</i>	(StrOpt) Driver to use for controlling virtualization. Options include: libvirt.LibvirtDriver, xenapi.XenAPIDriver, fake.FakeDriver, baremetal.BareMetalDriver, vmwareapi.VMwareVCDriver, hyperv.HyperVDriver
compute_manager = <i>nova.compute.manager.ComputeManager</i>	(StrOpt) Full class name for the Manager for compute
compute_monitors =	(ListOpt) A list of monitors that can be used for getting compute metrics.
compute_resources = <i>vcpu</i>	(ListOpt) The names of the extra resources to track.
compute_stats_class = <i>nova.compute.stats.Stats</i>	(StrOpt) Class that will manage stats for the local compute host
console_host = <i>localhost</i>	(StrOpt) Console proxy host to use to connect to instances on this host.
console_manager = <i>nova.console.manager.ConsoleProxyManager</i>	(StrOpt) Full class name for the Manager for console proxy
default_flavor = <i>m1.small</i>	(StrOpt) Default flavor to use for the EC2 API only. The Nova API does not support a default flavor.
default_notification_level = <i>INFO</i>	(StrOpt) Default notification level for outgoing notifications
enable_instance_password = <i>True</i>	(BoolOpt) Enables returning of the instance password by the relevant server API calls such as create, rebuild or rescue, If the hypervisor does not support password injection then the password returned will not be correct
heal_instance_info_cache_interval = <i>60</i>	(IntOpt) Number of seconds between instance network information cache updates
image_cache_manager_interval = <i>2400</i>	(IntOpt) Number of seconds to wait between runs of the image cache manager. Set to -1 to disable. Setting this to 0 will run at the default rate.
image_cache_subdirectory_name = <i>_base</i>	(StrOpt) Where cached images are stored under \$instances_path. This is NOT the full path - only a folder name. For per-compute-host cached images, set to _base_\$my_ip

Configuration option = Default value	Description
instance_build_timeout = 0	(IntOpt) Amount of time in seconds an instance can be in BUILD before going into ERROR status. Set to 0 to disable.
instance_delete_interval = 300	(IntOpt) Interval in seconds for retrying failed instance file deletes. Set to -1 to disable. Setting this to 0 will run at the default rate.
instance_usage_audit = <i>False</i>	(BoolOpt) Generate periodic compute.instance.exists notifications
instance_usage_audit_period = <i>month</i>	(StrOpt) Time period to generate instance usages for. Time period must be hour, day, month or year
instances_path = <i>\$state_path/instances</i>	(StrOpt) Where instances are stored on disk
max_concurrent_builds = 10	(IntOpt) Maximum number of instance builds to run concurrently
maximum_instance_delete_attempts = 5	(IntOpt) The number of times to attempt to reap an instance's files.
reboot_timeout = 0	(IntOpt) Automatically hard reboot an instance if it has been stuck in a rebooting state longer than N seconds. Set to 0 to disable.
reclaim_instance_interval = 0	(IntOpt) Interval in seconds for reclaiming deleted instances
rescue_timeout = 0	(IntOpt) Automatically unrescue an instance after N seconds. Set to 0 to disable.
resize_confirm_window = 0	(IntOpt) Automatically confirm resizes after N seconds. Set to 0 to disable.
resume_guests_state_on_host_boot = <i>False</i>	(BoolOpt) Whether to start guests that were running before the host rebooted
running_deleted_instance_action = <i>reap</i>	(StrOpt) Action to take if a running deleted instance is detected. Valid options are 'noop', 'log', 'shutdown', or 'reap'. Set to 'noop' to take no action.
running_deleted_instance_poll_interval = 1800	(IntOpt) Number of seconds to wait between runs of the cleanup task.
running_deleted_instance_timeout = 0	(IntOpt) Number of seconds after being deleted when a running instance should be considered eligible for cleanup.

Configuration option = Default value	Description
shelved_offload_time = 0	(IntOpt) Time in seconds before a shelved instance is eligible for removing from a host. -1 never offload, 0 offload when shelved
shelved_poll_interval = 3600	(IntOpt) Interval in seconds for polling shelved instances to offload. Set to -1 to disable. Setting this to 0 will run at the default rate.
shutdown_timeout = 60	(IntOpt) Total amount of time to wait in seconds for an instance to perform a clean shutdown.
sync_power_state_interval = 600	(IntOpt) Interval to sync power states between the database and the hypervisor. Set to -1 to disable. Setting this to 0 will run at the default rate.
vif_plugging_is_fatal = <i>True</i>	(BoolOpt) Fail instance boot if vif plugging fails
vif_plugging_timeout = 300	(IntOpt) Number of seconds to wait for neutron vif plugging events to arrive before continuing or failing (see <code>vif_plugging_is_fatal</code>). If this is set to zero and <code>vif_plugging_is_fatal</code> is <i>False</i> , events should not be expected to arrive at all.

Table 3.19. Description of conductor configuration options

Configuration option = Default value	Description
[DEFAULT]	
migrate_max_retries = -1	(IntOpt) Number of times to retry live-migration before failing. If set to -1, try until out of hosts. If set to 0, only try once, no retries.
[conductor]	
manager = <i>nova.conductor.manager.ConductorManager</i>	(StrOpt) Full class name for the Manager for conductor
topic = <i>conductor</i>	(StrOpt) The topic on which conductor nodes listen
use_local = <i>False</i>	(BoolOpt) Perform nova-conductor operations locally

Configuration option = Default value	Description
workers = <i>None</i>	(IntOpt) Number of workers for OpenStack Conductor service. The default will be the number of CPUs available.

Table 3.20. Description of config drive configuration options

Configuration option = Default value	Description
[DEFAULT]	
config_drive_format = <i>iso9660</i>	(StrOpt) Config drive format. One of iso9660 (default) or vfat
config_drive_skip_versions = <i>1.0 2007-01-19 2007-03-01 2007-08-29 2007-10-10 2007-12-15 2008-02-01 2008-09-01</i>	(StrOpt) List of metadata versions to skip placing into the config drive
force_config_drive = <i>None</i>	(StrOpt) Set to "always" to force injection to take place on a config drive. NOTE: The "always" will be deprecated in the Liberty release cycle.
mkisofs_cmd = <i>genisoimage</i>	(StrOpt) Name and optionally path of the tool used for ISO image creation
[hyperv]	
config_drive_cdrom = <i>False</i>	(BoolOpt) Attaches the Config Drive image as a cdrom drive instead of a disk drive
config_drive_inject_password = <i>False</i>	(BoolOpt) Sets the admin password in the config drive image

Table 3.21. Description of console configuration options

Configuration option = Default value	Description
[DEFAULT]	
console_public_hostname = <i>localhost</i>	(StrOpt) Publicly visible name for this console host
console_token_ttl = <i>600</i>	(IntOpt) How many seconds before deleting tokens
consoleauth_manager = <i>nova.consoleauth.manager.ConsoleAuthManager</i>	(StrOpt) Manager for console auth

Table 3.22. Description of database configuration options

Configuration option = Default value	Description
[DEFAULT]	
db_driver = <i>nova.db</i>	(StrOpt) The driver to use for database access
[api_database]	
connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the Nova API database.
connection_debug = <i>0</i>	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(BoolOpt) Add Python stack traces to SQL as comment strings.
idle_timeout = <i>3600</i>	(IntOpt) Timeout before idle SQL connections are reaped.
max_overflow = <i>None</i>	(IntOpt) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = <i>None</i>	(IntOpt) Maximum number of SQL connections to keep open in a pool.
max_retries = <i>10</i>	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
mysql_sql_mode = <i>TRADITIONAL</i>	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode=
pool_timeout = <i>None</i>	(IntOpt) If set, use this value for pool_timeout with SQLAlchemy.
retry_interval = <i>10</i>	(IntOpt) Interval between retries of opening a SQL connection.
slave_connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_synchronous = <i>True</i>	(BoolOpt) If True, SQLite uses synchronous mode.
[database]	

Configuration option = Default value	Description
backend = <i>sqlalchemy</i>	(StrOpt) The back end to use for the database.
connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(BoolOpt) Add Python stack traces to SQL as comment strings.
db_inc_retry_interval = <i>True</i>	(BoolOpt) If True, increases the interval between retries of a database operation up to db_max_retry_interval .
db_max_retries = <i>20</i>	(IntOpt) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = <i>10</i>	(IntOpt) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.
db_retry_interval = <i>1</i>	(IntOpt) Seconds between retries of a database transaction.
idle_timeout = <i>3600</i>	(IntOpt) Timeout before idle SQL connections are reaped.
max_overflow = <i>None</i>	(IntOpt) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = <i>None</i>	(IntOpt) Maximum number of SQL connections to keep open in a pool.
max_retries = <i>10</i>	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = <i>1</i>	(IntOpt) Minimum number of SQL connections to keep open in a pool.
mysql_sql_mode = <i>TRADITIONAL</i>	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode =

Configuration option = Default value	Description
pool_timeout = <i>None</i>	(IntOpt) If set, use this value for pool_timeout with SQLAlchemy.
retry_interval = <i>10</i>	(IntOpt) Interval between retries of opening a SQL connection.
slave_connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = <i>oslo.sqlite</i>	(StrOpt) The file name to use with SQLite.
sqlite_synchronous = <i>True</i>	(BoolOpt) If True, SQLite uses synchronous mode.
use_db_reconnect = <i>False</i>	(BoolOpt) Enable the experimental use of database reconnect on connection lost.
use_tpool = <i>False</i>	(BoolOpt) Enable the experimental use of thread pooling for all DB API calls

Table 3.23. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
backdoor_port = <i>None</i>	(StrOpt) Enable eventlet backdoor. Acceptable values are 0, <port>, and <start>:<end>, where 0 results in listening on a random tcp port number; <port> results in listening on the specified port number (and not enabling backdoor if that port is in use); and <start>:<end> results in listening on the smallest unused port number within the specified range of port numbers. The chosen port is displayed in the service's log file.
[guestfs]	
debug = <i>False</i>	(BoolOpt) Enable guestfs debug

Table 3.24. Description of EC2 configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
ec2_dmz_host = <i>\$my_ip</i>	(StrOpt) The internal IP address of the EC2 API server
ec2_host = <i>\$my_ip</i>	(StrOpt) The IP address of the EC2 API server
ec2_listen = <i>0.0.0.0</i>	(StrOpt) The IP address on which the EC2 API will listen.
ec2_listen_port = <i>8773</i>	(IntOpt) The port on which the EC2 API will listen.
ec2_path = <i>/</i>	(StrOpt) The path prefix used to call the ec2 API server
ec2_port = <i>8773</i>	(IntOpt) The port of the EC2 API server
ec2_private_dns_show_ip = <i>False</i>	(BoolOpt) Return the IP address as private dns hostname in describe instances
ec2_scheme = <i>http</i>	(StrOpt) The protocol to use when connecting to the EC2 API server (http, https)
ec2_strict_validation = <i>True</i>	(BoolOpt) Validate security group names according to EC2 specification
ec2_timestamp_expiry = <i>300</i>	(IntOpt) Time in seconds before ec2 timestamp expires
ec2_workers = <i>None</i>	(IntOpt) Number of workers for EC2 API service. The default will be equal to the number of CPUs available.
keystone_ec2_insecure = <i>False</i>	(BoolOpt) Disable SSL certificate verification.
keystone_ec2_url = <i>http://localhost:5000/v2.0/ec2tokens</i>	(StrOpt) URL to get token from ec2 request.
lockout_attempts = <i>5</i>	(IntOpt) Number of failed auths before lockout.
lockout_minutes = <i>15</i>	(IntOpt) Number of minutes to lockout if triggered.
lockout_window = <i>15</i>	(IntOpt) Number of minutes for lockout window.
region_list =	(ListOpt) List of region=fqdn pairs separated by commas

Table 3.25. Description of ephemeral storage encryption configuration options

Configuration option = Default value	Description
[ephemeral_storage_encryption]	
cipher = <i>aes-xts-plain64</i>	(StrOpt) The cipher and mode to be used to encrypt ephemeral storage. Which ciphers are available ciphers depends on kernel support. See <code>/proc/crypto</code> for the list of available options.
enabled = <i>False</i>	(BoolOpt) Whether to encrypt ephemeral storage
key_size = <i>512</i>	(IntOpt) The bit length of the encryption key to be used to encrypt ephemeral storage (in XTS mode only half of the bits are used for encryption key)

Table 3.26. Description of fping configuration options

Configuration option = Default value	Description
[DEFAULT]	
fping_path = <i>/usr/sbin/fping</i>	(StrOpt) Full path to fping.

Table 3.27. Description of glance configuration options

Configuration option = Default value	Description
[DEFAULT]	
osapi_glance_link_prefix = <i>None</i>	(StrOpt) Base URL that will be presented to users in links to glance resources
[glance]	
allowed_direct_url_schemes =	(ListOpt) A list of url scheme that can be downloaded directly via the direct_url. Currently supported schemes: <code>[file]</code> .
api_insecure = <i>False</i>	(BoolOpt) Allow to perform insecure SSL (https) requests to glance
api_servers = <i>None</i>	(ListOpt) A list of the glance api servers available to nova. Prefix with <code>https://</code> for ssl-based glance api servers. (<code>[hostname ip]:port</code>)
host = <i>\$my_ip</i>	(StrOpt) Default glance hostname or IP address

Configuration option = Default value	Description
num_retries = 0	(IntOpt) Number of retries when uploading / downloading an image to / from glance.
port = 9292	(IntOpt) Default glance port
protocol = <i>http</i>	(StrOpt) Default protocol to use when connecting to glance. Set to https for SSL.
[image_file_url]	
filesystems =	(ListOpt) List of file systems that are configured in this file in the image_file_url:<list entry name> sections

Table 3.28. Description of hypervisor configuration options

Configuration option = Default value	Description
[DEFAULT]	
default_ephemeral_format = <i>None</i>	(StrOpt) The default format an ephemeral_volume will be formatted with on creation.
force_raw_images = <i>True</i>	(BoolOpt) Force backing images to raw format
preallocate_images = <i>none</i>	(StrOpt) VM image preallocation mode: "none" => no storage provisioning is done up front, "space" => storage is fully allocated at instance start
timeout_nbd = 10	(IntOpt) Amount of time, in seconds, to wait for NBD device start up.
use_cow_images = <i>True</i>	(BoolOpt) Whether to use cow images
vcpu_pin_set = <i>None</i>	(StrOpt) Defines which pcpus that instance vcpus can use. For example, "4-12,^8,15"
virt_mkfs = []	(MultiStrOpt) Name of the mkfs commands for ephemeral device. The format is <os_type>=<mkfs command>

Table 3.29. Description of bare metal configuration options

Configuration option = Default value	Description
[ironic]	

Configuration option = Default value	Description
<code>admin_auth_token = None</code>	(StrOpt) Ironic keystone auth token.
<code>admin_password = None</code>	(StrOpt) Ironic keystone admin password.
<code>admin_tenant_name = None</code>	(StrOpt) Ironic keystone tenant name.
<code>admin_url = None</code>	(StrOpt) Keystone public API endpoint.
<code>admin_username = None</code>	(StrOpt) Ironic keystone admin name
<code>api_endpoint = None</code>	(StrOpt) URL for Ironic API endpoint.
<code>api_max_retries = 60</code>	(IntOpt) How many retries when a request does conflict.
<code>api_retry_interval = 2</code>	(IntOpt) How often to retry in seconds when a request does conflict
<code>api_version = 1</code>	(IntOpt) Version of Ironic API service endpoint.
<code>client_log_level = None</code>	(StrOpt) Log level override for ironicclient. Set this in order to override the global "default_log_levels", "verbose", and "debug" settings. DEPRECATED: use standard logging configuration.

Table 3.30. Description of IPv6 configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>fixed_range_v6 = fd00::/48</code>	(StrOpt) Fixed IPv6 address block
<code>gateway_v6 = None</code>	(StrOpt) Default IPv6 gateway
<code>ipv6_backend = rfc2462</code>	(StrOpt) Backend to use for IPv6 generation
<code>use_ipv6 = False</code>	(BoolOpt) Use IPv6

Table 3.31. Description of key manager configuration options

Configuration option = Default value	Description
[keymgr]	

Configuration option = Default value	Description
api_class = <i>nova.keymgr.conf_key_mgr.ConfKeyManager</i>	(StrOpt) The full class name of the key manager API class
fixed_key = <i>None</i>	(StrOpt) Fixed key returned by key manager, specified in hex

Table 3.32. Description of LDAP configuration options

Configuration option = Default value	Description
[DEFAULT]	
ldap_dns_base_dn = <i>ou=hosts,dc=example,dc=org</i>	(StrOpt) Base DN for DNS entries in LDAP
ldap_dns_password = <i>password</i>	(StrOpt) Password for LDAP DNS
ldap_dns_servers = <i>['dns.example.org']</i>	(MultiStrOpt) DNS Servers for LDAP DNS driver
ldap_dns_soa_expiry = <i>86400</i>	(StrOpt) Expiry interval (in seconds) for LDAP DNS driver Statement of Authority
ldap_dns_soa_hostmaster = <i>hostmaster@example.org</i>	(StrOpt) Hostmaster for LDAP DNS driver Statement of Authority
ldap_dns_soa_minimum = <i>7200</i>	(StrOpt) Minimum interval (in seconds) for LDAP DNS driver Statement of Authority
ldap_dns_soa_refresh = <i>1800</i>	(StrOpt) Refresh interval (in seconds) for LDAP DNS driver Statement of Authority
ldap_dns_soa_retry = <i>3600</i>	(StrOpt) Retry interval (in seconds) for LDAP DNS driver Statement of Authority
ldap_dns_url = <i>ldap://ldap.example.com:389</i>	(StrOpt) URL for LDAP server which will store DNS entries
ldap_dns_user = <i>uid=admin,ou=people,dc=example,dc=org</i>	(StrOpt) User for LDAP DNS

Table 3.33. Description of Libvirt configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
remove_unused_base_images = <i>True</i>	(BoolOpt) Should unused base images be removed?
remove_unused_original_minimum_age_seconds = <i>86400</i>	(IntOpt) Unused unresized base images younger than this will not be removed
[libvirt]	
block_migration_flag = <i>VIR_MIGRATE_UNDEFINE_SOURCE,</i> <i>VIR_MIGRATE_PEER2PEER, VIR_MIGRATE_LIVE,</i> <i>VIR_MIGRATE_TUNNELLED,</i> <i>VIR_MIGRATE_NON_SHARED_INC</i>	(StrOpt) Migration flags to be set for block migration
checksum_base_images = <i>False</i>	(BoolOpt) Write a checksum for files in <i>_base</i> to disk
checksum_interval_seconds = <i>3600</i>	(IntOpt) How frequently to checksum base images
connection_uri =	(StrOpt) Override the default libvirt URI (which is dependent on <i>virt_type</i>)
cpu_mode = <i>None</i>	(StrOpt) Set to "host-model" to clone the host CPU feature flags; to "host-passthrough" to use the host CPU model exactly; to "custom" to use a named CPU model; to "none" to not set any CPU model. If <i>virt_type</i> ="kvm qemu", it will default to "host-model", otherwise it will default to "none"
cpu_model = <i>None</i>	(StrOpt) Set to a named libvirt CPU model (see names listed in /usr/share/libvirt/cpu_map.xml). Only has effect if <i>cpu_mode</i> ="custom" and <i>virt_type</i> ="kvm qemu"
disk_cachemodes =	(ListOpt) Specific cachemodes to use for different disk types e.g: file=directsync,block=none
disk_prefix = <i>None</i>	(StrOpt) Override the default disk prefix for the devices attached to a server, which is dependent on <i>virt_type</i> . (valid options are: sd, xvd, uvd, vd)
gid_maps =	(ListOpt) List of guid targets and ranges.Syntax is guest-gid:host-gid:countMaximum of 5 allowed.
hw_disk_discard = <i>None</i>	(StrOpt) Discard option for nova managed disks (valid options are: ignore, unmap). Need Libvirt(1.0.6) Qemu1.5 (raw format) Qemu1.6(qcow2 format)

Configuration option = Default value	Description
hw_machine_type = <i>None</i>	(ListOpt) For qemu or KVM guests, set this option to specify a default machine type per host architecture. You can find a list of supported machine types in your environment by checking the output of the "virsh capabilities" command. The format of the value for this config option is host-arch=machine-type. For example: x86_64=machinetype1,armv7l=machinetype2
image_info_filename_pattern = <i>\$instances_path/\$image_cache_subdirectory_name/%(image)s.info</i>	(StrOpt) Allows image information files to be stored in non-standard locations
images_rbd_ceph_conf =	(StrOpt) Path to the ceph configuration file to use
images_rbd_pool = <i>rbd</i>	(StrOpt) The RADOS pool in which rbd volumes are stored
images_type = <i>default</i>	(StrOpt) VM Images format. Acceptable values are: raw, qcow2, lvm, rbd, default. If default is specified, then use_cow_images flag is used instead of this one.
images_volume_group = <i>None</i>	(StrOpt) LVM Volume Group that is used for VM images, when you specify images_type=lvm.
inject_key = <i>False</i>	(BoolOpt) Inject the ssh public key at boot time
inject_partition = <i>-2</i>	(IntOpt) The partition to inject to : -2 => disable, -1 => inspect (libguestfs only), 0 => not partitioned, >0 => partition number
inject_password = <i>False</i>	(BoolOpt) Inject the admin password at boot time, without an agent.
iscsi_iface = <i>None</i>	(StrOpt) The iSCSI transport iface to use to connect to target in case offload support is desired. Supported transports are be2iscsi, bnx2i, cxgb3i, cxgb4i, qla4xxx and ocs. Default format is transport_name.hwaddress and can be generated manually or via iscsiadm -m iface
iscsi_use_multipath = <i>False</i>	(BoolOpt) Use multipath connection of the iSCSI volume
iser_use_multipath = <i>False</i>	(BoolOpt) Use multipath connection of the iSER volume

Configuration option = Default value	Description
mem_stats_period_seconds = 10	(IntOpt) A number of seconds to memory usage statistics period. Zero or negative value mean to disable memory usage statistics.
remove_unused_kernels = False	(BoolOpt) Should unused kernel images be removed? This is only safe to enable if all compute nodes have been updated to support this option. This will be enabled by default in future.
remove_unused_resized_minimum_age_seconds = 3600	(IntOpt) Unused resized base images younger than this will not be removed
rescue_image_id = None	(StrOpt) Rescue ami image. This will not be used if an image id is provided by the user.
rescue_kernel_id = None	(StrOpt) Rescue aki image
rescue_ramdisk_id = None	(StrOpt) Rescue ari image
rng_dev_path = None	(StrOpt) A path to a device that will be used as source of entropy on the host. Permitted options are: /dev/random or /dev/hwrng
snapshot_compression = False	(BoolOpt) Compress snapshot images when possible. This currently applies exclusively to qcow2 images
snapshot_image_format = None	(StrOpt) Snapshot image format (valid options are : raw, qcow2, vmdk, vdi). Defaults to same as source image
snapshots_directory = <i>\$instances_path/snapshots</i>	(StrOpt) Location where libvirt driver will store snapshots before uploading them to image service
sparse_logical_volumes = False	(BoolOpt) Create sparse logical volumes (with virtualsize) if this flag is set to True.
sysinfo_serial = auto	(StrOpt) The data source used to the populate the host "serial" UUID exposed to guest in the virtual BIOS. Permitted options are "hardware", "os", "none" or "auto" (default).
uid_maps =	(ListOpt) List of uid targets and ranges. Syntax is guest-uid:host-uid:count . Maximum of 5 allowed.
use_usb_tablet = True	(BoolOpt) Sync virtual and real mouse cursors (Not applicable to Red Hat Enterprise Linux VMs)

Configuration option = Default value	Description
<code>use_virtio_for_bridges = True</code>	(BoolOpt) Use virtio for bridge interfaces with KVM/QEMU
<code>virt_type = kvm</code>	(StrOpt) Libvirt domain type (valid options are: kvm, lxc, qemu, uml, xen and parallels)
<code>volume_clear = zero</code>	(StrOpt) Method used to wipe old volumes (valid options are: none, zero, shred)
<code>volume_clear_size = 0</code>	(IntOpt) Size in MiB to wipe at start of old volumes. 0 => all
<code>wait_soft_reboot_seconds = 120</code>	(IntOpt) Number of seconds to wait for instance to shut down after soft reboot request is made. Fall back to hard reboot if instance does not shut down within this window.

Table 3.34. Description of live migration configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>live_migration_retry_count = 30</code>	(IntOpt) Number of 1 second retries needed in live_migration
[libvirt]	
<code>live_migration_bandwidth = 0</code>	(IntOpt) Maximum bandwidth to be used during migration, in Mbps
<code>live_migration_flag = VIR_MIGRATE_UNDEFINE_SOURCE, VIR_MIGRATE_PEER2PEER, VIR_MIGRATE_LIVE, VIR_MIGRATE_TUNNELLED</code>	(StrOpt) Migration flags to be set for live migration
<code>live_migration_uri = qemu+tcp://%s/system</code>	(StrOpt) Migration target URI (any included "%s" is replaced with the migration target hostname)

Table 3.35. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
debug = <i>False</i>	(BoolOpt) Print debugging output (set logging level to DEBUG instead of default WARNING level).
default_log_levels = <i>amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystonemiddleware=WARN, routes.middleware=WARN, stevedore=WARN</i>	(ListOpt) List of logger=LEVEL pairs.
fatal_deprecations = <i>False</i>	(BoolOpt) Enables or disables fatal status of deprecations.
fatal_exception_format_errors = <i>False</i>	(BoolOpt) Make exception message format errors fatal
instance_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation.
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(StrOpt) Format string for <code>%(asctime)s</code> in log records. Default: <code>%(default)s</code> .
log_dir = <i>None</i>	(StrOpt) (Optional) The base directory used for relative <code>--log-file</code> paths.
log_file = <i>None</i>	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout.
log_format = <i>None</i>	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecated. Use <code>logging_context_format_string</code> and <code>logging_default_format_string</code> instead.

Configuration option = Default value	Description
log_config_append = <i>None</i>	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation.
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(StrOpt) Format string for <i>%(asctime)s</i> in log records. Default: <i>%(default)s</i> .
log_dir = <i>None</i>	(StrOpt) (Optional) The base directory used for relative <i>--log-file</i> paths.
log_file = <i>None</i>	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout.
log_format = <i>None</i>	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecated. Use <i>logging_context_format_string</i> and <i>logging_default_format_string</i> instead.
logging_context_format_string = <i>%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%(message)s</i>	(StrOpt) Format string to use for log messages with context.
logging_debug_format_suffix = <i>%(funcName)s %(pathname)s:%(lineno)d</i>	(StrOpt) Data to append to log format when level is DEBUG.
logging_default_format_string = <i>%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [-] %(instance)s%(message)s</i>	(StrOpt) Format string to use for log messages without context.
logging_exception_prefix = <i>%(asctime)s.%(msecs)03d %(process)d TRACE %(name)s %(instance)s</i>	(StrOpt) Prefix each line of exception output with this format.
publish_errors = <i>False</i>	(BoolOpt) Enables or disables publication of error events.
syslog_log_facility = <i>LOG_USER</i>	(StrOpt) Syslog facility to receive log lines.
syslog_log_facility = <i>LOG_USER</i>	(StrOpt) Syslog facility to receive log lines.
use_syslog = <i>False</i>	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED during I, and will change in J to honor RFC5424.

Configuration option = Default value	Description
<code>use_syslog_rfc_format = False</code>	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in I, and will be removed in J.
<code>use_stderr = True</code>	(BoolOpt) Log output to standard error.
<code>use_syslog = False</code>	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED during I, and will change in J to honor RFC5424.
<code>use_syslog_rfc_format = False</code>	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in I, and will be removed in J.
<code>verbose = False</code>	(BoolOpt) Print more verbose output (set logging level to INFO instead of default WARNING level).

Table 3.36. Description of metadata configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>metadata_cache_expiration = 15</code>	(IntOpt) Time in seconds to cache metadata; 0 to disable metadata caching entirely (not recommended). Increasing this should improve response times of the metadata API when under heavy load. Higher values may increase memory usage and result in longer times for host metadata changes to take effect.
<code>metadata_host = \$my_ip</code>	(StrOpt) The IP address for the metadata API server
<code>metadata_listen = 0.0.0.0</code>	(StrOpt) The IP address on which the metadata API will listen.
<code>metadata_listen_port = 8775</code>	(IntOpt) The port on which the metadata API will listen.
<code>metadata_manager = nova.api.manager.MetadataManager</code>	(StrOpt) OpenStack metadata service manager
<code>metadata_port = 8775</code>	(IntOpt) The port for the metadata API port

Configuration option = Default value	Description
metadata_workers = <i>None</i>	(IntOpt) Number of workers for metadata service. The default will be the number of CPUs available.
vendordata_driver = <i>nova.api.metadata.vendordata_json.JsonFileVendorData</i>	(StrOpt) Driver to use for vendor data
vendordata_jsonfile_path = <i>None</i>	(StrOpt) File to load JSON formatted vendor data from

Table 3.37. Description of network configuration options

Configuration option = Default value	Description
[DEFAULT]	
allow_same_net_traffic = <i>True</i>	(BoolOpt) Whether to allow network traffic from same network
auto_assign_floating_ip = <i>False</i>	(BoolOpt) Autoassigning floating IP to VM
cnt_vpn_clients = <i>0</i>	(IntOpt) Number of addresses reserved for vpn clients
create_unique_mac_address_attempts = <i>5</i>	(IntOpt) Number of attempts to create unique mac address
default_access_ip_network_name = <i>None</i>	(StrOpt) Name of network to use to set access IPs for instances
default_floating_pool = <i>nova</i>	(StrOpt) Default pool for floating IPs
defer_iptables_apply = <i>False</i>	(BoolOpt) Whether to batch up the application of IPTables rules during a host restart and apply all at the end of the init phase
dhcp_domain = <i>novalocal</i>	(StrOpt) Domain to use for building the hostnames
dhcp_lease_time = <i>86400</i>	(IntOpt) Lifetime of a DHCP lease in seconds
dhcpbridge = <i>\$bindir/nova-dhcpbridge</i>	(StrOpt) Location of nova-dhcpbridge
dhcpbridge_flagfile = <i>['/etc/nova/nova-dhcpbridge.conf']</i>	(MultiStrOpt) Location of flagfiles for dhcpbridge
dns_server = <i>[]</i>	(MultiStrOpt) If set, uses specific DNS server for dnsmasq. Can be specified multiple times.

Configuration option = Default value	Description
dns_update_periodic_interval = -1	(IntOpt) Number of seconds to wait between runs of updates to DNS entries.
dnsmasq_config_file =	(StrOpt) Override the default dnsmasq settings with this file
ebtables_exec_attempts = 3	(IntOpt) Number of times to retry ebtables commands on failure.
ebtables_retry_interval = 1.0	(FloatOpt) Number of seconds to wait between ebtables retries.
firewall_driver = <i>None</i>	(StrOpt) Firewall driver (defaults to hypervisor specific iptables driver)
fixed_ip_disassociate_timeout = 600	(IntOpt) Seconds after which a deallocated IP is disassociated
flat_injected = <i>False</i>	(BoolOpt) Whether to attempt to inject network setup into guest
flat_interface = <i>None</i>	(StrOpt) FlatDhcp will bridge into this interface if set
flat_network_bridge = <i>None</i>	(StrOpt) Bridge for simple network instances
flat_network_dns = 8.8.4.4	(StrOpt) DNS server for simple network
floating_ip_dns_manager = <i>nova.network.noop_dns_driver.NoopDNSDriver</i>	(StrOpt) Full class name for the DNS Manager for floating IPs
force_dhcp_release = <i>True</i>	(BoolOpt) If True, send a dhcp release on instance termination
force_snat_range = []	(MultiStrOpt) Traffic to this range will always be snatted to the fallback ip, even if it would normally be bridged out of the node. Can be specified multiple times.
forward_bridge_interface = [<i>'all'</i>]	(MultiStrOpt) An interface that bridges can forward to. If this is set to all then all traffic will be forwarded. Can be specified multiple times.
gateway = <i>None</i>	(StrOpt) Default IPv4 gateway
injected_network_template = <i>\$pybasedir/nova/virt/interfaces.template</i>	(StrOpt) Template file for injected network

Configuration option = Default value	Description
instance_dns_domain =	(StrOpt) Full class name for the DNS Zone for instance IPs
instance_dns_manager = <i>nova.network.noop_dns_driver.NoopDNSDriver</i>	(StrOpt) Full class name for the DNS Manager for instance IPs
iptables_bottom_regex =	(StrOpt) Regular expression to match the iptables rule that should always be on the bottom.
iptables_drop_action = <i>DROP</i>	(StrOpt) The table that iptables to jump to when a packet is to be dropped.
iptables_top_regex =	(StrOpt) Regular expression to match the iptables rule that should always be on the top.
l3_lib = <i>nova.network.l3.LinuxNetL3</i>	(StrOpt) Indicates underlying L3 management library
linuxnet_interface_driver = <i>nova.network.linux_net.LinuxBridgeInterfaceDriver</i>	(StrOpt) Driver used to create ethernet devices.
linuxnet_ovs_integration_bridge = <i>br-int</i>	(StrOpt) Name of Open vSwitch bridge used with linuxnet
multi_host = <i>False</i>	(BoolOpt) Default value for multi_host in networks. Also, if set, some rpc network calls will be sent directly to host.
network_allocate_retries = <i>0</i>	(IntOpt) Number of times to retry network allocation on failures
network_api_class = <i>nova.network.api.API</i>	(StrOpt) The full class name of the network API class to use
network_device_mtu = <i>None</i>	(IntOpt) DEPRECATED: THIS VALUE SHOULD BE SET WHEN CREATING THE NETWORK. MTU setting for network interface.
network_driver = <i>nova.network.linux_net</i>	(StrOpt) Driver to use for network creation
network_manager = <i>nova.network.manager.VlanManager</i>	(StrOpt) Full class name for the Manager for network
network_size = <i>256</i>	(IntOpt) Number of addresses in each private subnet
network_topic = <i>network</i>	(StrOpt) The topic network nodes listen on

Configuration option = Default value	Description
networks_path = <i>\$state_path/networks</i>	(StrOpt) Location to keep network config files
num_networks = <i>1</i>	(IntOpt) Number of networks to support
ovs_vsctl_timeout = <i>120</i>	(IntOpt) Amount of time, in seconds, that ovs_vsctl should wait for a response from the database. 0 is to wait forever.
public_interface = <i>eth0</i>	(StrOpt) Interface for public IP addresses
routing_source_ip = <i>\$my_ip</i>	(StrOpt) Public IP of network host
security_group_api = <i>nova</i>	(StrOpt) The full class name of the security API class
send_arp_for_ha = <i>False</i>	(BoolOpt) Send gratuitous ARPs for HA setup
send_arp_for_ha_count = <i>3</i>	(IntOpt) Send this many gratuitous ARPs for HA setup
share_dhcp_address = <i>False</i>	(BoolOpt) DEPRECATED: THIS VALUE SHOULD BE SET WHEN CREATING THE NETWORK. If True in multi_host mode, all compute hosts share the same dhcp address. The same IP address used for DHCP will be added on each nova-network node which is only visible to the vms on the same host.
teardown_unused_network_gateway = <i>False</i>	(BoolOpt) If True, unused gateway devices (VLAN and bridge) are deleted in VLAN network mode with multi hosted networks
update_dns_entries = <i>False</i>	(BoolOpt) If True, when a DNS entry must be updated, it sends a fanout cast to all network hosts to update their DNS entries in multi host mode
use_network_dns_servers = <i>False</i>	(BoolOpt) If set, uses the dns1 and dns2 from the network ref. as dns servers.
use_neutron_default_nets = <i>False</i>	(StrOpt) Control for checking for default networks
use_single_default_gateway = <i>False</i>	(BoolOpt) Use single default gateway. Only first nic of vm will get default gateway from dhcp server
vlan_interface = <i>None</i>	(StrOpt) VLANs will bridge into this interface if set
vlan_start = <i>100</i>	(IntOpt) First VLAN for private networks
[vmware]	

Configuration option = Default value	Description
vlan_interface = <i>vmnic0</i>	(StrOpt) Physical ethernet adapter name for vlan networking

Table 3.38. Description of neutron configuration options

Configuration option = Default value	Description
[DEFAULT]	
neutron_default_tenant_id = <i>default</i>	(StrOpt) Default tenant id when creating neutron networks
[neutron]	
admin_auth_url = <i>http://localhost:5000/v2.0</i>	(StrOpt) Authorization URL for connecting to neutron in admin context. DEPRECATED: specify an auth_plugin and appropriate credentials instead.
admin_password = <i>None</i>	(StrOpt) Password for connecting to neutron in admin context DEPRECATED: specify an auth_plugin and appropriate credentials instead.
admin_tenant_id = <i>None</i>	(StrOpt) Tenant id for connecting to neutron in admin context DEPRECATED: specify an auth_plugin and appropriate credentials instead.
admin_tenant_name = <i>None</i>	(StrOpt) Tenant name for connecting to neutron in admin context. This option will be ignored if neutron_admin_tenant_id is set. Note that with Keystone V3 tenant names are only unique within a domain. DEPRECATED: specify an auth_plugin and appropriate credentials instead.
admin_user_id = <i>None</i>	(StrOpt) User id for connecting to neutron in admin context. DEPRECATED: specify an auth_plugin and appropriate credentials instead.
admin_username = <i>None</i>	(StrOpt) Username for connecting to neutron in admin context DEPRECATED: specify an auth_plugin and appropriate credentials instead.
allow_duplicate_networks = <i>False</i>	(BoolOpt) DEPRECATED: Allow an instance to have multiple vNICs attached to the same Neutron network. This option is deprecated in the 2015.1 release and will be removed in the 2015.2 release where the default behavior will be to always allow multiple ports from the same network to be attached to an instance.

Configuration option = Default value	Description
auth_plugin = <i>None</i>	(StrOpt) Name of the plugin to load
auth_section = <i>None</i>	(StrOpt) Config Section from which to load plugin specific options
auth_strategy = <i>keystone</i>	(StrOpt) Authorization strategy for connecting to neutron in admin context. DEPRECATED: specify an auth_plugin and appropriate credentials instead. If an auth_plugin is specified strategy will be ignored.
cafile = <i>None</i>	(StrOpt) PEM encoded Certificate Authority to use when verifying HTTPs connections.
certfile = <i>None</i>	(StrOpt) PEM encoded client certificate cert file
extension_sync_interval = <i>600</i>	(IntOpt) Number of seconds before querying neutron for extensions
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) PEM encoded client certificate key file
metadata_proxy_shared_secret =	(StrOpt) Shared secret to validate proxies Neutron metadata requests
ovs_bridge = <i>br-int</i>	(StrOpt) Name of Integration Bridge used by Open vSwitch
region_name = <i>None</i>	(StrOpt) Region name for connecting to neutron in admin context
service_metadata_proxy = <i>False</i>	(BoolOpt) Set flag to indicate Neutron will proxy metadata requests and resolve instance ids.
timeout = <i>None</i>	(IntOpt) Timeout value for http requests
url = <i>http://127.0.0.1:9696</i>	(StrOpt) URL for connecting to neutron

Table 3.39. Description of oslo_middleware configuration options

Configuration option = Default value	Description
[oslo_middleware]	
max_request_body_size = <i>114688</i>	(IntOpt) The maximum body size for each request, in bytes.

Table 3.40. Description of PCI configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>pci_alias = []</code>	(MultiStrOpt) An alias for a PCI passthrough device requirement. This allows users to specify the alias in the <code>extra_spec</code> for a flavor, without needing to repeat all the PCI property requirements. For example: <code>pci_alias = { "name": "QuicAssist", "product_id": "0443", "vendor_id": "8086", "device_type": "ACCEL" }</code> defines an alias for the Intel QuickAssist card. (multi valued)
<code>pci_passthrough_whitelist = []</code>	(MultiStrOpt) White list of PCI devices available to VMs. For example: <code>pci_passthrough_whitelist = [{"vendor_id": "8086", "product_id": "0443"}]</code>

Table 3.41. Description of periodic configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>periodic_enable = True</code>	(BoolOpt) Enable periodic tasks
<code>periodic_fuzzy_delay = 60</code>	(IntOpt) Range of seconds to randomly delay when starting the periodic task scheduler to reduce stampeding. (Disable by setting to 0)
<code>run_external_periodic_tasks = True</code>	(BoolOpt) Some periodic tasks can be run in a separate process. Should they run here?

Table 3.42. Description of policy configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>allow_instance_snapshots = True</code>	(BoolOpt) Permit instance snapshot operations.
<code>allow_migrate_to_same_host = False</code>	(BoolOpt) Allow migrate machine to the same host. Useful when testing in single-host environments.
<code>allow_resize_to_same_host = False</code>	(BoolOpt) Allow destination machine to match source for resize. Useful when testing in single-host environments.

Configuration option = Default value	Description
max_age = 0	(IntOpt) Number of seconds between subsequent usage refreshes. This defaults to 0(off) to avoid additional load but it is useful to turn on to help keep quota usage up to date and reduce the impact of out of sync usage issues. Note that quotas are not updated on a periodic task, they will update on a new reservation if max_age has passed since the last reservation
max_local_block_devices = 3	(IntOpt) Maximum number of devices that will result in a local image being created on the hypervisor node. Setting this to 0 means nova will allow only boot from volume. A negative number means unlimited.
osapi_compute_unique_server_name_scope =	(StrOpt) When set, compute API will consider duplicate hostnames invalid within the specified scope, regardless of case. Should be empty, "project" or "global".
osapi_max_limit = 1000	(IntOpt) The maximum number of items returned in a single response from a collection resource
password_length = 12	(IntOpt) Length of generated instance admin passwords
policy_default_rule = default	(StrOpt) Default rule. Enforced when a requested rule is not found.
policy_dirs = ['policy.d']	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the config_dir option, or absolute paths. The file defined by policy_file must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = policy.json	(StrOpt) The JSON file that defines policies.
reservation_expire = 86400	(IntOpt) Number of seconds until a reservation expires
resize_fs_using_block_device = False	(BoolOpt) Attempt to resize the filesystem by accessing the image over a block device. This is done by the host and may not be necessary if the image contains a recent version of cloud-init. Possible mechanisms require the nbd driver (for qcow and raw), or loop (for raw).

Configuration option = Default value	Description
until_refresh = 0	(IntOpt) Count of reservations until usage is refreshed. This defaults to 0(off) to avoid additional load but it is useful to turn on to help keep quota usage up to date and reduce the impact of out of sync usage issues.

Table 3.43. Description of Quobyte USP volume driver configuration options

Configuration option = Default value	Description
[libvirt]	
quobyte_client_cfg = None	(StrOpt) Path to a Quobyte Client configuration file.
quobyte_mount_point_base = \$state_path/mnt	(StrOpt) Directory where the Quobyte volume is mounted on the compute node

Table 3.44. Description of quota configuration options

Configuration option = Default value	Description
[DEFAULT]	
bandwidth_poll_interval = 600	(IntOpt) Interval to pull network bandwidth usage info. Not supported on all hypervisors. Set to -1 to disable. Setting this to 0 will run at the default rate.
enable_network_quota = False	(BoolOpt) Enables or disables quota checking for tenant networks
quota_cores = 20	(IntOpt) Number of instance cores allowed per project
quota_driver = nova.quota.DbQuotaDriver	(StrOpt) Default driver to use for quota checks
quota_fixed_ips = -1	(IntOpt) Number of fixed IPs allowed per project (this should be at least the number of instances allowed)
quota_floating_ips = 10	(IntOpt) Number of floating IPs allowed per project
quota_injected_file_content_bytes = 10240	(IntOpt) Number of bytes allowed per injected file
quota_injected_file_path_length = 255	(IntOpt) Length of injected file path

Configuration option = Default value	Description
quota_injected_files = 5	(IntOpt) Number of injected files allowed
quota_instances = 10	(IntOpt) Number of instances allowed per project
quota_key_pairs = 100	(IntOpt) Number of key pairs per user
quota_metadata_items = 128	(IntOpt) Number of metadata items allowed per instance
quota_networks = 3	(IntOpt) Number of private networks allowed per project
quota_ram = 51200	(IntOpt) Megabytes of instance RAM allowed per project
quota_security_group_rules = 20	(IntOpt) Number of security rules per security group
quota_security_groups = 10	(IntOpt) Number of security groups per project
quota_server_group_members = 10	(IntOpt) Number of servers per server group
quota_server_groups = 10	(IntOpt) Number of server groups per project
[cells]	
bandwidth_update_interval = 600	(IntOpt) Seconds between bandwidth updates for cells.

Table 3.45. Description of RDP configuration options

Configuration option = Default value	Description
[rdp]	
enabled = <i>False</i>	(BoolOpt) Enable RDP related features
html5_proxy_base_url = <i>http://127.0.0.1:6083/</i>	(StrOpt) Location of RDP html5 console proxy, in the form "http://127.0.0.1:6083/"

Table 3.46. Description of Redis configuration options

Configuration option = Default value	Description
[matchmaker_redis]	

Configuration option = Default value	Description
host = <i>127.0.0.1</i>	(StrOpt) Host to locate redis.
password = <i>None</i>	(StrOpt) Password for Redis server (optional).
port = <i>6379</i>	(IntOpt) Use this port to connect to redis host.
[matchmaker_ring]	
ringfile = <i>/etc/oslo/matchmaker_ring.json</i>	(StrOpt) Matchmaker ring file (JSON).

Table 3.47. Description of S3 configuration options

Configuration option = Default value	Description
[DEFAULT]	
buckets_path = <i>\$state_path/buckets</i>	(StrOpt) Path to S3 buckets
image_decryption_dir = <i>/tmp</i>	(StrOpt) Parent directory for tempdir used for image decryption
s3_access_key = <i>notchecked</i>	(StrOpt) Access key to use for S3 server for images
s3_affix_tenant = <i>False</i>	(BoolOpt) Whether to affix the tenant id to the access key when downloading from S3
s3_host = <i>\$my_ip</i>	(StrOpt) Hostname or IP for OpenStack to use when accessing the S3 api
s3_listen = <i>0.0.0.0</i>	(StrOpt) IP address for S3 API to listen
s3_listen_port = <i>3333</i>	(IntOpt) Port for S3 API to listen
s3_port = <i>3333</i>	(IntOpt) Port used when accessing the S3 api
s3_secret_key = <i>notchecked</i>	(StrOpt) Secret key to use for S3 server for images
s3_use_ssl = <i>False</i>	(BoolOpt) Whether to use SSL when talking to S3

Table 3.48. Description of scheduler configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
aggregate_image_properties_isolation_namespace = <i>None</i>	(StrOpt) Force the filter to consider only keys matching the given namespace.
aggregate_image_properties_isolation_separator = <i>.</i>	(StrOpt) The separator used between the namespace and keys
baremetal_scheduler_default_filters = <i>RetryFilter, AvailabilityZoneFilter, ComputeFilter, ComputeCapabilitiesFilter, ImagePropertiesFilter, ExactRamFilter, ExactDiskFilter, ExactCoreFilter</i>	(ListOpt) Which filter class names to use for filtering baremetal hosts when not specified in the request.
cpu_allocation_ratio = <i>16.0</i>	(FloatOpt) Virtual CPU to physical CPU allocation ratio which affects all CPU filters. This configuration specifies a global ratio for CoreFilter. For AggregateCoreFilter, it will fall back to this configuration value if no per-aggregate setting found.
disk_allocation_ratio = <i>1.0</i>	(FloatOpt) Virtual disk to physical disk allocation ratio
io_ops_weight_multiplier = <i>-1.0</i>	(FloatOpt) Multiplier used for weighing host io ops. Negative numbers mean a preference to choose light workload compute hosts.
isolated_hosts =	(ListOpt) Host reserved for specific images
isolated_images =	(ListOpt) Images to run on isolated host
max_instances_per_host = <i>50</i>	(IntOpt) Ignore hosts that have too many instances
max_io_ops_per_host = <i>8</i>	(IntOpt) Tells filters to ignore hosts that have this many or more instances currently in build, resize, snapshot, migrate, rescue or unshelve task states
ram_allocation_ratio = <i>1.5</i>	(FloatOpt) Virtual ram to physical ram allocation ratio which affects all ram filters. This configuration specifies a global ratio for RamFilter. For AggregateRamFilter, it will fall back to this configuration value if no per-aggregate setting found.
ram_weight_multiplier = <i>1.0</i>	(FloatOpt) Multiplier used for weighing ram. Negative numbers mean to stack vs spread.
reserved_host_disk_mb = <i>0</i>	(IntOpt) Amount of disk in MB to reserve for the host

Configuration option = Default value	Description
reserved_host_memory_mb = 512	(IntOpt) Amount of memory in MB to reserve for the host
restrict_isolated_hosts_to_isolated_images = <i>True</i>	(BoolOpt) Whether to force isolated hosts to run only isolated images
scheduler_available_filters = <i>['nova.scheduler.filters.all_filters']</i>	(MultiStrOpt) Filter classes available to the scheduler which may be specified more than once. An entry of "nova.scheduler.filters.all_filters" maps to all filters included with nova.
scheduler_default_filters = <i>RetryFilter, AvailabilityZoneFilter, RamFilter, ComputeFilter, ComputeCapabilitiesFilter, ImagePropertiesFilter, ServerGroupAntiAffinityFilter, ServerGroupAffinityFilter</i>	(ListOpt) Which filter class names to use for filtering hosts when not specified in the request.
scheduler_driver = <i>nova.scheduler.filter_scheduler.FilterScheduler</i>	(StrOpt) Default driver to use for the scheduler
scheduler_driver_task_period = 60	(IntOpt) How often (in seconds) to run periodic tasks in the scheduler driver of your choice. Note this is likely to interact with the value of <code>service_down_time</code> , but exactly how they interact will depend on your choice of scheduler driver.
scheduler_host_manager = <i>nova.scheduler.host_manager.HostManager</i>	(StrOpt) The scheduler host manager class to use
scheduler_host_subset_size = 1	(IntOpt) New instances will be scheduled on a host chosen randomly from a subset of the N best hosts. This property defines the subset size that a host is chosen from. A value of 1 chooses the first host returned by the weighing functions. This value must be at least 1. Any value less than 1 will be ignored, and 1 will be used instead
scheduler_instance_sync_interval = 120	(IntOpt) Waiting time interval (seconds) between sending the scheduler a list of current instance UUIDs to verify that its view of instances is in sync with nova. If the CONF option <code>`scheduler_tracks_instance_changes`</code> is False, changing this option will have no effect.
scheduler_json_config_location =	(StrOpt) Absolute path to scheduler configuration JSON file.
scheduler_manager = <i>nova.scheduler.manager.SchedulerManager</i>	(StrOpt) Full class name for the Manager for scheduler

Configuration option = Default value	Description
scheduler_max_attempts = 3	(IntOpt) Maximum number of attempts to schedule an instance
scheduler_topic = <i>scheduler</i>	(StrOpt) The topic scheduler nodes listen on
scheduler_tracks_instance_changes = <i>True</i>	(BoolOpt) Determines if the Scheduler tracks changes to instances to help with its filtering decisions.
scheduler_use_baremetal_filters = <i>False</i>	(BoolOpt) Flag to decide whether to use <code>baremetal_scheduler_default_filters</code> or not.
scheduler_weight_classes = <i>nova.scheduler.weights.all_weighters</i>	(ListOpt) Which weight class names to use for weighing hosts
[cells]	
ram_weight_multiplier = 10.0	(FloatOpt) Multiplier used for weighing ram. Negative numbers mean to stack vs spread.
scheduler_filter_classes = <i>nova.cells.filters.all_filters</i>	(ListOpt) Filter classes the cells scheduler should use. An entry of "nova.cells.filters.all_filters" maps to all cells filters included with nova.
scheduler_retries = 10	(IntOpt) How many retries when no cells are available.
scheduler_retry_delay = 2	(IntOpt) How often to retry in seconds when no cells are available.
scheduler_weight_classes = <i>nova.cells.weights.all_weighters</i>	(ListOpt) Weigher classes the cells scheduler should use. An entry of "nova.cells.weights.all_weighters" maps to all cell weighters included with nova.
[metrics]	
required = <i>True</i>	(BoolOpt) How to treat the unavailable metrics. When a metric is NOT available for a host, if it is set to be True, it would raise an exception, so it is recommended to use the scheduler filter <code>MetricFilter</code> to filter out those hosts. If it is set to be False, the unavailable metric would be treated as a negative factor in weighing process, the returned value would be set by the option <code>weight_of_unavailable</code> .
weight_multiplier = 1.0	(FloatOpt) Multiplier used for weighing metrics.

Configuration option = Default value	Description
weight_of_unavailable = -10000.0	(FloatOpt) The final weight value to be returned if required is set to False and any one of the metrics set by weight_setting is unavailable.
weight_setting =	(ListOpt) How the metrics are going to be weighed. This should be in the form of "<name1>=<ratio1>, <name2>=<ratio2>, ...", where <nameX> is one of the metrics to be weighed, and <ratioX> is the corresponding ratio. So for "name1=1.0, name2=-1.0" The final weight would be name1.value * 1.0 + name2.value * -1.0.

Table 3.49. Description of serial console configuration options

Configuration option = Default value	Description
[serial_console]	
base_url = ws://127.0.0.1:6083/	(StrOpt) Location of serial console proxy.
enabled = False	(BoolOpt) Enable serial console related features
listen = 127.0.0.1	(StrOpt) IP address on which instance serial console should listen
port_range = 10000:20000	(StrOpt) Range of TCP ports to use for serial ports on compute hosts
proxycient_address = 127.0.0.1	(StrOpt) The address to which proxy clients (like nova-serialproxy) should connect
serialproxy_host = 0.0.0.0	(StrOpt) Host on which to listen for incoming requests
serialproxy_port = 6083	(IntOpt) Port on which to listen for incoming requests

Table 3.50. Description of SPICE configuration options

Configuration option = Default value	Description
[spice]	
agent_enabled = True	(BoolOpt) Enable spice guest agent support
enabled = False	(BoolOpt) Enable spice related features

Configuration option = Default value	Description
html5proxy_base_url = <i>http://127.0.0.1:6082/spice_auto.html</i>	(StrOpt) Location of spice HTML5 console proxy, in the form "http://127.0.0.1:6082/spice_auto.html"
html5proxy_host = <i>0.0.0.0</i>	(StrOpt) Host on which to listen for incoming requests
html5proxy_port = <i>6082</i>	(IntOpt) Port on which to listen for incoming requests
keymap = <i>en-us</i>	(StrOpt) Keymap for spice
server_listen = <i>127.0.0.1</i>	(StrOpt) IP address on which instance spice server should listen
server_proxycient_address = <i>127.0.0.1</i>	(StrOpt) The address to which proxy clients (like nova-spicehtml5proxy) should connect

Table 3.51. Description of testing configuration options

Configuration option = Default value	Description
[DEFAULT]	
fake_call = <i>False</i>	(BoolOpt) If True, skip using the queue and make local calls
fake_network = <i>False</i>	(BoolOpt) If passed, use fake network devices and addresses
monkey_patch = <i>False</i>	(BoolOpt) Whether to log monkey patching
monkey_patch_modules = <i>nova.api.ec2.cloud:nova.notifications.notify_decorator,</i> <i>nova.compute.api:nova.notifications.notify_decorator</i>	(ListOpt) List of modules/decorators to monkey patch

Table 3.52. Description of trusted computing configuration options

Configuration option = Default value	Description
[trusted_computing]	
attestation_api_url = <i>/OpenAttestationWebServices/V1.0</i>	(StrOpt) Attestation web API URL
attestation_auth_blob = <i>None</i>	(StrOpt) Attestation authorization blob - must change

Configuration option = Default value	Description
attestation_auth_timeout = 60	(IntOpt) Attestation status cache valid period length
attestation_insecure_ssl = False	(BoolOpt) Disable SSL cert verification for Attestation service
attestation_port = 8443	(StrOpt) Attestation server port
attestation_server = None	(StrOpt) Attestation server HTTP
attestation_server_ca_file = None	(StrOpt) Attestation server Cert file for Identity verification

Table 3.53. Description of upgrade levels configuration options

Configuration option = Default value	Description
[cells]	
scheduler = <i>nova.cells.scheduler.CellsScheduler</i>	(StrOpt) Cells scheduler to use
[upgrade_levels]	
cells = None	(StrOpt) Set a version cap for messages sent to local cells services
cert = None	(StrOpt) Set a version cap for messages sent to cert services
compute = None	(StrOpt) Set a version cap for messages sent to compute services. If you plan to do a live upgrade from havana to icehouse, you should set this option to "icehouse-compat" before beginning the live upgrade procedure.
conductor = None	(StrOpt) Set a version cap for messages sent to conductor services
console = None	(StrOpt) Set a version cap for messages sent to console services
consoleauth = None	(StrOpt) Set a version cap for messages sent to consoleauth services
intercell = None	(StrOpt) Set a version cap for messages sent between cells services

Configuration option = Default value	Description
network = <i>None</i>	(StrOpt) Set a version cap for messages sent to network services
scheduler = <i>None</i>	(StrOpt) Set a version cap for messages sent to scheduler services

Table 3.54. Description of VMware configuration options

Configuration option = Default value	Description
[vmware]	
api_retry_count = 10	(IntOpt) The number of times to retry on failures, e.g., socket error, etc.
cache_prefix = <i>None</i>	(StrOpt) The prefix for Where cached images are stored. This is NOT the full path - only a folder prefix. This should only be used when a datastore cache should be shared between compute nodes. Note: this should only be used when the compute nodes have a shared file system.
cluster_name = <i>None</i>	(MultiStrOpt) Name of a VMware Cluster ComputeResource.
datastore_regex = <i>None</i>	(StrOpt) Regex to match the name of a datastore.
host_ip = <i>None</i>	(StrOpt) Hostname or IP address for connection to VMware VC host.
host_password = <i>None</i>	(StrOpt) Password for connection to VMware VC host.
host_port = 443	(IntOpt) Port for connection to VMware VC host.
host_username = <i>None</i>	(StrOpt) Username for connection to VMware VC host.
integration_bridge = <i>br-int</i>	(StrOpt) Name of Integration Bridge

Configuration option = Default value	Description
maximum_objects = <i>100</i>	(IntOpt) The maximum number of ObjectContent data objects that should be returned in a single result. A positive value will cause the operation to suspend the retrieval when the count of objects reaches the specified maximum. The server may still limit the count to something less than the configured value. Any remaining objects may be retrieved with additional requests.
pbm_default_policy = <i>None</i>	(StrOpt) The PBM default policy. If pbm_wsd1_location is set and there is no defined storage policy for the specific request then this policy will be used.
pbm_enabled = <i>False</i>	(BoolOpt) The PBM status.
pbm_wsd1_location = <i>None</i>	(StrOpt) PBM service WSDL file location URL. e.g. file:///opt/SDK/spbm/wsd1/pbmService.wsd1 Not setting this will disable storage policy based placement of instances.
task_poll_interval = <i>0.5</i>	(FloatOpt) The interval used for polling of remote tasks.
use_linked_clone = <i>True</i>	(BoolOpt) Whether to use linked clone
wsd1_location = <i>None</i>	(StrOpt) Optional VIM Service WSDL Location e.g http://<server>/vimService.wsd1. Optional override to default location for bug work-arounds

Table 3.55. Description of VNC configuration options

Configuration option = Default value	Description
[DEFAULT]	
daemon = <i>False</i>	(BoolOpt) Become a daemon (background process)
key = <i>None</i>	(StrOpt) SSL key file (if separate from cert)
novncproxy_base_url = <i>http://127.0.0.1:6080/vnc_auto.html</i>	(StrOpt) Location of VNC console proxy, in the form "http://127.0.0.1:6080/vnc_auto.html"
novncproxy_host = <i>0.0.0.0</i>	(StrOpt) Host on which to listen for incoming requests

Configuration option = Default value	Description
novncproxy_port = 6080	(IntOpt) Port on which to listen for incoming requests
record = <i>False</i>	(BoolOpt) Record sessions to FILE. [session_number]
source_is_ipv6 = <i>False</i>	(BoolOpt) Source is ipv6
ssl_only = <i>False</i>	(BoolOpt) Disallow non-encrypted connections
vnc_enabled = <i>True</i>	(BoolOpt) Enable VNC related features
vnc_keymap = <i>en-us</i>	(StrOpt) Keymap for VNC
vncserver_listen = 127.0.0.1	(StrOpt) IP address on which instance vncservers should listen
vncserver_proxyclient_address = 127.0.0.1	(StrOpt) The address to which proxy clients (like nova-xvpvncproxy) should connect
web = <i>/usr/share/spice-html5</i>	(StrOpt) Run webserver on same port. Serve files from DIR.
[vmware]	
vnc_port = 5900	(IntOpt) VNC starting port
vnc_port_total = 10000	(IntOpt) Total number of VNC ports

Table 3.56. Description of volumes configuration options

Configuration option = Default value	Description
[DEFAULT]	
block_device_allocate_retries = 60	(IntOpt) Number of times to retry block device allocation on failures
block_device_allocate_retries_interval = 3	(IntOpt) Waiting time interval (seconds) between block device allocation retries on failures
my_block_storage_ip = <i>\$my_ip</i>	(StrOpt) Block storage IP address of this host
volume_api_class = <i>nova.volume.cinder.API</i>	(StrOpt) The full class name of the volume API class to use

Configuration option = Default value	Description
volume_usage_poll_interval = 0	(IntOpt) Interval in seconds for gathering volume usages
[cinder]	
cafile = <i>None</i>	(StrOpt) PEM encoded Certificate Authority to use when verifying HTTPs connections.
catalog_info = <i>volumev2:cinderv2:publicURL</i>	(StrOpt) Info to match when looking for cinder in the service catalog. Format is: separated values of the form: <service_type>:<service_name>:<endpoint_type>
certfile = <i>None</i>	(StrOpt) PEM encoded client certificate cert file
cross_az_attach = <i>True</i>	(BoolOpt) Allow attach between instance and volume in different availability zones.
endpoint_template = <i>None</i>	(StrOpt) Override service catalog lookup with template for cinder endpoint e.g. http://localhost:8776/v1/%(project_id)s
http_retries = 3	(IntOpt) Number of cinderclient retries on failed http calls
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) PEM encoded client certificate key file
os_region_name = <i>None</i>	(StrOpt) Region name of this node
timeout = <i>None</i>	(IntOpt) Timeout value for http requests
[hyperv]	
force_volumeutils_v1 = <i>False</i>	(BoolOpt) Force V1 volume utility class
volume_attach_retry_count = 10	(IntOpt) The number of times to retry to attach a volume
volume_attach_retry_interval = 5	(IntOpt) Interval between volume attachment attempts, in seconds
[libvirt]	

Configuration option = Default value	Description
glusterfs_mount_point_base = <i>\$state_path/mnt</i>	(StrOpt) Directory where the glusterfs volume is mounted on the compute node
nfs_mount_options = <i>None</i>	(StrOpt) Mount options passed to the NFS client. See section of the nfs man page for details
nfs_mount_point_base = <i>\$state_path/mnt</i>	(StrOpt) Directory where the NFS volume is mounted on the compute node
num_aoe_discover_tries = 3	(IntOpt) Number of times to rediscover AoE target to find volume
num_iscsi_scan_tries = 5	(IntOpt) Number of times to rescan iSCSI target to find volume
num_iser_scan_tries = 5	(IntOpt) Number of times to rescan iSER target to find volume
qemu_allowed_storage_drivers =	(ListOpt) Protocols listed here will be accessed directly from QEMU. Currently supported protocols: [gluster]
rbd_secret_uuid = <i>None</i>	(StrOpt) The libvirt UUID of the secret for the rbd_uservolumes
rbd_user = <i>None</i>	(StrOpt) The RADOS client name for accessing rbd volumes
scalality_sofs_config = <i>None</i>	(StrOpt) Path or URL to Scalality SOFS configuration file
scalality_sofs_mount_point = <i>\$state_path/scalality</i>	(StrOpt) Base dir where Scalality SOFS shall be mounted
smbfs_mount_options =	(StrOpt) Mount options passed to the SMBFS client. See mount.cifs man page for details. Note that the libvirt-qemu uid and gid must be specified.
smbfs_mount_point_base = <i>\$state_path/mnt</i>	(StrOpt) Directory where the SMBFS shares are mounted on the compute node
[xenserver]	
block_device_creation_timeout = 10	(IntOpt) Time to wait for a block device to be created

Table 3.57. Description of VPN configuration options

Configuration option = Default value	Description
[DEFAULT]	
boot_script_template = <i>\$pybasedir/nova/cloudpipe/bootscript.template</i>	(StrOpt) Template for cloudpipe instance boot script
dmz_cidr =	(ListOpt) A list of dmz ranges that should be accepted
dmz_mask = 255.255.255.0	(StrOpt) Netmask to push into openvpn config
dmz_net = 10.0.0.0	(StrOpt) Network to push into openvpn config
vpn_flavor = m1.tiny	(StrOpt) Flavor for vpn instances
vpn_image_id = 0	(StrOpt) Image ID used when starting up a cloudpipe vpn server
vpn_ip = \$my_ip	(StrOpt) Public IP for the cloudpipe VPN servers
vpn_key_suffix = -vpn	(StrOpt) Suffix to add to project name for vpn key and secgroups
vpn_start = 1000	(IntOpt) First Vpn port for private networks

Table 3.58. Description of Xen configuration options

Configuration option = Default value	Description
[DEFAULT]	
console_driver = <i>nova.console.xvp.XVPConsoleProxy</i>	(StrOpt) Driver to use for the console proxy
console_xvp_conf = /etc/xvp.conf	(StrOpt) Generated XVP conf file
console_xvp_conf_template = <i>\$pybasedir/nova/console/xvp.conf.template</i>	(StrOpt) XVP conf template
console_xvp_log = /var/log/xvp.log	(StrOpt) XVP log file
console_xvp_multiplex_port = 5900	(IntOpt) Port for XVP to multiplex VNC connections on
console_xvp_pid = /var/run/xvp.pid	(StrOpt) XVP master process pid file

Configuration option = Default value	Description
stub_compute = <i>False</i>	(BoolOpt) Stub calls to compute worker for tests
[libvirt]	
xen_hvmloader_path = <i>/usr/lib/xen/boot/hvmloader</i>	(StrOpt) Location where the Xen hvmloader is kept
[xenserver]	
agent_path = <i>usr/sbin/xen-update-networking</i>	(StrOpt) Specifies the path in which the XenAPI guest agent should be located. If the agent is present, network configuration is not injected into the image. Used if <code>compute_driver=xenapi.XenAPIDriver</code> and <code>flat_injected=True</code>
agent_resetnetwork_timeout = 60	(IntOpt) Number of seconds to wait for agent reply to resetnetwork request
agent_timeout = 30	(IntOpt) Number of seconds to wait for agent reply
agent_version_timeout = 300	(IntOpt) Number of seconds to wait for agent to be fully operational
cache_images = <i>all</i>	(StrOpt) Cache glance images locally. <code>`all`</code> will cache all images, <code>`some`</code> will only cache images that have the image_property <code>`cache_in_nova=True`</code> , and <code>`none`</code> turns off caching entirely
check_host = <i>True</i>	(BoolOpt) Ensure compute service is running on host XenAPI connects to.
connection_concurrent = 5	(IntOpt) Maximum number of concurrent XenAPI connections. Used only if <code>compute_driver=xenapi.XenAPIDriver</code>
connection_password = <i>None</i>	(StrOpt) Password for connection to XenServer/Xen Cloud Platform. Used only if <code>compute_driver=xenapi.XenAPIDriver</code>
connection_url = <i>None</i>	(StrOpt) URL for connection to XenServer/Xen Cloud Platform. A special value of <code>unix://local</code> can be used to connect to the local unix socket. Required if <code>compute_driver=xenapi.XenAPIDriver</code>
connection_username = <i>root</i>	(StrOpt) Username for connection to XenServer/Xen Cloud Platform. Used only if <code>compute_driver=xenapi.XenAPIDriver</code>

Configuration option = Default value	Description
default_os_type = <i>linux</i>	(StrOpt) Default OS type
disable_agent = <i>False</i>	(BoolOpt) Disables the use of the XenAPI agent in any image regardless of what image properties are present.
image_compression_level = <i>None</i>	(IntOpt) Compression level for images, e.g., 9 for gzip -9. Range is 1-9, 9 being most compressed but most CPU intensive on dom0.
image_upload_handler = <i>nova.virt.xenapi.image.glance.GlanceStore</i>	(StrOpt) Dom0 plugin driver used to handle image uploads.
introduce_vdi_retry_wait = <i>20</i>	(IntOpt) Number of seconds to wait for an SR to settle if the VDI does not exist when first introduced
ipxe_boot_menu_url = <i>None</i>	(StrOpt) URL to the iPXE boot menu
ipxe_mkisofs_cmd = <i>mkisofs</i>	(StrOpt) Name and optionally path of the tool used for ISO image creation
ipxe_network_name = <i>None</i>	(StrOpt) Name of network to use for booting iPXE ISOs
iqn_prefix = <i>iqn.2010-10.org.openstack</i>	(StrOpt) IQN Prefix
login_timeout = <i>10</i>	(IntOpt) Timeout in seconds for XenAPI login.
max_kernel_ramdisk_size = <i>16777216</i>	(IntOpt) Maximum size in bytes of kernel or ramdisk images
num_vbd_unplug_retries = <i>10</i>	(IntOpt) Maximum number of retries to unplug VBD. if <=0, should try once and no retry
ovs_integration_bridge = <i>xapi1</i>	(StrOpt) Name of Integration Bridge used by Open vSwitch
remap_vbd_dev = <i>False</i>	(BoolOpt) Used to enable the remapping of VBD dev
remap_vbd_dev_prefix = <i>sd</i>	(StrOpt) Specify prefix to remap VBD dev to (ex. /dev/xvdb -> /dev/sdb)
running_timeout = <i>60</i>	(IntOpt) Number of seconds to wait for instance to go to running state

Configuration option = Default value	Description
sparse_copy = <i>True</i>	(BoolOpt) Whether to use sparse_copy for copying data on a resize down (False will use standard dd). This speeds up resizes down considerably since large runs of zeros will not have to be rsynced
sr_base_path = <i>/var/run/sr-mount</i>	(StrOpt) Base path to the storage repository
sr_matching_filter = <i>default-sr:true</i>	(StrOpt) Filter for finding the SR to be used to install guest instances on. To use the Local Storage in default XenServer/XCP installations set this flag to other-config:i18n-key=local-storage. To select an SR with a different matching criteria, you could set it to other-config:my_favorite_sr=true. On the other hand, to fall back on the Default SR, as displayed by XenCenter, set this flag to: default-sr:true
target_host = <i>None</i>	(StrOpt) The iSCSI Target Host
target_port = <i>3260</i>	(StrOpt) The iSCSI Target Port, default is port 3260
torrent_base_url = <i>None</i>	(StrOpt) Base URL for torrent files.
torrent_download_stall_cutoff = <i>600</i>	(IntOpt) Number of seconds a download can remain at the same progress percentage w/o being considered a stall
torrent_images = <i>none</i>	(StrOpt) Whether or not to download images via Bit Torrent (all some none).
torrent_listen_port_end = <i>6891</i>	(IntOpt) End of port range to listen on
torrent_listen_port_start = <i>6881</i>	(IntOpt) Beginning of port range to listen on
torrent_max_last_accessed = <i>86400</i>	(IntOpt) Cached torrent files not accessed within this number of seconds can be reaped
torrent_max_seeder_processes_per_host = <i>1</i>	(IntOpt) Maximum number of seeder processes to run concurrently within a given dom0. (-1 = no limit)
torrent_seed_chance = <i>1.0</i>	(FloatOpt) Probability that peer will become a seeder. (1.0 = 100%)
torrent_seed_duration = <i>3600</i>	(IntOpt) Number of seconds after downloading an image via BitTorrent that it should be seeded for other peers.

Configuration option = Default value	Description
use_agent_default = <i>False</i>	(BoolOpt) Determines if the XenAPI agent should be used when the image used does not contain a hint to declare if the agent is present or not. The hint is a glance property "xenapi_use_agent" that has the value "True" or "False". Note that waiting for the agent when it is not present will significantly increase server boot times.
use_join_force = <i>True</i>	(BoolOpt) To use for hosts with different CPUs
vhd_coalesce_max_attempts = <i>20</i>	(IntOpt) Max number of times to poll for VHD to coalesce. Used only if compute_driver=xenapi.XenAPIDriver
vhd_coalesce_poll_interval = <i>5.0</i>	(FloatOpt) The interval used for polling of coalescing vhd's. Used only if compute_driver=xenapi.XenAPIDriver
vif_driver = <i>nova.virt.xenapi.vif.XenAPIBridgeDriver</i>	(StrOpt) The XenAPI VIF driver using XenServer Network APIs.

Table 3.59. Description of XCP VNC proxy configuration options

Configuration option = Default value	Description
[DEFAULT]	
xvpvncproxy_base_url = <i>http://127.0.0.1:6081/console</i>	(StrOpt) Location of nova xvp VNC console proxy, in the form "http://127.0.0.1:6081/console"
xvpvncproxy_host = <i>0.0.0.0</i>	(StrOpt) Address that the XCP VNC proxy should bind to
xvpvncproxy_port = <i>6081</i>	(IntOpt) Port that the XCP VNC proxy should bind to

Table 3.60. Description of Zookeeper configuration options

Configuration option = Default value	Description
[zookeeper]	
address = <i>None</i>	(StrOpt) The ZooKeeper addresses for servicegroup service in the format of host1:port,host2:port,host3:port

Configuration option = Default value	Description
<code>recv_timeout = 4000</code>	(IntOpt) The <code>recv_timeout</code> parameter for the zk session
<code>sg_prefix = /servicegroups</code>	(StrOpt) The prefix used in ZooKeeper to store ephemeral nodes
<code>sg_retry_interval = 5</code>	(IntOpt) Number of seconds to wait until retrying to join the session

3.17.2. Additional sample configuration files

Files in this section can be found in `/etc/nova`.

3.17.2.1. `api-paste.ini`

The Compute service stores its API configuration settings in the `api-paste.ini` file.

```

#####
# Metadata #
#####
[composite:metadata]
use = egg:Paste#urlmap
/: meta

[pipeline:meta]
pipeline = ec2faultwrap logrequest metaapp

[app:metaapp]
paste.app_factory =
nova.api.metadata.handler:MetadataRequestHandler.factory

#####
# EC2 #
#####

[composite:ec2]
use = egg:Paste#urlmap
/: ec2cloud

[composite:ec2cloud]
use = call:nova.api.auth:pipeline_factory
noauth = ec2faultwrap logrequest ec2noauth cloudrequest validator
ec2executor
noauth2 = ec2faultwrap logrequest ec2noauth cloudrequest validator
ec2executor
keystone = ec2faultwrap logrequest ec2keystoneauth cloudrequest validator
ec2executor

[filter:ec2faultwrap]
paste.filter_factory = nova.api.ec2:FaultWrapper.factory

```

```

[filter:logrequest]
paste.filter_factory = nova.api.ec2:RequestLogging.factory

[filter:ec2lockout]
paste.filter_factory = nova.api.ec2:Lockout.factory

[filter:ec2keystoneauth]
paste.filter_factory = nova.api.ec2:EC2KeystoneAuth.factory

[filter:ec2noauth]
paste.filter_factory = nova.api.ec2:NoAuth.factory

[filter:cloudrequest]
controller = nova.api.ec2.cloud.CloudController
paste.filter_factory = nova.api.ec2:Requestify.factory

[filter:authorizer]
paste.filter_factory = nova.api.ec2:Authorizer.factory

[filter:validator]
paste.filter_factory = nova.api.ec2:Validator.factory

[app:ec2executor]
paste.app_factory = nova.api.ec2:Executor.factory

#####
# OpenStack #
#####

[composite:osapi_compute]
use = call:nova.api.openstack.urlmap:urlmap_factory
/: oscomputeversions
/v1.1: openstack_compute_api_v2
/v2: openstack_compute_api_v2
/v2.1: openstack_compute_api_v21
/v3: openstack_compute_api_v3

[composite:openstack_compute_api_v2]
use = call:nova.api.auth:pipeline_factory
noauth = compute_req_id faultwrap sizelimit noauth ratelimit
osapi_compute_app_v2
noauth2 = compute_req_id faultwrap sizelimit noauth2 ratelimit
osapi_compute_app_v2
keystone = compute_req_id faultwrap sizelimit authtoken keystonecontext
ratelimit osapi_compute_app_v2
keystone_nolimit = compute_req_id faultwrap sizelimit authtoken
keystonecontext osapi_compute_app_v2

[composite:openstack_compute_api_v21]
use = call:nova.api.auth:pipeline_factory_v21
noauth = compute_req_id faultwrap sizelimit noauth osapi_compute_app_v21
noauth2 = compute_req_id faultwrap sizelimit noauth2 osapi_compute_app_v21
keystone = compute_req_id faultwrap sizelimit authtoken keystonecontext
osapi_compute_app_v21

[composite:openstack_compute_api_v3]

```

```

use = call:nova.api.auth:pipeline_factory_v21
noauth = request_id faultwrap sizelimit noauth_v3 osapi_compute_app_v3
noauth2 = request_id faultwrap sizelimit noauth_v3 osapi_compute_app_v3
keystone = request_id faultwrap sizelimit authtoken keystonecontext
osapi_compute_app_v3

[filter:request_id]
paste.filter_factory = oslo.middleware:RequestId.factory

[filter:compute_req_id]
paste.filter_factory =
nova.api.compute_req_id:ComputeReqIdMiddleware.factory

[filter:faultwrap]
paste.filter_factory = nova.api.openstack:FaultWrapper.factory

[filter:noauth]
paste.filter_factory = nova.api.openstack.auth:NoAuthMiddlewareOld.factory

[filter:noauth2]
paste.filter_factory = nova.api.openstack.auth:NoAuthMiddleware.factory

[filter:noauth_v3]
paste.filter_factory = nova.api.openstack.auth:NoAuthMiddlewareV3.factory

[filter:ratelimit]
paste.filter_factory =
nova.api.openstack.compute.limits:RateLimitingMiddleware.factory

[filter:sizelimit]
paste.filter_factory = oslo.middleware:RequestBodySizeLimiter.factory

[app:osapi_compute_app_v2]
paste.app_factory = nova.api.openstack.compute:APIRouter.factory

[app:osapi_compute_app_v21]
paste.app_factory = nova.api.openstack.compute:APIRouterV21.factory

[app:osapi_compute_app_v3]
paste.app_factory = nova.api.openstack.compute:APIRouterV3.factory

[pipeline:oscomputeversions]
pipeline = faultwrap oscomputeversionapp

[app:oscomputeversionapp]
paste.app_factory = nova.api.openstack.compute.versions:Versions.factory

#####
# Shared #
#####

[filter:keystonecontext]
paste.filter_factory = nova.api.auth:NovaKeystoneContext.factory

[filter:authtoken]
paste.filter_factory = keystonemiddleware.auth_token:filter_factory

```

3.17.2.2. policy.json

The `policy.json` file defines additional access controls that apply to the Compute service.

```
{
  "context_is_admin": "role:admin",
  "admin_or_owner": "is_admin:True or project_id:$(project_id)s",
  "default": "rule:admin_or_owner",

  "cells_scheduler_filter:TargetCellFilter": "is_admin:True",

  "compute:create": "",
  "compute:create:attach_network": "",
  "compute:create:attach_volume": "",
  "compute:create:forced_host": "is_admin:True",
  "compute:get_all": "",
  "compute:get_all_tenants": "",
  "compute:start": "rule:admin_or_owner",
  "compute:stop": "rule:admin_or_owner",
  "compute:unlock_override": "rule:admin_api",

  "compute:shelve": "",
  "compute:shelve_offload": "",
  "compute:unshelve": "",
  "compute:resize": "",
  "compute:confirm_resize": "",
  "compute:revert_resize": "",
  "compute:rebuild": "",
  "compute:reboot": "",

  "compute:volume_snapshot_create": "",
  "compute:volume_snapshot_delete": "",

  "admin_api": "is_admin:True",
  "compute_extension:accounts": "rule:admin_api",
  "compute_extension:admin_actions": "rule:admin_api",
  "compute_extension:admin_actions:pause": "rule:admin_or_owner",
  "compute_extension:admin_actions:unpause": "rule:admin_or_owner",
  "compute_extension:admin_actions:suspend": "rule:admin_or_owner",
  "compute_extension:admin_actions:resume": "rule:admin_or_owner",
  "compute_extension:admin_actions:lock": "rule:admin_or_owner",
  "compute_extension:admin_actions:unlock": "rule:admin_or_owner",
  "compute_extension:admin_actions:resetNetwork": "rule:admin_api",
  "compute_extension:admin_actions:injectNetworkInfo": "rule:admin_api",
  "compute_extension:admin_actions:createBackup": "rule:admin_or_owner",
  "compute_extension:admin_actions:migrateLive": "rule:admin_api",
  "compute_extension:admin_actions:resetState": "rule:admin_api",
  "compute_extension:admin_actions:migrate": "rule:admin_api",
  "compute_extension:aggregates": "rule:admin_api",
  "compute_extension:agents": "rule:admin_api",
  "compute_extension:attach_interfaces": "",
```

```

"compute_extension:baremetal_nodes": "rule:admin_api",
"compute_extension:cells": "rule:admin_api",
"compute_extension:cells:create": "rule:admin_api",
"compute_extension:cells:delete": "rule:admin_api",
"compute_extension:cells:update": "rule:admin_api",
"compute_extension:cells:sync_instances": "rule:admin_api",
"compute_extension:certificates": "",
"compute_extension:cloudpipe": "rule:admin_api",
"compute_extension:cloudpipe_update": "rule:admin_api",
"compute_extension:console_output": "",
"compute_extension:consoles": "",
"compute_extension:createserverext": "",
"compute_extension:deferred_delete": "",
"compute_extension:disk_config": "",
"compute_extension:evacuate": "rule:admin_api",
"compute_extension:extended_server_attributes": "rule:admin_api",
"compute_extension:extended_status": "",
"compute_extension:extended_availability_zone": "",
"compute_extension:extended_ips": "",
"compute_extension:extended_ips_mac": "",
"compute_extension:extended_vif_net": "",
"compute_extension:extended_volumes": "",
"compute_extension:fixed_ips": "rule:admin_api",
"compute_extension:flavor_access": "",
"compute_extension:flavor_access:addTenantAccess": "rule:admin_api",
"compute_extension:flavor_access:removeTenantAccess":
"rule:admin_api",
"compute_extension:flavor_disabled": "",
"compute_extension:flavor_rxtx": "",
"compute_extension:flavor_swap": "",
"compute_extension:flavorextradata": "",
"compute_extension:flavorextraspecs:index": "",
"compute_extension:flavorextraspecs:show": "",
"compute_extension:flavorextraspecs:create": "rule:admin_api",
"compute_extension:flavorextraspecs:update": "rule:admin_api",
"compute_extension:flavorextraspecs:delete": "rule:admin_api",
"compute_extension:flavormanage": "rule:admin_api",
"compute_extension:floating_ip_dns": "",
"compute_extension:floating_ip_pools": "",
"compute_extension:floating_ips": "",
"compute_extension:floating_ips_bulk": "rule:admin_api",
"compute_extension:fping": "",
"compute_extension:fping:all_tenants": "rule:admin_api",
"compute_extension:hide_server_addresses": "is_admin:False",
"compute_extension:hosts": "rule:admin_api",
"compute_extension:hypervisors": "rule:admin_api",
"compute_extension:image_size": "",
"compute_extension:instance_actions": "",
"compute_extension:instance_actions:events": "rule:admin_api",
"compute_extension:instance_usage_audit_log": "rule:admin_api",
"compute_extension:keypairs": "",
"compute_extension:keypairs:index": "",
"compute_extension:keypairs:show": "",
"compute_extension:keypairs:create": "",
"compute_extension:keypairs:delete": "",
"compute_extension:multinic": "",

```

```

"compute_extension:networks": "rule:admin_api",
"compute_extension:networks:view": "",
"compute_extension:networks_associate": "rule:admin_api",
"compute_extension:quotas:show": "",
"compute_extension:quotas:update": "rule:admin_api",
"compute_extension:quotas:delete": "rule:admin_api",
"compute_extension:quota_classes": "",
"compute_extension:rescue": "",
"compute_extension:security_group_default_rules": "rule:admin_api",
"compute_extension:security_groups": "",
"compute_extension:server_diagnostics": "rule:admin_api",
"compute_extension:server_groups": "",
"compute_extension:server_password": "",
"compute_extension:server_usage": "",
"compute_extension:services": "rule:admin_api",
"compute_extension:shelve": "",
"compute_extension:shelveOffload": "rule:admin_api",
"compute_extension:simple_tenant_usage:show": "rule:admin_or_owner",
"compute_extension:simple_tenant_usage:list": "rule:admin_api",
"compute_extension:unshelve": "",
"compute_extension:users": "rule:admin_api",
"compute_extension:virtual_interfaces": "",
"compute_extension:virtual_storage_arrays": "",
"compute_extension:volumes": "",
"compute_extension:volume_attachments:index": "",
"compute_extension:volume_attachments:show": "",
"compute_extension:volume_attachments:create": "",
"compute_extension:volume_attachments:update": "",
"compute_extension:volume_attachments:delete": "",
"compute_extension:volumetypes": "",
"compute_extension:availability_zone:list": "",
"compute_extension:availability_zone:detail": "rule:admin_api",
"compute_extension:used_limits_for_admin": "rule:admin_api",
"compute_extension:migrations:index": "rule:admin_api",
"compute_extension:os-assisted-volume-snapshots:create":
"rule:admin_api",
  "compute_extension:os-assisted-volume-snapshots:delete":
"rule:admin_api",
  "compute_extension:console_auth_tokens": "rule:admin_api",
  "compute_extension:os-server-external-events:create":
"rule:admin_api",

"network:get_all": "",
"network:get": "",
"network:create": "",
"network:delete": "",
"network:associate": "",
"network:disassociate": "",
"network:get_vifs_by_instance": "",
"network:allocate_for_instance": "",
"network:deallocate_for_instance": "",
"network:validate_networks": "",
"network:get_instance_uuids_by_ip_filter": "",
"network:get_instance_id_by_floating_address": "",
"network:setup_networks_on_host": "",
"network:get_backdoor_port": "",

```



```

"network:get_floating_ip": "",
"network:get_floating_ip_pools": "",
"network:get_floating_ip_by_address": "",
"network:get_floating_ips_by_project": "",
"network:get_floating_ips_by_fixed_address": "",
"network:allocate_floating_ip": "",
"network:associate_floating_ip": "",
"network:disassociate_floating_ip": "",
"network:release_floating_ip": "",
"network:migrate_instance_start": "",
"network:migrate_instance_finish": "",

"network:get_fixed_ip": "",
"network:get_fixed_ip_by_address": "",
"network:add_fixed_ip_to_instance": "",
"network:remove_fixed_ip_from_instance": "",
"network:add_network_to_project": "",
"network:get_instance_nw_info": "",

"network:get_dns_domains": "",
"network:add_dns_entry": "",
"network:modify_dns_entry": "",
"network:delete_dns_entry": "",
"network:get_dns_entries_by_address": "",
"network:get_dns_entries_by_name": "",
"network:create_private_dns_domain": "",
"network:create_public_dns_domain": "",
"network:delete_dns_domain": "",
"network:attach_external_network": "rule:admin_api",

"os_compute_api:servers:start": "rule:admin_or_owner",
"os_compute_api:servers:stop": "rule:admin_or_owner",
"os_compute_api:os-access-ips:discoverable": "",
"os_compute_api:os-access-ips": "",
"os_compute_api:os-admin-actions": "rule:admin_api",
"os_compute_api:os-admin-actions:discoverable": "",
"os_compute_api:os-admin-actions:reset_network": "rule:admin_api",
"os_compute_api:os-admin-actions:inject_network_info":
"rule:admin_api",
"os_compute_api:os-admin-actions:reset_state": "rule:admin_api",
"os_compute_api:os-admin-password": "",
"os_compute_api:os-admin-password:discoverable": "",
"os_compute_api:os-aggregates:discoverable": "",
"os_compute_api:os-aggregates:index": "rule:admin_api",
"os_compute_api:os-aggregates:create": "rule:admin_api",
"os_compute_api:os-aggregates:show": "rule:admin_api",
"os_compute_api:os-aggregates:update": "rule:admin_api",
"os_compute_api:os-aggregates:delete": "rule:admin_api",
"os_compute_api:os-aggregates:add_host": "rule:admin_api",
"os_compute_api:os-aggregates:remove_host": "rule:admin_api",
"os_compute_api:os-aggregates:set_metadata": "rule:admin_api",
"os_compute_api:os-agents": "rule:admin_api",
"os_compute_api:os-agents:discoverable": "",
"os_compute_api:os-attach-interfaces": "",
"os_compute_api:os-attach-interfaces:discoverable": "",

```

```

"os_compute_api:os-baremetal-nodes": "rule:admin_api",
"os_compute_api:os-baremetal-nodes:discoverable": "",
"os_compute_api:os-block-device-mapping-v1:discoverable": "",
"os_compute_api:os-cells": "rule:admin_api",
"os_compute_api:os-cells:create": "rule:admin_api",
"os_compute_api:os-cells:delete": "rule:admin_api",
"os_compute_api:os-cells:update": "rule:admin_api",
"os_compute_api:os-cells:sync_instances": "rule:admin_api",
"os_compute_api:os-cells:discoverable": "",
"os_compute_api:os-certificates:create": "",
"os_compute_api:os-certificates:show": "",
"os_compute_api:os-certificates:discoverable": "",
"os_compute_api:os-cloudpipe": "rule:admin_api",
"os_compute_api:os-cloudpipe:discoverable": "",
"os_compute_api:os-consoles:discoverable": "",
"os_compute_api:os-consoles:create": "",
"os_compute_api:os-consoles:delete": "",
"os_compute_api:os-consoles:index": "",
"os_compute_api:os-consoles:show": "",
"os_compute_api:os-console-output:discoverable": "",
"os_compute_api:os-console-output": "",
"os_compute_api:os-remote-consoles": "",
"os_compute_api:os-remote-consoles:discoverable": "",
"os_compute_api:os-create-backup:discoverable": "",
"os_compute_api:os-create-backup": "rule:admin_or_owner",
"os_compute_api:os-deferred-delete": "",
"os_compute_api:os-deferred-delete:discoverable": "",
"os_compute_api:os-disk-config": "",
"os_compute_api:os-disk-config:discoverable": "",
"os_compute_api:os-evacuate": "rule:admin_api",
"os_compute_api:os-evacuate:discoverable": "",
"os_compute_api:os-extended-server-attributes": "rule:admin_api",
"os_compute_api:os-extended-server-attributes:discoverable": "",
"os_compute_api:os-extended-status": "",
"os_compute_api:os-extended-status:discoverable": "",
"os_compute_api:os-extended-availability-zone": "",
"os_compute_api:os-extended-availability-zone:discoverable": "",
"os_compute_api:extension_info:discoverable": "",
"os_compute_api:os-extended-volumes": "",
"os_compute_api:os-extended-volumes:discoverable": "",
"os_compute_api:os-fixed-ips": "rule:admin_api",
"os_compute_api:os-fixed-ips:discoverable": "",
"os_compute_api:os-flavor-access": "",
"os_compute_api:os-flavor-access:discoverable": "",
"os_compute_api:os-flavor-access:remove_tenant_access":
"rule:admin_api",
"os_compute_api:os-flavor-access:add_tenant_access": "rule:admin_api",
"os_compute_api:os-flavor-rxtx": "",
"os_compute_api:os-flavor-rxtx:discoverable": "",
"os_compute_api:flavors:discoverable": "",
"os_compute_api:os-flavor-extra-specs:discoverable": "",
"os_compute_api:os-flavor-extra-specs:index": "",
"os_compute_api:os-flavor-extra-specs:show": "",
"os_compute_api:os-flavor-extra-specs:create": "rule:admin_api",
"os_compute_api:os-flavor-extra-specs:update": "rule:admin_api",
"os_compute_api:os-flavor-extra-specs:delete": "rule:admin_api",

```

```

"os_compute_api:os-flavor-manage:discoverable": "",
"os_compute_api:os-flavor-manage": "rule:admin_api",
"os_compute_api:os-floating-ip-dns": "",
"os_compute_api:os-floating-ip-dns:discoverable": "",
"os_compute_api:os-floating-ip-pools": "",
"os_compute_api:os-floating-ip-pools:discoverable": "",
"os_compute_api:os-floating-ips": "",
"os_compute_api:os-floating-ips:discoverable": "",
"os_compute_api:os-floating-ips-bulk": "rule:admin_api",
"os_compute_api:os-floating-ips-bulk:discoverable": "",
"os_compute_api:os-fping": "",
"os_compute_api:os-fping:discoverable": "",
"os_compute_api:os-fping:all_tenants": "rule:admin_api",
"os_compute_api:os-hide-server-addresses": "is_admin:False",
"os_compute_api:os-hide-server-addresses:discoverable": "",
"os_compute_api:os-hosts": "rule:admin_api",
"os_compute_api:os-hosts:discoverable": "",
"os_compute_api:os-hypervisors": "rule:admin_api",
"os_compute_api:os-hypervisors:discoverable": "",
"os_compute_api:images:discoverable": "",
"os_compute_api:image-size": "",
"os_compute_api:image-size:discoverable": "",
"os_compute_api:os-instance-actions": "",
"os_compute_api:os-instance-actions:discoverable": "",
"os_compute_api:os-instance-actions:events": "rule:admin_api",
"os_compute_api:os-instance-usage-audit-log": "rule:admin_api",
"os_compute_api:os-instance-usage-audit-log:discoverable": "",
"os_compute_api:ips:discoverable": "",
"os_compute_api:ips:index": "rule:admin_or_owner",
"os_compute_api:ips:show": "rule:admin_or_owner",
"os_compute_api:os-keypairs:discoverable": "",
"os_compute_api:os-keypairs": "",
"os_compute_api:os-keypairs:index": "",
"os_compute_api:os-keypairs:show": "",
"os_compute_api:os-keypairs:create": "",
"os_compute_api:os-keypairs:delete": "",
"os_compute_api:limits:discoverable": "",
"os_compute_api:os-lock-server:discoverable": "",
"os_compute_api:os-lock-server:lock": "rule:admin_or_owner",
"os_compute_api:os-lock-server:unlock": "rule:admin_or_owner",
"os_compute_api:os-migrate-server:discoverable": "",
"os_compute_api:os-migrate-server:migrate": "rule:admin_api",
"os_compute_api:os-migrate-server:migrate_live": "rule:admin_api",
"os_compute_api:os-multinic": "",
"os_compute_api:os-multinic:discoverable": "",
"os_compute_api:os-networks": "rule:admin_api",
"os_compute_api:os-networks:view": "",
"os_compute_api:os-networks:discoverable": "",
"os_compute_api:os-networks-associate": "rule:admin_api",
"os_compute_api:os-networks-associate:discoverable": "",
"os_compute_api:os-pause-server:discoverable": "",
"os_compute_api:os-pause-server:pause": "rule:admin_or_owner",
"os_compute_api:os-pause-server:unpause": "rule:admin_or_owner",
"os_compute_api:os-pci:pci_servers": "",
"os_compute_api:os-pci:discoverable": "",
"os_compute_api:os-pci:index": "rule:admin_api",

```

```

"os_compute_api:os-pci:detail": "rule:admin_api",
"os_compute_api:os-pci:show": "rule:admin_api",
"os_compute_api:os-personality:discoverable": "",
"os_compute_api:os-preserve-ephemeral-rebuild:discoverable": "",
"os_compute_api:os-quota-sets:discoverable": "",
"os_compute_api:os-quota-sets:show": "",
"os_compute_api:os-quota-sets:update": "rule:admin_api",
"os_compute_api:os-quota-sets:delete": "rule:admin_api",
"os_compute_api:os-quota-sets:detail": "rule:admin_api",
"os_compute_api:os-quota-class-sets": "",
"os_compute_api:os-quota-class-sets:discoverable": "",
"os_compute_api:os-rescue": "",
"os_compute_api:os-rescue:discoverable": "",
"os_compute_api:os-scheduler-hints:discoverable": "",
"os_compute_api:os-security-group-default-rules:discoverable": "",
"os_compute_api:os-security-group-default-rules": "rule:admin_api",
"os_compute_api:os-security-groups": "",
"os_compute_api:os-security-groups:discoverable": "",
"os_compute_api:os-server-diagnostics": "rule:admin_api",
"os_compute_api:os-server-diagnostics:discoverable": "",
"os_compute_api:os-server-password": "",
"os_compute_api:os-server-password:discoverable": "",
"os_compute_api:os-server-usage": "",
"os_compute_api:os-server-usage:discoverable": "",
"os_compute_api:os-server-groups": "",
"os_compute_api:os-server-groups:discoverable": "",
"os_compute_api:os-services": "rule:admin_api",
"os_compute_api:os-services:discoverable": "",
"os_compute_api:server-metadata:discoverable": "",
"os_compute_api:server-metadata:index": "rule:admin_or_owner",
"os_compute_api:server-metadata:show": "rule:admin_or_owner",
"os_compute_api:server-metadata:delete": "rule:admin_or_owner",
"os_compute_api:server-metadata:create": "rule:admin_or_owner",
"os_compute_api:server-metadata:update": "rule:admin_or_owner",
"os_compute_api:server-metadata:update_all": "rule:admin_or_owner",
"os_compute_api:servers:discoverable": "",
"os_compute_api:os-shelve:shelve": "",
"os_compute_api:os-shelve:shelve:discoverable": "",
"os_compute_api:os-shelve:shelve_offload": "rule:admin_api",
"os_compute_api:os-simple-tenant-usage:discoverable": "",
"os_compute_api:os-simple-tenant-usage:show": "rule:admin_or_owner",
"os_compute_api:os-simple-tenant-usage:list": "rule:admin_api",
"os_compute_api:os-suspend-server:discoverable": "",
"os_compute_api:os-suspend-server:suspend": "rule:admin_or_owner",
"os_compute_api:os-suspend-server:resume": "rule:admin_or_owner",
"os_compute_api:os-tenant-networks": "rule:admin_or_owner",
"os_compute_api:os-tenant-networks:discoverable": "",
"os_compute_api:os-shelve:unshelve": "",
"os_compute_api:os-user-data:discoverable": "",
"os_compute_api:os-virtual-interfaces": "",
"os_compute_api:os-virtual-interfaces:discoverable": "",
"os_compute_api:os-volumes": "",
"os_compute_api:os-volumes:discoverable": "",
"os_compute_api:os-volumes-attachments:index": "",
"os_compute_api:os-volumes-attachments:show": "",
"os_compute_api:os-volumes-attachments:create": "",

```

```

"os_compute_api:os-volumes-attachments:update": "",
"os_compute_api:os-volumes-attachments:delete": "",
"os_compute_api:os-volumes-attachments:discoverable": "",
"os_compute_api:os-availability-zone:list": "",
"os_compute_api:os-availability-zone:discoverable": "",
"os_compute_api:os-availability-zone:detail": "rule:admin_api",
"os_compute_api:os-used-limits": "rule:admin_api",
"os_compute_api:os-used-limits:discoverable": "",
"os_compute_api:os-migrations:index": "rule:admin_api",
"os_compute_api:os-migrations:discoverable": "",
"os_compute_api:os-assisted-volume-snapshots:create":
"rule:admin_api",
  "os_compute_api:os-assisted-volume-snapshots:delete":
"rule:admin_api",
  "os_compute_api:os-assisted-volume-snapshots:discoverable": "",
  "os_compute_api:os-console-auth-tokens": "rule:admin_api",
  "os_compute_api:os-server-external-events:create": "rule:admin_api"
}

```

3.17.2.3. rootwrap.conf

The `rootwrap.conf` file defines configuration values used by the `rootwrap` script when the Compute service needs to escalate its privileges to those of the root user.

It is also possible to disable the root wrapper, and default to `sudo` only. Configure the *`disable_rootwrap`* option in the `[workaround]` section of the `nova.conf` configuration file.

```

# Configuration for nova-rootwrap
# This file should be owned by (and only-writeable by) the root user

[DEFAULT]
# List of directories to load filter definitions from (separated by ',').
# These directories MUST all be only writeable by root !
filters_path=/etc/nova/rootwrap.d,/usr/share/nova/rootwrap

# List of directories to search executables in, in case filters do not
# explicitly specify a full path (separated by ',')
# If not specified, defaults to system PATH environment variable.
# These directories MUST all be only writeable by root !
exec_dirs=/sbin,/usr/sbin,/bin,/usr/bin

# Enable logging to syslog
# Default value is False
use_syslog=False

# Which syslog facility to use.
# Valid values include auth, authpriv, syslog, local0, local1...
# Default value is 'syslog'
syslog_log_facility=syslog

# Which messages to log.
# INFO means log all usage

```

```
# ERROR means only log unsuccessful attempts
syslog_log_level=ERROR
```

3.18. NEW, UPDATED AND DEPRECATED OPTIONS IN KILO FOR OPENSTACK COMPUTE

Table 3.61. New options

Option = default value	(Type) Help string
[DEFAULT] ebtables_exec_attempts = 3	(IntOpt) Number of times to retry ebtables commands on failure.
[DEFAULT] ebtables_retry_interval = 1.0	(FloatOpt) Number of seconds to wait between ebtables retries.
[DEFAULT] io_ops_weight_multiplier = -1.0	(FloatOpt) Multiplier used for weighing host io ops. Negative numbers mean a preference to choose light workload compute hosts.
[DEFAULT] keystone_ec2_insecure = False	(BoolOpt) Disable SSL certificate verification.
[DEFAULT] log-config-append = None	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation.
[DEFAULT] log-date-format = %Y-%m-%d %H:%M:%S	(StrOpt) Format string for %(asctime)s in log records. Default: %(default)s .
[DEFAULT] log-dir = None	(StrOpt) (Optional) The base directory used for relative --log-file paths.
[DEFAULT] log-file = None	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout.
[DEFAULT] log-format = None	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecated. Use logging_context_format_string and logging_default_format_string instead.
[DEFAULT] max_concurrent_builds = 10	(IntOpt) Maximum number of instance builds to run concurrently

Option = default value	(Type) Help string
[DEFAULT] metadata_cache_expiration = 15	(IntOpt) Time in seconds to cache metadata; 0 to disable metadata caching entirely (not recommended). Increasing this should improve response times of the metadata API when under heavy load. Higher values may increase memory usage and result in longer times for host metadata changes to take effect.
[DEFAULT] my_block_storage_ip = \$my_ip	(StrOpt) Block storage IP address of this host
[DEFAULT] novncproxy_host = 0.0.0.0	(StrOpt) Host on which to listen for incoming requests
[DEFAULT] novncproxy_port = 6080	(IntOpt) Port on which to listen for incoming requests
[DEFAULT] policy_dirs = ['policy.d']	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the config_dir option, or absolute paths. The file defined by policy_file must exist for these directories to be searched. Missing or empty directories are ignored.
[DEFAULT] quota_networks = 3	(IntOpt) Number of private networks allowed per project
[DEFAULT] scheduler_instance_sync_interval = 120	(IntOpt) Waiting time interval (seconds) between sending the scheduler a list of current instance UUIDs to verify that its view of instances is in sync with nova. If the CONF option `scheduler_tracks_instance_changes` is False, changing this option will have no effect.
[DEFAULT] scheduler_tracks_instance_changes = True	(BoolOpt) Determines if the Scheduler tracks changes to instances to help with its filtering decisions.
[DEFAULT] syslog-log-facility = LOG_USER	(StrOpt) Syslog facility to receive log lines.
[DEFAULT] use-syslog = False	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED during I, and will change in J to honor RFC5424.
[DEFAULT] use-syslog-rfc-format = False	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in I, and will be removed in J.

Option = default value	(Type) Help string
[api_database] connection = None	(StrOpt) The SQLAlchemy connection string to use to connect to the Nova API database.
[api_database] connection_debug = 0	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
[api_database] connection_trace = False	(BoolOpt) Add Python stack traces to SQL as comment strings.
[api_database] idle_timeout = 3600	(IntOpt) Timeout before idle SQL connections are reaped.
[api_database] max_overflow = None	(IntOpt) If set, use this value for max_overflow with SQLAlchemy.
[api_database] max_pool_size = None	(IntOpt) Maximum number of SQL connections to keep open in a pool.
[api_database] max_retries = 10	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
[api_database] mysql_sql_mode = TRADITIONAL	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode=
[api_database] pool_timeout = None	(IntOpt) If set, use this value for pool_timeout with SQLAlchemy.
[api_database] retry_interval = 10	(IntOpt) Interval between retries of opening a SQL connection.
[api_database] slave_connection = None	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
[api_database] sqlite_synchronous = True	(BoolOpt) If True, SQLite uses synchronous mode.
[barbican] cafile = None	(StrOpt) PEM encoded Certificate Authority to use when verifying HTTPs connections.
[barbican] catalog_info = key-manager:barbican:public	(StrOpt) Info to match when looking for barbican in the service catalog. Format is: separated values of the form: <service_type>:<service_name>:<endpoint_type>
[barbican] certfile = None	(StrOpt) PEM encoded client certificate cert file

Option = default value	(Type) Help string
[barbican] endpoint_template = None	(StrOpt) Override service catalog lookup with template for barbican endpoint e.g. http://localhost:9311/v1/%(project_id)s
[barbican] insecure = False	(BoolOpt) Verify HTTPS connections.
[barbican] keyfile = None	(StrOpt) PEM encoded client certificate key file
[barbican] os_region_name = None	(StrOpt) Region name of this node
[barbican] timeout = None	(IntOpt) Timeout value for http requests
[cinder] cafile = None	(StrOpt) PEM encoded Certificate Authority to use when verifying HTTPs connections.
[cinder] certfile = None	(StrOpt) PEM encoded client certificate cert file
[cinder] insecure = False	(BoolOpt) Verify HTTPS connections.
[cinder] keyfile = None	(StrOpt) PEM encoded client certificate key file
[cinder] timeout = None	(IntOpt) Timeout value for http requests
[database] backend = sqlalchemy	(StrOpt) The back end to use for the database.
[database] connection = None	(StrOpt) The SQLAlchemy connection string to use to connect to the database.
[database] connection_debug = 0	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
[database] connection_trace = False	(BoolOpt) Add Python stack traces to SQL as comment strings.
[database] db_inc_retry_interval = True	(BoolOpt) If True, increases the interval between retries of a database operation up to db_max_retry_interval.
[database] db_max_retries = 20	(IntOpt) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
[database] db_max_retry_interval = 10	(IntOpt) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.

Option = default value	(Type) Help string
[database] db_retry_interval = 1	(IntOpt) Seconds between retries of a database transaction.
[database] idle_timeout = 3600	(IntOpt) Timeout before idle SQL connections are reaped.
[database] max_overflow = None	(IntOpt) If set, use this value for max_overflow with SQLAlchemy.
[database] max_pool_size = None	(IntOpt) Maximum number of SQL connections to keep open in a pool.
[database] max_retries = 10	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
[database] min_pool_size = 1	(IntOpt) Minimum number of SQL connections to keep open in a pool.
[database] mysql_sql_mode = TRADITIONAL	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode=
[database] pool_timeout = None	(IntOpt) If set, use this value for pool_timeout with SQLAlchemy.
[database] retry_interval = 10	(IntOpt) Interval between retries of opening a SQL connection.
[database] slave_connection = None	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
[database] sqlite_db = oslo.sqlite	(StrOpt) The file name to use with SQLite.
[database] sqlite_synchronous = True	(BoolOpt) If True, SQLite uses synchronous mode.
[database] use_db_reconnect = False	(BoolOpt) Enable the experimental use of database reconnect on connection lost.
[guestfs] debug = False	(BoolOpt) Enable guestfs debug

Option = default value	(Type) Help string
[libvirt] iscsi_iface = None	(StrOpt) The iSCSI transport iface to use to connect to target in case offload support is desired. Supported transports are be2iscsi, bnx2i, cxgb3i, cxgb4i, qla4xxx and ocs. Default format is transport_name.hwaddress and can be generated manually or via iscsiadm -m iface
[libvirt] quobyte_client_cfg = None	(StrOpt) Path to a Quobyte Client configuration file.
[libvirt] quobyte_mount_point_base = \$state_path/mnt	(StrOpt) Directory where the Quobyte volume is mounted on the compute node
[libvirt] smbfs_mount_options =	(StrOpt) Mount options passed to the SMBFS client. See mount.cifs man page for details. Note that the libvirt-qemu uid and gid must be specified.
[libvirt] smbfs_mount_point_base = \$state_path/mnt	(StrOpt) Directory where the SMBFS shares are mounted on the compute node
[neutron] auth_plugin = None	(StrOpt) Name of the plugin to load
[neutron] auth_section = None	(StrOpt) Config Section from which to load plugin specific options
[neutron] cafile = None	(StrOpt) PEM encoded Certificate Authority to use when verifying HTTPs connections.
[neutron] certfile = None	(StrOpt) PEM encoded client certificate cert file
[neutron] insecure = False	(BoolOpt) Verify HTTPS connections.
[neutron] keyfile = None	(StrOpt) PEM encoded client certificate key file
[neutron] timeout = None	(IntOpt) Timeout value for http requests
[oslo_concurrency] disable_process_locking = False	(BoolOpt) Enables or disables inter-process locks.
[oslo_concurrency] lock_path = None	(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging_amqp] allow_insecure_clients = False	(BoolOpt) Accept clients using either SSL or plain TCP

Option = default value	(Type) Help string
[oslo_messaging_amqp] broadcast_prefix = broadcast	(StrOpt) address prefix used when broadcasting to all servers
[oslo_messaging_amqp] container_name = None	(StrOpt) Name for the AMQP container
[oslo_messaging_amqp] group_request_prefix = unicast	(StrOpt) address prefix when sending to any server in group
[oslo_messaging_amqp] idle_timeout = 0	(IntOpt) Timeout for inactive connections (in seconds)
[oslo_messaging_amqp] server_request_prefix = exclusive	(StrOpt) address prefix used when sending to a specific server
[oslo_messaging_amqp] ssl_ca_file =	(StrOpt) CA certificate PEM file for verifying server certificate
[oslo_messaging_amqp] ssl_cert_file =	(StrOpt) Identifying certificate PEM file to present to clients
[oslo_messaging_amqp] ssl_key_file =	(StrOpt) Private key PEM file used to sign cert_file certificate
[oslo_messaging_amqp] ssl_key_password = None	(StrOpt) Password for decrypting ssl_key_file (if encrypted)
[oslo_messaging_amqp] trace = False	(BoolOpt) Debug: dump AMQP frames to stdout
[oslo_messaging_qpid] amqp_auto_delete = False	(BoolOpt) Auto-delete queues in AMQP.
[oslo_messaging_qpid] amqp_durable_queues = False	(BoolOpt) Use durable queues in AMQP.
[oslo_messaging_qpid] qpid_heartbeat = 60	(IntOpt) Seconds between connection keepalive heartbeats.
[oslo_messaging_qpid] qpid_hostname = localhost	(StrOpt) Qpid broker hostname.
[oslo_messaging_qpid] qpid_hosts = \$qpid_hostname:\$qpid_port	(ListOpt) Qpid HA cluster host:port pairs.
[oslo_messaging_qpid] qpid_password =	(StrOpt) Password for Qpid connection.
[oslo_messaging_qpid] qpid_port = 5672	(IntOpt) Qpid broker port.
[oslo_messaging_qpid] qpid_protocol = tcp	(StrOpt) Transport to use, either 'tcp' or 'ssl'.

Option = default value	(Type) Help string
[oslo_messaging_qpid] qpid_receiver_capacity = 1	(IntOpt) The number of prefetched messages held by receiver.
[oslo_messaging_qpid] qpid_sasl_mechanisms =	(StrOpt) Space separated list of SASL mechanisms to use for auth.
[oslo_messaging_qpid] qpid_tcp_nodelay = True	(BoolOpt) Whether to disable the Nagle algorithm.
[oslo_messaging_qpid] qpid_topology_version = 1	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
[oslo_messaging_qpid] qpid_username =	(StrOpt) Username for Qpid connection.
[oslo_messaging_qpid] rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
[oslo_messaging_rabbit] amqp_auto_delete = False	(BoolOpt) Auto-delete queues in AMQP.
[oslo_messaging_rabbit] amqp_durable_queues = False	(BoolOpt) Use durable queues in AMQP.
[oslo_messaging_rabbit] fake_rabbit = False	(BoolOpt) Deprecated, use rpc_backend=kombu+memory or rpc_backend=fake
[oslo_messaging_rabbit] heartbeat_rate = 2	(IntOpt) How often times during the heartbeat_timeout_threshold to check the heartbeat.
[oslo_messaging_rabbit] heartbeat_timeout_threshold = 0	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disables the heartbeat, >0 enables it. Enabling heartbeats requires kombu>=3.0.7 and amqp>=1.4.0). EXPERIMENTAL
[oslo_messaging_rabbit] kombu_reconnect_delay = 1.0	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
[oslo_messaging_rabbit] kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
[oslo_messaging_rabbit] kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
[oslo_messaging_rabbit] kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).

Option = default value	(Type) Help string
[oslo_messaging_rabbit] kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 are also available.
[oslo_messaging_rabbit] rabbit_ha_queues = False	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
[oslo_messaging_rabbit] rabbit_host = localhost	(StrOpt) The RabbitMQ broker address where a single node is used.
[oslo_messaging_rabbit] rabbit_hosts = \$rabbit_host:\$rabbit_port	(ListOpt) RabbitMQ HA cluster host:port pairs.
[oslo_messaging_rabbit] rabbit_login_method = AMQPLAIN	(StrOpt) The RabbitMQ login method.
[oslo_messaging_rabbit] rabbit_max_retries = 0	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
[oslo_messaging_rabbit] rabbit_password = guest	(StrOpt) The RabbitMQ password.
[oslo_messaging_rabbit] rabbit_port = 5672	(IntOpt) The RabbitMQ broker port where a single node is used.
[oslo_messaging_rabbit] rabbit_retry_backoff = 2	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
[oslo_messaging_rabbit] rabbit_retry_interval = 1	(IntOpt) How frequently to retry connecting with RabbitMQ.
[oslo_messaging_rabbit] rabbit_use_ssl = False	(BoolOpt) Connect over SSL for RabbitMQ.
[oslo_messaging_rabbit] rabbit_userid = guest	(StrOpt) The RabbitMQ userid.
[oslo_messaging_rabbit] rabbit_virtual_host = /	(StrOpt) The RabbitMQ virtual host.
[oslo_messaging_rabbit] rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
[oslo_middleware] max_request_body_size = 114688	(IntOpt) The maximum body size for each request, in bytes.
[serial_console] serialproxy_host = 0.0.0.0	(StrOpt) Host on which to listen for incoming requests
[serial_console] serialproxy_port = 6083	(IntOpt) Port on which to listen for incoming requests

Option = default value	(Type) Help string
[vmware] cache_prefix = None	(StrOpt) The prefix for Where cached images are stored. This is NOT the full path - only a folder prefix. This should only be used when a datastore cache should be shared between compute nodes. Note: this should only be used when the compute nodes have a shared file system.
[vmware] pbm_default_policy = None	(StrOpt) The PBM default policy. If pbm_wsdl_location is set and there is no defined storage policy for the specific request then this policy will be used.
[vmware] pbm_enabled = False	(BoolOpt) The PBM status.
[vmware] pbm_wsdl_location = None	(StrOpt) PBM service WSDL file location URL. e.g. file:///opt/SDK/spbm/wsdl/pbmService.wsdl Not setting this will disable storage policy based placement of instances.
[workarounds] destroy_after_evacuate = True	(BoolOpt) Whether to destroy instances on startup when it is suspected that they have previously been evacuated. This can result in data loss if undesired. See https://launchpad.net/bugs/1419785
[workarounds] disable_libvirt_livesnapshot = True	(BoolOpt) When using libvirt 1.2.2 fails live snapshots intermittently under load. This config option provides mechanism to disable livesnapshot while this is resolved. See https://bugs.launchpad.net/nova/+bug/1334398
[workarounds] disable_rootwrap = False	(BoolOpt) This option allows a fallback to sudo for performance reasons. For example see https://bugs.launchpad.net/nova/+bug/1415106

Table 3.62. New default values

Option	Previous default value	New default value
[DEFAULT] client_socket_timeout	0	900

Option	Previous default value	New default value
[DEFAULT] default_log_levels	amqp=WARN, amqplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, keystone.middleware=WARN, routes.middleware=WARN, stevedore=WARN	amqp=WARN, amqplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystone.middleware=WARN, routes.middleware=WARN, stevedore=WARN
[DEFAULT] ec2_path	/services/Cloud	/
[DEFAULT] multi_instance_display_name_template	%(name)s-%(uuid)s	%(name)s-%(count)d
[DEFAULT] rpc_zmq_matchmaker	oslo.messaging._drivers.matchmaker.MatchMakerLocalhost	local
[cinder] catalog_info	volume:cinder:publicURL	volumev2:cinderv2:publicURL

Table 3.63. Deprecated options

Deprecated option	New Option
[DEFAULT] network_device_mtu	None
[DEFAULT] log-format	None
[DEFAULT] use-syslog	None
[cinder] http_timeout	[cinder] timeout
[DEFAULT] use_syslog	None
[ironic] client_log_level	None
[neutron] admin_username	None
[DEFAULT] osapi_max_request_body_size	[oslo_middleware] max_request_body_size

Deprecated option	New Option
[neutron] ca_certificates_file	[neutron] cafile
[neutron] auth_strategy	None
[neutron] admin_user_id	None
[neutron] admin_tenant_id	None
[DEFAULT] log_format	None
[cinder] api_insecure	[cinder] insecure
[neutron] admin_tenant_name	None
[neutron] admin_password	None
[DEFAULT] share_dhcp_address	None
[neutron] api_insecure	[neutron] insecure
[cinder] ca_certificates_file	[cinder] cafile
[neutron] admin_auth_url	None
[neutron] url_timeout	[neutron] timeout
[neutron] allow_duplicate_networks	None

CHAPTER 4. DASHBOARD

This chapter describes how to configure the OpenStack dashboard with Apache web server.

4.1. CONFIGURE THE DASHBOARD

You can configure the dashboard for a simple HTTP deployment.

You can configure the dashboard for a secured HTTPS deployment. While the standard installation uses a non-encrypted HTTP channel, you can enable SSL support for the dashboard.

Also, you can configure the size of the VNC window in the dashboard.

4.1.1. Configure the dashboard for HTTP

You can configure the dashboard for a simple HTTP deployment. The standard installation uses a non-encrypted HTTP channel.

1. Specify the host for your OpenStack Identity Service endpoint in the `/etc/openstack-dashboard/local_settings` file with the `OPENSTACK_HOST` setting.

The following example shows this setting:

```
import os

from django.utils.translation import ugettext_lazy as _

DEBUG = False
TEMPLATE_DEBUG = DEBUG
PROD = True
USE_SSL = False

SITE_BRANDING = 'OpenStack Dashboard'

# WEBROOT is the location relative to Webserver root
# should end with a slash.
WEBROOT = '/dashboard/'

# Required for Django 1.5.
# If horizon is running in production (DEBUG is False), set this
# with the list of host/domain names that the application can serve.
# For more information see:
# https://docs.djangoproject.com/en/dev/ref/settings/#allowed-hosts
#ALLOWED_HOSTS = ['horizon.example.com', ]

ALLOWED_HOSTS = HOST_NAME

# Specify a regular expression to validate user passwords.
# HORIZON_CONFIG = {
#     "password_validator": {
#         "regex": '.*',
#         "help_text": _("Your password does not meet the
requirements.")
#     }
# }
```

```

LOCAL_PATH = os.path.dirname(os.path.abspath(__file__))

CACHES = {
    'default': {
        'BACKEND' : 'django.core.cache.backends.memcached.MemcachedCache',
        'LOCATION' : '127.0.0.1:11211'
        'SESSION_ENGINE' =
    'django.contrib.sessions.backends.cache'
    }
}

# Send email to the console by default
EMAIL_BACKEND = 'django.core.mail.backends.console.EmailBackend'
# Or send them to /dev/null
#EMAIL_BACKEND = 'django.core.mail.backends.dummy.EmailBackend'

# Configure these for your outgoing email host
# EMAIL_HOST = 'smtp.my-company.com'
# EMAIL_PORT = 25
# EMAIL_HOST_USER = 'djangomail'
# EMAIL_HOST_PASSWORD = 'top-secret!'

# For multiple regions uncomment this configuration, and add
(endpoint, title).
# AVAILABLE_REGIONS = [
#     ('http://cluster1.example.com:5000/v2.0', 'cluster1'),
#     ('http://cluster2.example.com:5000/v2.0', 'cluster2'),
# ]

OPENSTACK_HOST = "127.0.0.1"
OPENSTACK_KEYSTONE_URL = "http://%s:5000/v2.0" % OPENSTACK_HOST
OPENSTACK_KEYSTONE_DEFAULT_ROLE = "Member"

# The OPENSTACK_KEYSTONE_BACKEND settings can be used to identify
the
# capabilities of the auth backend for Keystone.
# If Keystone has been configured to use LDAP as the auth backend
then set
# can_edit_user to False and name to 'ldap'.
#
# TODO(tres): Remove these once Keystone has an API to identify auth
backend.
OPENSTACK_KEYSTONE_BACKEND = {
    'name': 'native',
    'can_edit_user': True
}

# OPENSTACK_ENDPOINT_TYPE specifies the endpoint type to use for the
endpoints
# in the Keystone service catalog. Use this setting when Horizon is
running
# external to the OpenStack environment. The default is
'internalURL'.
#OPENSTACK_ENDPOINT_TYPE = "publicURL"

```

```

# The number of Swift containers and objects to display on a single
page before
# providing a paging element (a "more" link) to paginate results.
API_RESULT_LIMIT = 1000

# If you have external monitoring links, eg:
# EXTERNAL_MONITORING = [
#     ['Nagios', 'http://foo.com'],
#     ['Ganglia', 'http://bar.com'],
# ]

LOGGING = {
    'version': 1,
    # When set to True this will disable all logging except
    # for loggers specified in this configuration dictionary.
    Note that
        # if nothing is specified here and disable_existing_loggers
is True,
        # django.db.backends will still log unless it is disabled
explicitly.
    'disable_existing_loggers': False,
    'handlers': {
        'null': {
            'level': 'DEBUG',
            'class': 'django.utils.log.NullHandler',
        },
        'console': {
            # Set the level to "DEBUG" for verbose output
logging.
            'level': 'INFO',
            'class': 'logging.StreamHandler',
        },
    },
    'loggers': {
        # Logging from django.db.backends is VERY verbose, send
to null
        # by default.
        'django.db.backends': {
            'handlers': ['null'],
            'propagate': False,
        },
        'horizon': {
            'handlers': ['console'],
            'propagate': False,
        },
        'novaclient': {
            'handlers': ['console'],
            'propagate': False,
        },
        'keystoneclient': {
            'handlers': ['console'],
            'propagate': False,
        },
        'nose.plugins.manager': {
            'handlers': ['console'],
            'propagate': False,

```

```
}
    }
}
```

The service catalog configuration in the Identity Service determines whether a service appears in the dashboard..

2. Restart Apache http server.

```
# systemctl restart httpd
```

Next, restart memcached:

```
# systemctl restart memcached
```

4.1.2. Configure the dashboard for HTTPS

You can configure the dashboard for a secured HTTPS deployment. While the standard installation uses a non-encrypted HTTP channel, you can enable SSL support for the dashboard.

This example uses the `http://openstack.example.com` domain. Use a domain that fits your current setup.

1. In the `/etc/openstack-dashboard/local_settings` file, update the following options:

```
USE_SSL = True
CSRF_COOKIE_SECURE = True
SESSION_COOKIE_SECURE = True
SESSION_COOKIE_HTTPONLY = True
```

To enable HTTPS, the `USE_SSL = True` option is required.

The other options require that HTTPS is enabled; these options defend against cross-site scripting.

2. Edit the `/etc/httpd/conf.d/openstack-dashboard.conf` file as shown in [Example 4.2](#), “After”:

Example 4.1. Before

```
WSGIScriptAlias / /usr/share/openstack-
dashboard/openstack_dashboard/wsgi/django.wsgi
WSGIDaemonProcess horizon user=apache group=apache processes=3
threads=10
Alias /static /usr/share/openstack-
dashboard/openstack_dashboard/static/
<Directory /usr/share/openstack-
dashboard/openstack_dashboard/wsgi>
# For Apache http server 2.2 and earlier:
Order allow,deny
Allow from all
```

```
# For Apache http server 2.4 and later:
# Require all granted
</Directory>
```

Example 4.2. After

```
<VirtualHost *:80>
ServerName openstack.example.com
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
</IfModule>
<IfModule !mod_rewrite.c>
RedirectPermanent / https://openstack.example.com
</IfModule>
</VirtualHost>
<VirtualHost *:443>
ServerName openstack.example.com

SSLEngine On
# Remember to replace certificates and keys with valid paths in
your environment
SSLCertificateFile /etc/httpd/SSL/openstack.example.com.crt
SSLCACertificateFile /etc/httpd/SSL/openstack.example.com.crt
SSLCertificateKeyFile /etc/httpd/SSL/openstack.example.com.key
SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown

# HTTP Strict Transport Security (HSTS) enforces that all
communications
# with a server go over SSL. This mitigates the threat from
attacks such
# as SSL-Strip which replaces links on the wire, stripping away
https prefixes
# and potentially allowing an attacker to view confidential
information on the
# wire
Header add Strict-Transport-Security "max-age=15768000"

WSGIScriptAlias / /usr/share/openstack-
dashboard/openstack_dashboard/wsgi/django.wsgi
WSGIDaemonProcess horizon user=apache group=apache processes=3
threads=10
Alias /static /usr/share/openstack-
dashboard/openstack_dashboard/static/
<Directory /usr/share/openstack-
dashboard/openstack_dashboard/wsgi>
# For Apache http server 2.2 and earlier:
Order allow,deny
Allow from all

# For Apache http server 2.4 and later:
```

```
# Require all granted
</Directory>
</VirtualHost>
```

In this configuration, the Apache HTTP server listens on port 443 and redirects all non-secure requests to the HTTPS protocol. The secured section defines the private key, public key, and certificate to use.

3. Restart the Apache HTTP server.

```
# systemctl restart httpd
```

4. Restart memcached:

```
# systemctl restart memcached
```

If you try to access the dashboard through HTTP, the browser redirects you to the HTTPS page.

NOTE

Configuring the dashboard for HTTPS also requires enabling SSL for the **noVNC** proxy service. On the controller node, add the following additional options to the **[DEFAULT]** section of the `/etc/nova/nova.conf` file:

```
[DEFAULT]
...
ssl_only = true
cert = /etc/apache2/SSL/openstack.example.com.crt
key = /etc/apache2/SSL/openstack.example.com.key
```

On the compute nodes, ensure the **novncproxy_base_url** option points to a URL with an HTTPS scheme:

```
[DEFAULT]
...
novncproxy_base_url =
https://controller:6080/vnc_auto.html
```

4.2. ADDITIONAL SAMPLE CONFIGURATION FILES

Find the following files in `/etc/openstack-dashboard`.

4.2.1. keystone_policy.json

The `keystone_policy.json` file defines additional access controls for the dashboard that apply to the Identity service.

**NOTE**

The `keystone_policy.json` file must match the Identity service `/etc/keystone/policy.json` policy file.

```
{
  "admin_required": [
    [
      "role:admin"
    ],
    [
      "is_admin:1"
    ]
  ],
  "service_role": [
    [
      "role:service"
    ]
  ],
  "service_or_admin": [
    [
      "rule:admin_required"
    ],
    [
      "rule:service_role"
    ]
  ],
  "owner": [
    [
      "user_id:%(user_id)s"
    ]
  ],
  "admin_or_owner": [
    [
      "rule:admin_required"
    ],
    [
      "rule:owner"
    ]
  ],
  "default": [
    [
      "rule:admin_required"
    ]
  ],
  "identity:get_service": [
    [
      "rule:admin_required"
    ]
  ],
  "identity:list_services": [
    [
      "rule:admin_required"
    ]
  ],
}
```



```

"identity:create_service": [
  [
    "rule:admin_required"
  ]
],
"identity:update_service": [
  [
    "rule:admin_required"
  ]
],
"identity:delete_service": [
  [
    "rule:admin_required"
  ]
],
"identity:get_endpoint": [
  [
    "rule:admin_required"
  ]
],
"identity:list_endpoints": [
  [
    "rule:admin_required"
  ]
],
"identity:create_endpoint": [
  [
    "rule:admin_required"
  ]
],
"identity:update_endpoint": [
  [
    "rule:admin_required"
  ]
],
"identity:delete_endpoint": [
  [
    "rule:admin_required"
  ]
],
"identity:get_domain": [
  [
    "rule:admin_required"
  ]
],
"identity:list_domains": [
  [
    "rule:admin_required"
  ]
],
"identity:create_domain": [
  [
    "rule:admin_required"
  ]
],
"identity:update_domain": [

```

```
[
    "rule:admin_required"
],
"identity:delete_domain": [
    "rule:admin_required"
],
"identity:get_project": [
    "rule:admin_required"
],
"identity:list_projects": [
    "rule:admin_required"
],
"identity:list_user_projects": [
    "rule:admin_or_owner"
],
"identity:create_project": [
    "rule:admin_required"
],
"identity:update_project": [
    "rule:admin_required"
],
"identity:delete_project": [
    "rule:admin_required"
],
"identity:get_user": [
    "rule:admin_required"
],
"identity:list_users": [
    "rule:admin_required"
],
"identity:create_user": [
    "rule:admin_required"
],
"identity:update_user": [
```

```

        "rule:admin_or_owner"
    ],
    "identity:delete_user": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:get_group": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:list_groups": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:list_groups_for_user": [
        [
            "rule:admin_or_owner"
        ]
    ],
    "identity:create_group": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:update_group": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:delete_group": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:list_users_in_group": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:remove_user_from_group": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:check_user_in_group": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:add_user_to_group": [
        [
            "rule:admin_required"
        ]
    ]

```

```
    ],
    "identity:get_credential": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:list_credentials": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:create_credential": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:update_credential": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:delete_credential": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:get_role": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:list_roles": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:create_role": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:update_role": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:delete_role": [
        [
            "rule:admin_required"
        ]
    ],
    "identity:check_grant": [
        [
            "rule:admin_required"
        ]
    ]
}
```

```

],
"identity:list_grants": [
  [
    "rule:admin_required"
  ]
],
"identity:create_grant": [
  [
    "rule:admin_required"
  ]
],
"identity:revoke_grant": [
  [
    "rule:admin_required"
  ]
],
"identity:list_role_assignments": [
  [
    "rule:admin_required"
  ]
],
"identity:get_policy": [
  [
    "rule:admin_required"
  ]
],
"identity:list_policies": [
  [
    "rule:admin_required"
  ]
],
"identity:create_policy": [
  [
    "rule:admin_required"
  ]
],
"identity:update_policy": [
  [
    "rule:admin_required"
  ]
],
"identity:delete_policy": [
  [
    "rule:admin_required"
  ]
],
"identity:check_token": [
  [
    "rule:admin_required"
  ]
],
"identity:validate_token": [
  [
    "rule:service_or_admin"
  ]
],
],

```

```

    "identity:validate_token_head": [
        [
            "rule:service_or_admin"
        ]
    ],
    "identity:revocation_list": [
        [
            "rule:service_or_admin"
        ]
    ],
    "identity:revoke_token": [
        [
            "rule:admin_or_owner"
        ]
    ],
    "identity:create_trust": [
        [
            "user_id:$(trust.trustor_user_id)s"
        ]
    ],
    "identity:get_trust": [
        [
            "rule:admin_or_owner"
        ]
    ],
    "identity:list_trusts": [
        [
            "@"
        ]
    ],
    "identity:list_roles_for_trust": [
        [
            "@"
        ]
    ],
    "identity:check_role_for_trust": [
        [
            "@"
        ]
    ],
    "identity:get_role_for_trust": [
        [
            "@"
        ]
    ],
    "identity:delete_trust": [
        [
            "@"
        ]
    ]
}

```

4.2.2. nova_policy.json

The `nova_policy.json` file defines additional access controls for the dashboard that apply to the Compute service.

**NOTE**

The `nova_policy.json` file must match the Compute `/etc/nova/policy.json` policy file.

```
{
  "context_is_admin": "role:admin",
  "admin_or_owner": "is_admin:True or project_id:%(project_id)s",
  "default": "rule:admin_or_owner",
  "cells_scheduler_filter:TargetCellFilter": "is_admin:True",
  "compute:create": "",
  "compute:create:attach_network": "",
  "compute:create:attach_volume": "",
  "compute:create:forced_host": "is_admin:True",
  "compute:get": "",
  "compute:get_all": "",
  "compute:get_all_tenants": "",
  "compute:update": "",
  "compute:get_instance_metadata": "",
  "compute:get_all_instance_metadata": "",
  "compute:get_all_instance_system_metadata": "",
  "compute:update_instance_metadata": "",
  "compute:delete_instance_metadata": "",
  "compute:get_instance_faults": "",
  "compute:get_diagnostics": "",
  "compute:get_instance_diagnostics": "",
  "compute:start": "rule:admin_or_owner",
  "compute:stop": "rule:admin_or_owner",
  "compute:get_lock": "",
  "compute:lock": "",
  "compute:unlock": "",
  "compute:unlock_override": "rule:admin_api",
  "compute:get_vnc_console": "",
  "compute:get_spice_console": "",
  "compute:get_rdp_console": "",
  "compute:get_serial_console": "",
  "compute:get_mks_console": "",
  "compute:get_console_output": "",
  "compute:reset_network": "",
  "compute:inject_network_info": "",
  "compute:add_fixed_ip": "",
  "compute:remove_fixed_ip": "",
  "compute:attach_volume": "",
  "compute:detach_volume": "",
  "compute:swap_volume": "",
  "compute:attach_interface": "",
  "compute:detach_interface": "",
  "compute:set_admin_password": "",
  "compute:rescue": "",
  "compute:unrescue": "",
  "compute:suspend": "",
  "compute:resume": "",
  "compute:pause": "",
  "compute:unpause": "",
  "compute:shelve": "",
  "compute:shelve_offload": "",
```

```

"compute:unshelve": "",
"compute:snapshot": "",
"compute:snapshot_volume_backed": "",
"compute:backup": "",
"compute:resize": "",
"compute:confirm_resize": "",
"compute:revert_resize": "",
"compute:rebuild": "",
"compute:reboot": "",
"compute:delete": "rule:admin_or_owner",
"compute:soft_delete": "rule:admin_or_owner",
"compute:force_delete": "rule:admin_or_owner",
"compute:security_groups:add_to_instance": "",
"compute:security_groups:remove_from_instance": "",
"compute:delete": "",
"compute:soft_delete": "",
"compute:force_delete": "",
"compute:restore": "",
"compute:volume_snapshot_create": "",
"compute:volume_snapshot_delete": "",
"admin_api": "is_admin:True",
"compute_extension:accounts": "rule:admin_api",
"compute_extension:admin_actions": "rule:admin_api",
"compute_extension:admin_actions:pause": "rule:admin_or_owner",
"compute_extension:admin_actions:unpause": "rule:admin_or_owner",
"compute_extension:admin_actions:suspend": "rule:admin_or_owner",
"compute_extension:admin_actions:resume": "rule:admin_or_owner",
"compute_extension:admin_actions:lock": "rule:admin_or_owner",
"compute_extension:admin_actions:unlock": "rule:admin_or_owner",
"compute_extension:admin_actions:resetNetwork": "rule:admin_api",
"compute_extension:admin_actions:injectNetworkInfo": "rule:admin_api",
"compute_extension:admin_actions:createBackup": "rule:admin_or_owner",
"compute_extension:admin_actions:migrateLive": "rule:admin_api",
"compute_extension:admin_actions:resetState": "rule:admin_api",
"compute_extension:admin_actions:migrate": "rule:admin_api",
"compute_extension:v3:os-admin-actions": "rule:admin_api",
"compute_extension:v3:os-admin-actions:pause": "rule:admin_or_owner",
"compute_extension:v3:os-admin-actions:unpause":
"rule:admin_or_owner",
  "compute_extension:v3:os-admin-actions:suspend":
"rule:admin_or_owner",
  "compute_extension:v3:os-admin-actions:resume": "rule:admin_or_owner",
  "compute_extension:v3:os-admin-actions:lock": "rule:admin_or_owner",
  "compute_extension:v3:os-admin-actions:unlock": "rule:admin_or_owner",
  "compute_extension:v3:os-admin-actions:reset_network":
"rule:admin_api",
  "compute_extension:v3:os-admin-actions:inject_network_info":
"rule:admin_api",
  "compute_extension:v3:os-admin-actions:create_backup":
"rule:admin_or_owner",
  "compute_extension:v3:os-admin-actions:migrate_live":
"rule:admin_api",
  "compute_extension:v3:os-admin-actions:reset_state": "rule:admin_api",
  "compute_extension:v3:os-admin-actions:migrate": "rule:admin_api",
  "compute_extension:v3:os-admin-password": "",
  "compute_extension:aggregates": "rule:admin_api",

```



```

"compute_extension:v3:os-aggregates": "rule:admin_api",
"compute_extension:agents": "rule:admin_api",
"compute_extension:v3:os-agents": "rule:admin_api",
"compute_extension:attach_interfaces": "",
"compute_extension:v3:os-attach-interfaces": "",
"compute_extension:baremetal_nodes": "rule:admin_api",
"compute_extension:v3:os-baremetal-nodes": "rule:admin_api",
"compute_extension:cells": "rule:admin_api",
"compute_extension:v3:os-cells": "rule:admin_api",
"compute_extension:cells:create": "rule:admin_api",
"compute_extension:cells:delete": "rule:admin_api",
"compute_extension:cells:update": "rule:admin_api",
"compute_extension:cells:sync_instances": "rule:admin_api",
"compute_extension:certificates": "",
"compute_extension:v3:os-certificates": "",
"compute_extension:cloudpipe": "rule:admin_api",
"compute_extension:cloudpipe_update": "rule:admin_api",
"compute_extension:config_drive": "",
"compute_extension:console_output": "",
"compute_extension:v3:consoles:discoverable": "",
"compute_extension:v3:os-console-output": "",
"compute_extension:consoles": "",
"compute_extension:v3:os-remote-consoles": "",
"compute_extension:coverage_ext": "rule:admin_api",
"compute_extension:v3:os-coverage": "rule:admin_api",
"compute_extension:createserverext": "",
"compute_extension:deferred_delete": "",
"compute_extension:v3:os-deferred-delete": "",
"compute_extension:disk_config": "",
"compute_extension:evacuate": "rule:admin_api",
"compute_extension:v3:os-evacuate": "rule:admin_api",
"compute_extension:extended_server_attributes": "rule:admin_api",
"compute_extension:v3:os-extended-server-attributes":
"rule:admin_api",
"compute_extension:extended_status": "",
"compute_extension:v3:os-extended-status": "",
"compute_extension:extended_availability_zone": "",
"compute_extension:v3:os-extended-availability-zone": "",
"compute_extension:extended_ips": "",
"compute_extension:extended_ips_mac": "",
"compute_extension:extended_vif_net": "",
"compute_extension:v3:extension_info:discoverable": "",
"compute_extension:extended_volumes": "",
"compute_extension:v3:os-extended-volumes": "",
"compute_extension:v3:os-extended-volumes:attach": "",
"compute_extension:v3:os-extended-volumes:detach": "",
"compute_extension:fixed_ips": "rule:admin_api",
"compute_extension:v3:os-fixed-ips:discoverable": "",
"compute_extension:v3:os-fixed-ips": "rule:admin_api",
"compute_extension:flavor_access": "",
"compute_extension:v3:os-flavor-access": "",
"compute_extension:flavor_access:addTenantAccess": "rule:admin_api",
"compute_extension:flavor_access:removeTenantAccess":
"rule:admin_api",
"compute_extension:flavor_disabled": "",
"compute_extension:v3:os-flavor-disabled": "",

```

```

"compute_extension:flavor_rxtx": "",
"compute_extension:v3:os-flavor-rxtx": "",
"compute_extension:flavor_swap": "",
"compute_extension:flavorextradata": "",
"compute_extension:flavorextraspecs:index": "",
"compute_extension:flavorextraspecs:show": "",
"compute_extension:flavorextraspecs:create": "rule:admin_api",
"compute_extension:flavorextraspecs:update": "rule:admin_api",
"compute_extension:flavorextraspecs:delete": "rule:admin_api",
"compute_extension:v3:flavor-extra-specs:index": "",
"compute_extension:v3:flavor-extra-specs:show": "",
"compute_extension:v3:flavor-extra-specs:create": "rule:admin_api",
"compute_extension:v3:flavor-extra-specs:update": "rule:admin_api",
"compute_extension:v3:flavor-extra-specs:delete": "rule:admin_api",
"compute_extension:flavormanage": "rule:admin_api",
"compute_extension:floating_ip_dns": "",
"compute_extension:floating_ip_pools": "",
"compute_extension:floating_ips": "",
"compute_extension:floating_ips_bulk": "rule:admin_api",
"compute_extension:fping": "",
"compute_extension:fping:all_tenants": "rule:admin_api",
"compute_extension:hide_server_addresses": "is_admin:False",
"compute_extension:v3:os-hide-server-addresses": "is_admin:False",
"compute_extension:hosts": "rule:admin_api",
"compute_extension:v3:os-hosts": "rule:admin_api",
"compute_extension:hypervisors": "rule:admin_api",
"compute_extension:v3:os-hypervisors": "rule:admin_api",
"compute_extension:image_size": "",
"compute_extension:v3:os-image-metadata": "",
"compute_extension:v3:os-images": "",
"compute_extension:instance_actions": "",
"compute_extension:v3:os-instance-actions": "",
"compute_extension:instance_actions:events": "rule:admin_api",
"compute_extension:v3:os-instance-actions:events": "rule:admin_api",
"compute_extension:instance_usage_audit_log": "rule:admin_api",
"compute_extension:v3:os-instance-usage-audit-log": "rule:admin_api",
"compute_extension:v3:ips:discoverable": "",
"compute_extension:keypairs": "",
"compute_extension:keypairs:index": "",
"compute_extension:keypairs:show": "",
"compute_extension:keypairs:create": "",
"compute_extension:keypairs:delete": "",
"compute_extension:v3:os-keypairs:discoverable": "",
"compute_extension:v3:os-keypairs": "",
"compute_extension:v3:os-keypairs:index": "",
"compute_extension:v3:os-keypairs:show": "",
"compute_extension:v3:os-keypairs:create": "",
"compute_extension:v3:os-keypairs:delete": "",
"compute_extension:multinic": "",
"compute_extension:v3:os-multinic": "",
"compute_extension:networks": "rule:admin_api",
"compute_extension:networks:view": "",
"compute_extension:networks_associate": "rule:admin_api",
"compute_extension:quotas:show": "",
"compute_extension:quotas:update": "rule:admin_api",
"compute_extension:quotas:delete": "rule:admin_api",

```

```

"compute_extension:v3:os-quota-sets:show": "",
"compute_extension:v3:os-quota-sets:update": "rule:admin_api",
"compute_extension:v3:os-quota-sets:delete": "rule:admin_api",
"compute_extension:quota_classes": "",
"compute_extension:v3:os-quota-class-sets": "",
"compute_extension:rescue": "",
"compute_extension:v3:os-rescue": "",
"compute_extension:security_group_default_rules": "rule:admin_api",
"compute_extension:security_groups": "",
"compute_extension:v3:os-security-groups": "",
"compute_extension:server_diagnostics": "rule:admin_api",
"compute_extension:v3:os-server-diagnostics": "rule:admin_api",
"compute_extension:server_password": "",
"compute_extension:v3:os-server-password": "",
"compute_extension:server_usage": "",
"compute_extension:v3:os-server-usage": "",
"compute_extension:services": "rule:admin_api",
"compute_extension:v3:os-services": "rule:admin_api",
"compute_extension:v3:servers:discoverable": "",
"compute_extension:shelve": "",
"compute_extension:shelveOffload": "rule:admin_api",
"compute_extension:v3:os-shelve:shelve": "",
"compute_extension:v3:os-shelve:shelve_offload": "rule:admin_api",
"compute_extension:simple_tenant_usage:show": "rule:admin_or_owner",
"compute_extension:v3:os-simple-tenant-usage:show":
"rule:admin_or_owner",
"compute_extension:simple_tenant_usage:list": "rule:admin_api",
"compute_extension:v3:os-simple-tenant-usage:list": "rule:admin_api",
"compute_extension:unshelve": "",
"compute_extension:v3:os-shelve:unshelve": "",
"compute_extension:users": "rule:admin_api",
"compute_extension:virtual_interfaces": "",
"compute_extension:virtual_storage_arrays": "",
"compute_extension:volumes": "",
"compute_extension:volume_attachments:index": "",
"compute_extension:volume_attachments:show": "",
"compute_extension:volume_attachments:create": "",
"compute_extension:volume_attachments:update": "",
"compute_extension:volume_attachments:delete": "",
"compute_extension:volumetypes": "",
"compute_extension:availability_zone:list": "",
"compute_extension:v3:os-availability-zone:list": "",
"compute_extension:availability_zone:detail": "rule:admin_api",
"compute_extension:v3:os-availability-zone:detail": "rule:admin_api",
"compute_extension:used_limits_for_admin": "rule:admin_api",
"compute_extension:v3:os-used-limits": "",
"compute_extension:v3:os-used-limits:tenant": "rule:admin_api",
"compute_extension:migrations:index": "rule:admin_api",
"compute_extension:v3:os-migrations:index": "rule:admin_api",
"compute_extension:os-assisted-volume-snapshots:create":
"rule:admin_api",
"compute_extension:os-assisted-volume-snapshots:delete":
"rule:admin_api",
"compute_extension:console_auth_tokens": "rule:admin_api",
"compute_extension:os-server-external-events:create":
"rule:admin_api",

```

```

"volume:create": "",
"volume:get_all": "",
"volume:get_volume_metadata": "",
"volume:get_snapshot": "",
"volume:get_all_snapshots": "",
"volume_extension:types_manage": "rule:admin_api",
"volume_extension:types_extra_specs": "rule:admin_api",
"volume_extension:volume_admin_actions:reset_status":
"rule:admin_api",
"volume_extension:snapshot_admin_actions:reset_status":
"rule:admin_api",
"volume_extension:volume_admin_actions:force_delete":
"rule:admin_api",
"network:get_all": "",
"network:get": "",
"network:create": "",
"network:delete": "",
"network:associate": "",
"network:disassociate": "",
"network:get_vifs_by_instance": "",
"network:allocate_for_instance": "",
"network:deallocate_for_instance": "",
"network:validate_networks": "",
"network:get_instance_uuids_by_ip_filter": "",
"network:get_instance_id_by_floating_address": "",
"network:setup_networks_on_host": "",
"network:get_backdoor_port": "",
"network:get_floating_ip": "",
"network:get_floating_ip_pools": "",
"network:get_floating_ip_by_address": "",
"network:get_floating_ips_by_project": "",
"network:get_floating_ips_by_fixed_address": "",
"network:allocate_floating_ip": "",
"network:deallocate_floating_ip": "",
"network:associate_floating_ip": "",
"network:disassociate_floating_ip": "",
"network:release_floating_ip": "",
"network:migrate_instance_start": "",
"network:migrate_instance_finish": "",
"network:get_fixed_ip": "",
"network:get_fixed_ip_by_address": "",
"network:add_fixed_ip_to_instance": "",
"network:remove_fixed_ip_from_instance": "",
"network:add_network_to_project": "",
"network:get_instance_nw_info": "",
"network:get_dns_domains": "",
"network:add_dns_entry": "",
"network:modify_dns_entry": "",
"network:delete_dns_entry": "",
"network:get_dns_entries_by_address": "",
"network:get_dns_entries_by_name": "",
"network:create_private_dns_domain": "",
"network:create_public_dns_domain": "",
"network:delete_dns_domain": "",
"network:attach_external_network": "rule:admin_api",
"network:get_vif_by_mac_address": "",

```

```

"os_compute_api:servers:detail:get_all_tenants": "is_admin:True",
"os_compute_api:servers:index:get_all_tenants": "is_admin:True",
"os_compute_api:servers:confirm_resize": "",
"os_compute_api:servers:create": "",
"os_compute_api:servers:create:attach_network": "",
"os_compute_api:servers:create:attach_volume": "",
"os_compute_api:servers:create:forced_host": "rule:admin_api",
"os_compute_api:servers:delete": "",
"os_compute_api:servers:update": "",
"os_compute_api:servers:detail": "",
"os_compute_api:servers:index": "",
"os_compute_api:servers:reboot": "",
"os_compute_api:servers:rebuild": "",
"os_compute_api:servers:resize": "",
"os_compute_api:servers:revert_resize": "",
"os_compute_api:servers:show": "",
"os_compute_api:servers:create_image": "",
"os_compute_api:servers:create_image:allow_volume_backed": "",
"os_compute_api:servers:start": "rule:admin_or_owner",
"os_compute_api:servers:stop": "rule:admin_or_owner",
"os_compute_api:os-access-ips:discoverable": "",
"os_compute_api:os-access-ips": "",
"os_compute_api:os-admin-actions": "rule:admin_api",
"os_compute_api:os-admin-actions:discoverable": "",
"os_compute_api:os-admin-actions:reset_network": "rule:admin_api",
"os_compute_api:os-admin-actions:inject_network_info":
"rule:admin_api",
"os_compute_api:os-admin-actions:reset_state": "rule:admin_api",
"os_compute_api:os-admin-password": "",
"os_compute_api:os-admin-password:discoverable": "",
"os_compute_api:os-aggregates:discoverable": "",
"os_compute_api:os-aggregates:index": "rule:admin_api",
"os_compute_api:os-aggregates:create": "rule:admin_api",
"os_compute_api:os-aggregates:show": "rule:admin_api",
"os_compute_api:os-aggregates:update": "rule:admin_api",
"os_compute_api:os-aggregates:delete": "rule:admin_api",
"os_compute_api:os-aggregates:add_host": "rule:admin_api",
"os_compute_api:os-aggregates:remove_host": "rule:admin_api",
"os_compute_api:os-aggregates:set_metadata": "rule:admin_api",
"os_compute_api:os-agents": "rule:admin_api",
"os_compute_api:os-agents:discoverable": "",
"os_compute_api:os-attach-interfaces": "",
"os_compute_api:os-attach-interfaces:discoverable": "",
"os_compute_api:os-baremetal-nodes": "rule:admin_api",
"os_compute_api:os-baremetal-nodes:discoverable": "",
"os_compute_api:os-block-device-mapping-v1:discoverable": "",
"os_compute_api:os-cells": "rule:admin_api",
"os_compute_api:os-cells:create": "rule:admin_api",
"os_compute_api:os-cells:delete": "rule:admin_api",
"os_compute_api:os-cells:update": "rule:admin_api",
"os_compute_api:os-cells:sync_instances": "rule:admin_api",
"os_compute_api:os-cells:discoverable": "",
"os_compute_api:os-certificates:create": "",
"os_compute_api:os-certificates:show": "",
"os_compute_api:os-certificates:discoverable": "",

```

```

"os_compute_api:os-cloudpipe": "rule:admin_api",
"os_compute_api:os-cloudpipe:discoverable": "",
"os_compute_api:os-config-drive": "",
"os_compute_api:os-consoles:discoverable": "",
"os_compute_api:os-consoles:create": "",
"os_compute_api:os-consoles:delete": "",
"os_compute_api:os-consoles:index": "",
"os_compute_api:os-consoles:show": "",
"os_compute_api:os-console-output:discoverable": "",
"os_compute_api:os-console-output": "",
"os_compute_api:os-remote-consoles": "",
"os_compute_api:os-remote-consoles:discoverable": "",
"os_compute_api:os-create-backup:discoverable": "",
"os_compute_api:os-create-backup": "rule:admin_or_owner",
"os_compute_api:os-deferred-delete": "",
"os_compute_api:os-deferred-delete:discoverable": "",
"os_compute_api:os-disk-config": "",
"os_compute_api:os-disk-config:discoverable": "",
"os_compute_api:os-evacuate": "rule:admin_api",
"os_compute_api:os-evacuate:discoverable": "",
"os_compute_api:os-extended-server-attributes": "rule:admin_api",
"os_compute_api:os-extended-server-attributes:discoverable": "",
"os_compute_api:os-extended-status": "",
"os_compute_api:os-extended-status:discoverable": "",
"os_compute_api:os-extended-availability-zone": "",
"os_compute_api:os-extended-availability-zone:discoverable": "",
"os_compute_api:extensions": "",
"os_compute_api:extension_info:discoverable": "",
"os_compute_api:os-extended-volumes": "",
"os_compute_api:os-extended-volumes:discoverable": "",
"os_compute_api:os-fixed-ips": "rule:admin_api",
"os_compute_api:os-fixed-ips:discoverable": "",
"os_compute_api:os-flavor-access": "",
"os_compute_api:os-flavor-access:discoverable": "",
"os_compute_api:os-flavor-access:remove_tenant_access":
"rule:admin_api",
"os_compute_api:os-flavor-access:add_tenant_access": "rule:admin_api",
"os_compute_api:os-flavor-rxtx": "",
"os_compute_api:os-flavor-rxtx:discoverable": "",
"os_compute_api:flavors:discoverable": "",
"os_compute_api:os-flavor-extra-specs:discoverable": "",
"os_compute_api:os-flavor-extra-specs:index": "",
"os_compute_api:os-flavor-extra-specs:show": "",
"os_compute_api:os-flavor-extra-specs:create": "rule:admin_api",
"os_compute_api:os-flavor-extra-specs:update": "rule:admin_api",
"os_compute_api:os-flavor-extra-specs:delete": "rule:admin_api",
"os_compute_api:os-flavor-manage:discoverable": "",
"os_compute_api:os-flavor-manage": "rule:admin_api",
"os_compute_api:os-floating-ip-dns": "",
"os_compute_api:os-floating-ip-dns:discoverable": "",
"os_compute_api:os-floating-ip-dns:domain:update": "rule:admin_api",
"os_compute_api:os-floating-ip-dns:domain:delete": "rule:admin_api",
"os_compute_api:os-floating-ip-pools": "",
"os_compute_api:os-floating-ip-pools:discoverable": "",
"os_compute_api:os-floating-ips": "",
"os_compute_api:os-floating-ips:discoverable": "",

```

```

"os_compute_api:os-floating-ips-bulk": "rule:admin_api",
"os_compute_api:os-floating-ips-bulk:discoverable": "",
"os_compute_api:os-fping": "",
"os_compute_api:os-fping:discoverable": "",
"os_compute_api:os-fping:all_tenants": "rule:admin_api",
"os_compute_api:os-hide-server-addresses": "is_admin:False",
"os_compute_api:os-hide-server-addresses:discoverable": "",
"os_compute_api:os-hosts": "rule:admin_api",
"os_compute_api:os-hosts:discoverable": "",
"os_compute_api:os-hypervisors": "rule:admin_api",
"os_compute_api:os-hypervisors:discoverable": "",
"os_compute_api:images:discoverable": "",
"os_compute_api:image-size": "",
"os_compute_api:image-size:discoverable": "",
"os_compute_api:os-instance-actions": "",
"os_compute_api:os-instance-actions:discoverable": "",
"os_compute_api:os-instance-actions:events": "rule:admin_api",
"os_compute_api:os-instance-usage-audit-log": "rule:admin_api",
"os_compute_api:os-instance-usage-audit-log:discoverable": "",
"os_compute_api:ips:discoverable": "",
"os_compute_api:ips:index": "rule:admin_or_owner",
"os_compute_api:ips:show": "rule:admin_or_owner",
"os_compute_api:os-keypairs:discoverable": "",
"os_compute_api:os-keypairs": "",
"os_compute_api:os-keypairs:index": "rule:admin_api or user_id:%
(user_id)s",
"os_compute_api:os-keypairs:show": "rule:admin_api or user_id:%
(user_id)s",
"os_compute_api:os-keypairs:create": "rule:admin_api or user_id:%
(user_id)s",
"os_compute_api:os-keypairs:delete": "rule:admin_api or user_id:%
(user_id)s",
"os_compute_api:limits:discoverable": "",
"os_compute_api:limits": "",
"os_compute_api:os-lock-server:discoverable": "",
"os_compute_api:os-lock-server:lock": "rule:admin_or_owner",
"os_compute_api:os-lock-server:unlock": "rule:admin_or_owner",
"os_compute_api:os-lock-server:unlock:unlock_override":
"rule:admin_api",
"os_compute_api:os-migrate-server:discoverable": "",
"os_compute_api:os-migrate-server:migrate": "rule:admin_api",
"os_compute_api:os-migrate-server:migrate_live": "rule:admin_api",
"os_compute_api:os-multinic": "",
"os_compute_api:os-multinic:discoverable": "",
"os_compute_api:os-networks": "rule:admin_api",
"os_compute_api:os-networks:view": "",
"os_compute_api:os-networks:discoverable": "",
"os_compute_api:os-networks:associate": "rule:admin_api",
"os_compute_api:os-networks:associate:discoverable": "",
"os_compute_api:os-pause-server:discoverable": "",
"os_compute_api:os-pause-server:pause": "rule:admin_or_owner",
"os_compute_api:os-pause-server:unpause": "rule:admin_or_owner",
"os_compute_api:os-pci:pci_servers": "",
"os_compute_api:os-pci:discoverable": "",
"os_compute_api:os-pci:index": "rule:admin_api",
"os_compute_api:os-pci:detail": "rule:admin_api",

```

```

"os_compute_api:os-pci:show": "rule:admin_api",
"os_compute_api:os-personality:discoverable": "",
"os_compute_api:os-preserve-ephemeral-rebuild:discoverable": "",
"os_compute_api:os-quota-sets:discoverable": "",
"os_compute_api:os-quota-sets:show": "rule:admin_or_owner",
"os_compute_api:os-quota-sets:defaults": "",
"os_compute_api:os-quota-sets:update": "rule:admin_api",
"os_compute_api:os-quota-sets:delete": "rule:admin_api",
"os_compute_api:os-quota-sets:detail": "rule:admin_api",
"os_compute_api:os-quota-class-sets:update": "rule:admin_api",
"os_compute_api:os-quota-class-sets:show": "is_admin:True or
quota_class:%(quota_class)s",
"os_compute_api:os-quota-class-sets:discoverable": "",
"os_compute_api:os-rescue": "",
"os_compute_api:os-rescue:discoverable": "",
"os_compute_api:os-scheduler-hints:discoverable": "",
"os_compute_api:os-security-group-default-rules:discoverable": "",
"os_compute_api:os-security-group-default-rules": "rule:admin_api",
"os_compute_api:os-security-groups": "",
"os_compute_api:os-security-groups:discoverable": "",
"os_compute_api:os-server-diagnostics": "rule:admin_api",
"os_compute_api:os-server-diagnostics:discoverable": "",
"os_compute_api:os-server-password": "",
"os_compute_api:os-server-password:discoverable": "",
"os_compute_api:os-server-usage": "",
"os_compute_api:os-server-usage:discoverable": "",
"os_compute_api:os-server-groups": "",
"os_compute_api:os-server-groups:discoverable": "",
"os_compute_api:os-services": "rule:admin_api",
"os_compute_api:os-services:discoverable": "",
"os_compute_api:server-metadata:discoverable": "",
"os_compute_api:server-metadata:index": "rule:admin_or_owner",
"os_compute_api:server-metadata:show": "rule:admin_or_owner",
"os_compute_api:server-metadata:delete": "rule:admin_or_owner",
"os_compute_api:server-metadata:create": "rule:admin_or_owner",
"os_compute_api:server-metadata:update": "rule:admin_or_owner",
"os_compute_api:server-metadata:update_all": "rule:admin_or_owner",
"os_compute_api:servers:discoverable": "",
"os_compute_api:os-shelve:shelve": "",
"os_compute_api:os-shelve:shelve:discoverable": "",
"os_compute_api:os-shelve:shelve_offload": "rule:admin_api",
"os_compute_api:os-simple-tenant-usage:discoverable": "",
"os_compute_api:os-simple-tenant-usage:show": "rule:admin_or_owner",
"os_compute_api:os-simple-tenant-usage:list": "rule:admin_api",
"os_compute_api:os-suspend-server:discoverable": "",
"os_compute_api:os-suspend-server:suspend": "rule:admin_or_owner",
"os_compute_api:os-suspend-server:resume": "rule:admin_or_owner",
"os_compute_api:os-tenant-networks": "rule:admin_or_owner",
"os_compute_api:os-tenant-networks:discoverable": "",
"os_compute_api:os-shelve:unshelve": "",
"os_compute_api:os-user-data:discoverable": "",
"os_compute_api:os-virtual-interfaces": "",
"os_compute_api:os-virtual-interfaces:discoverable": "",
"os_compute_api:os-volumes": "",
"os_compute_api:os-volumes:discoverable": "",
"os_compute_api:os-volumes-attachments:index": "",

```



```

"os_compute_api:os-volumes-attachments:show": "",
"os_compute_api:os-volumes-attachments:create": "",
"os_compute_api:os-volumes-attachments:update": "",
"os_compute_api:os-volumes-attachments:delete": "",
"os_compute_api:os-volumes-attachments:discoverable": "",
"os_compute_api:os-availability-zone:list": "",
"os_compute_api:os-availability-zone:discoverable": "",
"os_compute_api:os-availability-zone:detail": "rule:admin_api",
"os_compute_api:os-used-limits": "rule:admin_api",
"os_compute_api:os-used-limits:discoverable": "",
"os_compute_api:os-migrations:index": "rule:admin_api",
"os_compute_api:os-migrations:discoverable": "",
"os_compute_api:os-assisted-volume-snapshots:create":
"rule:admin_api",
  "os_compute_api:os-assisted-volume-snapshots:delete":
"rule:admin_api",
  "os_compute_api:os-assisted-volume-snapshots:discoverable": "",
  "os_compute_api:os-console-auth-tokens": "rule:admin_api",
  "os_compute_api:os-server-external-events:create": "rule:admin_api"
}

```

4.3. DASHBOARD LOG FILES

The dashboard is served to users through the Apache web server (**httpd**).

As a result, dashboard-related logs appear in files in the **/var/log/httpd** directory on the system where the dashboard is hosted.

Log file names are based on the installer used and how the log files are named is defined in **/etc/httpd/conf.d/** file, which is the Dashboard **httpd** configuration file, which is again dependent on installer.

The following table describes these files:

Table 4.1. Dashboard/httpd log files

Log file	Description
access_log	Logs all attempts to access the web server.
error_log	Logs all unsuccessful attempts to access the web server, along with the reason that each attempt failed.

CHAPTER 5. DATABASE SERVICE

The Database service provides a scalable and reliable Cloud Database-as-a-Service functionality for both relational and non-relational database engines.

The following tables provide a comprehensive list of the Database service configuration options.

Table 5.1. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
admin_roles = <i>admin</i>	(ListOpt) Roles to add to an admin user.
api_paste_config = <i>api-paste.ini</i>	(StrOpt) File name for the paste.deploy config for trove-api.
bind_host = <i>0.0.0.0</i>	(StrOpt) IP address the API server will listen on.
bind_port = <i>8779</i>	(IntOpt) Port the API server will listen on.
black_list_regex = <i>None</i>	(StrOpt) Exclude IP addresses that match this regular expression.
db_api_implementation = <i>trove.db.sqlalchemy.api</i>	(StrOpt) API Implementation for Trove database access.
hostname_require_valid_ip = <i>True</i>	(BoolOpt) Require user hostnames to be valid IP addresses.
http_delete_rate = <i>200</i>	(IntOpt) Maximum number of HTTP 'DELETE' requests (per minute).
http_get_rate = <i>200</i>	(IntOpt) Maximum number of HTTP 'GET' requests (per minute).
http_mgmt_post_rate = <i>200</i>	(IntOpt) Maximum number of management HTTP 'POST' requests (per minute).
http_post_rate = <i>200</i>	(IntOpt) Maximum number of HTTP 'POST' requests (per minute).
http_put_rate = <i>200</i>	(IntOpt) Maximum number of HTTP 'PUT' requests (per minute).
injected_config_location = <i>/etc/trove/conf.d</i>	(StrOpt) Path to folder on the Guest where config files will be injected during instance creation.
instances_page_size = <i>20</i>	(IntOpt) Page size for listing instances.

Configuration option = Default value	Description
max_header_line = 16384	(IntOpt) Maximum line size of message headers to be accepted. max_header_line may need to be increased when using large tokens (typically those generated by the Keystone v3 API with big service catalogs).
os_region_name = RegionOne	(StrOpt) Region name of this node. Used when searching catalog.
region = LOCAL_DEV	(StrOpt) The region this service is located.
tcp_keepidle = 600	(IntOpt) Sets the value of TCP_KEEPIDLE in seconds for each server socket. Not supported on OS X.
trove_api_workers = None	(IntOpt) Number of workers for the API service. The default will be the number of CPUs available.
trove_auth_url = http://0.0.0.0:5000/v2.0	(StrOpt) Trove authentication URL.
trove_conductor_workers = None	(IntOpt) Number of workers for the Conductor service. The default will be the number of CPUs available.
trove_security_group_name_prefix = SecGroup	(StrOpt) Prefix to use when creating Security Groups.
trove_security_group_rule_cidr = 0.0.0.0/0	(StrOpt) CIDR to use when creating Security Group Rules.
trove_security_groups_support = True	(BoolOpt) Whether Trove should add Security Groups on create.
users_page_size = 20	(IntOpt) Page size for listing users.

Table 5.2. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = None	(StrOpt) Service user password.
admin_tenant_name = admin	(StrOpt) Service tenant name.

Configuration option = Default value	Description
admin_token = <i>None</i>	(StrOpt) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(StrOpt) Service username.
auth_admin_prefix =	(StrOpt) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
auth_host = <i>127.0.0.1</i>	(StrOpt) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_plugin = <i>None</i>	(StrOpt) Name of the plugin to load
auth_port = <i>35357</i>	(IntOpt) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_protocol = <i>https</i>	(StrOpt) Protocol of the admin Identity API endpoint (http or https). Deprecated, use <code>identity_uri</code> .
auth_section = <i>None</i>	(StrOpt) Config Section from which to load plugin specific options
auth_uri = <i>None</i>	(StrOpt) Complete public Identity API endpoint.
auth_version = <i>None</i>	(StrOpt) API version of the admin Identity API endpoint.
cache = <i>None</i>	(StrOpt) Env key for the swift cache.
cafile = <i>None</i>	(StrOpt) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate
check_revocations_for_cached = <i>False</i>	(BoolOpt) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.

Configuration option = Default value	Description
delay_auth_decision = <i>False</i>	(BoolOpt) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(StrOpt) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(ListOpt) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(IntOpt) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(IntOpt) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(StrOpt) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/
include_service_catalog = <i>True</i>	(BoolOpt) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate

Configuration option = Default value	Description
<code>memcache_pool_conn_get_timeout = 10</code>	(IntOpt) (Optional) Number of seconds that an operation will wait to get a memcached client connection from the pool.
<code>memcache_pool_dead_retry = 300</code>	(IntOpt) (Optional) Number of seconds memcached server is considered dead before it is tried again.
<code>memcache_pool_maxsize = 10</code>	(IntOpt) (Optional) Maximum total number of open connections to every memcached server.
<code>memcache_pool_socket_timeout = 3</code>	(IntOpt) (Optional) Socket timeout in seconds for communicating with a memcached server.
<code>memcache_pool_unused_timeout = 60</code>	(IntOpt) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
<code>memcache_secret_key = None</code>	(StrOpt) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
<code>memcache_security_strategy = None</code>	(StrOpt) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. Acceptable values are MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.
<code>memcache_use_advanced_pool = False</code>	(BoolOpt) (Optional) Use the advanced (eventlet safe) memcached client pool. The advanced pool will only work under python 2.x.
<code>region_name = None</code>	(StrOpt) The region in which the identity server can be found.
<code>revocation_cache_time = 10</code>	(IntOpt) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
<code>signing_dir = None</code>	(StrOpt) Directory used to cache files related to PKI tokens.

Configuration option = Default value	Description
token_cache_time = 300	(IntOpt) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 5.3. Description of backup configuration options

Configuration option = Default value	Description
[DEFAULT]	
backup_aes_cbc_key = <i>default_aes_cbc_key</i>	(StrOpt) Default OpenSSL aes_cbc key.
backup_chunk_size = 65536	(IntOpt) Chunk size (in bytes) to stream to the Swift container. This should be in multiples of 128 bytes, since this is the size of an md5 digest block allowing the process to update the file checksum during streaming. See: http://stackoverflow.com/questions/1131220/
backup_runner = <i>trove.guestagent.backup.backup_types.InnoBackupEx</i>	(StrOpt) Runner to use for backups.
backup_runner_options = {}	(DictOpt) Additional options to be passed to the backup runner.
backup_segment_max_size = 2147483648	(IntOpt) Maximum size (in bytes) of each segment of the backup file.
backup_swift_container = <i>database_backups</i>	(StrOpt) Swift container to put backups in.
backup_use_gzip_compression = <i>True</i>	(BoolOpt) Compress backups using gzip.
backup_use_openssl_encryption = <i>True</i>	(BoolOpt) Encrypt backups using OpenSSL.
backup_use_snet = <i>False</i>	(BoolOpt) Send backup files over snet.
backups_page_size = 20	(IntOpt) Page size for listing backups.

Table 5.4. Description of CA and SSL configuration options

Configuration option = Default value	Description
[ssl]	

Configuration option = Default value	Description
ca_file = <i>None</i>	(StrOpt) CA certificate file to use to verify connecting clients
cert_file = <i>None</i>	(StrOpt) Certificate file to use when starting the server securely
key_file = <i>None</i>	(StrOpt) Private key file to use when starting the server securely

Table 5.5. Description of clients configuration options

Configuration option = Default value	Description
[DEFAULT]	
remote_cinder_client = <i>trove.common.remote.cinder_client</i>	(StrOpt) Client to send Cinder calls to.
remote_dns_client = <i>trove.common.remote.dns_client</i>	(StrOpt) Client to send DNS calls to.
remote_guest_client = <i>trove.common.remote.guest_client</i>	(StrOpt) Client to send Guest Agent calls to.
remote_heat_client = <i>trove.common.remote.heat_client</i>	(StrOpt) Client to send Heat calls to.
remote_neutron_client = <i>trove.common.remote.neutron_client</i>	(StrOpt) Client to send Neutron calls to.
remote_nova_client = <i>trove.common.remote.nova_client</i>	(StrOpt) Client to send Nova calls to.
remote_swift_client = <i>trove.common.remote.swift_client</i>	(StrOpt) Client to send Swift calls to.

Table 5.6. Description of cluster configuration options

Configuration option = Default value	Description
[DEFAULT]	
cluster_delete_time_out = <i>180</i>	(IntOpt) Maximum time (in seconds) to wait for a cluster delete.

Configuration option = Default value	Description
cluster_usage_timeout = 36000	(IntOpt) Maximum time (in seconds) to wait for a cluster to become active.
clusters_page_size = 20	(IntOpt) Page size for listing clusters.

Table 5.7. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
configurations_page_size = 20	(IntOpt) Page size for listing configurations.
databases_page_size = 20	(IntOpt) Page size for listing databases.
default_datastore = None	(StrOpt) The default datastore id or name to use if one is not provided by the user. If the default value is None, the field becomes required in the instance create request.
default_neutron_networks =	(ListOpt) List of IDs for management networks which should be attached to the instance regardless of what NICs are specified in the create API call.
default_password_length = 36	(IntOpt) Character length of generated passwords.
executor_thread_pool_size = 64	(IntOpt) Size of executor thread pool.
expected_filetype_suffixes = json	(ListOpt) Filetype endings not to be reattached to an ID by the utils method <code>correct_id_with_req</code> .
host = 0.0.0.0	(StrOpt) Host to listen for RPC messages.
memcached_servers = None	(ListOpt) Memcached servers or None for in process cache.
pybasedir = /usr/lib/python2.7/site-packages/trove	(StrOpt) Directory where the Trove python module is installed.
pydev_path = None	(StrOpt) Set path to pydevd library, used if pydevd is not found in python sys.path.
taskmanager_queue = taskmanager	(StrOpt) Message queue name the Taskmanager will listen to.
template_path = /etc/trove/templates/	(StrOpt) Path which leads to datastore templates.

Configuration option = Default value	Description
timeout_wait_for_service = 120	(IntOpt) Maximum time (in seconds) to wait for a service to become alive.
usage_timeout = 900	(IntOpt) Maximum time (in seconds) to wait for a Guest to become active.
[keystone_authtoken]	
memcached_servers = None	(ListOpt) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.

Table 5.8. Description of Compute configuration options

Configuration option = Default value	Description
[DEFAULT]	
ip_regex = None	(StrOpt) List IP addresses that match this regular expression.
nova_compute_endpoint_type = publicURL	(StrOpt) Service endpoint type to use when searching catalog.
nova_compute_service_type = compute	(StrOpt) Service type to use when searching catalog.
nova_compute_url = None	(StrOpt) URL without the tenant segment.
root_grant = ALL	(ListOpt) Permissions to grant to the 'root' user.
root_grant_option = True	(BoolOpt) Assign the 'root' user GRANT permissions.

Table 5.9. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
backlog = 4096	(IntOpt) Number of backlog requests to configure the socket with
pydev_debug = disabled	(StrOpt) Enable or disable pydev remote debugging. If value is 'auto' tries to connect to remote debugger server, but in case of error continues running with debugging disabled.

Configuration option = Default value	Description
pydev_debug_host = <i>None</i>	(StrOpt) Pydev debug server host (localhost by default).
pydev_debug_port = <i>None</i>	(IntOpt) Pydev debug server port (5678 by default).
[profiler]	
enabled = <i>False</i>	(BoolOpt) If False fully disable profiling feature.
trace_sqlalchemy = <i>True</i>	(BoolOpt) If False doesn't trace SQL requests.

Table 5.10. Description of DNS configuration options

Configuration option = Default value	Description
[DEFAULT]	
dns_account_id =	(StrOpt) Tenant ID for DNSaaS.
dns_auth_url =	(StrOpt) Authentication URL for DNSaaS.
dns_domain_id =	(StrOpt) Domain ID used for adding DNS entries.
dns_domain_name =	(StrOpt) Domain name used for adding DNS entries.
dns_driver = <i>trove.dns.driver.DnsDriver</i>	(StrOpt) Driver for DNSaaS.
dns_endpoint_url = <i>0.0.0.0</i>	(StrOpt) Endpoint URL for DNSaaS.
dns_hostname =	(StrOpt) Hostname used for adding DNS entries.
dns_instance_entry_factory = <i>trove.dns.driver.DnsInstanceEntryFactory</i>	(StrOpt) Factory for adding DNS entries.
dns_management_base_url =	(StrOpt) Management URL for DNSaaS.
dns_passkey =	(StrOpt) Passkey for DNSaaS.
dns_region =	(StrOpt) Region name for DNSaaS.
dns_service_type =	(StrOpt) Service Type for DNSaaS.
dns_time_out = <i>120</i>	(IntOpt) Maximum time (in seconds) to wait for a DNS entry add.

Configuration option = Default value	Description
dns_ttl = 300	(IntOpt) Time (in seconds) before a refresh of DNS information occurs.
dns_username =	(StrOpt) Username for DNSaaS.
trove_dns_support = <i>False</i>	(BoolOpt) Whether Trove should add DNS entries on create (using Designate DNSaaS).

Table 5.11. Description of guest agent configuration options

Configuration option = Default value	Description
[DEFAULT]	
agent_call_high_timeout = 60	(IntOpt) Maximum time (in seconds) to wait for Guest Agent 'slow' requests (such as restarting the database).
agent_call_low_timeout = 5	(IntOpt) Maximum time (in seconds) to wait for Guest Agent 'quick' requests (such as retrieving a list of users or databases).
agent_heartbeat_expiry = 60	(IntOpt) Time (in seconds) after which a guest is considered unreachable
agent_heartbeat_time = 10	(IntOpt) Maximum time (in seconds) for the Guest Agent to reply to a heartbeat request.
agent_replication_snapshot_timeout = 36000	(IntOpt) Maximum time (in seconds) to wait for taking a Guest Agent replication snapshot.
guest_config = <i>/etc/trove/trove-guestagent.conf</i>	(StrOpt) Path to the Guest Agent config file to be injected during instance creation.
guest_id = <i>None</i>	(StrOpt) ID of the Guest Instance.
guest_info = <i>guest_info.conf</i>	(StrOpt) The guest info filename found in the injected config location. If a full path is specified then it will be used as the path to the guest info file
ignore_dbs = <i>mysql, information_schema, performance_schema</i>	(ListOpt) Databases to exclude when listing databases.
ignore_users = <i>os_admin, root</i>	(ListOpt) Users to exclude when listing users.
mount_options = <i>defaults,noatime</i>	(StrOpt) Options to use when mounting a volume.

Configuration option = Default value	Description
storage_namespace = <i>trove.guestagent.strategies.storage.swift</i>	(StrOpt) Namespace to load the default storage strategy from.
storage_strategy = <i>SwiftStorage</i>	(StrOpt) Default strategy to store backups.
usage_sleep_time = 5	(IntOpt) Time to sleep during the check for an active Guest.

Table 5.12. Description of Orchestration module configuration options

Configuration option = Default value	Description
[DEFAULT]	
heat_endpoint_type = <i>publicURL</i>	(StrOpt) Service endpoint type to use when searching catalog.
heat_service_type = <i>orchestration</i>	(StrOpt) Service type to use when searching catalog.
heat_time_out = 60	(IntOpt) Maximum time (in seconds) to wait for a Heat request to complete.
heat_url = <i>None</i>	(StrOpt) URL without the tenant segment.

Table 5.13. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(BoolOpt) Print debugging output (set logging level to DEBUG instead of default INFO level).
default_log_levels = <i>amqp=WARN, amqpplib=WARN, boto=WARN, qpidd=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystone.middleware=WARN, routes.middleware=WARN, stevedore=WARN, taskflow=WARN</i>	(ListOpt) List of logger=LEVEL pairs. This option is ignored if log_config_append is set.
fatal_deprecations = <i>False</i>	(BoolOpt) Enables or disables fatal status of deprecations.

Configuration option = Default value	Description
format_options = <i>-m 5</i>	(StrOpt) Options to use when formatting a volume.
instance_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, log_format).
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(StrOpt) Format string for <code>%(asctime)s</code> in log records. Default: <code>%(default)s</code> . This option is ignored if log_config_append is set.
log_dir = <i>None</i>	(StrOpt) (Optional) The base directory used for relative <code>--log-file</code> paths. This option is ignored if log_config_append is set.
log_file = <i>None</i>	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout. This option is ignored if log_config_append is set.
log_format = <i>None</i>	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecated, use logging_context_format_string and logging_default_format_string instead. This option is ignored if log_config_append is set.
logging_context_format_string = <i>%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%request_id)s %(user_identity)s] %(instance)s%(message)s</i>	(StrOpt) Format string to use for log messages with context.
logging_debug_format_suffix = <i>%(funcName)s %(pathname)s:%(lineno)d</i>	(StrOpt) Data to append to log format when level is DEBUG.
logging_default_format_string = <i>%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [-] %(instance)s%(message)s</i>	(StrOpt) Format string to use for log messages without context.

Configuration option = Default value	Description
logging_exception_prefix = <code>%(asctime)s.%(msecs)03d %(process)d ERROR %(name)s %(instance)s</code>	(StrOpt) Prefix each line of exception output with this format.
network_label_regex = <code>^private\$</code>	(StrOpt) Regular expression to match Trove network labels.
publish_errors = <code>False</code>	(BoolOpt) Enables or disables publication of error events.
syslog_log_facility = <code>LOG_USER</code>	(StrOpt) Syslog facility to receive log lines. This option is ignored if <code>log_config_append</code> is set.
use_stderr = <code>True</code>	(BoolOpt) Log output to standard error. This option is ignored if <code>log_config_append</code> is set.
use_syslog = <code>False</code>	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if <code>log_config_append</code> is set.
use_syslog_rfc_format = <code>True</code>	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in Kilo, and will be removed in Mitaka, along with this option. This option is ignored if <code>log_config_append</code> is set.
verbose = <code>True</code>	(BoolOpt) If set to false, will disable INFO logging level, making WARNING the default.
watch_log_file = <code>False</code>	(BoolOpt) (Optional) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if <code>log-file</code> option is specified and Linux platform is used. This option is ignored if <code>log_config_append</code> is set.

Table 5.14. Description of network configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
network_driver = <i>trove.network.nova.NovaNetwork</i>	(StrOpt) Describes the actual network manager used for the management of network attributes (security groups, floating IPs, etc.).
neutron_endpoint_type = <i>publicURL</i>	(StrOpt) Service endpoint type to use when searching catalog.
neutron_service_type = <i>network</i>	(StrOpt) Service type to use when searching catalog.
neutron_url = <i>None</i>	(StrOpt) URL without the tenant segment.

Table 5.15. Description of nova configuration options

Configuration option = Default value	Description
[DEFAULT]	
nova_proxy_admin_pass =	(StrOpt) Admin password used to connect to Nova.
nova_proxy_admin_tenant_id =	(StrOpt) Admin tenant ID used to connect to Nova.
nova_proxy_admin_tenant_name =	(StrOpt) Admin tenant name used to connect to Nova.
nova_proxy_admin_user =	(StrOpt) Admin username used to connect to Nova.

Table 5.16. Description of quota configuration options

Configuration option = Default value	Description
[DEFAULT]	
max_accepted_volume_size = 5	(IntOpt) Default maximum volume size (in GB) for an instance.
max_backups_per_user = 50	(IntOpt) Default maximum number of backups created by a tenant.
max_instances_per_user = 5	(IntOpt) Default maximum number of instances per tenant.
max_volumes_per_user = 20	(IntOpt) Default maximum volume capacity (in GB) spanning across all Trove volumes per tenant.
quota_driver = <i>trove.quota.quota.DbQuotaDriver</i>	(StrOpt) Default driver to use for quota checks.

Table 5.17. Description of Redis configuration options

Configuration option = Default value	Description
[DEFAULT]	
password =	(StrOpt) Password for Redis server (optional).
port = 6379	(IntOpt) Use this port to connect to redis host.
[matchmaker_redis]	
host = 127.0.0.1	(StrOpt) Host to locate redis.
password =	(StrOpt) Password for Redis server (optional).
port = 6379	(IntOpt) Use this port to connect to redis host.

Table 5.18. Description of swift configuration options

Configuration option = Default value	Description
[DEFAULT]	
swift_endpoint_type = <i>publicURL</i>	(StrOpt) Service endpoint type to use when searching catalog.
swift_service_type = <i>object-store</i>	(StrOpt) Service type to use when searching catalog.
swift_url = <i>None</i>	(StrOpt) URL ending in AUTH_.

Table 5.19. Description of taskmanager configuration options

Configuration option = Default value	Description
[DEFAULT]	
cloudinit_location = <i>/etc/trove/cloudinit</i>	(StrOpt) Path to folder with cloudinit scripts.
datastore_manager = <i>None</i>	(StrOpt) Manager class in the Guest Agent, set up by the Taskmanager on instance provision.
datastore_registry_ext = {}	(DictOpt) Extension for default datastore managers. Allows the use of custom managers for each of the datastores supported by Trove.
exists_notification_interval = 3600	(IntOpt) Seconds to wait between pushing events.

Configuration option = Default value	Description
exists_notification_transformer = <i>None</i>	(StrOpt) Transformer for exists notifications.
reboot_time_out = <i>120</i>	(IntOpt) Maximum time (in seconds) to wait for a server reboot.
resize_time_out = <i>600</i>	(IntOpt) Maximum time (in seconds) to wait for a server resize.
restore_usage_timeout = <i>36000</i>	(IntOpt) Maximum time (in seconds) to wait for a Guest instance restored from a backup to become active.
revert_time_out = <i>600</i>	(IntOpt) Maximum time (in seconds) to wait for a server resize revert.
server_delete_time_out = <i>60</i>	(IntOpt) Maximum time (in seconds) to wait for a server delete.
state_change_wait_time = <i>180</i>	(IntOpt) Maximum time (in seconds) to wait for a state change.
update_status_on_fail = <i>True</i>	(BoolOpt) Set the service and instance task statuses to ERROR when an instance fails to become active within the configured usage_timeout.
usage_sleep_time = <i>5</i>	(IntOpt) Time to sleep during the check for an active Guest.
use_heat = <i>False</i>	(BoolOpt) Use Heat for provisioning.
use_nova_server_config_drive = <i>False</i>	(BoolOpt) Use config drive for file injection when booting instance.
use_nova_server_volume = <i>False</i>	(BoolOpt) Whether to provision a Cinder volume for the Nova instance.
verify_swift_checksum_on_restore = <i>True</i>	(BoolOpt) Enable verification of Swift checksum before starting restore. Makes sure the checksum of original backup matches the checksum of the Swift backup file.

Table 5.20. Description of upgrades configuration options

Configuration option = Default value	Description
[upgrade_levels]	

Configuration option = Default value	Description
conductor = <i>icehouse</i>	(StrOpt) Set a version cap for messages sent to conductor services
guestagent = <i>icehouse</i>	(StrOpt) Set a version cap for messages sent to guestagent services
taskmanager = <i>icehouse</i>	(StrOpt) Set a version cap for messages sent to taskmanager services

Table 5.21. Description of volume configuration options

Configuration option = Default value	Description
[DEFAULT]	
block_device_mapping = <i>vdb</i>	(StrOpt) Block device to map onto the created instance.
cinder_endpoint_type = <i>publicURL</i>	(StrOpt) Service endpoint type to use when searching catalog.
cinder_service_type = <i>volumev2</i>	(StrOpt) Service type to use when searching catalog.
cinder_url = <i>None</i>	(StrOpt) URL without the tenant segment.
cinder_volume_type = <i>None</i>	(StrOpt) Volume type to use when provisioning a Cinder volume.
device_path = <i>/dev/vdb</i>	(StrOpt) Device path for volume if volume support is enabled.
trove_volume_support = <i>True</i>	(BoolOpt) Whether to provision a Cinder volume for datadir.
volume_format_timeout = <i>120</i>	(IntOpt) Maximum time (in seconds) to wait for a volume format.
volume_fstype = <i>ext3</i>	(StrOpt) File system type used to format a volume.
volume_time_out = <i>60</i>	(IntOpt) Maximum time (in seconds) to wait for a volume attach.

5.1. CONFIGURE THE DATABASE

Use the options to configure the used databases:

Table 5.22. Description of MariaDB database configuration options

Configuration option = Default value	Description
[mariadb]	
backup_incremental_strategy = <i>{'InnoDBBackupEx': 'InnoDBBackupExIncremental'}</i>	(DictOpt) Incremental Backup Runner based on the default strategy. For strategies that do not implement an incremental backup, the runner will use the default full backup.
backup_namespace = <i>trove.guestagent.strategies.backup.mysql_impl</i>	(StrOpt) Namespace to load backup strategies from.
backup_strategy = <i>InnoDBBackupEx</i>	(StrOpt) Default strategy to perform backups.
device_path = <i>/dev/vdb</i>	(StrOpt) Device path for volume if volume support is enabled.
mount_point = <i>/var/lib/mysql</i>	(StrOpt) Filesystem path for mounting volumes if volume support is enabled.
replication_namespace = <i>trove.guestagent.strategies.replication.mysql_binlog</i>	(StrOpt) Namespace to load replication strategies from.
replication_strategy = <i>MysqlBinlogReplication</i>	(StrOpt) Default strategy for replication.
restore_namespace = <i>trove.guestagent.strategies.restore.mysql_impl</i>	(StrOpt) Namespace to load restore strategies from.
root_controller = <i>trove.extensions.common.service.DefaultRootController</i>	(StrOpt) Root controller implementation for mysql.
root_on_create = <i>False</i>	(BoolOpt) Enable the automatic creation of the root user for the service during instance-create. The generated password for the root user is immediately returned in the response of instance-create as the 'password' field.
tcp_ports = <i>3306</i>	(ListOpt) List of TCP ports and/or port ranges to open in the security group (only applicable if <i>trove_security_groups_support</i> is True).
udp_ports =	(ListOpt) List of UDP ports and/or port ranges to open in the security group (only applicable if <i>trove_security_groups_support</i> is True).
usage_timeout = <i>400</i>	(IntOpt) Maximum time (in seconds) to wait for a Guest to become active.

Configuration option = Default value	Description
volume_support = <i>True</i>	(BoolOpt) Whether to provision a Cinder volume for datadir.

Table 5.23. Description of MySQL database configuration options

Configuration option = Default value	Description
[mysql]	
backup_incremental_strategy = <i>{'InnoDBBackupEx': 'InnoDBBackupExIncremental'}</i>	(DictOpt) Incremental Backup Runner based on the default strategy. For strategies that do not implement an incremental backup, the runner will use the default full backup.
backup_namespace = <i>trove.guestagent.strategies.backup.mysql_impl</i>	(StrOpt) Namespace to load backup strategies from.
backup_strategy = <i>InnoDBBackupEx</i>	(StrOpt) Default strategy to perform backups.
device_path = <i>/dev/vdb</i>	(StrOpt) Device path for volume if volume support is enabled.
mount_point = <i>/var/lib/mysql</i>	(StrOpt) Filesystem path for mounting volumes if volume support is enabled.
replication_namespace = <i>trove.guestagent.strategies.replication.mysql_gtid</i>	(StrOpt) Namespace to load replication strategies from.
replication_strategy = <i>MySQLGTIDReplication</i>	(StrOpt) Default strategy for replication.
restore_namespace = <i>trove.guestagent.strategies.restore.mysql_impl</i>	(StrOpt) Namespace to load restore strategies from.
root_controller = <i>trove.extensions.common.service.DefaultRootController</i>	(StrOpt) Root controller implementation for mysql.
root_on_create = <i>False</i>	(BoolOpt) Enable the automatic creation of the root user for the service during instance-create. The generated password for the root user is immediately returned in the response of instance-create as the 'password' field.
tcp_ports = <i>3306</i>	(ListOpt) List of TCP ports and/or port ranges to open in the security group (only applicable if <i>trove_security_groups_support</i> is True).

Configuration option = Default value	Description
udp_ports =	(ListOpt) List of UDP ports and/or port ranges to open in the security group (only applicable if <code>trove_security_groups_support</code> is <code>True</code>).
usage_timeout = 400	(IntOpt) Maximum time (in seconds) to wait for a Guest to become active.
volume_support = True	(BoolOpt) Whether to provision a Cinder volume for datadir.

5.2. CONFIGURE THE RPC MESSAGING SYSTEM

OpenStack projects use an open standard for messaging middleware known as AMQP. This messaging middleware enables the OpenStack services that run on multiple servers to talk to each other. OpenStack Trove RPC supports two implementations of AMQP: **RabbitMQ** and **Qpid**.

5.2.1. Configure RabbitMQ

Use these options to configure the **RabbitMQ** messaging system:

Table 5.24. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = False	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = False	(BoolOpt) Use durable queues in AMQP.
fake_rabbit = False	(BoolOpt) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
heartbeat_rate = 2	(IntOpt) How often times during the <code>heartbeat_timeout_threshold</code> we check the heartbeat.
heartbeat_timeout_threshold = 60	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disables the heartbeat). EXPERIMENTAL
kombu_reconnect_delay = 1.0	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_reconnect_timeout = 60	(IntOpt) How long to wait before considering a reconnect attempt to have failed. This value should not be longer than <code>rpc_response_timeout</code> .

Configuration option = Default value	Description
kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
rabbit_ha_queues = False	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
rabbit_host = localhost	(StrOpt) The RabbitMQ broker address where a single node is used.
rabbit_hosts = \$rabbit_host:\$rabbit_port	(ListOpt) RabbitMQ HA cluster host:port pairs.
rabbit_login_method = AMQPLAIN	(StrOpt) The RabbitMQ login method.
rabbit_max_retries = 0	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = guest	(StrOpt) The RabbitMQ password.
rabbit_port = 5672	(IntOpt) The RabbitMQ broker port where a single node is used.
rabbit_retry_backoff = 2	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
rabbit_retry_interval = 1	(IntOpt) How frequently to retry connecting with RabbitMQ.
rabbit_use_ssl = False	(BoolOpt) Connect over SSL for RabbitMQ.
rabbit_userid = guest	(StrOpt) The RabbitMQ userid.
rabbit_virtual_host = /	(StrOpt) The RabbitMQ virtual host.

Configuration option = Default value	Description
<code>send_single_reply = False</code>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies: first one with the payload, a second one to ensure the other has finished to send the payload. This option defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

5.2.2. Configure Qpid

Use these options to configure the Qpid messaging system:

Table 5.25. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
<code>amqp_auto_delete = False</code>	(BoolOpt) Auto-delete queues in AMQP.
<code>amqp_durable_queues = False</code>	(BoolOpt) Use durable queues in AMQP.
<code>qpid_heartbeat = 60</code>	(IntOpt) Seconds between connection keepalive heartbeats.
<code>qpid_hostname = localhost</code>	(StrOpt) Qpid broker hostname.
<code>qpid_hosts = \$qpid_hostname:\$qpid_port</code>	(ListOpt) Qpid HA cluster host:port pairs.
<code>qpid_password =</code>	(StrOpt) Password for Qpid connection.
<code>qpid_port = 5672</code>	(IntOpt) Qpid broker port.
<code>qpid_protocol = tcp</code>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
<code>qpid_receiver_capacity = 1</code>	(IntOpt) The number of prefetched messages held by receiver.
<code>qpid_sasl_mechanisms =</code>	(StrOpt) Space separated list of SASL mechanisms to use for auth.
<code>qpid_tcp_nodelay = True</code>	(BoolOpt) Whether to disable the Nagle algorithm.

Configuration option = Default value	Description
qpid_topology_version = 1	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
qpid_username =	(StrOpt) Username for Qpid connection.
send_single_reply = False	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies: first one with the payload, a second one to ensure the other has finished to send the payload. This option defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

5.2.3. Configure messaging

Use these common options to configure the **RabbitMQ**, and **Qpid** messaging drivers:

Table 5.26. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
conductor_manager = <i>trove.conductor.manager.Manager</i>	(StrOpt) Qualified class name to use for conductor manager.
conductor_queue = <i>trove-conductor</i>	(StrOpt) Message queue name the Conductor will listen on.
control_exchange = <i>openstack</i>	(StrOpt) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
notification_driver = []	(MultiStrOpt) The Drivers(s) to handle sending notifications. Possible values are messaging , messagingv2 , routing , log , test , and noop .
notification_service_id = {'mysql': '2f3ff068-2bfb-4f70-9a9d-a6bb65bc084b', 'mariadb': '7a4f82cc-10d2-4bc6-aadc-d9aacc2a3cb5'}	(DictOpt) Unique ID to tag notification events.
notification_topics = <i>notifications</i>	(ListOpt) AMQP topic used for OpenStack notifications.

Configuration option = Default value	Description
transport_url = <i>None</i>	(StrOpt) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the <code>rpc_backend</code> option and driver specific configuration.

Table 5.27. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
num_tries = 3	(IntOpt) Number of times to check if a volume exists.
report_interval = 30	(IntOpt) The interval (in seconds) which periodic tasks are run.
rpc_backend = <i>rabbit</i>	(StrOpt) The messaging driver to use, defaults to rabbit. Other drivers include qpid and zmq.
rpc_cast_timeout = 30	(IntOpt) Seconds to wait before a cast expires (TTL). Only supported by impl_zmq.
rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
rpc_poll_timeout = 1	(IntOpt) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
rpc_response_timeout = 60	(IntOpt) Seconds to wait for a response from a call.
[oslo_concurrency]	
disable_process_locking = <i>False</i>	(BoolOpt) Enables or disables inter-process locks.
lock_path = <i>None</i>	(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable <code>OSLO_LOCK_PATH</code> . If external locks are used, a lock path must be set.
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(BoolOpt) Accept clients using either SSL or plain TCP

Configuration option = Default value	Description
broadcast_prefix = <i>broadcast</i>	(StrOpt) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(StrOpt) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(StrOpt) address prefix when sending to any server in group
idle_timeout = <i>0</i>	(IntOpt) Timeout for inactive connections (in seconds)
password =	(StrOpt) Password for message broker authentication
sasl_config_dir =	(StrOpt) Path to directory that contains the SASL configuration
sasl_config_name =	(StrOpt) Name of configuration file (without .conf suffix)
sasl_mechanisms =	(StrOpt) Space separated list of acceptable SASL mechanisms
server_request_prefix = <i>exclusive</i>	(StrOpt) address prefix used when sending to a specific server
ssl_ca_file =	(StrOpt) CA certificate PEM file to verify server certificate
ssl_cert_file =	(StrOpt) Identifying certificate PEM file to present to clients
ssl_key_file =	(StrOpt) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(StrOpt) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>	(BoolOpt) Debug: dump AMQP frames to stdout
username =	(StrOpt) User name for message broker authentication

5.3. NEW, UPDATED AND DEPRECATED OPTIONS IN LIBERTY FOR DATABASE SERVICE

Table 5.28. New options

Option = default value	(Type) Help string
[DEFAULT] executor_thread_pool_size = 64	(IntOpt) Size of executor thread pool.
[DEFAULT] exists_notification_interval = 3600	(IntOpt) Seconds to wait between pushing events.
[DEFAULT] nova_proxy_admin_tenant_id =	(StrOpt) Admin tenant ID used to connect to Nova.
[DEFAULT] password =	(StrOpt) Password for Redis server (optional).
[DEFAULT] port = 6379	(IntOpt) Use this port to connect to redis host.
[DEFAULT] rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
[DEFAULT] rpc_poll_timeout = 1	(IntOpt) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
[DEFAULT] rpc_zmq_all_req_rep = True	(BoolOpt) Use REQ/REP pattern for all methods CALL/CAST/FANOUT.
[DEFAULT] rpc_zmq_concurrency = eventlet	(StrOpt) Type of concurrency used. Either "native" or "eventlet"
[DEFAULT] timeout_wait_for_service = 120	(IntOpt) Maximum time (in seconds) to wait for a service to become alive.
[DEFAULT] watch_log_file = False	(BoolOpt) (Optional) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log-file option is specified and Linux platform is used. This option is ignored if log_config_append is set.
[DEFAULT] zmq_use_broker = True	(BoolOpt) Shows whether zmq-messaging uses broker or not.
[keystone_authtoken] region_name = None	(StrOpt) The region in which the identity server can be found.
[mariadb] backup_incremental_strategy = {'InnoDBBackupEx': 'InnoDBBackupExIncremental'}	(DictOpt) Incremental Backup Runner based on the default strategy. For strategies that do not implement an incremental backup, the runner will use the default full backup.
[mariadb] backup_namespace = trove.guestagent.strategies.backup.mysql_impl	(StrOpt) Namespace to load backup strategies from.

Option = default value	(Type) Help string
[mariadb] backup_strategy = InnoDBBackupEx	(StrOpt) Default strategy to perform backups.
[mariadb] device_path = /dev/vdb	(StrOpt) Device path for volume if volume support is enabled.
[mariadb] mount_point = /var/lib/mysql	(StrOpt) Filesystem path for mounting volumes if volume support is enabled.
[mariadb] replication_namespace = trove.guestagent.strategies.replication.mysql_binlog	(StrOpt) Namespace to load replication strategies from.
[mariadb] replication_strategy = MySQLBinlogReplication	(StrOpt) Default strategy for replication.
[mariadb] restore_namespace = trove.guestagent.strategies.restore.mysql_impl	(StrOpt) Namespace to load restore strategies from.
[mariadb] root_controller = trove.extensions.common.service.DefaultRootController	(StrOpt) Root controller implementation for mysql.
[mariadb] root_on_create = False	(BoolOpt) Enable the automatic creation of the root user for the service during instance-create. The generated password for the root user is immediately returned in the response of instance-create as the 'password' field.
[mariadb] tcp_ports = 3306	(ListOpt) List of TCP ports and/or port ranges to open in the security group (only applicable if trove_security_groups_support is True).
[mariadb] udp_ports =	(ListOpt) List of UDP ports and/or port ranges to open in the security group (only applicable if trove_security_groups_support is True).
[mariadb] usage_timeout = 400	(IntOpt) Maximum time (in seconds) to wait for a Guest to become active.
[mariadb] volume_support = True	(BoolOpt) Whether to provision a Cinder volume for datadir.
[mysql] root_controller = trove.extensions.common.service.DefaultRootController	(StrOpt) Root controller implementation for mysql.
[oslo_messaging_amqp] password =	(StrOpt) Password for message broker authentication

Option = default value	(Type) Help string
[oslo_messaging_amqp] sasl_config_dir =	(StrOpt) Path to directory that contains the SASL configuration
[oslo_messaging_amqp] sasl_config_name =	(StrOpt) Name of configuration file (without .conf suffix)
[oslo_messaging_amqp] sasl_mechanisms =	(StrOpt) Space separated list of acceptable SASL mechanisms
[oslo_messaging_amqp] username =	(StrOpt) User name for message broker authentication
[oslo_messaging_qpid] send_single_reply = False	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>
[oslo_messaging_rabbit] kombu_reconnect_timeout = 60	(IntOpt) How long to wait before considering a reconnect attempt to have failed. This value should not be longer than rpc_response_timeout.
[oslo_messaging_rabbit] send_single_reply = False	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>
[pxc] api_strategy = trove.common.strategies.cluster.experimental.pxc.api.PXCAPIStrategy	(StrOpt) Class that implements datastore-specific API logic.
[pxc] backup_incremental_strategy = {'InnoBackupEx': 'InnoBackupExIncremental'}	(DictOpt) Incremental Backup Runner based on the default strategy. For strategies that do not implement an incremental backup, the runner will use the default full backup.

Option = default value	(Type) Help string
[pxc] backup_namespace = trove.guestagent.strategies.backup.mysql_impl	(StrOpt) Namespace to load backup strategies from.
[pxc] backup_strategy = InnoDBBackupEx	(StrOpt) Default strategy to perform backups.
[pxc] cluster_support = True	(BoolOpt) Enable clusters to be created and managed.
[pxc] device_path = /dev/vdb	(StrOpt) Device path for volume if volume support is enabled.
[pxc] guestagent_strategy = trove.common.strategies.cluster.experimental.pxc.guestagent.PXCGuestAgentStrategy	(StrOpt) Class that implements datastore-specific Guest Agent API logic.
[pxc] ignore_users = os_admin, root, clusterrepuser	(ListOpt) Users to exclude when listing users.
[pxc] min_cluster_member_count = 3	(IntOpt) Minimum number of members in PXC cluster.
[pxc] mount_point = /var/lib/mysql	(StrOpt) Filesystem path for mounting volumes if volume support is enabled.
[pxc] replication_namespace = trove.guestagent.strategies.replication.mysql_gtid	(StrOpt) Namespace to load replication strategies from.
[pxc] replication_strategy = MySQLGTIDReplication	(StrOpt) Default strategy for replication.
[pxc] replication_user = slave_user	(StrOpt) Userid for replication slave.
[pxc] restore_namespace = trove.guestagent.strategies.restore.mysql_impl	(StrOpt) Namespace to load restore strategies from.
[pxc] root_controller = trove.extensions.common.service.DefaultRootController	(StrOpt) Root controller implementation for pxc.
[pxc] root_on_create = False	(BoolOpt) Enable the automatic creation of the root user for the service during instance-create. The generated password for the root user is immediately returned in the response of instance-create as the 'password' field.
[pxc] taskmanager_strategy = trove.common.strategies.cluster.experimental.pxc.taskmanager.PXCTaskManagerStrategy	(StrOpt) Class that implements datastore-specific task manager logic.

Option = default value	(Type) Help string
[pxc] tcp_ports = 3306, 4444, 4567, 4568	(ListOpt) List of TCP ports and/or port ranges to open in the security group (only applicable if trove_security_groups_support is True).
[pxc] udp_ports =	(ListOpt) List of UDP ports and/or port ranges to open in the security group (only applicable if trove_security_groups_support is True).
[pxc] usage_timeout = 450	(IntOpt) Maximum time (in seconds) to wait for a Guest to become active.
[pxc] volume_support = True	(BoolOpt) Whether to provision a Cinder volume for datadir.
[redis] api_strategy = trove.common.strategies.cluster.experimental.redis.api.RedisAPIStrategy	(StrOpt) Class that implements datastore-specific API logic.
[redis] cluster_support = True	(BoolOpt) Enable clusters to be created and managed.
[redis] guestagent_strategy = trove.common.strategies.cluster.experimental.redis.guestagent.RedisGuestAgentStrategy	(StrOpt) Class that implements datastore-specific Guest Agent API logic.
[redis] replication_namespace = trove.guestagent.strategies.replication.experimental.redis_sync	(StrOpt) Namespace to load replication strategies from.
[redis] root_controller = trove.extensions.common.service.DefaultRootController	(StrOpt) Root controller implementation for redis.
[redis] taskmanager_strategy = trove.common.strategies.cluster.experimental.redis.taskmanager.RedisTaskManagerStrategy	(StrOpt) Class that implements datastore-specific task manager logic.

Table 5.29. New default values

Option	Previous default value	New default value
[DEFAULT] cluster_usage_timeout	675	36000

Option	Previous default value	New default value
[DEFAULT] default_log_levels	amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.conne ctionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, keystonemiddleware=WARN, routes.middleware=WARN, stevedore=WARN	amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.conne ctionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retr y=WARN, urllib3.util.retry=WARN, keystonemiddleware=WARN, routes.middleware=WARN, stevedore=WARN, taskflow=WARN
[DEFAULT] ignore_dbs	lost+found, #mysql50#lost+found, mysql, information_schema	mysql, information_schema, performance_schema
[DEFAULT] logging_exception_prefix	%(asctime)s.%(msecs)03d % (process)d TRACE %(name)s % (instance)s	%(asctime)s.%(msecs)03d % (process)d ERROR %(name)s % (instance)s
[DEFAULT] notification_service_id	{'vertica': 'a8d805ae-a3b2-c4fd- gb23-b62cee5201ae', 'db2': 'e040cd37-263d-4869-aaa6- c62aa97523b5', 'postgres': 'ac277e0d-4f21-40aa-b347- 1ea31e571720', 'mysql': '2f3ff068-2bfb-4f70-9a9d- a6bb65bc084b', 'couchbase': 'fa62fe68-74d9-4779-a24e- 36f19602c415', 'mongodb': 'c8c907af-7375-456f-b929- b637ff9209ee', 'couchdb': 'f0a9ab7b-66f7-4352-93d7- 071521d44c7c', 'redis': 'b216ffc5- 1947-456c-a4cf-70f94c05f7d0', 'cassandra': '459a230d-4e97- 4344-9067-2a54a310b0ed'}	{'mongodb': 'c8c907af-7375- 456f-b929-b637ff9209ee', 'percona': 'fd1723f5-68d2-409c- 994f-a4a197892a17', 'mysql': '2f3ff068-2bfb-4f70-9a9d- a6bb65bc084b', 'pxc': '75a628c3-f81b-4ffb-b10a- 4087c26bc854', 'db2': 'e040cd37-263d-4869-aaa6- c62aa97523b5', 'cassandra': '459a230d-4e97-4344-9067- 2a54a310b0ed', 'mariadb': '7a4f82cc-10d2-4bc6-aadc- d9aacc2a3cb5', 'postgres': 'ac277e0d-4f21-40aa-b347- 1ea31e571720', 'couchbase': 'fa62fe68-74d9-4779-a24e- 36f19602c415', 'couchdb': 'f0a9ab7b-66f7-4352-93d7- 071521d44c7c', 'redis': 'b216ffc5- 1947-456c-a4cf-70f94c05f7d0', 'vertica': 'a8d805ae-a3b2-c4fd- gb23-b62cee5201ae'}
[DEFAULT] report_interval	10	30

Option	Previous default value	New default value
[DEFAULT] rpc_zmq_matchmaker	local	redis
[DEFAULT] usage_timeout	600	900
[DEFAULT] use_syslog_rfc_format	False	True
[DEFAULT] verbose	False	True
[matchmaker_redis] password	None	
[oslo_messaging_rabbit] heartbeat_timeout_threshold	0	60
[redis] backup_namespace	None	trove.guestagent.strategies.back up.experimental.redis_impl
[redis] backup_strategy	None	RedisBackup
[redis] replication_strategy	None	RedisSyncReplication
[redis] restore_namespace	None	trove.guestagent.strategies.resto re.experimental.redis_impl
[redis] tcp_ports	6379	6379,16379
[redis] volume_support	False	True

Table 5.30. Deprecated options

Deprecated option	New Option
[DEFAULT] use_syslog	None
[DEFAULT] rpc_thread_pool_size	[DEFAULT] executor_thread_pool_size
[DEFAULT] log_format	None

CHAPTER 6. DATA PROCESSING SERVICE

The Data processing service (sahara) provides a scalable data-processing stack and associated management interfaces.

The following tables provide a comprehensive list of the Data processing service configuration options.

Table 6.1. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>openstack</i>	(String) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
transport_url = <i>None</i>	(String) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the <code>rpc_backend</code> option and driver specific configuration.

Table 6.2. Description of API configuration options

Configuration option = Default value	Description
[oslo_middleware]	
max_request_body_size = <i>114688</i>	(Integer) The maximum body size for each request, in bytes.
secure_proxy_ssl_header = <i>X-Forwarded-Proto</i>	(String) DEPRECATED: The HTTP Header that will be used to determine what the original request protocol scheme was, even if it was hidden by an SSL termination proxy.
[retries]	
retries_number = <i>5</i>	(Integer) Number of times to retry the request to client before failing
retry_after = <i>10</i>	(Integer) Time between the retries to client (in seconds).
[service_auth]	
admin_password = <i>password</i>	(String) The service admin password
admin_project_domain = <i>admin</i>	(String) The admin project domain name

Configuration option = Default value	Description
admin_tenant_name = <i>admin</i>	(String) The service admin tenant name
admin_user = <i>admin</i>	(String) The service admin user name
admin_user_domain = <i>admin</i>	(String) The admin user domain name
auth_url = <i>http://127.0.0.1:5000/v2.0</i>	(String) Authentication endpoint
auth_version = <i>2</i>	(String) The auth version used to authenticate
endpoint_type = <i>public</i>	(String) The endpoint_type to be used
region = <i>RegionOne</i>	(String) The deployment region
service_name = <i>Ibaas</i>	(String) The name of the service

Table 6.3. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(String) Service user password.
admin_tenant_name = <i>admin</i>	(String) Service tenant name.
admin_token = <i>None</i>	(String) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(String) Service username.
auth_admin_prefix =	(String) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
auth_host = <i>127.0.0.1</i>	(String) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_port = <i>35357</i>	(Integer) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .

Configuration option = Default value	Description
auth_protocol = <i>https</i>	(String) Protocol of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_section = <i>None</i>	(Unknown) Config Section from which to load plugin specific options
auth_type = <i>None</i>	(Unknown) Authentication type to load
auth_uri = <i>None</i>	(String) Complete public Identity API endpoint.
auth_version = <i>None</i>	(String) API version of the admin Identity API endpoint.
cache = <i>None</i>	(String) Env key for the swift cache.
cafile = <i>None</i>	(String) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(String) Required if identity server requires client certificate
check_revocations_for_cached = <i>False</i>	(Boolean) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.
delay_auth_decision = <i>False</i>	(Boolean) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(String) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.

Configuration option = Default value	Description
hash_algorithms = <i>md5</i>	(List) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard <code>hashlib.new()</code> . The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(Integer) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(Integer) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(String) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. <code>https://localhost:35357/</code>
include_service_catalog = <i>True</i>	(Boolean) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(Boolean) Verify HTTPS connections.
keyfile = <i>None</i>	(String) Required if identity server requires client certificate
memcache_pool_conn_get_timeout = <i>10</i>	(Integer) (Optional) Number of seconds that an operation will wait to get a memcached client connection from the pool.
memcache_pool_dead_retry = <i>300</i>	(Integer) (Optional) Number of seconds memcached server is considered dead before it is tried again.
memcache_pool_maxsize = <i>10</i>	(Integer) (Optional) Maximum total number of open connections to every memcached server.
memcache_pool_socket_timeout = <i>3</i>	(Integer) (Optional) Socket timeout in seconds for communicating with a memcached server.
memcache_pool_unused_timeout = <i>60</i>	(Integer) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.

Configuration option = Default value	Description
<code>memcache_secret_key = None</code>	(String) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
<code>memcache_security_strategy = None</code>	(String) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.
<code>memcache_use_advanced_pool = False</code>	(Boolean) (Optional) Use the advanced (eventlet safe) memcached client pool. The advanced pool will only work under python 2.x.
<code>region_name = None</code>	(String) The region in which the identity server can be found.
<code>revocation_cache_time = 10</code>	(Integer) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
<code>signing_dir = None</code>	(String) Directory used to cache files related to PKI tokens.
<code>token_cache_time = 300</code>	(Integer) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 6.4. Description of clients configuration options

Configuration option = Default value	Description
[cinder]	
<code>api_insecure = False</code>	(Boolean) Allow to perform insecure SSL requests to cinder.
<code>api_version = 2</code>	(Integer) Version of the Cinder API to use.
<code>ca_file = None</code>	(String) Location of ca certificates file to use for cinder client requests.

Configuration option = Default value	Description
endpoint_type = <i>internalURL</i>	(String) Endpoint type for cinder client requests
[heat]	
api_insecure = <i>False</i>	(Boolean) Allow to perform insecure SSL requests to heat.
ca_file = <i>None</i>	(String) Location of ca certificates file to use for heat client requests.
endpoint_type = <i>internalURL</i>	(String) Endpoint type for heat client requests
[keystone]	
api_insecure = <i>False</i>	(Boolean) Allow to perform insecure SSL requests to keystone.
ca_file = <i>None</i>	(String) Location of ca certificates file to use for keystone client requests.
endpoint_type = <i>internalURL</i>	(String) Endpoint type for keystone client requests
[manila]	
api_insecure = <i>True</i>	(Boolean) Allow to perform insecure SSL requests to manila.
api_version = <i>1</i>	(Integer) Version of the manila API to use.
ca_file = <i>None</i>	(String) Location of ca certificates file to use for manila client requests.
[neutron]	
api_insecure = <i>False</i>	(Boolean) Allow to perform insecure SSL requests to neutron.
ca_file = <i>None</i>	(String) Location of ca certificates file to use for neutron client requests.
endpoint_type = <i>internalURL</i>	(String) Endpoint type for neutron client requests
[nova]	
api_insecure = <i>False</i>	(Boolean) Allow to perform insecure SSL requests to nova.

Configuration option = Default value	Description
ca_file = <i>None</i>	(String) Location of ca certificates file to use for nova client requests.
endpoint_type = <i>internalURL</i>	(String) Endpoint type for nova client requests
[swift]	
api_insecure = <i>False</i>	(Boolean) Allow to perform insecure SSL requests to swift.
ca_file = <i>None</i>	(String) Location of ca certificates file to use for swift client requests.
endpoint_type = <i>internalURL</i>	(String) Endpoint type for swift client requests

Table 6.5. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
admin_project_domain_name = <i>default</i>	(String) The name of the domain for the service project(ex. tenant).
admin_user_domain_name = <i>default</i>	(String) The name of the domain to which the admin user belongs.
api_workers = <i>1</i>	(Integer) Number of workers for Sahara API service (0 means all-in-one-thread configuration).
cleanup_time_for_incomplete_clusters = <i>0</i>	(Integer) Maximal time (in hours) for clusters allowed to be in states other than "Active", "Deleting" or "Error". If a cluster is not in "Active", "Deleting" or "Error" state and last update of it was longer than "cleanup_time_for_incomplete_clusters" hours ago then it will be deleted automatically. (0 value means that automatic clean up is disabled).
cluster_remote_threshold = <i>70</i>	(Integer) The same as global_remote_threshold, but for a single cluster.
compute_topology_file = <i>etc/sahara/compute.topology</i>	(String) File with nova compute topology. It should contain mapping between nova computes and racks.

Configuration option = Default value	Description
coordinator_heartbeat_interval = 1	(Integer) Interval size between heartbeat execution in seconds. Heartbeats are executed to make sure that connection to the coordination server is active.
default_ntp_server = <i>pool.ntp.org</i>	(String) Default ntp server for time sync
disable_event_log = <i>False</i>	(Boolean) Disables event log feature.
enable_data_locality = <i>False</i>	(Boolean) Enables data locality for hadoop cluster. Also enables data locality for Swift used by hadoop. If enabled, 'compute_topology' and 'swift_topology' configuration parameters should point to OpenStack and Swift topology correspondingly.
enable_hypervisor_awareness = <i>True</i>	(Boolean) Enables four-level topology for data locality. Works only if corresponding plugin supports such mode.
executor_thread_pool_size = 64	(Integer) Size of executor thread pool.
global_remote_threshold = 100	(Integer) Maximum number of remote operations that will be running at the same time. Note that each remote operation requires its own process to run.
hash_ring_replicas_count = 40	(Integer) Number of points that belongs to each member on a hash ring. The larger number leads to a better distribution.
heat_enable_wait_condition = <i>True</i>	(Boolean) Enable wait condition feature to reduce polling during cluster creation
heat_stack_tags = <i>data-processing-cluster</i>	(List) List of tags to be used during operating with stack.
infrastructure_engine = <i>heat</i>	(String) DEPRECATED: An engine which will be used to provision infrastructure for Hadoop cluster.
job_binary_max_KB = 5120	(Integer) Maximum length of job binary data in kilobytes that may be stored or retrieved in a single operation.
job_canceling_timeout = 300	(Integer) Timeout for canceling job execution (in seconds). Sahara will try to cancel job execution during this time.
job_workflow_postfix =	(String) Postfix for storing jobs in hdfs. Will be added to '/user/<hdfs user>/' path.

Configuration option = Default value	Description
memcached_servers = <i>None</i>	(List) Memcached servers or None for in process cache.
min_transient_cluster_active_time = 30	(Integer) Minimal "lifetime" in seconds for a transient cluster. Cluster is guaranteed to be "alive" within this time period.
node_domain = <i>novalocal</i>	(String) The suffix of the node's FQDN. In nova-network that is the dhcp_domain config parameter.
os_region_name = <i>None</i>	(String) Region name used to get services endpoints.
periodic_coordinator_backend_url = <i>None</i>	(String) The backend URL to use for distributed periodic tasks coordination.
periodic_enable = <i>True</i>	(Boolean) Enable periodic tasks.
periodic_fuzzy_delay = 60	(Integer) Range in seconds to randomly delay when starting the periodic task scheduler to reduce stampeding. (Disable by setting to 0).
periodic_interval_max = 60	(Integer) Max interval size between periodic tasks execution in seconds.
periodic_workers_number = 1	(Integer) Number of threads to run periodic tasks.
plugins = <i>vanilla, spark, cdh, ambari</i>	(List) List of plugins to be loaded. Sahara preserves the order of the list when returning it.
proxy_command =	(String) Proxy command used to connect to instances. If set, this command should open a netcat socket, that Sahara will use for SSH and HTTP connections. Use {host} and {port} to describe the destination. Other available keywords: {tenant_id}, {network_id}, {router_id}.
remote = <i>ssh</i>	(String) A method for Sahara to execute commands on VMs.
rootwrap_command = <i>sudo sahara-rootwrap /etc/sahara/rootwrap.conf</i>	(String) Rootwrap command to leverage. Use in conjunction with use_rootwrap=True
swift_topology_file = <i>etc/sahara/swift.topology</i>	(String) File with Swift topology. It should contain mapping between Swift nodes and racks.
use_barbican_key_manager = <i>False</i>	(Boolean) Enable the usage of the OpenStack Key Management service provided by barbican.

Configuration option = Default value	Description
use_floating_ips = <i>True</i>	(Boolean) If set to True, Sahara will use floating IPs to communicate with instances. To make sure that all instances have floating IPs assigned in Nova Network set "auto_assign_floating_ip=True" in nova.conf. If Neutron is used for networking, make sure that all Node Groups have "floating_ip_pool" parameter defined.
use_identity_api_v3 = <i>True</i>	(Boolean) Enables Sahara to use Keystone API v3. If that flag is disabled, per-job clusters will not be terminated automatically.
use_namespaces = <i>False</i>	(Boolean) Use network namespaces for communication (only valid to use in conjunction with use_neutron=True).
use_neutron = <i>False</i>	(Boolean) Use Neutron Networking (False indicates the use of Nova networking).
use_rootwrap = <i>False</i>	(Boolean) Use rootwrap facility to allow non-root users to run the sahara-all server instance and access private network IPs (only valid to use in conjunction with use_namespaces=True)
[castellan]	
barbican_api_endpoint = <i>None</i>	(String) The endpoint to use for connecting to the barbican api controller. By default, castellan will use the URL from the service catalog.
barbican_api_version = <i>v1</i>	(String) Version of the barbican API, for example: "v1"
[certificates]	
barbican_auth = <i>barbican_acl_auth</i>	(String) Name of the Barbican authentication method to use
cert_manager_type = <i>barbican</i>	(String) Certificate Manager plugin. Defaults to barbican.
[cluster_verifications]	
verification_enable = <i>True</i>	(Boolean) Option to enable verifications for all clusters

Configuration option = Default value	Description
verification_periodic_interval = 600	(Integer) Interval between two consecutive periodic tasks for verifications, in seconds.
[conductor]	
use_local = <i>True</i>	(Boolean) Perform sahara-conductor operations locally.
[keystone_authtoken]	
memcached_servers = <i>None</i>	(List) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.

Table 6.6. Description of CORS configuration options

Configuration option = Default value	Description
[cors]	
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = 3600	(Integer) Maximum cache age of CORS preflight requests.
[cors.subdomain]	
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.

Configuration option = Default value	Description
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(Integer) Maximum cache age of CORS preflight requests.

Table 6.7. Description of database configuration options

Configuration option = Default value	Description
[DEFAULT]	
db_driver = <i>sahara.db</i>	(String) Driver to use for database access.
[database]	
backend = <i>sqlalchemy</i>	(String) The back end to use for the database.
connection = <i>None</i>	(String) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(Integer) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(Boolean) Add Python stack traces to SQL as comment strings.
db_inc_retry_interval = <i>True</i>	(Boolean) If True, increases the interval between retries of a database operation up to db_max_retry_interval .
db_max_retries = <i>20</i>	(Integer) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = <i>10</i>	(Integer) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.

Configuration option = Default value	Description
db_retry_interval = 1	(Integer) Seconds between retries of a database transaction.
idle_timeout = 3600	(Integer) Timeout before idle SQL connections are reaped.
max_overflow = 50	(Integer) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = None	(Integer) Maximum number of SQL connections to keep open in a pool.
max_retries = 10	(Integer) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = 1	(Integer) Minimum number of SQL connections to keep open in a pool.
mysql_sql_mode = <i>TRADITIONAL</i>	(String) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode=
pool_timeout = None	(Integer) If set, use this value for pool_timeout with SQLAlchemy.
retry_interval = 10	(Integer) Interval between retries of opening a SQL connection.
slave_connection = None	(String) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = <i>oslo.sqlite</i>	(String) The file name to use with SQLite.
sqlite_synchronous = <i>True</i>	(Boolean) If True, SQLite uses synchronous mode.
use_db_reconnect = <i>False</i>	(Boolean) Enable the experimental use of database reconnect on connection lost.

Table 6.8. Description of domain configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
proxy_user_domain_name = <i>None</i>	(String) The domain Sahara will use to create new proxy users for Swift object access.
proxy_user_role_names = <i>Member</i>	(List) A list of the role names that the proxy user should assume through trust for Swift object access.
use_domain_for_proxy_users = <i>False</i>	(Boolean) Enables Sahara to use a domain for creating temporary proxy users to access Swift. If this is enabled a domain must be created for Sahara to use.

Table 6.9. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(Boolean) If set to true, the logging level will be set to DEBUG instead of the default INFO level.
default_log_levels = <i>amqpplib=WARN, qpid.messaging=INFO, stevedore=INFO, eventlet.wsgi.server=WARN, sqlalchemy=WARN, boto=WARN, suds=INFO, keystone=INFO, paramiko=WARN, requests=WARN, iso8601=WARN, oslo_messaging=INFO, neutronclient=INFO</i>	(List) List of package logging levels in logger=LEVEL pairs. This option is ignored if log_config_append is set.
fatal_deprecations = <i>False</i>	(Boolean) Enables or disables fatal status of deprecations.
instance_format = <i>"[instance: %(uuid)s] "</i>	(String) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(String) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(String) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, logging_context_format_string).

Configuration option = Default value	Description
log_date_format = %Y-%m-%d %H:%M:%S	(String) Defines the format string for %%(asctime)s in log records. Default: %(default)s . This option is ignored if log_config_append is set.
log_dir = None	(String) (Optional) The base directory used for relative log_file paths. This option is ignored if log_config_append is set.
log_file = None	(String) (Optional) Name of log file to send logging output to. If no default is set, logging will go to stderr as defined by use_stderr. This option is ignored if log_config_append is set.
logging_context_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [% (request_id)s %(user_identity)s] % (instance)s%(message)s	(String) Format string to use for log messages with context.
logging_debug_format_suffix = % (funcName)s %(pathname)s:%(lineno)d	(String) Additional data to append to log message when logging level for the message is DEBUG.
logging_default_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [-] %(instance)s%(message)s	(String) Format string to use for log messages when context is undefined.
logging_exception_prefix = %(asctime)s.% (msecs)03d %(process)d ERROR %(name)s % (instance)s	(String) Prefix each line of exception output with this format.
logging_user_identity_format = %(user)s (tenant)s %(domain)s %(user_domain)s % (project_domain)s	(String) Defines the format string for % (user_identity)s that is used in logging_context_format_string.
publish_errors = False	(Boolean) Enables or disables publication of error events.
syslog_log_facility = LOG_USER	(String) Syslog facility to receive log lines. This option is ignored if log_config_append is set.
use_stderr = True	(Boolean) Log output to standard error. This option is ignored if log_config_append is set.
use_syslog = False	(Boolean) Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.

Configuration option = Default value	Description
verbose = <i>True</i>	(Boolean) DEPRECATED: If set to false, the logging level will be set to WARNING instead of the default INFO level.
watch_log_file = <i>False</i>	(Boolean) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log_file option is specified and Linux platform is used. This option is ignored if log_config_append is set.

Table 6.10. Description of Auth options for Swift access for VM configuration options

Configuration option = Default value	Description
[object_store_access]	
public_identity_ca_file = <i>None</i>	(String) Location of ca certificate file to use for identity client requests via public endpoint
public_object_store_ca_file = <i>None</i>	(String) Location of ca certificate file to use for object-store client requests via public endpoint

Table 6.11. Description of policy configuration options

Configuration option = Default value	Description
[oslo_policy]	
policy_default_rule = <i>default</i>	(String) Default rule. Enforced when a requested rule is not found.
policy_dirs = <i>['policy.d']</i>	(Multi-valued) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the config_dir option, or absolute paths. The file defined by policy_file must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(String) The JSON file that defines policies.

Table 6.12. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	

Configuration option = Default value	Description
<code>amqp_auto_delete = False</code>	(BoolOpt) Auto-delete queues in AMQP.
<code>amqp_durable_queues = False</code>	(BoolOpt) Use durable queues in AMQP.
<code>qpid_heartbeat = 60</code>	(IntOpt) Seconds between connection keepalive heartbeats.
<code>qpid_hostname = localhost</code>	(StrOpt) Qpid broker hostname.
<code>qpid_hosts = \$qpid_hostname:\$qpid_port</code>	(ListOpt) Qpid HA cluster host:port pairs.
<code>qpid_password =</code>	(StrOpt) Password for Qpid connection.
<code>qpid_port = 5672</code>	(IntOpt) Qpid broker port.
<code>qpid_protocol = tcp</code>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
<code>qpid_receiver_capacity = 1</code>	(IntOpt) The number of prefetched messages held by receiver.
<code>qpid_sasl_mechanisms =</code>	(StrOpt) Space separated list of SASL mechanisms to use for auth.
<code>qpid_tcp_nodelay = True</code>	(BoolOpt) Whether to disable the Nagle algorithm.
<code>qpid_topology_version = 1</code>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
<code>qpid_username =</code>	(StrOpt) Username for Qpid connection.
<code>send_single_reply = False</code>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

Table 6.13. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = <i>False</i>	(Boolean) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(Boolean) Use durable queues in AMQP.
channel_max = <i>None</i>	(Integer) Maximum number of channels to allow
default_notification_exchange = <i>\${control_exchange}_notification</i>	(String) Exchange name for for sending notifications
default_notification_retry_attempts = <i>-1</i>	(Integer) Reconnecting retry count in case of connectivity problem during sending notification, -1 means infinite retry.
default_rpc_exchange = <i>\${control_exchange}_rpc</i>	(String) Exchange name for sending RPC messages
default_rpc_retry_attempts = <i>-1</i>	(Integer) Reconnecting retry count in case of connectivity problem during sending RPC message, -1 means infinite retry. If actual retry attempts in not 0 the rpc request could be processed more then one time
fake_rabbit = <i>False</i>	(Boolean) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
frame_max = <i>None</i>	(Integer) The maximum byte size for an AMQP frame
heartbeat_interval = <i>1</i>	(Integer) How often to send heartbeats for consumer's connections
heartbeat_rate = <i>2</i>	(Integer) How often times during the <code>heartbeat_timeout_threshold</code> we check the heartbeat.
heartbeat_timeout_threshold = <i>60</i>	(Integer) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disable the heartbeat). EXPERIMENTAL
host_connection_reconnect_delay = <i>0.25</i>	(Floating point) Set delay for reconnection to some host which has connection error
kombu_compression = <i>None</i>	(String) EXPERIMENTAL: Possible values are: <code>gzip</code> , <code>bz2</code> . If not set compression will not be used. This option may notbe available in future versions.

Configuration option = Default value	Description
kombu_failover_strategy = <i>round-robin</i>	(String) Determines how the next RabbitMQ node is chosen in case the one we are currently connected to becomes unavailable. Takes effect only if more than one RabbitMQ node is provided in config.
kombu_missing_consumer_retry_timeout = 60	(Integer) How long to wait a missing client before abandoning to send it its replies. This value should not be longer than <code>rpc_response_timeout</code> .
kombu_reconnect_delay = 1.0	(Floating point) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_ssl_ca_certs =	(String) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(String) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(String) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(String) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
notification_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to notification listener.
notification_persistence = <i>False</i>	(Boolean) Persist notification messages.
notification_retry_delay = 0.25	(Floating point) Reconnecting retry delay in case of connectivity problem during sending notification message
pool_max_overflow = 0	(Integer) Maximum number of connections to create above <code>pool_max_size</code> .
pool_max_size = 10	(Integer) Maximum number of connections to keep queued.
pool_recycle = 600	(Integer) Lifetime of a connection (since creation) in seconds or None for no recycling. Expired connections are closed on acquire.
pool_stale = 60	(Integer) Threshold at which inactive (since release) connections are considered stale in seconds or None for no staleness. Stale connections are closed on acquire.

Configuration option = Default value	Description
pool_timeout = 30	(Integer) Default number of seconds to wait for a connections to available
rabbit_ha_queues = <i>False</i>	(Boolean) Try to use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database. In RabbitMQ 3.0, queue mirroring is no longer controlled by the x-ha-policy argument when declaring a queue. If you just want to make sure that all queues (except those with auto-generated names) are mirrored across all nodes, run: "rabbitmqctl set_policy HA '^(?!amq\\.\\.)*' '{\"ha-mode\": \"all\"}' "
rabbit_host = <i>localhost</i>	(String) The RabbitMQ broker address where a single node is used.
rabbit_hosts = <i>\$rabbit_host:\$rabbit_port</i>	(List) RabbitMQ HA cluster host:port pairs.
rabbit_interval_max = 30	(Integer) Maximum interval of RabbitMQ connection retries. Default is 30 seconds.
rabbit_login_method = <i>AMQPLAIN</i>	(String) The RabbitMQ login method.
rabbit_max_retries = 0	(Integer) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = <i>guest</i>	(String) The RabbitMQ password.
rabbit_port = 5672	(Port number) The RabbitMQ broker port where a single node is used.
rabbit_qos_prefetch_count = 0	(Integer) Specifies the number of messages to prefetch. Setting to zero allows unlimited messages.
rabbit_retry_backoff = 2	(Integer) How long to backoff for between retries when connecting to RabbitMQ.
rabbit_retry_interval = 1	(Integer) How frequently to retry connecting with RabbitMQ.
rabbit_transient_queues_ttl = 1800	(Integer) Positive integer representing duration in seconds for queue TTL (x-expires). Queues which are unused for the duration of the TTL are automatically deleted. The parameter affects only reply and fanout queues.
rabbit_use_ssl = <i>False</i>	(Boolean) Connect over SSL for RabbitMQ.

Configuration option = Default value	Description
<code>rabbit_userid = guest</code>	(String) The RabbitMQ userid.
<code>rabbit_virtual_host = /</code>	(String) The RabbitMQ virtual host.
<code>rpc_listener_prefetch_count = 100</code>	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc listener.
<code>rpc_queue_expiration = 60</code>	(Integer) Time to live for rpc queues without consumers in seconds.
<code>rpc_reply_exchange = \${control_exchange}_rpc_reply</code>	(String) Exchange name for receiving RPC replies
<code>rpc_reply_listener_prefetch_count = 100</code>	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc reply listener.
<code>rpc_reply_retry_attempts = -1</code>	(Integer) Reconnecting retry count in case of connectivity problem during sending reply. -1 means infinite retry during rpc_timeout
<code>rpc_reply_retry_delay = 0.25</code>	(Floating point) Reconnecting retry delay in case of connectivity problem during sending reply.
<code>rpc_retry_delay = 0.25</code>	(Floating point) Reconnecting retry delay in case of connectivity problem during sending RPC message
<code>socket_timeout = 0.25</code>	(Floating point) Set socket timeout in seconds for connection's socket
<code>ssl = None</code>	(Boolean) Enable SSL
<code>ssl_options = None</code>	(Dict) Arguments passed to ssl.wrap_socket
<code>tcp_user_timeout = 0.25</code>	(Floating point) Set TCP_USER_TIMEOUT in seconds for connection's socket

Table 6.14. Description of Redis configuration options

Configuration option = Default value	Description
<code>[matchmaker_redis]</code>	
<code>check_timeout = 20000</code>	(Integer) Time in ms to wait before the transaction is killed.
<code>host = 127.0.0.1</code>	(String) Host to locate redis.

Configuration option = Default value	Description
password =	(String) Password for Redis server (optional).
port = 6379	(Port number) Use this port to connect to redis host.
sentinel_group_name = oslo-messaging-zeromq	(String) Redis replica set name.
sentinel_hosts =	(List) List of Redis Sentinel hosts (fault tolerance mode) e.g. [host:port, host1:port ...]
socket_timeout = 1000	(Integer) Timeout in ms on blocking socket operations
wait_timeout = 500	(Integer) Time in ms to wait between connection attempts.

Table 6.15. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
rpc_backend = rabbit	(String) The messaging driver to use, defaults to rabbit. Other drivers include amqp and zmq.
rpc_cast_timeout = -1	(Integer) Seconds to wait before a cast expires (TTL). The default value of -1 specifies an infinite linger period. The value of 0 specifies no linger period. Pending messages shall be discarded immediately when the socket is closed. Only supported by impl_zmq.
rpc_conn_pool_size = 30	(Integer) Size of RPC connection pool.
rpc_poll_timeout = 1	(Integer) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
rpc_response_timeout = 60	(Integer) Seconds to wait for a response from a call.
[oslo_concurrency]	
disable_process_locking = False	(Boolean) Enables or disables inter-process locks.

Configuration option = Default value	Description
lock_path = <i>None</i>	(String) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging]	
event_stream_topic = <i>neutron_lbaas_event</i>	(String) topic name for receiving events from a queue
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(Boolean) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>	(String) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(String) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(String) address prefix when sending to any server in group
idle_timeout = <i>0</i>	(Integer) Timeout for inactive connections (in seconds)
password =	(String) Password for message broker authentication
sasl_config_dir =	(String) Path to directory that contains the SASL configuration
sasl_config_name =	(String) Name of configuration file (without .conf suffix)
sasl_mechanisms =	(String) Space separated list of acceptable SASL mechanisms
server_request_prefix = <i>exclusive</i>	(String) address prefix used when sending to a specific server
ssl_ca_file =	(String) CA certificate PEM file to verify server certificate

Configuration option = Default value	Description
ssl_cert_file =	(String) Identifying certificate PEM file to present to clients
ssl_key_file =	(String) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(String) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>	(Boolean) Debug: dump AMQP frames to stdout
username =	(String) User name for message broker authentication
[oslo_messaging_notifications]	
driver = []	(Multi-valued) The Drivers(s) to handle sending notifications. Possible values are messaging, messagingv2, routing, log, test, noop
enable = <i>False</i>	(Boolean) Enables sending notifications to Ceilometer
level = <i>INFO</i>	(String) Notification level for outgoing notifications
publisher_id = <i>None</i>	(String) Notification publisher_id for outgoing notifications
topics = <i>notifications</i>	(List) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(String) A URL representing the messaging driver to use for notifications. If not set, we fall back to the same configuration used for RPC.

Table 6.16. Description of timeouts configuration options

Configuration option = Default value	Description
[timeouts]	
delete_instances_timeout = <i>10800</i>	(Integer) Wait for instances to be deleted, in seconds
detach_volume_timeout = <i>300</i>	(Integer) Timeout for detaching volumes from instance, in seconds
ips_assign_timeout = <i>10800</i>	(Integer) Assign IPs timeout, in seconds

Configuration option = Default value	Description
<code>wait_until_accessible = 10800</code>	(Integer) Wait for instance accessibility, in seconds

6.1. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR DATA PROCESSING SERVICE

Table 6.17. New options

Configuration option = Default value	Description
[DEFAULT] <code>coordinator_heartbeat_interval = 1</code>	(IntOpt) Interval size between heartbeat execution in seconds. Heartbeats are executed to make sure that connection to the coordination server is active.
[DEFAULT] <code>hash_ring_replicas_count = 40</code>	(IntOpt) Number of points that belongs to each member on a hash ring. The larger number leads to a better distribution.
[DEFAULT] <code>periodic_coordinator_backend_url = None</code>	(StrOpt) The backend URL to use for distributed periodic tasks coordination.
[DEFAULT] <code>periodic_workers_number = 1</code>	(IntOpt) Number of threads to run periodic tasks.
[DEFAULT] <code>use_barbican_key_manager = False</code>	(BoolOpt) Enable the usage of the OpenStack Key Management service provided by barbican.
[castellan] <code>barbican_api_endpoint = None</code>	(StrOpt) The endpoint to use for connecting to the barbican api controller. By default, castellan will use the URL from the service catalog.
[castellan] <code>barbican_api_version = v1</code>	(StrOpt) Version of the barbican API, for example: "v1"
[cluster_verifications] <code>verification_enable = True</code>	(BoolOpt) Option to enable verifications for all clusters
[cluster_verifications] <code>verification_periodic_interval = 600</code>	(IntOpt) Interval between two consecutive periodic tasks for verifications, in seconds.
[oslo_messaging_notifications] <code>enable = False</code>	(BoolOpt) Enables sending notifications to Ceilometer
[oslo_messaging_notifications] <code>level = INFO</code>	(StrOpt) Notification level for outgoing notifications

Configuration option = Default value	Description
<code>[oslo_messaging_notifications]</code> <code>publisher_id = None</code>	(StrOpt) Notification publisher_id for outgoing notifications

Table 6.18. New default values

Option	Previous default value	New default value
<code>[DEFAULT] api_workers</code>	<i>0</i>	<i>1</i>
<code>[DEFAULT] plugins</code>	<i>hdp, cdh</i>	<i>cdh</i>

Table 6.19. Deprecated options

Configuration option = Default value	Description
<code>[DEFAULT] enable_notifications</code>	<i>[oslo_messaging_notifications] enable</i>
<code>[DEFAULT] notification_level</code>	<i>[oslo_messaging_notifications] level</i>
<code>[DEFAULT] notification_publisher_id</code>	<i>[oslo_messaging_notifications] publisher_id</i>
<code>[DEFAULT] use_syslog</code>	<i>None</i>

CHAPTER 7. IDENTITY SERVICE

This chapter details the OpenStack Identity service configuration options.

7.1. IDENTITY SERVICE CONFIGURATION FILE

The Identity service is configured in the `/etc/keystone/keystone.conf` file.

The following tables provide a comprehensive list of the Identity service options.

Table 7.1. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
admin_endpoint = <i>None</i>	(StrOpt) The base admin endpoint URL for Keystone that is advertised to clients (NOTE: this does NOT affect how Keystone listens for connections). Defaults to the base host URL of the request. E.g. a request to <code>http://server:35357/v3/users</code> will default to <code>http://server:35357</code> . You should only need to set this value if the base URL contains a path (e.g. <code>/prefix/v3</code>) or the endpoint should be found on a different server.
admin_token = <i>ADMIN</i>	(StrOpt) A "shared secret" that can be used to bootstrap Keystone. This "token" does not represent a user, and carries no explicit authorization. To disable in production (highly recommended), remove <code>AdminTokenAuthMiddleware</code> from your paste application pipelines (for example, in <code>keystone-paste.ini</code>).
domain_id_immutable = <i>True</i>	(BoolOpt) Set this to false if you want to enable the ability for user, group and project entities to be moved between domains by updating their <code>domain_id</code> . Allowing such movement is not recommended if the scope of a domain admin is being restricted by use of an appropriate policy file (see <code>policy.v3cloudsample</code> as an example).
list_limit = <i>None</i>	(IntOpt) The maximum number of entities that will be returned in a collection, with no limit set by default. This global limit may be then overridden for a specific driver, by specifying a <code>list_limit</code> in the appropriate section (e.g. <code>[assignment]</code>).
max_param_size = <i>64</i>	(IntOpt) Limit the sizes of user & project ID/names.

Configuration option = Default value	Description
max_project_tree_depth = 5	(IntOpt) Maximum depth of the project hierarchy. WARNING: setting it to a large value may adversely impact performance.
max_token_size = 8192	(IntOpt) Similar to max_param_size, but provides an exception for token values.
member_role_id = 9fe2ff9ee4384b1894a90878d3e92bab	(StrOpt) Similar to the member_role_name option, this represents the default role ID used to associate users with their default projects in the v2 API. This will be used as the explicit role where one is not specified by the v2 API.
member_role_name = _member_	(StrOpt) This is the role name used in combination with the member_role_id option; see that option for more detail.
public_endpoint = None	(StrOpt) The base public endpoint URL for Keystone that is advertised to clients (NOTE: this does NOT affect how Keystone listens for connections). Defaults to the base host URL of the request. E.g. a request to http://server:5000/v3/users will default to http://server:5000. You should only need to set this value if the base URL contains a path (e.g. /prefix/v3) or the endpoint should be found on a different server.
secure_proxy_ssl_header = None	(StrOpt) The HTTP header used to determine the scheme for the original request, even if it was removed by an SSL terminating proxy. Typical value is "HTTP_X_FORWARDED_PROTO".
strict_password_check = False	(BoolOpt) If set to true, strict password length checking is performed for password manipulation. If a password exceeds the maximum length, the operation will fail with an HTTP 403 Forbidden error. If set to false, passwords are automatically truncated to the maximum length.
[endpoint_filter]	
driver = sql	(StrOpt) Entrypoint for the endpoint filter backend driver in the keystone.endpoint_filter namespace.
return_all_endpoints_if_no_filter = True	(BoolOpt) Toggle to return all active endpoints if no filter exists.
[endpoint_policy]	

Configuration option = Default value	Description
driver = <i>sql</i>	(StrOpt) Entrypoint for the endpoint policy backend driver in the keystone.endpoint_policy namespace.
enabled = <i>True</i>	(BoolOpt) Enable endpoint_policy functionality.
[eventlet_server]	
admin_bind_host = <i>0.0.0.0</i>	(StrOpt) The IP address of the network interface for the admin service to listen on.
admin_port = <i>35357</i>	(IntOpt) The port number which the admin service listens on.
admin_workers = <i>None</i>	(IntOpt) The number of worker processes to serve the admin eventlet application. Defaults to number of CPUs (minimum of 2).
client_socket_timeout = <i>900</i>	(IntOpt) Timeout for socket operations on a client connection. If an incoming connection is idle for this number of seconds it will be closed. A value of '0' means wait forever.
public_bind_host = <i>0.0.0.0</i>	(StrOpt) The IP address of the network interface for the public service to listen on.
public_port = <i>5000</i>	(IntOpt) The port number which the public service listens on.
public_workers = <i>None</i>	(IntOpt) The number of worker processes to serve the public eventlet application. Defaults to number of CPUs (minimum of 2).
tcp_keepalive = <i>False</i>	(BoolOpt) Set this to true if you want to enable TCP_KEEPALIVE on server sockets, i.e. sockets used by the Keystone wsgi server for client connections.
tcp_keepidle = <i>600</i>	(IntOpt) Sets the value of TCP_KEEPIDLE in seconds for each server socket. Only applies if tcp_keepalive is true.
wsgi_keep_alive = <i>True</i>	(BoolOpt) If set to false, disables keepalives on the server; all connections will be closed after serving one request.
[oslo_middleware]	

Configuration option = Default value	Description
max_request_body_size = <i>114688</i>	(IntOpt) The maximum body size for each request, in bytes.
secure_proxy_ssl_header = <i>X-Forwarded-Proto</i>	(StrOpt) The HTTP Header that will be used to determine what the original request protocol scheme was, even if it was hidden by an SSL termination proxy.
[paste_deploy]	
config_file = <i>keystone-paste.ini</i>	(StrOpt) Name of the paste configuration file that defines the available pipelines.
[resource]	
cache_time = <i>None</i>	(IntOpt) TTL (in seconds) to cache resource data. This has no effect unless global caching is enabled.
caching = <i>True</i>	(BoolOpt) Toggle for resource caching. This has no effect unless global caching is enabled.
driver = <i>None</i>	(StrOpt) Entrypoint for the resource backend driver in the keystone.resource namespace. Supplied drivers are ldap and sql. If a resource driver is not specified, the assignment driver will choose the resource driver.
list_limit = <i>None</i>	(IntOpt) Maximum number of entities that will be returned in a resource collection.

Table 7.2. Description of assignment configuration options

Configuration option = Default value	Description
[assignment]	
driver = <i>None</i>	(StrOpt) Entrypoint for the assignment backend driver in the keystone.assignment namespace. Supplied drivers are ldap and sql. If an assignment driver is not specified, the identity driver will choose the assignment driver.

Table 7.3. Description of authorization configuration options

Configuration option = Default value	Description
[auth]	
external = <i>None</i>	(StrOpt) Entrypoint for the external (REMOTE_USER) auth plugin module in the keystone.auth.external namespace. Supplied drivers are DefaultDomain and Domain. The default driver is DefaultDomain.
methods = <i>external, password, token, oauth1</i>	(ListOpt) Allowed authentication methods.
oauth1 = <i>None</i>	(StrOpt) Entrypoint for the oAuth1.0 auth plugin module in the keystone.auth.oauth1 namespace.
password = <i>None</i>	(StrOpt) Entrypoint for the password auth plugin module in the keystone.auth.password namespace.
token = <i>None</i>	(StrOpt) Entrypoint for the token auth plugin module in the keystone.auth.token namespace.

Table 7.4. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(StrOpt) Service user password.
admin_tenant_name = <i>admin</i>	(StrOpt) Service tenant name.
admin_token = <i>None</i>	(StrOpt) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use `admin_user` and `admin_password` instead.
admin_user = <i>None</i>	(StrOpt) Service username.
auth_admin_prefix =	(StrOpt) Prefix to prepend at the beginning of the path. Deprecated, use identity_uri.
auth_host = <i>127.0.0.1</i>	(StrOpt) Host providing the admin Identity API endpoint. Deprecated, use identity_uri.
auth_plugin = <i>None</i>	(StrOpt) Name of the plugin to load

Configuration option = Default value	Description
auth_port = 35357	(IntOpt) Port of the admin Identity API endpoint. Deprecated, use identity_uri.
auth_protocol = https	(StrOpt) Protocol of the admin Identity API endpoint (http or https). Deprecated, use identity_uri.
auth_section = None	(StrOpt) Config Section from which to load plugin specific options
auth_uri = None	(StrOpt) Complete public Identity API endpoint.
auth_version = None	(StrOpt) API version of the admin Identity API endpoint.
cache = None	(StrOpt) Env key for the swift cache.
cafile = None	(StrOpt) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = None	(StrOpt) Required if identity server requires client certificate
check_revocations_for_cached = False	(BoolOpt) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.
delay_auth_decision = False	(BoolOpt) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = permissive	(StrOpt) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.

Configuration option = Default value	Description
hash_algorithms = <i>md5</i>	(ListOpt) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(IntOpt) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(IntOpt) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(StrOpt) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/
include_service_catalog = <i>True</i>	(BoolOpt) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate
memcache_pool_conn_get_timeout = <i>10</i>	(IntOpt) (Optional) Number of seconds that an operation will wait to get a memcached client connection from the pool.
memcache_pool_dead_retry = <i>300</i>	(IntOpt) (Optional) Number of seconds memcached server is considered dead before it is tried again.
memcache_pool_maxsize = <i>10</i>	(IntOpt) (Optional) Maximum total number of open connections to every memcached server.
memcache_pool_socket_timeout = <i>3</i>	(IntOpt) (Optional) Socket timeout in seconds for communicating with a memcached server.
memcache_pool_unused_timeout = <i>60</i>	(IntOpt) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.

Configuration option = Default value	Description
memcache_secret_key = <i>None</i>	(StrOpt) (Optional, mandatory if memcache_security_strategy is defined) This string is used for key derivation.
memcache_security_strategy = <i>None</i>	(StrOpt) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. Acceptable values are MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, auth_token will raise an exception on initialization.
memcache_use_advanced_pool = <i>False</i>	(BoolOpt) (Optional) Use the advanced (eventlet safe) memcached client pool. The advanced pool will only work under python 2.x.
region_name = <i>None</i>	(StrOpt) The region in which the identity server can be found.
revocation_cache_time = <i>10</i>	(IntOpt) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
signing_dir = <i>None</i>	(StrOpt) Directory used to cache files related to PKI tokens.
token_cache_time = <i>300</i>	(IntOpt) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 7.5. Description of CA and SSL configuration options

Configuration option = Default value	Description
[eventlet_server_ssl]	
ca_certs = <i>/etc/keystone/ssl/certs/ca.pem</i>	(StrOpt) Path of the CA cert file for SSL.
cert_required = <i>False</i>	(BoolOpt) Require client certificate.
certfile = <i>/etc/keystone/ssl/certs/keystone.pem</i>	(StrOpt) Path of the certfile for SSL. For non-production environments, you may be interested in using <code>`keystone-manage ssl_setup`</code> to generate self-signed certificates.

Configuration option = Default value	Description
enable = <i>False</i>	(BoolOpt) Toggle for SSL support on the Keystone eventlet servers.
keyfile = <i>/etc/keystone/ssl/private/keystonekey.pem</i>	(StrOpt) Path of the keyfile for SSL.
[signing]	
ca_certs = <i>/etc/keystone/ssl/certs/ca.pem</i>	(StrOpt) Path of the CA for token signing.
ca_key = <i>/etc/keystone/ssl/private/cakey.pem</i>	(StrOpt) Path of the CA key for token signing.
cert_subject = <i>/C=US/ST=Unset/L=Unset/O=Unset/CN=www.example.com</i>	(StrOpt) Certificate subject (auto generated certificate) for token signing.
certfile = <i>/etc/keystone/ssl/certs/signing_cert.pem</i>	(StrOpt) Path of the certfile for token signing. For non-production environments, you may be interested in using <code>keystone-manage pki_setup</code> to generate self-signed certificates.
key_size = <i>2048</i>	(IntOpt) Key size (in bits) for token signing cert (auto generated certificate).
keyfile = <i>/etc/keystone/ssl/private/signing_key.pem</i>	(StrOpt) Path of the keyfile for token signing.
valid_days = <i>3650</i>	(IntOpt) Days the token signing cert is valid for (auto generated certificate).
[ssl]	
ca_key = <i>/etc/keystone/ssl/private/cakey.pem</i>	(StrOpt) Path of the CA key file for SSL.
cert_subject = <i>/C=US/ST=Unset/L=Unset/O=Unset/CN=localhost</i>	(StrOpt) SSL certificate subject (auto generated certificate).
key_size = <i>1024</i>	(IntOpt) SSL key length (in bits) (auto generated certificate).
valid_days = <i>3650</i>	(IntOpt) Days the certificate is valid for once signed (auto generated certificate).

Table 7.6. Description of catalog configuration options

Configuration option = Default value	Description
[catalog]	
cache_time = <i>None</i>	(IntOpt) Time to cache catalog data (in seconds). This has no effect unless global and catalog caching are enabled.
caching = <i>True</i>	(BoolOpt) Toggle for catalog caching. This has no effect unless global caching is enabled.
driver = <i>sql</i>	(StrOpt) Entrypoint for the catalog backend driver in the keystone.catalog namespace. Supplied drivers are kvs, sql, templated, and endpoint_filter.sql
list_limit = <i>None</i>	(IntOpt) Maximum number of entities that will be returned in a catalog collection.
template_file = <i>default_catalog.templates</i>	(StrOpt) Catalog template file name for use with the template catalog backend.

Table 7.7. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
executor_thread_pool_size = <i>64</i>	(IntOpt) Size of executor thread pool.
memcached_servers = <i>None</i>	(ListOpt) Memcached servers or None for in process cache.
[keystone_authtoken]	
memcached_servers = <i>None</i>	(ListOpt) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.
[oslo_concurrency]	
disable_process_locking = <i>False</i>	(BoolOpt) Enables or disables inter-process locks.
lock_path = <i>None</i>	(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.

Table 7.8. Description of CORS configuration options

Configuration option = Default value	Description
[cors]	
allow_credentials = <i>True</i>	(BoolOpt) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(ListOpt) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(ListOpt) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(StrOpt) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(ListOpt) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(IntOpt) Maximum cache age of CORS preflight requests.
[cors.subdomain]	
allow_credentials = <i>True</i>	(BoolOpt) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(ListOpt) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(ListOpt) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(StrOpt) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(ListOpt) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(IntOpt) Maximum cache age of CORS preflight requests.

Table 7.9. Description of credential configuration options

Configuration option = Default value	Description
[credential]	
driver = <i>sql</i>	(StrOpt) Entrypoint for the credential backend driver in the keystone.credential namespace.

Table 7.10. Description of database configuration options

Configuration option = Default value	Description
[database]	
backend = <i>sqlalchemy</i>	(StrOpt) The back end to use for the database.
connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(BoolOpt) Add Python stack traces to SQL as comment strings.
db_inc_retry_interval = <i>True</i>	(BoolOpt) If True, increases the interval between retries of a database operation up to db_max_retry_interval .
db_max_retries = <i>20</i>	(IntOpt) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = <i>10</i>	(IntOpt) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.
db_retry_interval = <i>1</i>	(IntOpt) Seconds between retries of a database transaction.
idle_timeout = <i>3600</i>	(IntOpt) Timeout before idle SQL connections are reaped.
max_overflow = <i>None</i>	(IntOpt) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = <i>None</i>	(IntOpt) Maximum number of SQL connections to keep open in a pool.

Configuration option = Default value	Description
max_retries = 10	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = 1	(IntOpt) Minimum number of SQL connections to keep open in a pool.
mysql_sql_mode = <i>TRADITIONAL</i>	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: <code>mysql_sql_mode=</code>
pool_timeout = <i>None</i>	(IntOpt) If set, use this value for <code>pool_timeout</code> with SQLAlchemy.
retry_interval = 10	(IntOpt) Interval between retries of opening a SQL connection.
slave_connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = <i>oslo.sqlite</i>	(StrOpt) The file name to use with SQLite.
sqlite_synchronous = <i>True</i>	(BoolOpt) If True, SQLite uses synchronous mode.
use_db_reconnect = <i>False</i>	(BoolOpt) Enable the experimental use of database reconnect on connection lost.

Table 7.11. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
pydev_debug_host = <i>None</i>	(StrOpt) Host to connect to for remote debugger.
pydev_debug_port = <i>None</i>	(IntOpt) Port to connect to for remote debugger.
standard_threads = <i>False</i>	(BoolOpt) Do not monkey-patch threading system modules.
[audit]	
namespace = <i>openstack</i>	(StrOpt) namespace prefix for generated id

Table 7.12. Description of domain configuration options

Configuration option = Default value	Description
[domain_config]	
cache_time = 300	(IntOpt) TTL (in seconds) to cache domain config data. This has no effect unless domain config caching is enabled.
caching = <i>True</i>	(BoolOpt) Toggle for domain config caching. This has no effect unless global caching is enabled.
driver = <i>sql</i>	(StrOpt) Entrypoint for the domain config backend driver in the keystone.resource.domain_config namespace.

Table 7.13. Description of federation configuration options

Configuration option = Default value	Description
[federation]	
assertion_prefix =	(StrOpt) Value to be used when filtering assertion parameters from the environment.
driver = <i>sql</i>	(StrOpt) Entrypoint for the federation backend driver in the keystone.federation namespace.
federated_domain_name = <i>Federated</i>	(StrOpt) A domain name that is reserved to allow federated ephemeral users to have a domain concept. Note that an admin will not be able to create a domain with this name or update an existing domain to this name. You are not advised to change this value unless you really have to.
remote_id_attribute = <i>None</i>	(StrOpt) Value to be used to obtain the entity ID of the Identity Provider from the environment (e.g. if using the mod_shib plugin this value is `Shib-Identity-Provider`).
sso_callback_template = <i>/etc/keystone/sso_callback_template.html</i>	(StrOpt) Location of Single Sign-On callback handler, will return a token to a trusted dashboard host.

Configuration option = Default value	Description
trusted_dashboard = []	(MultiStrOpt) A list of trusted dashboard hosts. Before accepting a Single Sign-On request to return a token, the origin host must be a member of the trusted_dashboard list. This configuration option may be repeated for multiple values. For example: trusted_dashboard=http://acme.com/auth/websso trusted_dashboard=http://beta.com/auth/websso

Table 7.14. Description of Fernet tokens configuration options

Configuration option = Default value	Description
[fernet_tokens]	
key_repository = /etc/keystone/fernet-keys/	(StrOpt) Directory containing Fernet token keys.
max_active_keys = 3	(IntOpt) This controls how many keys are held in rotation by keystone-manage fernet_rotate before they are discarded. The default value of 3 means that keystone will maintain one staged key, one primary key, and one secondary key. Increasing this value means that additional secondary keys will be kept in the rotation.

Table 7.15. Description of identity configuration options

Configuration option = Default value	Description
[identity]	
cache_time = 600	(IntOpt) Time to cache identity data (in seconds). This has no effect unless global and identity caching are enabled.
caching = True	(BoolOpt) Toggle for identity caching. This has no effect unless global caching is enabled.
default_domain_id = default	(StrOpt) This references the domain to use for all Identity API v2 requests (which are not aware of domains). A domain with this ID will be created for you by keystone-manage db_sync in migration 008. The domain referenced by this ID cannot be deleted on the v3 API, to prevent accidentally breaking the v2 API. There is nothing special about this domain, other than the fact that it must exist to order to maintain support for your v2 clients.

Configuration option = Default value	Description
domain_config_dir = <i>/etc/keystone/domains</i>	(StrOpt) Path for Keystone to locate the domain specific identity configuration files if <code>domain_specific_drivers_enabled</code> is set to true.
domain_configurations_from_database = <i>False</i>	(BoolOpt) Extract the domain specific configuration options from the resource backend where they have been stored with the domain data. This feature is disabled by default (in which case the domain specific options will be loaded from files in the domain configuration directory); set to true to enable.
domain_specific_drivers_enabled = <i>False</i>	(BoolOpt) A subset (or all) of domains can have their own identity driver, each with their own partial configuration options, stored in either the resource backend or in a file in a domain configuration directory (depending on the setting of <code>domain_configurations_from_database</code>). Only values specific to the domain need to be specified in this manner. This feature is disabled by default; set to true to enable.
driver = <i>sql</i>	(StrOpt) Entrypoint for the identity backend driver in the <code>keystone.identity</code> namespace. Supplied drivers are <code>ldap</code> and <code>sql</code> .
list_limit = <i>None</i>	(IntOpt) Maximum number of entities that will be returned in an identity collection.
max_password_length = <i>4096</i>	(IntOpt) Maximum supported length for user passwords; decrease to improve performance.

Table 7.16. Description of KVS configuration options

Configuration option = Default value	Description
[kvs]	
backends =	(ListOpt) Extra <code>dogpile.cache</code> backend modules to register with the <code>dogpile.cache</code> library.
config_prefix = <i>keystone.kvs</i>	(StrOpt) Prefix for building the configuration dictionary for the KVS region. This should not need to be changed unless there is another <code>dogpile.cache</code> region with the same configuration name.
default_lock_timeout = <i>5</i>	(IntOpt) Default lock timeout (in seconds) for distributed locking.

Configuration option = Default value	Description
enable_key_mangler = <i>True</i>	(BoolOpt) Toggle to disable using a key-mangling function to ensure fixed length keys. This is toggleable for debugging purposes, it is highly recommended to always leave this set to true.

Table 7.17. Description of LDAP configuration options

Configuration option = Default value	Description
[ldap]	
alias_dereferencing = <i>default</i>	(StrOpt) The LDAP dereferencing option for queries. The "default" option falls back to using default dereferencing configured by your ldap.conf.
allow_subtree_delete = <i>False</i>	(BoolOpt) Delete subtrees using the subtree delete control. Only enable this option if your LDAP server supports subtree deletion.
auth_pool_connection_lifetime = <i>60</i>	(IntOpt) End user auth connection lifetime in seconds.
auth_pool_size = <i>100</i>	(IntOpt) End user auth connection pool size.
chase_referrals = <i>None</i>	(BoolOpt) Override the system's default referral chasing behavior for queries.
debug_level = <i>None</i>	(IntOpt) Sets the LDAP debugging level for LDAP calls. A value of 0 means that debugging is not enabled. This value is a bitmask, consult your LDAP documentation for possible values.
dumb_member = <i>cn=dumb,dc=nonexistent</i>	(StrOpt) DN of the "dummy member" to use when "use_dumb_member" is enabled.
group_additional_attribute_mapping =	(ListOpt) Additional attribute mappings for groups. Attribute mapping format is <ldap_attr>: <user_attr>, where ldap_attr is the attribute in the LDAP entry and user_attr is the Identity API attribute.
group_allow_create = <i>True</i>	(BoolOpt) Allow group creation in LDAP backend.
group_allow_delete = <i>True</i>	(BoolOpt) Allow group deletion in LDAP backend.
group_allow_update = <i>True</i>	(BoolOpt) Allow group update in LDAP backend.

Configuration option = Default value	Description
group_attribute_ignore =	(ListOpt) List of attributes stripped off the group on update.
group_desc_attribute = <i>description</i>	(StrOpt) LDAP attribute mapped to group description.
group_filter = <i>None</i>	(StrOpt) LDAP search filter for groups.
group_id_attribute = <i>cn</i>	(StrOpt) LDAP attribute mapped to group id.
group_member_attribute = <i>member</i>	(StrOpt) LDAP attribute mapped to show group membership.
group_name_attribute = <i>ou</i>	(StrOpt) LDAP attribute mapped to group name.
group_objectclass = <i>groupOfNames</i>	(StrOpt) LDAP objectclass for groups.
group_tree_dn = <i>None</i>	(StrOpt) Search base for groups. Defaults to the suffix value.
page_size = <i>0</i>	(IntOpt) Maximum results per page; a value of zero ("0") disables paging.
password = <i>None</i>	(StrOpt) Password for the BindDN to query the LDAP server.
pool_connection_lifetime = <i>600</i>	(IntOpt) Connection lifetime in seconds.
pool_connection_timeout = <i>-1</i>	(IntOpt) Connector timeout in seconds. Value -1 indicates indefinite wait for response.
pool_retry_delay = <i>0.1</i>	(FloatOpt) Time span in seconds to wait between two reconnect trials.
pool_retry_max = <i>3</i>	(IntOpt) Maximum count of reconnect trials.
pool_size = <i>10</i>	(IntOpt) Connection pool size.
project_additional_attribute_mapping =	(ListOpt) Additional attribute mappings for projects. Attribute mapping format is <ldap_attr>: <user_attr>, where ldap_attr is the attribute in the LDAP entry and user_attr is the Identity API attribute.
project_allow_create = <i>True</i>	(BoolOpt) Allow project creation in LDAP backend.

Configuration option = Default value	Description
project_allow_delete = <i>True</i>	(BoolOpt) Allow project deletion in LDAP backend.
project_allow_update = <i>True</i>	(BoolOpt) Allow project update in LDAP backend.
project_attribute_ignore =	(ListOpt) List of attributes stripped off the project on update.
project_desc_attribute = <i>description</i>	(StrOpt) LDAP attribute mapped to project description.
project_domain_id_attribute = <i>businessCategory</i>	(StrOpt) LDAP attribute mapped to project domain_id.
project_enabled_attribute = <i>enabled</i>	(StrOpt) LDAP attribute mapped to project enabled.
project_enabled_emulation = <i>False</i>	(BoolOpt) If true, Keystone uses an alternative method to determine if a project is enabled or not by checking if they are a member of the "project_enabled_emulation_dn" group.
project_enabled_emulation_dn = <i>None</i>	(StrOpt) DN of the group entry to hold enabled projects when using enabled emulation.
project_filter = <i>None</i>	(StrOpt) LDAP search filter for projects.
project_id_attribute = <i>cn</i>	(StrOpt) LDAP attribute mapped to project id.
project_member_attribute = <i>member</i>	(StrOpt) LDAP attribute mapped to project membership for user.
project_name_attribute = <i>ou</i>	(StrOpt) LDAP attribute mapped to project name.
project_objectclass = <i>groupOfNames</i>	(StrOpt) LDAP objectclass for projects.
project_tree_dn = <i>None</i>	(StrOpt) Search base for projects. Defaults to the suffix value.
query_scope = <i>one</i>	(StrOpt) The LDAP scope for queries, "one" represents oneLevel/singleLevel and "sub" represents subtree/wholeSubtree options.
role_additional_attribute_mapping =	(ListOpt) Additional attribute mappings for roles. Attribute mapping format is <ldap_attr>: <user_attr>, where ldap_attr is the attribute in the LDAP entry and user_attr is the Identity API attribute.

Configuration option = Default value	Description
role_allow_create = <i>True</i>	(BoolOpt) Allow role creation in LDAP backend.
role_allow_delete = <i>True</i>	(BoolOpt) Allow role deletion in LDAP backend.
role_allow_update = <i>True</i>	(BoolOpt) Allow role update in LDAP backend.
role_attribute_ignore =	(ListOpt) List of attributes stripped off the role on update.
role_filter = <i>None</i>	(StrOpt) LDAP search filter for roles.
role_id_attribute = <i>cn</i>	(StrOpt) LDAP attribute mapped to role id.
role_member_attribute = <i>roleOccupant</i>	(StrOpt) LDAP attribute mapped to role membership.
role_name_attribute = <i>ou</i>	(StrOpt) LDAP attribute mapped to role name.
role_objectclass = <i>organizationalRole</i>	(StrOpt) LDAP objectclass for roles.
role_tree_dn = <i>None</i>	(StrOpt) Search base for roles.
suffix = <i>cn=example,cn=com</i>	(StrOpt) LDAP server suffix
tls_cacertdir = <i>None</i>	(StrOpt) CA certificate directory path for communicating with LDAP servers.
tls_cacertfile = <i>None</i>	(StrOpt) CA certificate file path for communicating with LDAP servers.
tls_req_cert = <i>demand</i>	(StrOpt) Specifies what checks to perform on client certificates in an incoming TLS session.
url = <i>ldap://localhost</i>	(StrOpt) URL for connecting to the LDAP server.
use_auth_pool = <i>False</i>	(BoolOpt) Enable LDAP connection pooling for end user authentication. If use_pool is disabled, then this setting is meaningless and is not used at all.
use_dumb_member = <i>False</i>	(BoolOpt) If true, will add a dummy member to groups. This is required if the objectclass for groups requires the "member" attribute.
use_pool = <i>False</i>	(BoolOpt) Enable LDAP connection pooling.

Configuration option = Default value	Description
use_tls = <i>False</i>	(BoolOpt) Enable TLS for communicating with LDAP servers.
user = <i>None</i>	(StrOpt) User BindDN to query the LDAP server.
user_additional_attribute_mapping =	(ListOpt) List of additional LDAP attributes used for mapping additional attribute mappings for users. Attribute mapping format is <ldap_attr>: <user_attr>, where ldap_attr is the attribute in the LDAP entry and user_attr is the Identity API attribute.
user_allow_create = <i>True</i>	(BoolOpt) Allow user creation in LDAP backend.
user_allow_delete = <i>True</i>	(BoolOpt) Allow user deletion in LDAP backend.
user_allow_update = <i>True</i>	(BoolOpt) Allow user updates in LDAP backend.
user_attribute_ignore = <i>default_project_id</i>	(ListOpt) List of attributes stripped off the user on update.
user_default_project_id_attribute = <i>None</i>	(StrOpt) LDAP attribute mapped to default_project_id for users.
user_enabled_attribute = <i>enabled</i>	(StrOpt) LDAP attribute mapped to user enabled flag.
user_enabled_default = <i>True</i>	(StrOpt) Default value to enable users. This should match an appropriate int value if the LDAP server uses non-boolean (bitmask) values to indicate if a user is enabled or disabled. If this is not set to "True" the typical value is "512". This is typically used when "user_enabled_attribute = userAccountControl".
user_enabled_emulation = <i>False</i>	(BoolOpt) If true, Keystone uses an alternative method to determine if a user is enabled or not by checking if they are a member of the "user_enabled_emulation_dn" group.
user_enabled_emulation_dn = <i>None</i>	(StrOpt) DN of the group entry to hold enabled users when using enabled emulation.

Configuration option = Default value	Description
user_enabled_invert = <i>False</i>	(BoolOpt) Invert the meaning of the boolean enabled values. Some LDAP servers use a boolean lock attribute where "true" means an account is disabled. Setting "user_enabled_invert = true" will allow these lock attributes to be used. This setting will have no effect if "user_enabled_mask" or "user_enabled_emulation" settings are in use.
user_enabled_mask = <i>0</i>	(IntOpt) Bitmask integer to indicate the bit that the enabled value is stored in if the LDAP server represents "enabled" as a bit on an integer rather than a boolean. A value of "0" indicates the mask is not used. If this is not set to "0" the typical value is "2". This is typically used when "user_enabled_attribute = userAccountControl".
user_filter = <i>None</i>	(StrOpt) LDAP search filter for users.
user_id_attribute = <i>cn</i>	(StrOpt) LDAP attribute mapped to user id. WARNING: must not be a multivalued attribute.
user_mail_attribute = <i>mail</i>	(StrOpt) LDAP attribute mapped to user email.
user_name_attribute = <i>sn</i>	(StrOpt) LDAP attribute mapped to user name.
user_objectclass = <i>inetOrgPerson</i>	(StrOpt) LDAP objectclass for users.
user_pass_attribute = <i>userPassword</i>	(StrOpt) LDAP attribute mapped to password.
user_tree_dn = <i>None</i>	(StrOpt) Search base for users. Defaults to the suffix value.

Table 7.18. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(BoolOpt) Print debugging output (set logging level to DEBUG instead of default INFO level).

Configuration option = Default value	Description
default_log_levels = <i>amqp=WARN, amqplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystone.middleware=WARN, routes.middleware=WARN, stevedore=WARN, taskflow=WARN</i>	(ListOpt) List of logger=LEVEL pairs. This option is ignored if log_config_append is set.
fatal_deprecations = <i>False</i>	(BoolOpt) Enables or disables fatal status of deprecations.
instance_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, log_format).
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(StrOpt) Format string for %(asctime)s in log records. Default: %(default)s . This option is ignored if log_config_append is set.
log_dir = <i>None</i>	(StrOpt) (Optional) The base directory used for relative --log-file paths. This option is ignored if log_config_append is set.
log_file = <i>None</i>	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout. This option is ignored if log_config_append is set.
log_format = <i>None</i>	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecate, use logging_context_format_string and logging_default_format_string instead. This option is ignored if log_config_append is set.

Configuration option = Default value	Description
logging_context_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [% (request_id)s %(user_identity)s] % (instance)s%(message)s	(StrOpt) Format string to use for log messages with context.
logging_debug_format_suffix = % (funcName)s %(pathname)s:%(lineno)d	(StrOpt) Data to append to log format when level is DEBUG.
logging_default_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [-] %(instance)s%(message)s	(StrOpt) Format string to use for log messages without context.
logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d ERROR %(name)s % (instance)s	(StrOpt) Prefix each line of exception output with this format.
publish_errors = <i>False</i>	(BoolOpt) Enables or disables publication of error events.
syslog_log_facility = <i>LOG_USER</i>	(StrOpt) Syslog facility to receive log lines. This option is ignored if log_config_append is set.
use_stderr = <i>True</i>	(BoolOpt) Log output to standard error. This option is ignored if log_config_append is set.
use_syslog = <i>False</i>	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.
use_syslog_rfc_format = <i>True</i>	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in Kilo, and will be removed in Mitaka, along with this option. This option is ignored if log_config_append is set.
verbose = <i>True</i>	(BoolOpt) If set to false, will disable INFO logging level, making WARNING the default.
watch_log_file = <i>False</i>	(BoolOpt) (Optional) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log-file option is specified and Linux platform is used. This option is ignored if log_config_append is set.

Table 7.19. Description of mapping configuration options

Configuration option = Default value	Description
[identity_mapping]	
backward_compatible_ids = <i>True</i>	(BoolOpt) The format of user and group IDs changed in Juno for backends that do not generate UUIDs (e.g. LDAP), with keystone providing a hash mapping to the underlying attribute in LDAP. By default this mapping is disabled, which ensures that existing IDs will not change. Even when the mapping is enabled by using domain specific drivers, any users and groups from the default domain being handled by LDAP will still not be mapped to ensure their IDs remain backward compatible. Setting this value to False will enable the mapping for even the default LDAP driver. It is only safe to do this if you do not already have assignments for users and groups from the default LDAP domain, and it is acceptable for Keystone to provide the different IDs to clients than it did previously. Typically this means that the only time you can set this value to False is when configuring a fresh installation.
driver = <i>sql</i>	(StrOpt) Entrypoint for the identity mapping backend driver in the keystone.identity.id_mapping namespace.
generator = <i>sha256</i>	(StrOpt) Entrypoint for the public ID generator for user and group entities in the keystone.identity.id_generator namespace. The Keystone identity mapper only supports generators that produce no more than 64 characters.

Table 7.20. Description of memcache configuration options

Configuration option = Default value	Description
[memcache]	
servers = <i>localhost:11211</i>	(ListOpt) Memcache servers in the format of "host:port".
socket_timeout = <i>3</i>	(IntOpt) Timeout in seconds for every call to a server. This is used by the key value store system (e.g. token pooled memcached persistence backend).

Table 7.21. Description of OAuth configuration options

Configuration option = Default value	Description
[oauth1]	
access_token_duration = 86400	(IntOpt) Duration (in seconds) for the OAuth Access Token.
driver = <i>sql</i>	(StrOpt) Entrypoint for the OAuth backend driver in the keystone.oauth1 namespace.
request_token_duration = 28800	(IntOpt) Duration (in seconds) for the OAuth Request Token.

Table 7.22. Description of os_inherit configuration options

Configuration option = Default value	Description
[os_inherit]	
enabled = <i>False</i>	(BoolOpt) role-assignment inheritance to projects from owning domain or from projects higher in the hierarchy can be optionally enabled.

Table 7.23. Description of policy configuration options

Configuration option = Default value	Description
[oslo_policy]	
policy_default_rule = <i>default</i>	(StrOpt) Default rule. Enforced when a requested rule is not found.
policy_dirs = [<i>'policy.d'</i>]	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the config_dir option, or absolute paths. The file defined by policy_file must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(StrOpt) The JSON file that defines policies.
[policy]	
driver = <i>sql</i>	(StrOpt) Entrypoint for the policy backend driver in the keystone.policy namespace. Supplied drivers are rules and sql.

Configuration option = Default value	Description
list_limit = <i>None</i>	(IntOpt) Maximum number of entities that will be returned in a policy collection.

Table 7.24. Description of revoke configuration options

Configuration option = Default value	Description
[revoke]	
cache_time = <i>3600</i>	(IntOpt) Time to cache the revocation list and the revocation events (in seconds). This has no effect unless global and token caching are enabled.
caching = <i>True</i>	(BoolOpt) Toggle for revocation event caching. This has no effect unless global caching is enabled.
driver = <i>sql</i>	(StrOpt) Entrypoint for an implementation of the backend for persisting revocation events in the keystone.revoke namespace. Supplied drivers are kvs and sql.
expiration_buffer = <i>1800</i>	(IntOpt) This value (calculated in seconds) is added to token expiration before a revocation event may be removed from the backend.

Table 7.25. Description of role configuration options

Configuration option = Default value	Description
[role]	
cache_time = <i>None</i>	(IntOpt) TTL (in seconds) to cache role data. This has no effect unless global caching is enabled.
caching = <i>True</i>	(BoolOpt) Toggle for role caching. This has no effect unless global caching is enabled.
driver = <i>None</i>	(StrOpt) Entrypoint for the role backend driver in the keystone.role namespace. Supplied drivers are ldap and sql.
list_limit = <i>None</i>	(IntOpt) Maximum number of entities that will be returned in a role collection.

Table 7.26. Description of authorization configuration options

Configuration option = Default value	Description
[auth]	
saml2 = <i>keystone.auth.plugins.mapped.Mapped</i>	(StrOpt) The saml2 auth plugin module.

Table 7.27. Description of SAML configuration options

Configuration option = Default value	Description
[saml]	
assertion_expiration_time = 3600	(IntOpt) Default TTL, in seconds, for any generated SAML assertion created by Keystone.
certfile = <i>/etc/keystone/ssl/certs/signing_cert.pem</i>	(StrOpt) Path of the certfile for SAML signing. For non-production environments, you may be interested in using <code>keystone-manage pki_setup`</code> to generate self-signed certificates. Note, the path cannot contain a comma.
idp_contact_company = <i>None</i>	(StrOpt) Company of contact person.
idp_contact_email = <i>None</i>	(StrOpt) Email address of contact person.
idp_contact_name = <i>None</i>	(StrOpt) Given name of contact person
idp_contact_surname = <i>None</i>	(StrOpt) Surname of contact person.
idp_contact_telephone = <i>None</i>	(StrOpt) Telephone number of contact person.
idp_contact_type = <i>other</i>	(StrOpt) The contact type describing the main point of contact for the identity provider.
idp_entity_id = <i>None</i>	(StrOpt) Entity ID value for unique Identity Provider identification. Usually FQDN is set with a suffix. A value is required to generate IDP Metadata. For example: <code>https://keystone.example.com/v3/OS-FEDERATION/saml2/idp</code>
idp_lang = <i>en</i>	(StrOpt) Language used by the organization.
idp_metadata_path = <i>/etc/keystone/saml2_idp_metadata.xml</i>	(StrOpt) Path to the Identity Provider Metadata file. This file should be generated with the <code>keystone-manage saml_idp_metadata</code> command.
idp_organization_display_name = <i>None</i>	(StrOpt) Organization name to be displayed.

Configuration option = Default value	Description
idp_organization_name = <i>None</i>	(StrOpt) Organization name the installation belongs to.
idp_organization_url = <i>None</i>	(StrOpt) URL of the organization.
idp_sso_endpoint = <i>None</i>	(StrOpt) Identity Provider Single-Sign-On service value, required in the Identity Provider's metadata. A value is required to generate IDP Metadata. For example: https://keystone.example.com/v3/OS-FEDERATION/saml2/sso
keyfile = <i>/etc/keystone/ssl/private/signing_key.pem</i>	(StrOpt) Path of the keyfile for SAML signing. Note, the path cannot contain a comma.
relay_state_prefix = <i>ss:mem:</i>	(StrOpt) The prefix to use for the RelayState SAML attribute, used when generating ECP wrapped assertions.
xmlsec1_binary = <i>xmlsec1</i>	(StrOpt) Binary to be called for XML signing. Install the appropriate package, specify absolute path or adjust your PATH environment variable if the binary cannot be found.

Table 7.28. Description of security configuration options

Configuration option = Default value	Description
[DEFAULT]	
crypt_strength = <i>10000</i>	(IntOpt) The value passed as the keyword "rounds" to passlib's encrypt method.

Table 7.29. Description of token configuration options

Configuration option = Default value	Description
[token]	
allow_rescope_scoped_token = <i>True</i>	(BoolOpt) Allow rescoping of scoped token. Setting <code>allow_rescope_scoped_token</code> to false prevents a user from exchanging a scoped token for any other token.
bind =	(ListOpt) External auth mechanisms that should add bind information to token, e.g., kerberos,x509.

Configuration option = Default value	Description
cache_time = <i>None</i>	(IntOpt) Time to cache tokens (in seconds). This has no effect unless global and token caching are enabled.
caching = <i>True</i>	(BoolOpt) Toggle for token system caching. This has no effect unless global caching is enabled.
driver = <i>sql</i>	(StrOpt) Entrypoint for the token persistence backend driver in the keystone.token.persistence namespace. Supplied drivers are kvs, memcache, memcache_pool, and sql.
enforce_token_bind = <i>permissive</i>	(StrOpt) Enforcement policy on tokens presented to Keystone with bind information. One of disabled, permissive, strict, required or a specifically required bind mode, e.g., kerberos or x509 to require binding to that authentication.
expiration = <i>3600</i>	(IntOpt) Amount of time a token should remain valid (in seconds).
hash_algorithm = <i>md5</i>	(StrOpt) The hash algorithm to use for PKI tokens. This can be set to any algorithm that hashlib supports. WARNING: Before changing this value, the auth_token middleware must be configured with the hash_algorithms, otherwise token revocation will not be processed correctly.
provider = <i>uuid</i>	(StrOpt) Controls the token construction, validation, and revocation operations. Entrypoint in the keystone.token.provider namespace. Core providers are [fernet pkiz pki uuid].
revoke_by_id = <i>True</i>	(BoolOpt) Revoke token by token identifier. Setting revoke_by_id to true enables various forms of enumerating tokens, e.g. `list tokens for user`. These enumerations are processed to determine the list of tokens to revoke. Only disable if you are switching to using the Revoke extension with a backend other than KVS, which stores events in memory.

Table 7.30. Description of Tokenless Authorization configuration options

Configuration option = Default value	Description
[tokenless_auth]	

Configuration option = Default value	Description
issuer_attribute = <i>SSL_CLIENT_I_DN</i>	(StrOpt) The issuer attribute that is served as an IdP ID for the X.509 tokenless authorization along with the protocol to look up its corresponding mapping. It is the environment variable in the WSGI environment that references to the issuer of the client certificate.
protocol = <i>x509</i>	(StrOpt) The protocol name for the X.509 tokenless authorization along with the option issuer_attribute below can look up its corresponding mapping.
trusted_issuer = <i>[]</i>	(MultiStrOpt) The list of trusted issuers to further filter the certificates that are allowed to participate in the X.509 tokenless authorization. If the option is absent then no certificates will be allowed. The naming format for the attributes of a Distinguished Name(DN) must be separated by a comma and contain no spaces. This configuration option may be repeated for multiple values. For example: trusted_issuer=CN=john,OU=keystone,O=openstack trusted_issuer=CN=mary,OU=eng,O=abc

Table 7.31. Description of trust configuration options

Configuration option = Default value	Description
[trust]	
allow_redelegation = <i>False</i>	(BoolOpt) Enable redelegation feature.
driver = <i>sql</i>	(StrOpt) Entrypoint for the trust backend driver in the keystone.trust namespace.
enabled = <i>True</i>	(BoolOpt) Delegation and impersonation features can be optionally disabled.
max_redelegation_count = <i>3</i>	(IntOpt) Maximum depth of trust redelegation.

Table 7.32. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
rpc_backend = <i>rabbit</i>	(StrOpt) The messaging driver to use, defaults to rabbit. Other drivers include qpid and zmq.

Configuration option = Default value	Description
rpc_cast_timeout = 30	(IntOpt) Seconds to wait before a cast expires (TTL). Only supported by impl_zmq.
rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
rpc_poll_timeout = 1	(IntOpt) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
rpc_response_timeout = 60	(IntOpt) Seconds to wait for a response from a call.
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(BoolOpt) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>	(StrOpt) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(StrOpt) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(StrOpt) address prefix when sending to any server in group
idle_timeout = 0	(IntOpt) Timeout for inactive connections (in seconds)
password =	(StrOpt) Password for message broker authentication
sasl_config_dir =	(StrOpt) Path to directory that contains the SASL configuration
sasl_config_name =	(StrOpt) Name of configuration file (without .conf suffix)
sasl_mechanisms =	(StrOpt) Space separated list of acceptable SASL mechanisms
server_request_prefix = <i>exclusive</i>	(StrOpt) address prefix used when sending to a specific server
ssl_ca_file =	(StrOpt) CA certificate PEM file to verify server certificate
ssl_cert_file =	(StrOpt) Identifying certificate PEM file to present to clients

Configuration option = Default value	Description
ssl_key_file =	(StrOpt) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(StrOpt) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>	(BoolOpt) Debug: dump AMQP frames to stdout
username =	(StrOpt) User name for message broker authentication

Table 7.33. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>keystone</i>	(StrOpt) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the transport_url option.
default_publisher_id = <i>None</i>	(StrOpt) Default publisher_id for outgoing notifications
notification_driver = []	(MultiStrOpt) The Drivers(s) to handle sending notifications. Possible values are messaging, messagingv2, routing, log, test, noop
notification_format = <i>basic</i>	(StrOpt) Define the notification format for Identity Service events. A "basic" notification has information about the resource being operated on. A "cadf" notification has the same information, as well as information about the initiator of the event.
notification_topics = <i>notifications</i>	(ListOpt) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(StrOpt) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the rpc_backend option and driver specific configuration.

Table 7.34. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
qpid_heartbeat = <i>60</i>	(IntOpt) Seconds between connection keepalive heartbeats.
qpid_hostname = <i>localhost</i>	(StrOpt) Qpid broker hostname.
qpid_hosts = <i>\$qpid_hostname:\$qpid_port</i>	(ListOpt) Qpid HA cluster host:port pairs.
qpid_password =	(StrOpt) Password for Qpid connection.
qpid_port = <i>5672</i>	(IntOpt) Qpid broker port.
qpid_protocol = <i>tcp</i>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
qpid_receiver_capacity = <i>1</i>	(IntOpt) The number of prefetched messages held by receiver.
qpid_sasl_mechanisms =	(StrOpt) Space separated list of SASL mechanisms to use for auth.
qpid_tcp_nodelay = <i>True</i>	(BoolOpt) Whether to disable the Nagle algorithm.
qpid_topology_version = <i>1</i>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
qpid_username =	(StrOpt) Username for Qpid connection.
send_single_reply = <i>False</i>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

Table 7.35. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
fake_rabbit = <i>False</i>	(BoolOpt) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
heartbeat_rate = 2	(IntOpt) How often times during the <code>heartbeat_timeout_threshold</code> we check the heartbeat.
heartbeat_timeout_threshold = 60	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disable the heartbeat). EXPERIMENTAL
kombu_reconnect_delay = 1.0	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_reconnect_timeout = 60	(IntOpt) How long to wait before considering a reconnect attempt to have failed. This value should not be longer than <code>rpc_response_timeout</code> .
kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
rabbit_ha_queues = <i>False</i>	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
rabbit_host = <i>localhost</i>	(StrOpt) The RabbitMQ broker address where a single node is used.
rabbit_hosts = <i>\$rabbit_host:\$rabbit_port</i>	(ListOpt) RabbitMQ HA cluster host:port pairs.
rabbit_login_method = <i>AMQPLAIN</i>	(StrOpt) The RabbitMQ login method.

Configuration option = Default value	Description
<code>rabbit_max_retries = 0</code>	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
<code>rabbit_password = guest</code>	(StrOpt) The RabbitMQ password.
<code>rabbit_port = 5672</code>	(IntOpt) The RabbitMQ broker port where a single node is used.
<code>rabbit_retry_backoff = 2</code>	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
<code>rabbit_retry_interval = 1</code>	(IntOpt) How frequently to retry connecting with RabbitMQ.
<code>rabbit_use_ssl = False</code>	(BoolOpt) Connect over SSL for RabbitMQ.
<code>rabbit_userid = guest</code>	(StrOpt) The RabbitMQ userid.
<code>rabbit_virtual_host = /</code>	(StrOpt) The RabbitMQ virtual host.
<code>send_single_reply = False</code>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

Table 7.36. Description of Redis configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>host = 127.0.0.1</code>	(StrOpt) Host to locate redis.
<code>password =</code>	(StrOpt) Password for Redis server (optional).
<code>port = 6379</code>	(IntOpt) Use this port to connect to redis host.
[matchmaker_redis]	

Configuration option = Default value	Description
host = 127.0.0.1	(StrOpt) Host to locate redis.
password =	(StrOpt) Password for Redis server (optional).
port = 6379	(IntOpt) Use this port to connect to redis host.

7.2. IDENTITY SERVICE SAMPLE CONFIGURATION FILES

You can find the files described in this section in the `/etc/keystone` directory.

7.2.1. keystone.conf

Use the `keystone.conf` file to configure most Identity service options:

```
[DEFAULT]

#
# From keystone
#

# A "shared secret" that can be used to bootstrap Keystone. This "token"
# does
# not represent a user, and carries no explicit authorization. To disable
# in
# production (highly recommended), remove AdminTokenAuthMiddleware from
# your
# paste application pipelines (for example, in keystone-paste.ini).
# (string
# value)
#admin_token = ADMIN

# (Deprecated) The port which the OpenStack Compute service listens on.
# This
# option was only used for string replacement in the templated catalog
# backend.
# Templated catalogs should replace the "${compute_port}s" substitution
# with
# the static port of the compute service. As of Juno, this option is
# deprecated
# and will be removed in the L release. (integer value)
#compute_port = 8774

# The base public endpoint URL for Keystone that is advertised to clients
# (NOTE: this does NOT affect how Keystone listens for connections).
# Defaults
# to the base host URL of the request. E.g. a request to
# http://server:5000/v3/users will default to http://server:5000. You
# should
# only need to set this value if the base URL contains a path (e.g.
# /prefix/v3)
```

```

# or the endpoint should be found on a different server. (string value)
#public_endpoint = <None>

# The base admin endpoint URL for Keystone that is advertised to clients
(NOTE:
# this does NOT affect how Keystone listens for connections). Defaults to
the
# base host URL of the request. E.g. a request to
http://server:35357/v3/users
# will default to http://server:35357. You should only need to set this
value
# if the base URL contains a path (e.g. /prefix/v3) or the endpoint should
be
# found on a different server. (string value)
#admin_endpoint = <None>

# Maximum depth of the project hierarchy. WARNING: setting it to a large
value
# may adversely impact performance. (integer value)
#max_project_tree_depth = 5

# Limit the sizes of user & project ID/names. (integer value)
#max_param_size = 64

# Similar to max_param_size, but provides an exception for token values.
# (integer value)
#max_token_size = 8192

# Similar to the member_role_name option, this represents the default role
ID
# used to associate users with their default projects in the v2 API. This
will
# be used as the explicit role where one is not specified by the v2 API.
# (string value)
#member_role_id = 9fe2ff9ee4384b1894a90878d3e92bab

# This is the role name used in combination with the member_role_id
option; see
# that option for more detail. (string value)
#member_role_name = _member_

# The value passed as the keyword "rounds" to passlib's encrypt method.
# (integer value)
#crypt_strength = 40000

# The maximum number of entities that will be returned in a collection,
with no
# limit set by default. This global limit may be then overridden for a
specific
# driver, by specifying a list_limit in the appropriate section (e.g.
# [assignment]). (integer value)
#list_limit = <None>

# Set this to false if you want to enable the ability for user, group and
# project entities to be moved between domains by updating their
domain_id.

```

```

# Allowing such movement is not recommended if the scope of a domain admin
is
# being restricted by use of an appropriate policy file (see
# policy.v3cloudsample as an example). (boolean value)
#domain_id_immutable = true

# If set to true, strict password length checking is performed for
password
# manipulation. If a password exceeds the maximum length, the operation
will
# fail with an HTTP 403 Forbidden error. If set to false, passwords are
# automatically truncated to the maximum length. (boolean value)
#strict_password_check = false

# The HTTP header used to determine the scheme for the original request,
even
# if it was removed by an SSL terminating proxy. Typical value is
# "HTTP_X_FORWARDED_PROTO". (string value)
#secure_proxy_ssl_header = <None>

#
# From keystone.notifications
#

# Default publisher_id for outgoing notifications (string value)
#default_publisher_id = <None>

# Define the notification format for Identity Service events. A "basic"
# notification has information about the resource being operated on. A
"cadf"
# notification has the same information, as well as information about the
# initiator of the event. Valid options are: basic and cadf (string value)
#notification_format = basic

#
# From keystone.openstack.common.eventlet_backdoor
#

# Enable eventlet backdoor. Acceptable values are 0, <port>, and
# <start>:<end>, where 0 results in listening on a random tcp port number;
# <port> results in listening on the specified port number (and not
enabling
# backdoor if that port is in use); and <start>:<end> results in listening
on
# the smallest unused port number within the specified range of port
numbers.
# The chosen port is displayed in the service's log file. (string value)
#backdoor_port = <None>

#
# From oslo.log
#

# Print debugging output (set logging level to DEBUG instead of default
WARNING
# level). (boolean value)

```

```
#debug = false

# Print more verbose output (set logging level to INFO instead of default
# WARNING level). (boolean value)
#verbose = false

# The name of a logging configuration file. This file is appended to any
# existing logging configuration files. For details about logging
# configuration
# files, see the Python logging module documentation. (string value)
# Deprecated group/name - [DEFAULT]/log_config
#log_config_append = <None>

# DEPRECATED. A logging.Formatter log message format string which may use
# any
# of the available logging.LogRecord attributes. This option is
# deprecated.
# Please use logging_context_format_string and
# logging_default_format_string
# instead. (string value)
#log_format = <None>

# Format string for %(asctime)s in log records. Default: %(default)s .
# (string
# value)
#log_date_format = %Y-%m-%d %H:%M:%S

# (Optional) Name of log file to output to. If no default is set, logging
# will
# go to stdout. (string value)
# Deprecated group/name - [DEFAULT]/logfile
#log_file = <None>

# (Optional) The base directory used for relative --log-file paths.
# (string
# value)
# Deprecated group/name - [DEFAULT]/logdir
#log_dir = <None>

# Use syslog for logging. Existing syslog format is DEPRECATED during I,
# and
# will change in J to honor RFC5424. (boolean value)
#use_syslog = false

# (Optional) Enables or disables syslog rfc5424 format for logging. If
# enabled,
# prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The
# format without the APP-NAME is deprecated in I, and will be removed in
# J.
# (boolean value)
#use_syslog_rfc_format = false

# Syslog facility to receive log lines. (string value)
#syslog_log_facility = LOG_USER

# Log output to standard error. (boolean value)
```

```

#use_stderr = true

# Format string to use for log messages with context. (string value)
#logging_context_format_string = %(asctime)s.%(msecs)03d %(process)d %
(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%
(message)s

# Format string to use for log messages without context. (string value)
#logging_default_format_string = %(asctime)s.%(msecs)03d %(process)d %
(levelname)s %(name)s [-] %(instance)s%(message)s

# Data to append to log format when level is DEBUG. (string value)
#logging_debug_format_suffix = %(funcName)s %(pathname)s:%(lineno)d

# Prefix each line of exception output with this format. (string value)
#logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d TRACE %
(name)s %(instance)s

# List of logger=LEVEL pairs. (list value)
#default_log_levels =
amqp=WARN,amqplib=WARN,boto=WARN,qpid=WARN,sqlalchemy=WARN,suds=INFO,oslo.
messaging=INFO,iso8601=WARN,requests.packages.urllib3.connectionpool=WARN,
urllib3.connectionpool=WARN,websocket=WARN,requests.packages.urllib3.util.
retry=WARN,urllib3.util.retry=WARN,keystonemiddleware=WARN,routes.middlewa
re=WARN,stevedore=WARN

# Enables or disables publication of error events. (boolean value)
#publish_errors = false

# Enables or disables fatal status of deprecations. (boolean value)
#fatal_deprecations = false

# The format for an instance that is passed with the log message. (string
# value)
#instance_format = "[instance: %(uuid)s] "

# The format for an instance UUID that is passed with the log message.
(string
# value)
#instance_uuid_format = "[instance: %(uuid)s] "

#
# From oslo.messaging
#

# ZeroMQ bind address. Should be a wildcard (*), an ethernet interface, or
IP.
# The "host" option should point or resolve to this address. (string
value)
#rpc_zmq_bind_address = *

# MatchMaker driver. (string value)
#rpc_zmq_matchmaker =
oslo_messaging._drivers.matchmaker.MatchMakerLocalhost

# ZeroMQ receiver listening port. (integer value)

```

```
#rpc_zmq_port = 9501

# Number of ZeroMQ contexts, defaults to 1. (integer value)
#rpc_zmq_contexts = 1

# Maximum number of ingress messages to locally buffer per topic. Default
is
# unlimited. (integer value)
#rpc_zmq_topic_backlog = <None>

# Directory for holding IPC sockets. (string value)
#rpc_zmq_ipc_dir = /var/run/openstack

# Name of this node. Must be a valid hostname, FQDN, or IP address. Must
match
# "host" option, if running Nova. (string value)
#rpc_zmq_host = localhost

# Seconds to wait before a cast expires (TTL). Only supported by impl_zmq.
# (integer value)
#rpc_cast_timeout = 30

# Heartbeat frequency. (integer value)
#matchmaker_heartbeat_freq = 300

# Heartbeat time-to-live. (integer value)
#matchmaker_heartbeat_ttl = 600

# Size of RPC thread pool. (integer value)
#rpc_thread_pool_size = 64

# Driver or drivers to handle sending notifications. (multi valued)
#notification_driver =

# AMQP topic used for OpenStack notifications. (list value)
# Deprecated group/name - [rpc_notifier2]/topics
#notification_topics = notifications

# Seconds to wait for a response from a call. (integer value)
#rpc_response_timeout = 60

# A URL representing the messaging driver to use and its full
configuration. If
# not set, we fall back to the rpc_backend option and driver specific
# configuration. (string value)
#transport_url = <None>

# The messaging driver to use, defaults to rabbit. Other drivers include
qpid
# and zmq. (string value)
#rpc_backend = rabbit

# The default exchange under which topics are scoped. May be overridden by
an
# exchange name specified in the transport_url option. (string value)
#control_exchange = keystone
```

[assignment]

```
#
# From keystone
#

# Assignment backend driver. (string value)
#driver = <None>
```

[auth]

```
#
# From keystone
#

# Default auth methods. (list value)
#methods = external,password,token,oauth1

# The password auth plugin module. (string value)
#password = keystone.auth.plugins.password.Password

# The token auth plugin module. (string value)
#token = keystone.auth.plugins.token.Token

# The external (REMOTE_USER) auth plugin module. (string value)
#external = keystone.auth.plugins.external.DefaultDomain

# The OAuth1.0 auth plugin module. (string value)
#oauth1 = keystone.auth.plugins.oauth1.OAuth
```

[cache]

```
#
# From keystone
#

# Prefix for building the configuration dictionary for the cache region.
# This
# should not need to be changed unless there is another dogpile.cache
# region
# with the same configuration name. (string value)
#config_prefix = cache.keystone

# Default TTL, in seconds, for any cached item in the dogpile.cache
# region.
# This applies to any cached method that doesn't have an explicit cache
# expiration time defined for it. (integer value)
#expiration_time = 600

# Dogpile.cache backend module. It is recommended that Memcache with
# pooling
# (keystone.cache.memcache_pool) or Redis (dogpile.cache.redis) be used in
```

```
# production deployments. Small workloads (single process) like devstack
can
# use the dogpile.cache.memory backend. (string value)
#backend = keystone.common.cache.noop

# Arguments supplied to the backend module. Specify this option once per
# argument to be passed to the dogpile.cache backend. Example format:
# "<argname>:<value>". (multi valued)
#backend_argument =

# Proxy classes to import that will affect the way the dogpile.cache
backend
# functions. See the dogpile.cache documentation on changing-backend-
behavior.
# (list value)
#proxies =

# Global toggle for all caching using the should_cache_fn mechanism.
(boolean
# value)
#enabled = false

# Extra debugging from the cache backend (cache keys, get/set/delete/etc
# calls). This is only really useful if you need to see the specific
cache-
# backend get/set/delete calls with the keys/values. Typically this
should be
# left set to false. (boolean value)
#debug_cache_backend = false

# Memcache servers in the format of "host:port". (dogpile.cache.memcache
and
# keystone.cache.memcache_pool backends only). (list value)
#memcache_servers = localhost:11211

# Number of seconds memcached server is considered dead before it is tried
# again. (dogpile.cache.memcache and keystone.cache.memcache_pool backends
# only). (integer value)
#memcache_dead_retry = 300

# Timeout in seconds for every call to a server. (dogpile.cache.memcache
and
# keystone.cache.memcache_pool backends only). (integer value)
#memcache_socket_timeout = 3

# Max total number of open connections to every memcached server.
# (keystone.cache.memcache_pool backend only). (integer value)
#memcache_pool_maxsize = 10

# Number of seconds a connection to memcached is held unused in the pool
before
# it is closed. (keystone.cache.memcache_pool backend only). (integer
value)
#memcache_pool_unused_timeout = 60

# Number of seconds that an operation will wait to get a memcache client
```



```

# connection. (integer value)
#memcache_pool_connection_get_timeout = 10

[catalog]

#
# From keystone
#

# Catalog template file name for use with the template catalog backend.
(string
# value)
#template_file = default_catalog.templates

# Catalog backend driver. (string value)
#driver = keystone.catalog.backends.sql.Catalog

# Toggle for catalog caching. This has no effect unless global caching is
# enabled. (boolean value)
#caching = true

# Time to cache catalog data (in seconds). This has no effect unless
global and
# catalog caching are enabled. (integer value)
#cache_time = <None>

# Maximum number of entities that will be returned in a catalog
collection.
# (integer value)
#list_limit = <None>

[credential]

#
# From keystone
#

# Credential backend driver. (string value)
#driver = keystone.credential.backends.sql.Credential

[database]

#
# From oslo.db
#

# The file name to use with SQLite. (string value)
# Deprecated group/name - [DEFAULT]/sqlite_db
#sqlite_db = oslo.sqlite

# If True, SQLite uses synchronous mode. (boolean value)
# Deprecated group/name - [DEFAULT]/sqlite_synchronous
#sqlite_synchronous = true

```

```

# The back end to use for the database. (string value)
# Deprecated group/name - [DEFAULT]/db_backend
#backend = sqlalchemy

# The SQLAlchemy connection string to use to connect to the database.
(string
# value)
# Deprecated group/name - [DEFAULT]/sql_connection
# Deprecated group/name - [DATABASE]/sql_connection
# Deprecated group/name - [sql]/connection
#connection = <None>

# The SQLAlchemy connection string to use to connect to the slave
database.
# (string value)
#slave_connection = <None>

# The SQL mode to be used for MySQL sessions. This option, including the
# default, overrides any server-set SQL mode. To use whatever SQL mode is
set
# by the server configuration, set this to no value. Example:
mysql_sql_mode=
# (string value)
#mysql_sql_mode = TRADITIONAL

# Timeout before idle SQL connections are reaped. (integer value)
# Deprecated group/name - [DEFAULT]/sql_idle_timeout
# Deprecated group/name - [DATABASE]/sql_idle_timeout
# Deprecated group/name - [sql]/idle_timeout
#idle_timeout = 3600

# Minimum number of SQL connections to keep open in a pool. (integer
value)
# Deprecated group/name - [DEFAULT]/sql_min_pool_size
# Deprecated group/name - [DATABASE]/sql_min_pool_size
#min_pool_size = 1

# Maximum number of SQL connections to keep open in a pool. (integer
value)
# Deprecated group/name - [DEFAULT]/sql_max_pool_size
# Deprecated group/name - [DATABASE]/sql_max_pool_size
#max_pool_size = <None>

# Maximum number of database connection retries during startup. Set to -1
to
# specify an infinite retry count. (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_retries
# Deprecated group/name - [DATABASE]/sql_max_retries
#max_retries = 10

# Interval between retries of opening a SQL connection. (integer value)
# Deprecated group/name - [DEFAULT]/sql_retry_interval
# Deprecated group/name - [DATABASE]/reconnect_interval
#retry_interval = 10

```

```

# If set, use this value for max_overflow with SQLAlchemy. (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_overflow
# Deprecated group/name - [DATABASE]/sqlalchemy_max_overflow
#max_overflow = <None>

# Verbosity of SQL debugging information: 0=None, 100=Everything. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_connection_debug
#connection_debug = 0

# Add Python stack traces to SQL as comment strings. (boolean value)
# Deprecated group/name - [DEFAULT]/sql_connection_trace
#connection_trace = false

# If set, use this value for pool_timeout with SQLAlchemy. (integer value)
# Deprecated group/name - [DATABASE]/sqlalchemy_pool_timeout
#pool_timeout = <None>

# Enable the experimental use of database reconnect on connection lost.
# (boolean value)
#use_db_reconnect = false

# Seconds between retries of a database transaction. (integer value)
#db_retry_interval = 1

# If True, increases the interval between retries of a database operation
up to
# db_max_retry_interval. (boolean value)
#db_inc_retry_interval = true

# If db_inc_retry_interval is set, the maximum seconds between retries of
a
# database operation. (integer value)
#db_max_retry_interval = 10

# Maximum retries in case of connection error or deadlock error before
error is
# raised. Set to -1 to specify an infinite retry count. (integer value)
#db_max_retries = 20

[domain_config]

#
# From keystone
#

# Domain config backend driver. (string value)
#driver = keystone.resource.config_backends.sql.DomainConfig

# Toggle for domain config caching. This has no effect unless global
caching is
# enabled. (boolean value)
#caching = true

# TTL (in seconds) to cache domain config data. This has no effect unless

```

```
# domain config caching is enabled. (integer value)
#cache_time = 300

[endpoint_filter]

#
# From keystone
#

# Endpoint Filter backend driver (string value)
#driver = keystone.contrib.endpoint_filter.backends.sql.EndpointFilter

# Toggle to return all active endpoints if no filter exists. (boolean
value)
#return_all_endpoints_if_no_filter = true

[endpoint_policy]

#
# From keystone
#

# Endpoint policy backend driver (string value)
#driver = keystone.contrib.endpoint_policy.backends.sql.EndpointPolicy

[eventlet_server]

#
# From keystone
#

# The number of worker processes to serve the public eventlet application.
# Defaults to number of CPUs (minimum of 2). (integer value)
# Deprecated group/name - [DEFAULT]/public_workers
#public_workers = <None>

# The number of worker processes to serve the admin eventlet application.
# Defaults to number of CPUs (minimum of 2). (integer value)
# Deprecated group/name - [DEFAULT]/admin_workers
#admin_workers = <None>

# The IP address of the network interface for the public service to listen
on.
# (string value)
# Deprecated group/name - [DEFAULT]/bind_host
# Deprecated group/name - [DEFAULT]/public_bind_host
#public_bind_host = 0.0.0.0

# The port number which the public service listens on. (integer value)
# Deprecated group/name - [DEFAULT]/public_port
#public_port = 5000

# The IP address of the network interface for the admin service to listen
```

```

on.
# (string value)
# Deprecated group/name - [DEFAULT]/bind_host
# Deprecated group/name - [DEFAULT]/admin_bind_host
#admin_bind_host = 0.0.0.0

# The port number which the admin service listens on. (integer value)
# Deprecated group/name - [DEFAULT]/admin_port
#admin_port = 35357

# Set this to true if you want to enable TCP_KEEPALIVE on server sockets,
i.e.
# sockets used by the Keystone wsgi server for client connections.
(boolean
# value)
# Deprecated group/name - [DEFAULT]/tcp_keepalive
#tcp_keepalive = false

# Sets the value of TCP_KEEPIDLE in seconds for each server socket. Only
# applies if tcp_keepalive is true. (integer value)
# Deprecated group/name - [DEFAULT]/tcp_keepidle
#tcp_keepidle = 600

[eventlet_server_ssl]

#
# From keystone
#

# Toggle for SSL support on the Keystone eventlet servers. (boolean value)
# Deprecated group/name - [ssl]/enable
#enable = false

# Path of the certfile for SSL. For non-production environments, you may be
# interested in using `keystone-manage ssl_setup` to generate self-signed
# certificates. (string value)
# Deprecated group/name - [ssl]/certfile
#certfile = /etc/keystone/ssl/certs/keystone.pem

# Path of the keyfile for SSL. (string value)
# Deprecated group/name - [ssl]/keyfile
#keyfile = /etc/keystone/ssl/private/keystonekey.pem

# Path of the CA cert file for SSL. (string value)
# Deprecated group/name - [ssl]/ca_certs
#ca_certs = /etc/keystone/ssl/certs/ca.pem

# Require client certificate. (boolean value)
# Deprecated group/name - [ssl]/cert_required
#cert_required = false

[federation]

#

```

```

# From keystone
#

# Federation backend driver. (string value)
#driver = keystone.contrib.federation.backends.sql.Federation

# Value to be used when filtering assertion parameters from the
environment.
# (string value)
#assertion_prefix =

# Value to be used to obtain the entity ID of the Identity Provider from
the
# environment (e.g. if using the mod_shib plugin this value is `Shib-
Identity-
# Provider`). (string value)
#remote_id_attribute = <None>

# A domain name that is reserved to allow federated ephemeral users to
have a
# domain concept. Note that an admin will not be able to create a domain
with
# this name or update an existing domain to this name. You are not advised
to
# change this value unless you really have to. Changing this option to
empty
# string or None will not have any impact and default name will be used.
# (string value)
#federated_domain_name = Federated

# A list of trusted dashboard hosts. Before accepting a Single Sign-On
request
# to return a token, the origin host must be a member of the
trusted_dashboard
# list. This configuration option may be repeated for multiple values. For
# example: trusted_dashboard=http://acme.com
trusted_dashboard=http://beta.com
# (multi valued)
#trusted_dashboard =

# Location of Single Sign-On callback handler, will return a token to a
trusted
# dashboard host. (string value)
#sso_callback_template = /etc/keystone/sso_callback_template.html

[fernet_tokens]

#
# From keystone
#

# Directory containing Fernet token keys. (string value)
#key_repository = /etc/keystone/fernet-keys/

# This controls how many keys are held in rotation by keystone-manage

```

```

# fernet_rotate before they are discarded. The default value of 3 means
# that
# keystone will maintain one staged key, one primary key, and one secondary
# key. Increasing this value means that additional secondary keys will be
# kept
# in the rotation. (integer value)
#max_active_keys = 3

[identity]

#
# From keystone
#

# This references the domain to use for all Identity API v2 requests (which
# are
# not aware of domains). A domain with this ID will be created for you by
# keystone-manage db_sync in migration 008. The domain referenced by this
# ID
# cannot be deleted on the v3 API, to prevent accidentally breaking the v2
# API.
# There is nothing special about this domain, other than the fact that it
# must
# exist to order to maintain support for your v2 clients. (string value)
#default_domain_id = default

# A subset (or all) of domains can have their own identity driver, each
# with
# their own partial configuration options, stored in either the resource
# backend or in a file in a domain configuration directory (depending on
# the
# setting of domain_configurations_from_database). Only values specific to
# the
# domain need to be specified in this manner. This feature is disabled by
# default; set to true to enable. (boolean value)
#domain_specific_drivers_enabled = false

# Extract the domain specific configuration options from the resource
# backend
# where they have been stored with the domain data. This feature is
# disabled by
# default (in which case the domain specific options will be loaded from
# files
# in the domain configuration directory); set to true to enable. (boolean
# value)
#domain_configurations_from_database = false

# Path for Keystone to locate the domain specific identity configuration
# files
# if domain_specific_drivers_enabled is set to true. (string value)
#domain_config_dir = /etc/keystone/domains

# Identity backend driver. (string value)
#driver = keystone.identity.backends.sql.Identity

```

```

# Toggle for identity caching. This has no effect unless global caching is
# enabled. (boolean value)
#cacheing = true

# Time to cache identity data (in seconds). This has no effect unless
global
# and identity caching are enabled. (integer value)
#cache_time = 600

# Maximum supported length for user passwords; decrease to improve
performance.
# (integer value)
#max_password_length = 4096

# Maximum number of entities that will be returned in an identity
collection.
# (integer value)
#list_limit = <None>

[identity_mapping]

#
# From keystone
#

# Keystone Identity Mapping backend driver. (string value)
#driver = keystone.identity.mapping_backends.sql.Mapping

# Public ID generator for user and group entities. The Keystone identity
mapper
# only supports generators that produce no more than 64 characters.
(string
# value)
#generator = keystone.identity.id_generators.sha256.Generator

# The format of user and group IDs changed in Juno for backends that do
not
# generate UUIDs (e.g. LDAP), with keystone providing a hash mapping to
the
# underlying attribute in LDAP. By default this mapping is disabled, which
# ensures that existing IDs will not change. Even when the mapping is
enabled
# by using domain specific drivers, any users and groups from the default
# domain being handled by LDAP will still not be mapped to ensure their
IDs
# remain backward compatible. Setting this value to False will enable the
# mapping for even the default LDAP driver. It is only safe to do this if
you
# do not already have assignments for users and groups from the default
LDAP
# domain, and it is acceptable for Keystone to provide the different IDs
to
# clients than it did previously. Typically this means that the only time
you
# can set this value to False is when configuring a fresh installation.

```



```

# (boolean value)
#backward_compatible_ids = true

[kvs]

#
# From keystone
#

# Extra dogpile.cache backend modules to register with the dogpile.cache
# library. (list value)
#backends =

# Prefix for building the configuration dictionary for the KVS region. This
# should not need to be changed unless there is another dogpile.cache
# region
# with the same configuration name. (string value)
#config_prefix = keystone.kvs

# Toggle to disable using a key-mangling function to ensure fixed length
# keys.
# This is toggle-able for debugging purposes, it is highly recommended to
# always leave this set to true. (boolean value)
#enable_key_mangler = true

# Default lock timeout (in seconds) for distributed locking. (integer
# value)
#default_lock_timeout = 5

[ldap]

#
# From keystone
#

# URL for connecting to the LDAP server. (string value)
#url = ldap://localhost

# User BindDN to query the LDAP server. (string value)
#user = <None>

# Password for the BindDN to query the LDAP server. (string value)
#password = <None>

# LDAP server suffix (string value)
#suffix = cn=example,cn=com

# If true, will add a dummy member to groups. This is required if the
# objectclass for groups requires the "member" attribute. (boolean value)
#use_dumb_member = false

# DN of the "dummy member" to use when "use_dumb_member" is enabled.
# (string
# value)

```

```
#dumb_member = cn=dumb,dc=nonexistent

# Delete subtrees using the subtree delete control. Only enable this
option if
# your LDAP server supports subtree deletion. (boolean value)
#allow_subtree_delete = false

# The LDAP scope for queries, this can be either "one"
(onelevel/singleLevel)
# or "sub" (subtree/wholeSubtree). (string value)
#query_scope = one

# Maximum results per page; a value of zero ("0") disables paging.
(integer
# value)
#page_size = 0

# The LDAP dereferencing option for queries. This can be either "never",
# "searching", "always", "finding" or "default". The "default" option
falls
# back to using default dereferencing configured by your ldap.conf.
(string
# value)
#alias_dereferencing = default

# Sets the LDAP debugging level for LDAP calls. A value of 0 means that
# debugging is not enabled. This value is a bitmask, consult your LDAP
# documentation for possible values. (integer value)
#debug_level = <None>

# Override the system's default referral chasing behavior for queries.
(boolean
# value)
#chase_referrals = <None>

# Search base for users. (string value)
#user_tree_dn = <None>

# LDAP search filter for users. (string value)
#user_filter = <None>

# LDAP objectclass for users. (string value)
#user_objectclass = inetOrgPerson

# LDAP attribute mapped to user id. WARNING: must not be a multivalued
# attribute. (string value)
#user_id_attribute = cn

# LDAP attribute mapped to user name. (string value)
#user_name_attribute = sn

# LDAP attribute mapped to user email. (string value)
#user_mail_attribute = mail

# LDAP attribute mapped to password. (string value)
#user_pass_attribute = userPassword
```

```

# LDAP attribute mapped to user enabled flag. (string value)
#user_enabled_attribute = enabled

# Invert the meaning of the boolean enabled values. Some LDAP servers use
a
# boolean lock attribute where "true" means an account is disabled.
Setting
# "user_enabled_invert = true" will allow these lock attributes to be
used.
# This setting will have no effect if "user_enabled_mask" or
# "user_enabled_emulation" settings are in use. (boolean value)
#user_enabled_invert = false

# Bitmask integer to indicate the bit that the enabled value is stored in
if
# the LDAP server represents "enabled" as a bit on an integer rather than
a
# boolean. A value of "0" indicates the mask is not used. If this is not
set to
# "0" the typical value is "2". This is typically used when
# "user_enabled_attribute = userAccountControl". (integer value)
#user_enabled_mask = 0

# Default value to enable users. This should match an appropriate int
value if
# the LDAP server uses non-boolean (bitmask) values to indicate if a user
is
# enabled or disabled. If this is not set to "True" the typical value is
"512".
# This is typically used when "user_enabled_attribute =
userAccountControl".
# (string value)
#user_enabled_default = True

# List of attributes stripped off the user on update. (list value)
#user_attribute_ignore = default_project_id,tenants

# LDAP attribute mapped to default_project_id for users. (string value)
#user_default_project_id_attribute = <None>

# Allow user creation in LDAP backend. (boolean value)
#user_allow_create = true

# Allow user updates in LDAP backend. (boolean value)
#user_allow_update = true

# Allow user deletion in LDAP backend. (boolean value)
#user_allow_delete = true

# If true, Keystone uses an alternative method to determine if a user is
# enabled or not by checking if they are a member of the
# "user_enabled_emulation_dn" group. (boolean value)
#user_enabled_emulation = false

# DN of the group entry to hold enabled users when using enabled

```

```

emulation.
# (string value)
#user_enabled_emulation_dn = <None>

# List of additional LDAP attributes used for mapping additional attribute
# mappings for users. Attribute mapping format is <ldap_attr>:<user_attr>,
# where ldap_attr is the attribute in the LDAP entry and user_attr is the
# Identity API attribute. (list value)
#user_additional_attribute_mapping =

# Search base for projects (string value)
# Deprecated group/name - [ldap]/tenant_tree_dn
#project_tree_dn = <None>

# LDAP search filter for projects. (string value)
# Deprecated group/name - [ldap]/tenant_filter
#project_filter = <None>

# LDAP objectclass for projects. (string value)
# Deprecated group/name - [ldap]/tenant_objectclass
#project_objectclass = groupOfNames

# LDAP attribute mapped to project id. (string value)
# Deprecated group/name - [ldap]/tenant_id_attribute
#project_id_attribute = cn

# LDAP attribute mapped to project membership for user. (string value)
# Deprecated group/name - [ldap]/tenant_member_attribute
#project_member_attribute = member

# LDAP attribute mapped to project name. (string value)
# Deprecated group/name - [ldap]/tenant_name_attribute
#project_name_attribute = ou

# LDAP attribute mapped to project description. (string value)
# Deprecated group/name - [ldap]/tenant_desc_attribute
#project_desc_attribute = description

# LDAP attribute mapped to project enabled. (string value)
# Deprecated group/name - [ldap]/tenant_enabled_attribute
#project_enabled_attribute = enabled

# LDAP attribute mapped to project domain_id. (string value)
# Deprecated group/name - [ldap]/tenant_domain_id_attribute
#project_domain_id_attribute = businessCategory

# List of attributes stripped off the project on update. (list value)
# Deprecated group/name - [ldap]/tenant_attribute_ignore
#project_attribute_ignore =

# Allow project creation in LDAP backend. (boolean value)
# Deprecated group/name - [ldap]/tenant_allow_create
#project_allow_create = true

# Allow project update in LDAP backend. (boolean value)
# Deprecated group/name - [ldap]/tenant_allow_update

```

```

#project_allow_update = true

# Allow project deletion in LDAP backend. (boolean value)
# Deprecated group/name - [ldap]/tenant_allow_delete
#project_allow_delete = true

# If true, Keystone uses an alternative method to determine if a project
is
# enabled or not by checking if they are a member of the
# "project_enabled_emulation_dn" group. (boolean value)
# Deprecated group/name - [ldap]/tenant_enabled_emulation
#project_enabled_emulation = false

# DN of the group entry to hold enabled projects when using enabled
emulation.
# (string value)
# Deprecated group/name - [ldap]/tenant_enabled_emulation_dn
#project_enabled_emulation_dn = <None>

# Additional attribute mappings for projects. Attribute mapping format is
# <ldap_attr>:<user_attr>, where ldap_attr is the attribute in the LDAP
entry
# and user_attr is the Identity API attribute. (list value)
# Deprecated group/name - [ldap]/tenant_additional_attribute_mapping
#project_additional_attribute_mapping =

# Search base for roles. (string value)
#role_tree_dn = <None>

# LDAP search filter for roles. (string value)
#role_filter = <None>

# LDAP objectclass for roles. (string value)
#role_objectclass = organizationalRole

# LDAP attribute mapped to role id. (string value)
#role_id_attribute = cn

# LDAP attribute mapped to role name. (string value)
#role_name_attribute = ou

# LDAP attribute mapped to role membership. (string value)
#role_member_attribute = roleOccupant

# List of attributes stripped off the role on update. (list value)
#role_attribute_ignore =

# Allow role creation in LDAP backend. (boolean value)
#role_allow_create = true

# Allow role update in LDAP backend. (boolean value)
#role_allow_update = true

# Allow role deletion in LDAP backend. (boolean value)
#role_allow_delete = true

```

```
# Additional attribute mappings for roles. Attribute mapping format is
# <ldap_attr>:<user_attr>, where ldap_attr is the attribute in the LDAP
entry
# and user_attr is the Identity API attribute. (list value)
#role_additional_attribute_mapping =

# Search base for groups. (string value)
#group_tree_dn = <None>

# LDAP search filter for groups. (string value)
#group_filter = <None>

# LDAP objectclass for groups. (string value)
#group_objectclass = groupOfNames

# LDAP attribute mapped to group id. (string value)
#group_id_attribute = cn

# LDAP attribute mapped to group name. (string value)
#group_name_attribute = ou

# LDAP attribute mapped to show group membership. (string value)
#group_member_attribute = member

# LDAP attribute mapped to group description. (string value)
#group_desc_attribute = description

# List of attributes stripped off the group on update. (list value)
#group_attribute_ignore =

# Allow group creation in LDAP backend. (boolean value)
#group_allow_create = true

# Allow group update in LDAP backend. (boolean value)
#group_allow_update = true

# Allow group deletion in LDAP backend. (boolean value)
#group_allow_delete = true

# Additional attribute mappings for groups. Attribute mapping format is
# <ldap_attr>:<user_attr>, where ldap_attr is the attribute in the LDAP
entry
# and user_attr is the Identity API attribute. (list value)
#group_additional_attribute_mapping =

# CA certificate file path for communicating with LDAP servers. (string
value)
#tls_cacertfile = <None>

# CA certificate directory path for communicating with LDAP servers.
(string
# value)
#tls_cacertdir = <None>

# Enable TLS for communicating with LDAP servers. (boolean value)
#use_tls = false
```

```

# Valid options for tls_req_cert are demand, never, and allow. (string
value)
#tls_req_cert = demand

# Enable LDAP connection pooling. (boolean value)
#use_pool = false

# Connection pool size. (integer value)
#pool_size = 10

# Maximum count of reconnect trials. (integer value)
#pool_retry_max = 3

# Time span in seconds to wait between two reconnect trials. (floating
point
# value)
#pool_retry_delay = 0.1

# Connector timeout in seconds. Value -1 indicates indefinite wait for
# response. (integer value)
#pool_connection_timeout = -1

# Connection lifetime in seconds. (integer value)
#pool_connection_lifetime = 600

# Enable LDAP connection pooling for end user authentication. If use_pool
is
# disabled, then this setting is meaningless and is not used at all.
(boolean
# value)
#use_auth_pool = false

# End user auth connection pool size. (integer value)
#auth_pool_size = 100

# End user auth connection lifetime in seconds. (integer value)
#auth_pool_connection_lifetime = 60

[matchmaker_redis]

#
# From oslo.messaging
#

# Host to locate redis. (string value)
#host = 127.0.0.1

# Use this port to connect to redis host. (integer value)
#port = 6379

# Password for Redis server (optional). (string value)
#password = <None>

```

```
[matchmaker_ring]

#
# From oslo.messaging
#

# Matchmaker ring file (JSON). (string value)
# Deprecated group/name - [DEFAULT]/matchmaker_ringfile
#ringfile = /etc/oslo/matchmaker_ring.json


[memcache]

#
# From keystone
#

# Memcache servers in the format of "host:port". (list value)
#servers = localhost:11211

# Number of seconds memcached server is considered dead before it is tried
# again. This is used by the key value store system (e.g. token pooled
# memcached persistence backend). (integer value)
#dead_retry = 300

# Timeout in seconds for every call to a server. This is used by the key
# value
# store system (e.g. token pooled memcached persistence backend). (integer
# value)
#socket_timeout = 3

# Max total number of open connections to every memcached server. This is
# used
# by the key value store system (e.g. token pooled memcached persistence
# backend). (integer value)
#pool_maxsize = 10

# Number of seconds a connection to memcached is held unused in the pool
# before
# it is closed. This is used by the key value store system (e.g. token
# pooled
# memcached persistence backend). (integer value)
#pool_unused_timeout = 60

# Number of seconds that an operation will wait to get a memcache client
# connection. This is used by the key value store system (e.g. token
# pooled
# memcached persistence backend). (integer value)
#pool_connection_get_timeout = 10


[oauth1]

#
# From keystone
#
```



```

# Credential backend driver. (string value)
#driver = keystone.contrib.oauth1.backends.sql.OAuth1

# Duration (in seconds) for the OAuth Request Token. (integer value)
#request_token_duration = 28800

# Duration (in seconds) for the OAuth Access Token. (integer value)
#access_token_duration = 86400

[os_inherit]

#
# From keystone
#

# role-assignment inheritance to projects from owning domain or from
# projects
# higher in the hierarchy can be optionally enabled. (boolean value)
#enabled = false

[oslo_messaging_amqp]

#
# From oslo.messaging
#

# address prefix used when sending to a specific server (string value)
# Deprecated group/name - [amqp1]/server_request_prefix
#server_request_prefix = exclusive

# address prefix used when broadcasting to all servers (string value)
# Deprecated group/name - [amqp1]/broadcast_prefix
#broadcast_prefix = broadcast

# address prefix when sending to any server in group (string value)
# Deprecated group/name - [amqp1]/group_request_prefix
#group_request_prefix = unicast

# Name for the AMQP container (string value)
# Deprecated group/name - [amqp1]/container_name
#container_name = <None>

# Timeout for inactive connections (in seconds) (integer value)
# Deprecated group/name - [amqp1]/idle_timeout
#idle_timeout = 0

# Debug: dump AMQP frames to stdout (boolean value)
# Deprecated group/name - [amqp1]/trace
#trace = false

# CA certificate PEM file for verifying server certificate (string value)
# Deprecated group/name - [amqp1]/ssl_ca_file
#ssl_ca_file =

```

```

# Identifying certificate PEM file to present to clients (string value)
# Deprecated group/name - [amqp1]/ssl_cert_file
#ssl_cert_file =

# Private key PEM file used to sign cert_file certificate (string value)
# Deprecated group/name - [amqp1]/ssl_key_file
#ssl_key_file =

# Password for decrypting ssl_key_file (if encrypted) (string value)
# Deprecated group/name - [amqp1]/ssl_key_password
#ssl_key_password = <None>

# Accept clients using either SSL or plain TCP (boolean value)
# Deprecated group/name - [amqp1]/allow_insecure_clients
#allow_insecure_clients = false

[oslo_messaging_qpid]

#
# From oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
#amqp_durable_queues = false

# Auto-delete queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/amqp_auto_delete
#amqp_auto_delete = false

# Size of RPC connection pool. (integer value)
# Deprecated group/name - [DEFAULT]/rpc_conn_pool_size
#rpc_conn_pool_size = 30

# Qpid broker hostname. (string value)
# Deprecated group/name - [DEFAULT]/qpid_hostname
#qpid_hostname = localhost

# Qpid broker port. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_port
#qpid_port = 5672

# Qpid HA cluster host:port pairs. (list value)
# Deprecated group/name - [DEFAULT]/qpid_hosts
#qpid_hosts = $qpid_hostname:$qpid_port

# Username for Qpid connection. (string value)
# Deprecated group/name - [DEFAULT]/qpid_username
#qpid_username =

# Password for Qpid connection. (string value)
# Deprecated group/name - [DEFAULT]/qpid_password
#qpid_password =

```

```

# Space separated list of SASL mechanisms to use for auth. (string value)
# Deprecated group/name - [DEFAULT]/qpid_sasl_mechanisms
#qpid_sasl_mechanisms =

# Seconds between connection keepalive heartbeats. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_heartbeat
#qpid_heartbeat = 60

# Transport to use, either 'tcp' or 'ssl'. (string value)
# Deprecated group/name - [DEFAULT]/qpid_protocol
#qpid_protocol = tcp

# Whether to disable the Nagle algorithm. (boolean value)
# Deprecated group/name - [DEFAULT]/qpid_tcp_nodelay
#qpid_tcp_nodelay = true

# The number of prefetched messages held by receiver. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_receiver_capacity
#qpid_receiver_capacity = 1

# The qpid topology version to use. Version 1 is what was originally used
by
# impl_qpid. Version 2 includes some backwards-incompatible changes that
allow
# broker federation to work. Users should update to version 2 when they
are
# able to take everything down, as it requires a clean break. (integer
value)
# Deprecated group/name - [DEFAULT]/qpid_topology_version
#qpid_topology_version = 1

[oslo_messaging_rabbit]

#
# From oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
#amqp_durable_queues = false

# Auto-delete queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/amqp_auto_delete
#amqp_auto_delete = false

# Size of RPC connection pool. (integer value)
# Deprecated group/name - [DEFAULT]/rpc_conn_pool_size
#rpc_conn_pool_size = 30

# SSL version to use (valid only if SSL enabled). Valid values are TLSv1
and
# SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some
# distributions. (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_version
#kombu_ssl_version =

```

```
# SSL key file (valid only if SSL enabled). (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_keyfile
#kombu_ssl_keyfile =

# SSL cert file (valid only if SSL enabled). (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_certfile
#kombu_ssl_certfile =

# SSL certification authority file (valid only if SSL enabled). (string
value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_ca_certs
#kombu_ssl_ca_certs =

# How long to wait before reconnecting in response to an AMQP consumer
cancel
# notification. (floating point value)
# Deprecated group/name - [DEFAULT]/kombu_reconnect_delay
#kombu_reconnect_delay = 1.0

# The RabbitMQ broker address where a single node is used. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_host
#rabbit_host = localhost

# The RabbitMQ broker port where a single node is used. (integer value)
# Deprecated group/name - [DEFAULT]/rabbit_port
#rabbit_port = 5672

# RabbitMQ HA cluster host:port pairs. (list value)
# Deprecated group/name - [DEFAULT]/rabbit_hosts
#rabbit_hosts = $rabbit_host:$rabbit_port

# Connect over SSL for RabbitMQ. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_use_ssl
#rabbit_use_ssl = false

# The RabbitMQ userid. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_userid
#rabbit_userid = guest

# The RabbitMQ password. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_password
#rabbit_password = guest

# The RabbitMQ login method. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_login_method
#rabbit_login_method = AMQPPLAIN

# The RabbitMQ virtual host. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_virtual_host
#rabbit_virtual_host = /

# How frequently to retry connecting with RabbitMQ. (integer value)
#rabbit_retry_interval = 1

# How long to backoff for between retries when connecting to RabbitMQ.
```

```

(integer
# value)
# Deprecated group/name - [DEFAULT]/rabbit_retry_backoff
#rabbit_retry_backoff = 2

# Maximum number of RabbitMQ connection retries. Default is 0 (infinite
retry
# count). (integer value)
# Deprecated group/name - [DEFAULT]/rabbit_max_retries
#rabbit_max_retries = 0

# Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option,
you
# must wipe the RabbitMQ database. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_ha_queues
#rabbit_ha_queues = false

# Number of seconds after which the Rabbit broker is considered down if
# heartbeat's keep-alive fails (0 disable the heartbeat). (integer value)
#heartbeat_timeout_threshold = 60

# How often times during the heartbeat_timeout_threshold we check the
# heartbeat. (integer value)
#heartbeat_rate = 2

# Deprecated, use rpc_backend=kombu+memory or rpc_backend=fake (boolean
value)
# Deprecated group/name - [DEFAULT]/fake_rabbit
#fake_rabbit = false

[oslo_middleware]

#
# From oslo.middleware
#

# The maximum body size for each request, in bytes. (integer value)
# Deprecated group/name - [DEFAULT]/osapi_max_request_body_size
# Deprecated group/name - [DEFAULT]/max_request_body_size
#max_request_body_size = 114688

[oslo_policy]

#
# From oslo.policy
#

# The JSON file that defines policies. (string value)
# Deprecated group/name - [DEFAULT]/policy_file
#policy_file = policy.json

# Default rule. Enforced when a requested rule is not found. (string
value)
# Deprecated group/name - [DEFAULT]/policy_default_rule

```

```

#policy_default_rule = default

# Directories where policy configuration files are stored. They can be
relative
# to any directory in the search path defined by the config_dir option, or
# absolute paths. The file defined by policy_file must exist for these
# directories to be searched. Missing or empty directories are ignored.
(multi
# valued)
# Deprecated group/name - [DEFAULT]/policy_dirs
#policy_dirs = policy.d

[paste_deploy]

#
# From keystone
#

# Name of the paste configuration file that defines the available
pipelines.
# (string value)
#config_file = keystone-paste.ini

[policy]

#
# From keystone
#

# Policy backend driver. (string value)
#driver = keystone.policy.backends.sql.Policy

# Maximum number of entities that will be returned in a policy collection.
# (integer value)
#list_limit = <None>

[resource]

#
# From keystone
#

# Resource backend driver. If a resource driver is not specified, the
# assignment driver will choose the resource driver. (string value)
#driver = <None>

# Toggle for resource caching. This has no effect unless global caching is
# enabled. (boolean value)
# Deprecated group/name - [assignment]/caching
#caching = true

# TTL (in seconds) to cache resource data. This has no effect unless
global

```

```

# caching is enabled. (integer value)
# Deprecated group/name - [assignment]/cache_time
#cache_time = <None>

# Maximum number of entities that will be returned in a resource
collection.
# (integer value)
# Deprecated group/name - [assignment]/list_limit
#list_limit = <None>

[revoke]

#
# From keystone
#

# An implementation of the backend for persisting revocation events.
(string
# value)
#driver = keystone.contrib.revoke.backends.sql.Revoke

# This value (calculated in seconds) is added to token expiration before a
# revocation event may be removed from the backend. (integer value)
#expiration_buffer = 1800

# Toggle for revocation event caching. This has no effect unless global
caching
# is enabled. (boolean value)
#caching = true

# Time to cache the revocation list and the revocation events (in
seconds).
# This has no effect unless global and token caching are enabled. (integer
# value)
# Deprecated group/name - [token]/revocation_cache_time
#cache_time = 3600

[role]

#
# From keystone
#

# Role backend driver. (string value)
#driver = <None>

# Toggle for role caching. This has no effect unless global caching is
enabled.
# (boolean value)
#caching = true

# TTL (in seconds) to cache role data. This has no effect unless global
caching
# is enabled. (integer value)

```

```
#cache_time = <None>

# Maximum number of entities that will be returned in a role collection.
# (integer value)
#list_limit = <None>

[saml]

#
# From keystone
#

# Default TTL, in seconds, for any generated SAML assertion created by
# Keystone. (integer value)
#assertion_expiration_time = 3600

# Binary to be called for XML signing. Install the appropriate package,
# specify
# absolute path or adjust your PATH environment variable if the binary
# cannot
# be found. (string value)
#xmlsec1_binary = xmlsec1

# Path of the certfile for SAML signing. For non-production environments,
# you
# may be interested in using `keystone-manage pki_setup` to generate self-
# signed certificates. Note, the path cannot contain a comma. (string
# value)
#certfile = /etc/keystone/ssl/certs/signing_cert.pem

# Path of the keyfile for SAML signing. Note, the path cannot contain a
# comma.
# (string value)
#keyfile = /etc/keystone/ssl/private/signing_key.pem

# Entity ID value for unique Identity Provider identification. Usually
# FQDN is
# set with a suffix. A value is required to generate IDP Metadata. For
# example:
# https://keystone.example.com/v3/OS-FEDERATION/saml2/idp (string value)
#idp_entity_id = <None>

# Identity Provider Single-Sign-On service value, required in the Identity
# Provider's metadata. A value is required to generate IDP Metadata. For
# example: https://keystone.example.com/v3/OS-FEDERATION/saml2/sso (string
# value)
#idp_sso_endpoint = <None>

# Language used by the organization. (string value)
#idp_lang = en

# Organization name the installation belongs to. (string value)
#idp_organization_name = <None>

# Organization name to be displayed. (string value)
```



```

#idp_organization_display_name = <None>

# URL of the organization. (string value)
#idp_organization_url = <None>

# Company of contact person. (string value)
#idp_contact_company = <None>

# Given name of contact person (string value)
#idp_contact_name = <None>

# Surname of contact person. (string value)
#idp_contact_surname = <None>

# Email address of contact person. (string value)
#idp_contact_email = <None>

# Telephone number of contact person. (string value)
#idp_contact_telephone = <None>

# Contact type. Allowed values are: technical, support, administrative
billing,
# and other (string value)
#idp_contact_type = other

# Path to the Identity Provider Metadata file. This file should be
generated
# with the keystone-manage saml_idp_metadata command. (string value)
#idp_metadata_path = /etc/keystone/saml2_idp_metadata.xml

# The prefix to use for the RelayState SAML attribute, used when
generating ECP
# wrapped assertions. (string value)
#relay_state_prefix = ss:mem:

[signing]

#
# From keystone
#

# Path of the certfile for token signing. For non-production environments,
you
# may be interested in using `keystone-manage pki_setup` to generate self-
# signed certificates. (string value)
#certfile = /etc/keystone/ssl/certs/signing_cert.pem

# Path of the keyfile for token signing. (string value)
#keyfile = /etc/keystone/ssl/private/signing_key.pem

# Path of the CA for token signing. (string value)
#ca_certs = /etc/keystone/ssl/certs/ca.pem

# Path of the CA key for token signing. (string value)
#ca_key = /etc/keystone/ssl/private/cakey.pem

```

```

# Key size (in bits) for token signing cert (auto generated certificate).
# (integer value)
#key_size = 2048

# Days the token signing cert is valid for (auto generated certificate).
# (integer value)
#valid_days = 3650

# Certificate subject (auto generated certificate) for token signing.
(string
# value)
#cert_subject = /C=US/ST=Unset/L=Unset/O=Unset/CN=www.example.com

[ssl]

#
# From keystone
#

# Path of the CA key file for SSL. (string value)
#ca_key = /etc/keystone/ssl/private/cakey.pem

# SSL key length (in bits) (auto generated certificate). (integer value)
#key_size = 1024

# Days the certificate is valid for once signed (auto generated
certificate).
# (integer value)
#valid_days = 3650

# SSL certificate subject (auto generated certificate). (string value)
#cert_subject = /C=US/ST=Unset/L=Unset/O=Unset/CN=localhost

[token]

#
# From keystone
#

# External auth mechanisms that should add bind information to token,
e.g.,
# kerberos,x509. (list value)
#bind =

# Enforcement policy on tokens presented to Keystone with bind
information. One
# of disabled, permissive, strict, required or a specifically required
bind
# mode, e.g., kerberos or x509 to require binding to that authentication.
# (string value)
#enforce_token_bind = permissive

# Amount of time a token should remain valid (in seconds). (integer value)

```

```

#expiration = 3600

# Controls the token construction, validation, and revocation operations.
Core
# providers are "keystone.token.providers.
[fernet|pkiz|pki|uuid].Provider".
# (string value)
#provider = keystone.token.providers.uuid.Provider

# Token persistence backend driver. (string value)
#driver = keystone.token.persistence.backends.sql.Token

# Toggle for token system caching. This has no effect unless global
caching is
# enabled. (boolean value)
#caching = true

# Time to cache tokens (in seconds). This has no effect unless global and
token
# caching are enabled. (integer value)
#cache_time = <None>

# Revoke token by token identifier. Setting revoke_by_id to true enables
# various forms of enumerating tokens, e.g. `list tokens for user`. These
# enumerations are processed to determine the list of tokens to revoke.
Only
# disable if you are switching to using the Revoke extension with a
backend
# other than KVS, which stores events in memory. (boolean value)
#revoke_by_id = true

# Allow rescoping of scoped token. Setting allow_rescoped_scoped_token to
false
# prevents a user from exchanging a scoped token for any other token.
(boolean
# value)
#allow_rescope_scoped_token = true

# The hash algorithm to use for PKI tokens. This can be set to any
algorithm
# that hashlib supports. WARNING: Before changing this value, the
auth_token
# middleware must be configured with the hash_algorithms, otherwise token
# revocation will not be processed correctly. (string value)
#hash_algorithm = md5

[trust]

#
# From keystone
#

# Delegation and impersonation features can be optionally disabled.
(boolean
# value)

```

```
#enabled = true

# Enable redelegation feature. (boolean value)
#allow_redelegation = false

# Maximum depth of trust redelegation. (integer value)
#max_redelegation_count = 3

# Trust backend driver. (string value)
#driver = keystone.trust.backends.sql.Trust
```

7.2.2. keystone-paste.ini

Use the `keystone-paste.ini` file to configure the Web Service Gateway Interface (WSGI) middleware pipeline for the Identity service.

```
# Keystone PasteDeploy configuration file.

[filter:debug]
paste.filter_factory = keystone.common.wsgi.Debug.factory

[filter:request_id]
paste.filter_factory = oslo_middleware:RequestId.factory

[filter:build_auth_context]
paste.filter_factory = keystone.middleware:AuthContextMiddleware.factory

[filter:token_auth]
paste.filter_factory = keystone.middleware:TokenAuthMiddleware.factory

[filter:admin_token_auth]
paste.filter_factory =
keystone.middleware:AdminTokenAuthMiddleware.factory

[filter:json_body]
paste.filter_factory = keystone.middleware:JsonBodyMiddleware.factory

[filter:user_crud_extension]
paste.filter_factory = keystone.contrib.user_crud:CrudExtension.factory

[filter:crud_extension]
paste.filter_factory = keystone.contrib.admin_crud:CrudExtension.factory

[filter:ec2_extension]
paste.filter_factory = keystone.contrib.ec2:Ec2Extension.factory

[filter:ec2_extension_v3]
paste.filter_factory = keystone.contrib.ec2:Ec2ExtensionV3.factory

[filter:federation_extension]
paste.filter_factory =
keystone.contrib.federation.routers:FederationExtension.factory
```

```

[filter:oauth1_extension]
paste.filter_factory =
keystone.contrib.oauth1.routers:OAuth1Extension.factory

[filter:s3_extension]
paste.filter_factory = keystone.contrib.s3:S3Extension.factory

[filter:endpoint_filter_extension]
paste.filter_factory =
keystone.contrib.endpoint_filter.routers:EndpointFilterExtension.factory

[filter:endpoint_policy_extension]
paste.filter_factory =
keystone.contrib.endpoint_policy.routers:EndpointPolicyExtension.factory

[filter:simple_cert_extension]
paste.filter_factory =
keystone.contrib.simple_cert:SimpleCertExtension.factory

[filter:revoke_extension]
paste.filter_factory =
keystone.contrib.revoke.routers:RevokeExtension.factory

[filter:url_normalize]
paste.filter_factory = keystone.middleware:NormalizingFilter.factory

[filter:sizelimit]
paste.filter_factory =
oslo_middleware.sizelimit:RequestBodySizeLimiter.factory

[app:public_service]
paste.app_factory = keystone.service:public_app_factory

[app:service_v3]
paste.app_factory = keystone.service:v3_app_factory

[app:admin_service]
paste.app_factory = keystone.service:admin_app_factory

[pipeline:public_api]
# The last item in this pipeline must be public_service or an equivalent
# application. It cannot be a filter.
pipeline = sizelimit url_normalize request_id build_auth_context
token_auth admin_token_auth json_body ec2_extension user_crud_extension
public_service

[pipeline:admin_api]
# The last item in this pipeline must be admin_service or an equivalent
# application. It cannot be a filter.
pipeline = sizelimit url_normalize request_id build_auth_context
token_auth admin_token_auth json_body ec2_extension s3_extension
crud_extension admin_service

[pipeline:api_v3]
# The last item in this pipeline must be service_v3 or an equivalent

```

```
# application. It cannot be a filter.
pipeline = sizelimit url_normalize request_id build_auth_context
token_auth admin_token_auth json_body ec2_extension_v3 s3_extension
simple_cert_extension revoke_extension federation_extension
oauth1_extension endpoint_filter_extension endpoint_policy_extension
service_v3

[app:public_version_service]
paste.app_factory = keystone.service:public_version_app_factory

[app:admin_version_service]
paste.app_factory = keystone.service:admin_version_app_factory

[pipeline:public_version_api]
pipeline = sizelimit url_normalize public_version_service

[pipeline:admin_version_api]
pipeline = sizelimit url_normalize admin_version_service

[composite:main]
use = egg:Paste#urlmap
/v2.0 = public_api
/v3 = api_v3
/ = public_version_api

[composite:admin]
use = egg:Paste#urlmap
/v2.0 = admin_api
/v3 = api_v3
/ = admin_version_api
```

7.2.3. logging.conf

You can specify a special logging configuration file in the `keystone.conf` configuration file. For example, `/etc/keystone/logging.conf`.

For details, see the ([Python logging module documentation](#)).

```
[loggers]
keys=root,access

[handlers]
keys=production,file,access_file,devel

[formatters]
keys=minimal,normal,debug

#####
# Loggers #
#####
```

```

[logger_root]
level=WARNING
handlers=file

[logger_access]
level=INFO
qualname=access
handlers=access_file

#####
# Log Handlers #
#####

[handler_production]
class=handlers.SysLogHandler
level=ERROR
formatter=normal
args=('localhost', handlers.SYSLOG_UDP_PORT),
handlers.SysLogHandler.LOG_USER)

[handler_file]
class=handlers.WatchedFileHandler
level=WARNING
formatter=normal
args=('error.log',)

[handler_access_file]
class=handlers.WatchedFileHandler
level=INFO
formatter=minimal
args=('access.log',)

[handler_devel]
class=StreamHandler
level=NOTSET
formatter=debug
args=(sys.stdout,)

#####
# Log Formatters #
#####

[formatter_minimal]
format=%(message)s

[formatter_normal]
format=%(name)s: %(asctime)s %(levelname)s %(message)s

[formatter_debug]
format=%(name)s: %(asctime)s %(levelname)s %(module)s %(funcName)s %
(message)s

```

7.2.4. policy.json

Use the `policy.json` file to define additional access controls that apply to the Identity service.

```
{
  "admin_required": "role:admin or is_admin:1",
  "service_role": "role:service",
  "service_or_admin": "rule:admin_required or rule:service_role",
  "owner" : "user_id:%(user_id)s",
  "admin_or_owner": "rule:admin_required or rule:owner",
  "token_subject": "user_id:%(target.token.user_id)s",
  "admin_or_token_subject": "rule:admin_required or rule:token_subject",

  "default": "rule:admin_required",

  "identity:get_region": "",
  "identity:list_regions": "",
  "identity:create_region": "rule:admin_required",
  "identity:update_region": "rule:admin_required",
  "identity:delete_region": "rule:admin_required",

  "identity:get_service": "rule:admin_required",
  "identity:list_services": "rule:admin_required",
  "identity:create_service": "rule:admin_required",
  "identity:update_service": "rule:admin_required",
  "identity:delete_service": "rule:admin_required",

  "identity:get_endpoint": "rule:admin_required",
  "identity:list_endpoints": "rule:admin_required",
  "identity:create_endpoint": "rule:admin_required",
  "identity:update_endpoint": "rule:admin_required",
  "identity:delete_endpoint": "rule:admin_required",

  "identity:get_domain": "rule:admin_required",
  "identity:list_domains": "rule:admin_required",
  "identity:create_domain": "rule:admin_required",
  "identity:update_domain": "rule:admin_required",
  "identity:delete_domain": "rule:admin_required",

  "identity:get_project": "rule:admin_required",
  "identity:list_projects": "rule:admin_required",
  "identity:list_user_projects": "rule:admin_or_owner",
  "identity:create_project": "rule:admin_required",
  "identity:update_project": "rule:admin_required",
  "identity:delete_project": "rule:admin_required",

  "identity:get_user": "rule:admin_required",
  "identity:list_users": "rule:admin_required",
  "identity:create_user": "rule:admin_required",
  "identity:update_user": "rule:admin_required",
  "identity:delete_user": "rule:admin_required",
  "identity:change_password": "rule:admin_or_owner",

  "identity:get_group": "rule:admin_required",
  "identity:list_groups": "rule:admin_required",
  "identity:list_groups_for_user": "rule:admin_or_owner",
```



```

"identity:create_group": "rule:admin_required",
"identity:update_group": "rule:admin_required",
"identity:delete_group": "rule:admin_required",
"identity:list_users_in_group": "rule:admin_required",
"identity:remove_user_from_group": "rule:admin_required",
"identity:check_user_in_group": "rule:admin_required",
"identity:add_user_to_group": "rule:admin_required",

"identity:get_credential": "rule:admin_required",
"identity:list_credentials": "rule:admin_required",
"identity:create_credential": "rule:admin_required",
"identity:update_credential": "rule:admin_required",
"identity:delete_credential": "rule:admin_required",

"identity:ec2_get_credential": "rule:admin_required or (rule:owner and
user_id:%(target.credential.user_id)s)",
"identity:ec2_list_credentials": "rule:admin_or_owner",
"identity:ec2_create_credential": "rule:admin_or_owner",
"identity:ec2_delete_credential": "rule:admin_required or (rule:owner
and user_id:%(target.credential.user_id)s)",

"identity:get_role": "rule:admin_required",
"identity:list_roles": "rule:admin_required",
"identity:create_role": "rule:admin_required",
"identity:update_role": "rule:admin_required",
"identity:delete_role": "rule:admin_required",

"identity:check_grant": "rule:admin_required",
"identity:list_grants": "rule:admin_required",
"identity:create_grant": "rule:admin_required",
"identity:revoke_grant": "rule:admin_required",

"identity:list_role_assignments": "rule:admin_required",

"identity:get_policy": "rule:admin_required",
"identity:list_policies": "rule:admin_required",
"identity:create_policy": "rule:admin_required",
"identity:update_policy": "rule:admin_required",
"identity:delete_policy": "rule:admin_required",

"identity:check_token": "rule:admin_required",
"identity:validate_token": "rule:service_or_admin",
"identity:validate_token_head": "rule:service_or_admin",
"identity:revocation_list": "rule:service_or_admin",
"identity:revoke_token": "rule:admin_or_token_subject",

"identity:create_trust": "user_id:%(trust.trustor_user_id)s",
"identity:get_trust": "rule:admin_or_owner",
"identity:list_trusts": "",
"identity:list_roles_for_trust": "",
"identity:get_role_for_trust": "",
"identity:delete_trust": "",

"identity:create_consumer": "rule:admin_required",
"identity:get_consumer": "rule:admin_required",
"identity:list_consumers": "rule:admin_required",

```

```

"identity:delete_consumer": "rule:admin_required",
"identity:update_consumer": "rule:admin_required",

"identity:authorize_request_token": "rule:admin_required",
"identity:list_access_token_roles": "rule:admin_required",
"identity:get_access_token_role": "rule:admin_required",
"identity:list_access_tokens": "rule:admin_required",
"identity:get_access_token": "rule:admin_required",
"identity:delete_access_token": "rule:admin_required",

"identity:list_projects_for_endpoint": "rule:admin_required",
"identity:add_endpoint_to_project": "rule:admin_required",
"identity:check_endpoint_in_project": "rule:admin_required",
"identity:list_endpoints_for_project": "rule:admin_required",
"identity:remove_endpoint_from_project": "rule:admin_required",

"identity:create_endpoint_group": "rule:admin_required",
"identity:list_endpoint_groups": "rule:admin_required",
"identity:get_endpoint_group": "rule:admin_required",
"identity:update_endpoint_group": "rule:admin_required",
"identity:delete_endpoint_group": "rule:admin_required",
"identity:list_projects_associated_with_endpoint_group":
"rule:admin_required",
  "identity:list_endpoints_associated_with_endpoint_group":
"rule:admin_required",
  "identity:get_endpoint_group_in_project": "rule:admin_required",
  "identity:add_endpoint_group_to_project": "rule:admin_required",
  "identity:remove_endpoint_group_from_project": "rule:admin_required",

"identity:create_identity_provider": "rule:admin_required",
"identity:list_identity_providers": "rule:admin_required",
"identity:get_identity_providers": "rule:admin_required",
"identity:update_identity_provider": "rule:admin_required",
"identity:delete_identity_provider": "rule:admin_required",

"identity:create_protocol": "rule:admin_required",
"identity:update_protocol": "rule:admin_required",
"identity:get_protocol": "rule:admin_required",
"identity:list_protocols": "rule:admin_required",
"identity:delete_protocol": "rule:admin_required",

"identity:create_mapping": "rule:admin_required",
"identity:get_mapping": "rule:admin_required",
"identity:list_mappings": "rule:admin_required",
"identity:delete_mapping": "rule:admin_required",
"identity:update_mapping": "rule:admin_required",

"identity:create_service_provider": "rule:admin_required",
"identity:list_service_providers": "rule:admin_required",
"identity:get_service_provider": "rule:admin_required",
"identity:update_service_provider": "rule:admin_required",
"identity:delete_service_provider": "rule:admin_required",

"identity:get_auth_catalog": "",
"identity:get_auth_projects": "",
"identity:get_auth_domains": "",

```

```

    "identity:list_projects_for_groups": "",
    "identity:list_domains_for_groups": "",

    "identity:list_revoke_events": "",

    "identity:create_policy_association_for_endpoint":
"rule:admin_required",
    "identity:check_policy_association_for_endpoint":
"rule:admin_required",
    "identity:delete_policy_association_for_endpoint":
"rule:admin_required",
    "identity:create_policy_association_for_service":
"rule:admin_required",
    "identity:check_policy_association_for_service":
"rule:admin_required",
    "identity:delete_policy_association_for_service":
"rule:admin_required",
    "identity:create_policy_association_for_region_and_service":
"rule:admin_required",
    "identity:check_policy_association_for_region_and_service":
"rule:admin_required",
    "identity:delete_policy_association_for_region_and_service":
"rule:admin_required",
    "identity:get_policy_for_endpoint": "rule:admin_required",
    "identity:list_endpoints_for_policy": "rule:admin_required",

    "identity:create_domain_config": "rule:admin_required",
    "identity:get_domain_config": "rule:admin_required",
    "identity:update_domain_config": "rule:admin_required",
    "identity:delete_domain_config": "rule:admin_required"
}

```

7.2.5. Domain-specific configuration

The Identity service enables you to configure domain-specific authentication drivers. For example, you can configure a domain to have its own LDAP or SQL server.

By default, the option to configure domain-specific drivers is disabled.

To enable domain-specific drivers, set these options in the `[identity]` section in the `keystone.conf` file:

```

[identity]
domain_specific_drivers_enabled = True
domain_config_dir = /etc/keystone/domains

```

When you enable domain-specific drivers, the Identity service looks in the `domain_config_dir` directory for configuration files that are named as follows: `keystone.DOMAIN_NAME.conf`, where `DOMAIN_NAME` is the domain name.

Any options that you define in the domain-specific configuration file override options in the primary configuration file for the specified domain. Any domain without a domain-specific configuration file uses only the options in the primary configuration file.

7.3. NEW, UPDATED AND DEPRECATED OPTIONS IN KILO FOR OPENSTACK IDENTITY

Table 7.37. New options

Option = default value	(Type) Help string
[DEFAULT] executor_thread_pool_size = 64	(IntOpt) Size of executor thread pool.
[DEFAULT] host = 127.0.0.1	(StrOpt) Host to locate redis.
[DEFAULT] password =	(StrOpt) Password for Redis server (optional).
[DEFAULT] port = 6379	(IntOpt) Use this port to connect to redis host.
[DEFAULT] rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
[DEFAULT] rpc_poll_timeout = 1	(IntOpt) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
[DEFAULT] rpc_zmq_all_req_rep = True	(BoolOpt) Use REQ/REP pattern for all methods CALL/CAST/FANOUT.
[DEFAULT] rpc_zmq_concurrency = eventlet	(StrOpt) Type of concurrency used. Either "native" or "eventlet"
[DEFAULT] watch_log_file = False	(BoolOpt) (Optional) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log-file option is specified and Linux platform is used. This option is ignored if log_config_append is set.
[DEFAULT] zmq_use_broker = True	(BoolOpt) Shows whether zmq-messaging uses broker or not.
[cors] allow_credentials = True	(BoolOpt) Indicate that the actual request can include user credentials
[cors] allow_headers = Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma	(ListOpt) Indicate which header field names may be used during the actual request.
[cors] allow_methods = GET, POST, PUT, DELETE, OPTIONS	(ListOpt) Indicate which methods can be used during the actual request.

Option = default value	(Type) Help string
[cors] allowed_origin = None	(StrOpt) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
[cors] expose_headers = Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma	(ListOpt) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
[cors] max_age = 3600	(IntOpt) Maximum cache age of CORS preflight requests.
[cors.subdomain] allow_credentials = True	(BoolOpt) Indicate that the actual request can include user credentials
[cors.subdomain] allow_headers = Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma	(ListOpt) Indicate which header field names may be used during the actual request.
[cors.subdomain] allow_methods = GET, POST, PUT, DELETE, OPTIONS	(ListOpt) Indicate which methods can be used during the actual request.
[cors.subdomain] allowed_origin = None	(StrOpt) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
[cors.subdomain] expose_headers = Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma	(ListOpt) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
[cors.subdomain] max_age = 3600	(IntOpt) Maximum cache age of CORS preflight requests.
[endpoint_policy] enabled = True	(BoolOpt) Enable endpoint_policy functionality.
[keystone_authtoken] region_name = None	(StrOpt) The region in which the identity server can be found.
[oslo_messaging_amqp] password =	(StrOpt) Password for message broker authentication
[oslo_messaging_amqp] sasl_config_dir =	(StrOpt) Path to directory that contains the SASL configuration
[oslo_messaging_amqp] sasl_config_name =	(StrOpt) Name of configuration file (without .conf suffix)
[oslo_messaging_amqp] sasl_mechanisms =	(StrOpt) Space separated list of acceptable SASL mechanisms

Option = default value	(Type) Help string
[oslo_messaging_amqp] username =	(StrOpt) User name for message broker authentication
[oslo_messaging_qpid] send_single_reply = False	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>
[oslo_messaging_rabbit] kombu_reconnect_timeout = 60	(IntOpt) How long to wait before considering a reconnect attempt to have failed. This value should not be longer than rpc_response_timeout.
[oslo_messaging_rabbit] send_single_reply = False	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>
[oslo_middleware] secure_proxy_ssl_header = X-Forwarded-Proto	(StrOpt) The HTTP Header that will be used to determine what the original request protocol scheme was, even if it was hidden by an SSL termination proxy.
[tokenless_auth] issuer_attribute = SSL_CLIENT_I_DN	(StrOpt) The issuer attribute that is served as an IdP ID for the X.509 tokenless authorization along with the protocol to look up its corresponding mapping. It is the environment variable in the WSGI environment that references to the issuer of the client certificate.
[tokenless_auth] protocol = x509	(StrOpt) The protocol name for the X.509 tokenless authorization along with the option issuer_attribute below can look up its corresponding mapping.

Option = default value	(Type) Help string
[tokenless_auth] trusted_issuer = []	(MultiStrOpt) The list of trusted issuers to further filter the certificates that are allowed to participate in the X.509 tokenless authorization. If the option is absent then no certificates will be allowed. The naming format for the attributes of a Distinguished Name(DN) must be separated by a comma and contain no spaces. This configuration option may be repeated for multiple values. For example: trusted_issuer=CN=john,OU=keystone,O=openstack trusted_issuer=CN=mary,OU=eng,O=abc

Table 7.38. New default values

Option	Previous default value	New default value
[DEFAULT] crypt_strength	40000	10000
[DEFAULT] default_log_levels	amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystone.middleware=WARNING, routes.middleware=WARNING, stevedore=WARNING	amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystone.middleware=WARNING, routes.middleware=WARNING, stevedore=WARNING, taskflow=WARNING
[DEFAULT] logging_exception_prefix	%(asctime)s.%(msecs)03d %(process)d TRACE %(name)s %(instance)s	%(asctime)s.%(msecs)03d %(process)d ERROR %(name)s %(instance)s
[DEFAULT] rpc_zmq_matchmaker	local	redis
[DEFAULT] use_syslog_rfc_format	False	True
[DEFAULT] verbose	False	True
[auth] external	keystone.auth.plugins.external.DefaultDomain	None

Option	Previous default value	New default value
[auth] oauth1	keystone.auth.plugins.oauth1.OAuth	None
[auth] password	keystone.auth.plugins.password.Password	None
[auth] token	keystone.auth.plugins.token.Token	None
[catalog] driver	keystone.catalog.backends.sql.Catalog	sql
[credential] driver	keystone.credential.backends.sql.Credential	sql
[domain_config] driver	keystone.resource.config_backends.sql.DomainConfig	sql
[endpoint_filter] driver	keystone.contrib.endpoint_filter.backends.sql.EndpointFilter	sql
[endpoint_policy] driver	keystone.contrib.endpoint_policy.backends.sql.EndpointPolicy	sql
[federation] driver	keystone.contrib.federation.backends.sql.Federation	sql
[identity] driver	keystone.identity.backends.sql.Identity	sql
[identity_mapping] driver	keystone.identity.mapping_backends.sql.Mapping	sql
[identity_mapping] generator	keystone.identity.id_generators.sha256.Generator	sha256
[ldap] user_attribute_ignore	default_project_id, tenants	default_project_id
[matchmaker_redis] password	None	
[oauth1] driver	keystone.contrib.oauth1.backends.sql.OAuth1	sql
[oslo_messaging_rabbit] heartbeat_timeout_threshold	0	60

Option	Previous default value	New default value
[policy] driver	keystone.policy.backends.sql.Policy	sql
[revoke] driver	keystone.contrib.revoke.backends.sql.Revoke	sql
[token] driver	keystone.token.persistence.backends.sql.Token	sql
[token] provider	keystone.token.providers.uuid.Provider	uuid
[trust] driver	keystone.trust.backends.sql.Trust	sql

Table 7.39. Deprecated options

Deprecated option	New Option
[DEFAULT] use_syslog	None
[DEFAULT] log_format	None
[DEFAULT] rpc_thread_pool_size	[DEFAULT] executor_thread_pool_size

CHAPTER 8. IMAGE SERVICE

Compute relies on an external image service to store virtual machine images and maintain a catalog of available images. By default, Compute is configured to use the OpenStack Image service (glance), which is currently the only supported image service.

If your installation requires `euca2ools` to register new images, you must run the `nova-objectstore` service. This service provides an Amazon S3 front-end for Image service, which is required by `euca2ools`.

To customize the Compute service, use the configuration option settings documented in [Table 3.27, “Description of glance configuration options”](#) and [Table 3.47, “Description of S3 configuration options”](#).

You can modify many options in the OpenStack Image service. The following tables provide a comprehensive list.

Table 8.1. Description of authorization token configuration options

Configuration option = Default value	Description
<code>[keystone_authtoken]</code>	
<code>admin_password = None</code>	(String) Service user password.
<code>admin_tenant_name = admin</code>	(String) Service tenant name.
<code>admin_token = None</code>	(String) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
<code>admin_user = None</code>	(String) Service username.
<code>auth_admin_prefix =</code>	(String) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
<code>auth_host = 127.0.0.1</code>	(String) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
<code>auth_port = 35357</code>	(Integer) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
<code>auth_protocol = https</code>	(String) Protocol of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
<code>auth_section = None</code>	(Unknown) Config Section from which to load plugin specific options

Configuration option = Default value	Description
auth_type = <i>None</i>	(Unknown) Authentication type to load
auth_uri = <i>None</i>	(String) Complete public Identity API endpoint.
auth_version = <i>None</i>	(String) API version of the admin Identity API endpoint.
cache = <i>None</i>	(String) Env key for the swift cache.
cafile = <i>None</i>	(String) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(String) Required if identity server requires client certificate
check_revocations_for_cached = <i>False</i>	(Boolean) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.
delay_auth_decision = <i>False</i>	(Boolean) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(String) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(List) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(Integer) Request timeout value for communicating with Identity API server.

Configuration option = Default value	Description
http_request_max_retries = 3	(Integer) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(String) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/
include_service_catalog = <i>True</i>	(Boolean) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(Boolean) Verify HTTPS connections.
keyfile = <i>None</i>	(String) Required if identity server requires client certificate
memcache_pool_conn_get_timeout = 10	(Integer) (Optional) Number of seconds that an operation will wait to get a memcached client connection from the pool.
memcache_pool_dead_retry = 300	(Integer) (Optional) Number of seconds memcached server is considered dead before it is tried again.
memcache_pool_maxsize = 10	(Integer) (Optional) Maximum total number of open connections to every memcached server.
memcache_pool_socket_timeout = 3	(Integer) (Optional) Socket timeout in seconds for communicating with a memcached server.
memcache_pool_unused_timeout = 60	(Integer) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
memcache_secret_key = <i>None</i>	(String) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
memcache_security_strategy = <i>None</i>	(String) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.

Configuration option = Default value	Description
memcache_use_advanced_pool = <i>False</i>	(Boolean) (Optional) Use the advanced (eventlet safe) memcached client pool. The advanced pool will only work under python 2.x.
region_name = <i>None</i>	(String) The region in which the identity server can be found.
revocation_cache_time = <i>10</i>	(Integer) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
signing_dir = <i>None</i>	(String) Directory used to cache files related to PKI tokens.
token_cache_time = <i>300</i>	(Integer) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 8.2. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
allow_additional_image_properties = <i>True</i>	(Boolean) Whether to allow users to specify image properties beyond what the image schema provides
api_limit_max = <i>1000</i>	(Integer) Maximum permissible number of items that could be returned by a request
backlog = <i>4096</i>	(Integer) The backlog value that will be used when creating the TCP listener socket.
bind_host = <i>0.0.0.0</i>	(String) Address to bind the server. Useful when selecting a particular network interface.
bind_port = <i>None</i>	(Port number) The port on which the server will listen.
data_api = <i>glance.db.sqlalchemy.api</i>	(String) Python module path of data access API

Configuration option = Default value	Description
digest_algorithm = <i>sha256</i>	(String) Digest algorithm which will be used for digital signature. Use the command "openssl list-message-digest-algorithms" to get the available algorithms supported by the version of OpenSSL on the platform. Examples are "sha1", "sha256", "sha512", etc.
executor_thread_pool_size = <i>64</i>	(Integer) Size of executor thread pool.
image_location_quota = <i>10</i>	(Integer) Maximum number of locations allowed on an image. Negative values evaluate to unlimited.
image_member_quota = <i>128</i>	(Integer) Maximum number of image members per image. Negative values evaluate to unlimited.
image_property_quota = <i>128</i>	(Integer) Maximum number of properties allowed on an image. Negative values evaluate to unlimited.
image_tag_quota = <i>128</i>	(Integer) Maximum number of tags allowed on an image. Negative values evaluate to unlimited.
limit_param_default = <i>25</i>	(Integer) Default value for the number of items returned by a request if not specified explicitly in the request
memcached_servers = <i>None</i>	(List) Memcached servers or None for in process cache.
metadata_encryption_key = <i>None</i>	(String) AES key for encrypting store 'location' metadata. This includes, if used, Swift or S3 credentials. Should be set to a random string of length 16, 24 or 32 bytes
metadata_source_path = <i>/etc/glance/metadefs/</i>	(String) Path to the directory where json metadata files are stored
property_protection_file = <i>None</i>	(String) The location of the property protection file. This file contains the rules for property protections and the roles/policies associated with it. If this config value is not specified, by default, property protections won't be enforced. If a value is specified and the file is not found, then the glance-api service will not start.
property_protection_rule_format = <i>roles</i>	(String) This config value indicates whether "roles" or "policies" are used in the property protection file.

Configuration option = Default value	Description
show_image_direct_url = <i>False</i>	(Boolean) Whether to include the backend image storage location in image properties. Revealing storage location can be a security risk, so use this setting with caution!
user_storage_quota = <i>0</i>	(String) Set a system wide quota for every user. This value is the total capacity that a user can use across all storage systems. A value of 0 means unlimited. Optional unit can be specified for the value. Accepted units are B, KB, MB, GB and TB representing Bytes, KiloBytes, MegaBytes, GigaBytes and TeraBytes respectively. If no unit is specified then Bytes is assumed. Note that there should not be any space between value and unit and units are case sensitive.
workers = <i>None</i>	(Integer) The number of child process workers that will be created to service requests. The default will be equal to the number of CPUs available.
[glance_store]	
rootwrap_config = <i>/etc/glance/rootwrap.conf</i>	(String) Path to the rootwrap configuration file to use for running commands as root.
[image_format]	
container_formats = <i>ami, ari, aki, bare, ovf, ova, docker</i>	(List) Supported values for the 'container_format' image attribute
disk_formats = <i>ami, ari, aki, vhd, vmdk, raw, qcow2, vdi, iso</i>	(List) Supported values for the 'disk_format' image attribute
[keystone_auth_token]	
memcached_servers = <i>None</i>	(List) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.
[task]	
task_executor = <i>taskflow</i>	(String) Specifies which task executor to be used to run the task scripts.
task_time_to_live = <i>48</i>	(Integer) Time in hours for which a task lives after, either succeeding or failing

Configuration option = Default value	Description
work_dir = <i>None</i>	(String) Work dir for asynchronous task operations. The directory set here will be used to operate over images - normally before they are imported in the destination store. When providing work dir, make sure enough space is provided for concurrent tasks to run efficiently without running out of space. A rough estimation can be done by multiplying the number of <code>max_workers`</code> - or the N of workers running - by an average image size (e.g 500MB). The image size estimation should be done based on the average size in your deployment. Note that depending on the tasks running you may need to multiply this number by some factor depending on what the task does. For example, you may want to double the available size if image conversion is enabled. All this being said, remember these are just estimations and you should do them based on the worst case scenario and be prepared to act in case they were wrong.

Table 8.3. Description of CORS configuration options

Configuration option = Default value	Description
[cors]	
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(Integer) Maximum cache age of CORS preflight requests.
[cors.subdomain]	

Configuration option = Default value	Description
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(Integer) Maximum cache age of CORS preflight requests.

Table 8.4. Description of database configuration options

Configuration option = Default value	Description
[database]	
backend = <i>sqlalchemy</i>	(String) The back end to use for the database.
connection = <i>None</i>	(String) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(Integer) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(Boolean) Add Python stack traces to SQL as comment strings.
db_inc_retry_interval = <i>True</i>	(Boolean) If True, increases the interval between retries of a database operation up to db_max_retry_interval .
db_max_retries = <i>20</i>	(Integer) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = <i>10</i>	(Integer) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.

Configuration option = Default value	Description
db_retry_interval = 1	(Integer) Seconds between retries of a database transaction.
idle_timeout = 3600	(Integer) Timeout before idle SQL connections are reaped.
max_overflow = 50	(Integer) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = None	(Integer) Maximum number of SQL connections to keep open in a pool.
max_retries = 10	(Integer) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = 1	(Integer) Minimum number of SQL connections to keep open in a pool.
mysql_sql_mode = <i>TRADITIONAL</i>	(String) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode=
pool_timeout = None	(Integer) If set, use this value for pool_timeout with SQLAlchemy.
retry_interval = 10	(Integer) Interval between retries of opening a SQL connection.
slave_connection = None	(String) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = <i>oslo.sqlite</i>	(String) The file name to use with SQLite.
sqlite_synchronous = <i>True</i>	(Boolean) If True, SQLite uses synchronous mode.
use_db_reconnect = <i>False</i>	(Boolean) Enable the experimental use of database reconnect on connection lost.

Table 8.5. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
backdoor_port = <i>None</i>	(StrOpt) Enable eventlet backdoor. Acceptable values are 0, <port>, and <start>:<end>, where 0 results in listening on a random tcp port number; <port> results in listening on the specified port number (and not enabling backdoor if that port is in use); and <start>:<end> results in listening on the smallest unused port number within the specified range of port numbers. The chosen port is displayed in the service's log file.

Table 8.6. Description of flagmappings configuration options

Configuration option = Default value	Description
[DEFAULT]	
delayed_delete = <i>False</i>	(Boolean) Turn on/off delayed delete.
image_cache_dir = <i>None</i>	(String) Base directory that the image cache uses.
image_cache_driver = <i>sqlite</i>	(String) The driver to use for image cache management.
image_cache_max_size = <i>10737418240</i>	(Integer) The upper limit (the maximum size of accumulated cache in bytes) beyond which the cache pruner, if running, starts cleaning the image cache.
image_cache_sqlite_db = <i>cache.db</i>	(String) The path to the sqlite file database that will be used for image cache management.
image_cache_stall_time = <i>86400</i>	(Integer) The amount of time to let an incomplete image remain in the cache, before the cache cleaner, if running, will remove the incomplete image.
scrub_pool_size = <i>1</i>	(Integer) The size of thread pool to be used for scrubbing images. The default is one, which signifies serial scrubbing. Any value above one indicates the max number of images that may be scrubbed in parallel.
scrub_time = <i>0</i>	(Integer) The amount of time in seconds to delay before performing a delete.

Table 8.7. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(Boolean) If set to true, the logging level will be set to DEBUG instead of the default INFO level.
default_log_levels = <i>amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystone.middleware=WARNING, routes.middleware=WARN, stevedore=WARN, taskflow=WARN, keystoneauth=WARN, oslo.cache=INFO, dogpile.core.dogpile=INFO</i>	(List) List of package logging levels in logger=LEVEL pairs. This option is ignored if log_config_append is set.
fatal_deprecations = <i>False</i>	(Boolean) Enables or disables fatal status of deprecations.
instance_format = <i>"[instance: %(uuid)s]"</i>	(String) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s]"</i>	(String) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(String) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, logging_context_format_string).
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(String) Defines the format string for %(asctime)s in log records. Default: %(default)s . This option is ignored if log_config_append is set.
log_dir = <i>None</i>	(String) (Optional) The base directory used for relative log_file paths. This option is ignored if log_config_append is set.
log_file = <i>None</i>	(String) (Optional) Name of log file to send logging output to. If no default is set, logging will go to stderr as defined by use_stderr. This option is ignored if log_config_append is set.

Configuration option = Default value	Description
logging_context_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [% (request_id)s %(user_identity)s] % (instance)s%(message)s	(String) Format string to use for log messages with context.
logging_debug_format_suffix = % (funcName)s %(pathname)s:%(lineno)d	(String) Additional data to append to log message when logging level for the message is DEBUG.
logging_default_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [-] %(instance)s%(message)s	(String) Format string to use for log messages when context is undefined.
logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d ERROR %(name)s % (instance)s	(String) Prefix each line of exception output with this format.
logging_user_identity_format = %(user)s %(tenant)s %(domain)s %(user_domain)s % (project_domain)s	(String) Defines the format string for % (user_identity)s that is used in logging_context_format_string.
publish_errors = <i>False</i>	(Boolean) Enables or disables publication of error events.
syslog_log_facility = <i>LOG_USER</i>	(String) Syslog facility to receive log lines. This option is ignored if log_config_append is set.
use_stderr = <i>True</i>	(Boolean) Log output to standard error. This option is ignored if log_config_append is set.
use_syslog = <i>False</i>	(Boolean) Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.
verbose = <i>True</i>	(Boolean) DEPRECATED: If set to false, the logging level will be set to WARNING instead of the default INFO level.
watch_log_file = <i>False</i>	(Boolean) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log_file option is specified and Linux platform is used. This option is ignored if log_config_append is set.

Table 8.8. Description of policy configuration options

Configuration option = Default value	Description
[oslo_policy]	
policy_default_rule = <i>default</i>	(String) Default rule. Enforced when a requested rule is not found.
policy_dirs = <i>['policy.d']</i>	(Multi-valued) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the <code>config_dir</code> option, or absolute paths. The file defined by <code>policy_file</code> must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(String) The JSON file that defines policies.

Table 8.9. Description of profiler configuration options

Configuration option = Default value	Description
[profiler]	
enabled = <i>False</i>	(Boolean) If <i>False</i> fully disable profiling feature.
hmac_keys = <i>SECRET_KEY</i>	(String) Secret key to use to sign Glance API and Glance Registry services tracing messages.
trace_sqlalchemy = <i>False</i>	(Boolean) If <i>False</i> doesn't trace SQL requests.

Table 8.10. Description of Redis configuration options

Configuration option = Default value	Description
[matchmaker_redis]	
check_timeout = <i>20000</i>	(Integer) Time in ms to wait before the transaction is killed.
host = <i>127.0.0.1</i>	(String) Host to locate redis.
password =	(String) Password for Redis server (optional).
port = <i>6379</i>	(Port number) Use this port to connect to redis host.
sentinel_group_name = <i>oslo-messaging-zeromq</i>	(String) Redis replica set name.

Configuration option = Default value	Description
sentinel_hosts =	(List) List of Redis Sentinel hosts (fault tolerance mode) e.g. [host:port, host1:port ...]
socket_timeout = 1000	(Integer) Timeout in ms on blocking socket operations
wait_timeout = 500	(Integer) Time in ms to wait between connection attempts.

Table 8.11. Description of registry configuration options

Configuration option = Default value	Description
[DEFAULT]	
admin_password = None	(String) DEPRECATED: The administrators password. If "use_user_token" is not in effect, then admin credentials can be specified. This option was considered harmful and has been deprecated in M release. It will be removed in O release. For more information read OSSN-0060. Related functionality with uploading big images has been implemented with Keystone trusts support.
admin_tenant_name = None	(String) DEPRECATED: The tenant name of the administrative user. If "use_user_token" is not in effect, then admin tenant name can be specified. This option was considered harmful and has been deprecated in M release. It will be removed in O release. For more information read OSSN-0060. Related functionality with uploading big images has been implemented with Keystone trusts support.
admin_user = None	(String) DEPRECATED: The administrators user name. If "use_user_token" is not in effect, then admin credentials can be specified. This option was considered harmful and has been deprecated in M release. It will be removed in O release. For more information read OSSN-0060. Related functionality with uploading big images has been implemented with Keystone trusts support.

Configuration option = Default value	Description
auth_region = <i>None</i>	(String) DEPRECATED: The region for the authentication service. If "use_user_token" is not in effect and using keystone auth, then region name can be specified. This option was considered harmful and has been deprecated in M release. It will be removed in O release. For more information read OSSN-0060. Related functionality with uploading big images has been implemented with Keystone trusts support.
auth_strategy = <i>noauth</i>	(String) DEPRECATED: The strategy to use for authentication. If "use_user_token" is not in effect, then auth strategy can be specified. This option was considered harmful and has been deprecated in M release. It will be removed in O release. For more information read OSSN-0060. Related functionality with uploading big images has been implemented with Keystone trusts support.
auth_url = <i>None</i>	(String) DEPRECATED: The URL to the keystone service. If "use_user_token" is not in effect and using keystone auth, then URL of keystone can be specified. This option was considered harmful and has been deprecated in M release. It will be removed in O release. For more information read OSSN-0060. Related functionality with uploading big images has been implemented with Keystone trusts support.
registry_client_ca_file = <i>None</i>	(String) The path to the certifying authority cert file to use in SSL connections to the registry server, if any. Alternately, you may set the <code>GLANCE_CLIENT_CA_FILE</code> environment variable to a filepath of the CA cert file.
registry_client_cert_file = <i>None</i>	(String) The path to the cert file to use in SSL connections to the registry server, if any. Alternately, you may set the <code>GLANCE_CLIENT_CERT_FILE</code> environment variable to a filepath of the CA cert file
registry_client_insecure = <i>False</i>	(Boolean) When using SSL in connections to the registry server, do not require validation via a certifying authority. This is the registry's equivalent of specifying <code>--insecure</code> on the command line using <code>glanceclient</code> for the API.

Configuration option = Default value	Description
registry_client_key_file = <i>None</i>	(String) The path to the key file to use in SSL connections to the registry server, if any. Alternately, you may set the <code>GLANCE_CLIENT_KEY_FILE</code> environment variable to a filepath of the key file
registry_client_protocol = <i>http</i>	(String) The protocol to use for communication with the registry server. Either <code>http</code> or <code>https</code> .
registry_client_timeout = <i>600</i>	(Integer) The period of time, in seconds, that the API server will wait for a registry request to complete. A value of 0 implies no timeout.
registry_host = <i>0.0.0.0</i>	(String) Address to find the registry server.
registry_port = <i>9191</i>	(Port number) Port the registry server is listening on.

Table 8.12. Description of replicator configuration options

Configuration option = Default value	Description
[DEFAULT]	
args = <i>None</i>	(List) Arguments for the command
chunksize = <i>65536</i>	(Integer) Amount of data to transfer per HTTP write.
command = <i>None</i>	(String) Command to be given to replicator
dontreplicate = <i>created_at date deleted_at location updated_at</i>	(String) List of fields to not replicate.
mastertoken =	(String) Pass in your authentication token if you have one. This is the token used for the master.
metaonly = <i>False</i>	(Boolean) Only replicate metadata, not images.
slavetoken =	(String) Pass in your authentication token if you have one. This is the token used for the slave.
token =	(String) Pass in your authentication token if you have one. If you use this option the same token is used for both the master and the slave.

Table 8.13. Description of scrubber configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>wakeup_time = 300</code>	(Integer) Loop time between checking for new items to schedule for delete.

Table 8.14. Description of TaskFlow configuration options

Configuration option = Default value	Description
[taskflow_executor]	
<code>conversion_format = None</code>	(String) The format to which images will be automatically converted. When using the RBD backend, this should be set to 'raw'
<code>engine_mode = parallel</code>	(String) The mode in which the engine will run. Can be 'serial' or 'parallel'.
<code>max_workers = 10</code>	(Integer) The number of parallel activities executed at the same time by the engine. The value can be greater than one when the engine mode is 'parallel'.

Table 8.15. Description of testing configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>pydev_worker_debug_host = None</code>	(String) The hostname/IP of the pydev process listening for debug connections
<code>pydev_worker_debug_port = 5678</code>	(Port number) The port on which a pydev process is listening for connections.

8.1. CONFIGURE THE API

The Image service has two APIs: the user-facing API, and the registry API for internal requests that require access to the database.


Both of the APIs currently have the following major versions, v1 and v2. It is possible to run either or both versions, by setting appropriate values of `enable_v1_api`, `enable_v2_api`, `enable_v1_registry` and `enable_v2_registry`. If only the v2 API is used, running `glance-registry` is optional, as v2 of `glance-api` can connect directly to the database. If both the APIs are used, the v1 API still needs the registry and there is only one `data_api` specified in the config.

Tables of all the options used to configure the APIs, including enabling SSL and modifying WSGI settings are found below.

Table 8.16. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
admin_role = <i>admin</i>	(String) Role used to identify an authenticated user as administrator.
allow_anonymous_access = <i>False</i>	(Boolean) Allow unauthenticated users to access the API with read-only privileges. This only applies when using ContextMiddleware.
available_plugins =	(List) A list of artifacts that are allowed in the format name or name-version. Empty list means that any artifact can be loaded.
client_socket_timeout = <i>900</i>	(Integer) Timeout for client connections' socket operations. If an incoming connection is idle for this number of seconds it will be closed. A value of '0' means wait forever.
enable_v1_api = <i>True</i>	(Boolean) Deploy the v1 OpenStack Images API.
enable_v1_registry = <i>True</i>	(Boolean) Deploy the v1 OpenStack Registry API.
enable_v2_api = <i>True</i>	(Boolean) Deploy the v2 OpenStack Images API.
enable_v2_registry = <i>True</i>	(Boolean) Deploy the v2 OpenStack Registry API.
http_keepalive = <i>True</i>	(Boolean) If False, server will return the header "Connection: close", If True, server will return "Connection: Keep-Alive" in its responses. In order to close the client socket connection explicitly after the response is sent and read successfully by the client, you simply have to set this option to False when you create a wsgi server.
image_size_cap = <i>1099511627776</i>	(Integer) Maximum size of image a user can upload in bytes. Defaults to 1099511627776 bytes (1 TB).WARNING: this value should only be increased after careful consideration and must be set to a value under 8 EB (9223372036854775808).
load_enabled = <i>True</i>	(Boolean) When false, no artifacts can be loaded regardless of available_plugins. When true, artifacts can be loaded.

Configuration option = Default value	Description
location_strategy = <i>location_order</i>	(String) This value sets what strategy will be used to determine the image location order. Currently two strategies are packaged with Glance 'location_order' and 'store_type'.
max_header_line = <i>16384</i>	(Integer) Maximum line size of message headers to be accepted. max_header_line may need to be increased when using large tokens (typically those generated by the Keystone v3 API with big service catalogs
max_request_id_length = <i>64</i>	(Integer) Limits request ID length.
owner_is_tenant = <i>True</i>	(Boolean) When true, this option sets the owner of an image to be the tenant. Otherwise, the owner of the image will be the authenticated user issuing the request.
public_endpoint = <i>None</i>	(String) Public url to use for versions endpoint. The default is None, which will use the request's host_url attribute to populate the URL base. If Glance is operating behind a proxy, you will want to change this to represent the proxy's URL.
secure_proxy_ssl_header = <i>None</i>	(String) The HTTP header used to determine the scheme for the original request, even if it was removed by an SSL terminating proxy. Typical value is "HTTP_X_FORWARDED_PROTO".
send_identity_headers = <i>False</i>	(Boolean) Whether to pass through headers containing user and tenant information when making requests to the registry. This allows the registry to use the context middleware without keystone middleware's auth_token middleware, removing calls to the keystone auth service. It is recommended that when using this option, secure communication between glance api and glance registry is ensured by means other than auth_token middleware.

Configuration option = Default value	Description
<code>show_multiple_locations = False</code>	<p>(Boolean) Whether to include the backend image locations in image properties. For example, if using the file system store a URL of "file:///path/to/image" will be returned to the user in the 'direct_url' meta-data field. Revealing storage location can be a security risk, so use this setting with caution! Setting this to true overrides the <code>show_image_direct_url</code> option.</p> <div>  <div> <p>IMPORTANT</p> <p>If configured without the proper policy settings, a non-admin user of the Image Service can replace active image data (that is, switch out a current image without other users knowing). See the OSSN announcement (recommended actions) for configuration information: https://wiki.openstack.org/wiki/OS_SN/OSSN-0065</p> </div> </div>
<code>tcp_keepidle = 600</code>	<p>(Integer) The value for the socket option TCP_KEEPIDLE. This is the time in seconds that the connection must be idle before TCP starts sending keepalive probes.</p>
<code>use_user_token = True</code>	<p>(Boolean) DEPRECATED: Whether to pass through the user token when making requests to the registry. To prevent failures with token expiration during big files upload, it is recommended to set this parameter to False. If "use_user_token" is not in effect, then admin credentials can be specified. This option was considered harmful and has been deprecated in M release. It will be removed in O release. For more information read OSSN-0060. Related functionality with uploading big images has been implemented with Keystone trusts support.</p>
<code>[glance_store]</code>	

Configuration option = Default value	Description
default_store = <i>file</i>	(String) Default scheme to use to store image data. The scheme must be registered by one of the stores defined by the 'stores' config option.
store_capabilities_update_min_interval = 0	(Integer) Minimum interval seconds to execute updating dynamic storage capabilities based on backend status then. It's not a periodic routine, the update logic will be executed only when interval seconds elapsed and an operation of store has triggered. The feature will be enabled only when the option value greater then zero.
stores = <i>file, http</i>	(List) List of stores enabled. Valid stores are: cinder, file, http, rbd, sheepdog, swift, s3, vsphere
[oslo_middleware]	
max_request_body_size = 114688	(Integer) The maximum body size for each request, in bytes.
secure_proxy_ssl_header = <i>X-Forwarded-Proto</i>	(String) DEPRECATED: The HTTP Header that will be used to determine what the original request protocol scheme was, even if it was hidden by an SSL termination proxy.
[paste_deploy]	
config_file = <i>None</i>	(String) Name of the paste configuration file.
flavor = <i>None</i>	(String) Partial name of a pipeline in your paste configuration file with the service name removed. For example, if your paste section name is [pipeline:glance-api-keystone] use the value "keystone"
[store_type_location_strategy]	
store_type_preference =	(List) The store names to use to get store preference order. The name must be registered by one of the stores defined by the 'stores' config option. This option will be applied when you using 'store_type' option as image location strategy defined by the 'location_strategy' config option.

Table 8.17. Description of CA and SSL configuration options

Configuration option = Default value	Description
[DEFAULT]	
ca_file = <i>None</i>	(String) CA certificate file to use to verify connecting clients.
cert_file = <i>None</i>	(String) Certificate file to use when starting API server securely.
key_file = <i>None</i>	(String) Private key file to use when starting API server securely.

8.2. CONFIGURE THE RPC MESSAGING SYSTEM

OpenStack projects use an open standard for messaging middleware known as AMQP. This messaging middleware enables the OpenStack services that run on multiple servers to talk to each other. The OpenStack common library project, oslo, supports the following implementation of AMQP: **RabbitMQ**, and **Qpid**.

The following tables contain settings to configure the messaging middleware for the Image service:

Table 8.18. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>openstack</i>	(String) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
default_publisher_id = <i>image.localhost</i>	(String) Default publisher_id for outgoing notifications.
disabled_notifications =	(List) List of disabled notifications. A notification can be given either as a notification type to disable a single event, or as a notification group prefix to disable all events within a group. Example: if this config option is set to ["image.create", "metadef_namespace"], then "image.create" notification will not be sent after image is created and none of the notifications for metadefinition namespaces will be sent.
transport_url = <i>None</i>	(String) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the <code>rpc_backend</code> option and driver specific configuration.

Configuration option = Default value	Description
[oslo_messaging_notifications]	
driver = []	(Multi-valued) The Drivers(s) to handle sending notifications. Possible values are messaging, messagingv2, routing, log, test, noop
topics = notifications	(List) AMQP topic used for OpenStack notifications.
transport_url = None	(String) A URL representing the messaging driver to use for notifications. If not set, we fall back to the same configuration used for RPC.

Table 8.19. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
allowed_rpc_exception_modules = glance.common.exception, builtins, exceptions	(List) Modules of exceptions that are permitted to be recreated upon receiving exception data from an rpc call.
rpc_backend = rabbit	(String) The messaging driver to use, defaults to rabbit. Other drivers include amqp and zmq.
rpc_cast_timeout = -1	(Integer) Seconds to wait before a cast expires (TTL). The default value of -1 specifies an infinite linger period. The value of 0 specifies no linger period. Pending messages shall be discarded immediately when the socket is closed. Only supported by impl_zmq.
rpc_conn_pool_size = 30	(Integer) Size of RPC connection pool.
rpc_poll_timeout = 1	(Integer) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
rpc_response_timeout = 60	(Integer) Seconds to wait for a response from a call.
[oslo_concurrency]	
disable_process_locking = False	(Boolean) Enables or disables inter-process locks.

Configuration option = Default value	Description
lock_path = <i>None</i>	(String) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(Boolean) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>	(String) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(String) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(String) address prefix when sending to any server in group
idle_timeout = <i>0</i>	(Integer) Timeout for inactive connections (in seconds)
password =	(String) Password for message broker authentication
sasl_config_dir =	(String) Path to directory that contains the SASL configuration
sasl_config_name =	(String) Name of configuration file (without .conf suffix)
sasl_mechanisms =	(String) Space separated list of acceptable SASL mechanisms
server_request_prefix = <i>exclusive</i>	(String) address prefix used when sending to a specific server
ssl_ca_file =	(String) CA certificate PEM file to verify server certificate
ssl_cert_file =	(String) Identifying certificate PEM file to present to clients
ssl_key_file =	(String) Private key PEM file used to sign cert_file certificate

Configuration option = Default value	Description
ssl_key_password = <i>None</i>	(String) Password for decrypting <code>ssl_key_file</code> (if encrypted)
trace = <i>False</i>	(Boolean) Debug: dump AMQP frames to stdout
username =	(String) User name for message broker authentication
[oslo_messaging_rabbit]	
rpc_listener_prefetch_count = <i>100</i>	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc listener.
rpc_queue_expiration = <i>60</i>	(Integer) Time to live for rpc queues without consumers in seconds.
rpc_reply_exchange = <i>\${control_exchange}_rpc_reply</i>	(String) Exchange name for receiving RPC replies
rpc_reply_listener_prefetch_count = <i>100</i>	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc reply listener.
rpc_reply_retry_attempts = <i>-1</i>	(Integer) Reconnecting retry count in case of connectivity problem during sending reply. -1 means infinite retry during <code>rpc_timeout</code>
rpc_reply_retry_delay = <i>0.25</i>	(Floating point) Reconnecting retry delay in case of connectivity problem during sending reply.
rpc_retry_delay = <i>0.25</i>	(Floating point) Reconnecting retry delay in case of connectivity problem during sending RPC message
socket_timeout = <i>0.25</i>	(Floating point) Set socket timeout in seconds for connection's socket
ssl = <i>None</i>	(Boolean) Enable SSL
ssl_options = <i>None</i>	(Dict) Arguments passed to <code>ssl.wrap_socket</code>
tcp_user_timeout = <i>0.25</i>	(Floating point) Set <code>TCP_USER_TIMEOUT</code> in seconds for connection's socket

Table 8.20. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = <i>False</i>	(Boolean) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(Boolean) Use durable queues in AMQP.
channel_max = <i>None</i>	(Integer) Maximum number of channels to allow
default_notification_exchange = <i>\${control_exchange}_notification</i>	(String) Exchange name for for sending notifications
default_notification_retry_attempts = <i>-1</i>	(Integer) Reconnecting retry count in case of connectivity problem during sending notification, -1 means infinite retry.
default_rpc_exchange = <i>\${control_exchange}_rpc</i>	(String) Exchange name for sending RPC messages
default_rpc_retry_attempts = <i>-1</i>	(Integer) Reconnecting retry count in case of connectivity problem during sending RPC message, -1 means infinite retry. If actual retry attempts in not 0 the rpc request could be processed more then one time
fake_rabbit = <i>False</i>	(Boolean) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
frame_max = <i>None</i>	(Integer) The maximum byte size for an AMQP frame
heartbeat_interval = <i>1</i>	(Integer) How often to send heartbeats for consumer's connections
heartbeat_rate = <i>2</i>	(Integer) How often times during the <code>heartbeat_timeout_threshold</code> we check the heartbeat.
heartbeat_timeout_threshold = <i>60</i>	(Integer) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disable the heartbeat). EXPERIMENTAL
host_connection_reconnect_delay = <i>0.25</i>	(Floating point) Set delay for reconnection to some host which has connection error
kombu_compression = <i>None</i>	(String) EXPERIMENTAL: Possible values are: <code>gzip</code> , <code>bz2</code> . If not set compression will not be used. This option may notbe available in future versions.

Configuration option = Default value	Description
kombu_failover_strategy = <i>round-robin</i>	(String) Determines how the next RabbitMQ node is chosen in case the one we are currently connected to becomes unavailable. Takes effect only if more than one RabbitMQ node is provided in config.
kombu_missing_consumer_retry_timeout = 60	(Integer) How long to wait a missing client before abandoning to send it its replies. This value should not be longer than <code>rpc_response_timeout</code> .
kombu_reconnect_delay = 1.0	(Floating point) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_ssl_ca_certs =	(String) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(String) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(String) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(String) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
notification_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to notification listener.
notification_persistence = <i>False</i>	(Boolean) Persist notification messages.
notification_retry_delay = 0.25	(Floating point) Reconnecting retry delay in case of connectivity problem during sending notification message
pool_max_overflow = 0	(Integer) Maximum number of connections to create above <code>pool_max_size</code> .
pool_max_size = 10	(Integer) Maximum number of connections to keep queued.
pool_recycle = 600	(Integer) Lifetime of a connection (since creation) in seconds or None for no recycling. Expired connections are closed on acquire.
pool_stale = 60	(Integer) Threshold at which inactive (since release) connections are considered stale in seconds or None for no staleness. Stale connections are closed on acquire.

Configuration option = Default value	Description
pool_timeout = 30	(Integer) Default number of seconds to wait for a connections to available
rabbit_ha_queues = <i>False</i>	(Boolean) Try to use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database. In RabbitMQ 3.0, queue mirroring is no longer controlled by the x-ha-policy argument when declaring a queue. If you just want to make sure that all queues (except those with auto-generated names) are mirrored across all nodes, run: "rabbitmqctl set_policy HA '^(?!amq\\.).*' '{\"ha-mode\": \"all\"}' "
rabbit_host = <i>localhost</i>	(String) The RabbitMQ broker address where a single node is used.
rabbit_hosts = <i>\$rabbit_host:\$rabbit_port</i>	(List) RabbitMQ HA cluster host:port pairs.
rabbit_interval_max = 30	(Integer) Maximum interval of RabbitMQ connection retries. Default is 30 seconds.
rabbit_login_method = <i>AMQPLAIN</i>	(String) The RabbitMQ login method.
rabbit_max_retries = 0	(Integer) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = <i>guest</i>	(String) The RabbitMQ password.
rabbit_port = 5672	(Port number) The RabbitMQ broker port where a single node is used.
rabbit_qos_prefetch_count = 0	(Integer) Specifies the number of messages to prefetch. Setting to zero allows unlimited messages.
rabbit_retry_backoff = 2	(Integer) How long to backoff for between retries when connecting to RabbitMQ.
rabbit_retry_interval = 1	(Integer) How frequently to retry connecting with RabbitMQ.
rabbit_transient_queues_ttl = 1800	(Integer) Positive integer representing duration in seconds for queue TTL (x-expires). Queues which are unused for the duration of the TTL are automatically deleted. The parameter affects only reply and fanout queues.
rabbit_use_ssl = <i>False</i>	(Boolean) Connect over SSL for RabbitMQ.

Configuration option = Default value	Description
rabbit_userid = <i>guest</i>	(String) The RabbitMQ userid.
rabbit_virtual_host = <i>/</i>	(String) The RabbitMQ virtual host.

Table 8.21. Description of Qpid configuration options

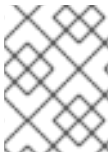
Configuration option = Default value	Description
[oslo_messaging_qpid]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
qpid_heartbeat = <i>60</i>	(IntOpt) Seconds between connection keepalive heartbeats.
qpid_hostname = <i>localhost</i>	(StrOpt) Qpid broker hostname.
qpid_hosts = <i>\$qpid_hostname:\$qpid_port</i>	(ListOpt) Qpid HA cluster host:port pairs.
qpid_password =	(StrOpt) Password for Qpid connection.
qpid_port = <i>5672</i>	(IntOpt) Qpid broker port.
qpid_protocol = <i>tcp</i>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
qpid_receiver_capacity = <i>1</i>	(IntOpt) The number of prefetched messages held by receiver.
qpid_sasl_mechanisms =	(StrOpt) Space separated list of SASL mechanisms to use for auth.
qpid_tcp_nodelay = <i>True</i>	(BoolOpt) Whether to disable the Nagle algorithm.
qpid_topology_version = <i>1</i>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
qpid_username =	(StrOpt) Username for Qpid connection.

Configuration option = Default value	Description
<code>send_single_reply = False</code>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

8.3. CONFIGURE IMAGE CACHE

You can configure the Image service API to have a local image cache. Caching of image files is transparent and can be performed by modifying some configuration parameters.

These parameters are configured in the PasteDeploy configuration file, `component-paste.ini`. You should not generally edit this file directly, as it ships with default options for all common deployment flavors. The PasteDeploy configuration file is stored with the rest of the `glance` configuration files, usually stored in `/etc/glance` or `/usr/share/glance`.



NOTE

Using `glance-cache` is deployment specific, and depending on the store that is being used, it is possible that having `glance-cache` may not provide any benefit.

Advantages and Disadvantages of having `glance-cache`

Advantage - It creates locality for the image data. It brings the data closer to the compute node and avoid having to download it every time from `glance-api`. Therefore, requests would be spread across the cache node and not go directly to `glance-api`.

Disadvantage - Depending on the store, duplication of the data in your cloud as the image data will be downloaded in the `glance-cache` node. Therefore, it is necessary to account for the extra storage when provisioning the cloud.

8.3.1. Enable the Image Cache

To enable the image cache parameters, the cache parameters must occur in the application pipeline after the appropriate context parameters. A pipeline is a series of middlewares that will be executed whenever a request hits the API.

The cache parameters should be in your `glance-api-paste.ini` in a section titled `[filter:cache]`. It should look like this:

```
[filter:cache]
paste.filter_factory = glance.api.middleware.cache:CacheFilter.factory
```

A ready-made application parameter including this filter is defined in the `glance-api-paste.ini` file as follows:

```
[pipeline:glance-api-caching]
pipeline = versionnegotiation context cache apiv1app
```

To enable the above application parameter, in your main `glance-api.conf` configuration file, select the appropriate deployment flavor as follows:

```
[paste_deploy]
flavor = caching
```

8.3.2. Enable the Image Cache Management

There is an optional `cachemanage` option that allows you to directly interact with cache images. Use this flavor in place of the `cache` flavor in your API config file.

```
[paste_deploy]
flavor = cachemanage
```



NOTE

For example, the following setting enables a pipeline (configured in the `.ini` file) that has both, `keystone` and `cachemanagement`, enabled.

```
[paste_deploy]
flavor = keystone+cachemanage
```

8.3.3. Configuration Options Affecting the Image Cache

These configuration options must be set in both the `glance-cache` and `glance-api` configuration files.

Table 8.22. Description of Image Cache configuration options

Configuration option = Default value	Description
<code>image_cache_dir = /var/lib/glance/image-cache</code>	Required when image cache middleware is enabled. This is the base directory the image cache can write files to. Make sure the directory is writable by the user running the <code>glance-api</code> server.

Configuration option = Default value	Description
<code>image_cache_driver = sqlite</code>	(Optional) The default sqlite cache driver has no special dependencies, other than the python-sqlite3 library, which is installed on almost all operating systems with modern versions of Python. It stores information about the cached files in a SQLite database. The xattr cache driver requires the python-xattr <code>>=0.6.0</code> library and requires that the filesystem containing image_cache_dir has access times tracked for all files (in other words, the noatime option CANNOT be set for that filesystem). In addition, user_xattr must be set on the filesystem's description line in fstab.
<code>image_cache_sqlite_db = cache.db</code>	(Optional) When using the sqlite cache driver, you can set the name of the database that will be used to store the cached images information. The database is always contained in the image_cache_dir .
<code>image_cache_max_size = 10737418240 (10 GB)</code>	(Optional) Size, in bytes, that the image cache should be constrained to. Images files are cached automatically in the local image cache, even if the writing of that image file would put the total cache size over this size. The glance-cache-pruner executable is what prunes the image cache to be equal to or less than this value. The glance-cache-pruner executable is designed to be run via cron on a regular basis.

8.3.4. Configure Image-Volume Cache

OpenStack Block Storage has an optional Image cache which can dramatically improve the performance of creating a volume from an image. The improvement depends on many factors, primarily on how quickly the configured back-end can clone a volume.

When a volume is first created from an image, a new cached image-volume will be created that is owned by the Block Storage internal tenant. Subsequent requests to create volumes from that image will clone the cached version instead of downloading the image contents and copying data to the volume.

The cache itself is configurable per back end and will contain the most recently used images.

The Image-Volume cache requires that the internal tenant be configured for the Block Storage services. This tenant will own the cached image-volumes so they can be managed like normal users including tools like volume quotas. This protects normal users from having to see the cached image-volumes, but does not make them globally hidden.

The administrator should configure the new internal **cinder** tenant manually for image cache, by setting the following parameter values in the **cinder.conf** file.

```
cinder_internal_tenant_project_id=PROJECT_ID
cinder_internal_tenant_user_id=USER_ID
```

This tenant will store the cached image volumes so they can be managed like normal users. It protects normal users from having to see the cached image volumes.



NOTE

The actual user and project that are configured for the internal tenant do not require any special privileges. They can be the Block Storage service tenant or can be any normal project and user.

8.4. SUPPORT FOR ISO IMAGES

You can load ISO images into the Image service. You can subsequently boot an ISO image using Compute.

Procedure 8.1. To load an ISO image to an Image service data store

1. In the Image service, run the following command:

```
$ glance image-create --name "rhel-server-7.0.iso"
--copy-from
https://access.redhat.com/downloads/content/69/ver=/rhel---
7/7.0/x86_64/product-downloads/rhel-server-7.0-x86_64-dvd.iso --is-
public True --container-format bare --disk-format iso
```

In this command, **rhel-server-7.0.iso** is the name for the ISO image after it is loaded to the Image service, and **rhel-server-7.0-x86_64-dvd.iso** is the name of the source ISO image.

2. Optionally, to confirm the upload in Image service (glance), run this command:

```
$ glance image-list
```

Procedure 8.2. To boot an instance from an ISO image

OpenStack supports booting instances using ISO images, but in order to make the instances created using ISO images functional, follow a few more steps:

1. Boot instance with ISO image using the following command:

```
$ nova boot
--image rhel-server-7.0-x86_64-dvd.iso
--block-device source=blank,dest=volume,size=10,shutdown=preserve
--nic net-id=NETWORK_UUID --flavor 3 INSTANCE_NAME
```

In this command, **rhel-server-7.0.iso** is the ISO image, and **INSTANCE_NAME** is the name of the new instance. **NETWORK_UUID** is a valid network id in your system

You will need Block Storage service and setting the parameter **shutdown=preserve** will preserve the volume even after the shutting down an instance.

2. After the instance is successfully launched, connect to the instance using remote console and

follow the instructions to install the system as using ISO images on regular computers. When the installation is complete and system reboots, the instance prompts you to install the operating system, which implies your instance is not usable.

Procedure 8.3. To make an instance that was booted from an ISO image functional

Run the following commands to ensure the instances you created using ISO images are functional:

1. Delete the instance you just created:

```
$ nova delete INSTANCE_NAME
```

2. After you delete an instance, the system you just installed using your ISO image remains because the parameter `shutdown=preserve` was set, run the following command:

```
$ cinder list
```

You get a list with all the volumes in your system. In this list, you can find the volume that is attached to your ISO created instance, with the `false` bootable property.

3. Upload the volume to glance:

```
$ cinder upload-to-image VOLUME_UUID IMAGE_NAME
```

The `VOLUME_UUID` is the uuid of the volume that is attached to your ISO created instance, and the `IMAGE_NAME` is the name that you give to your new image.

4. After the image is successfully uploaded, you can now use the new image to boot instances, the instance launched using this image will contain the system you just installed using the ISO image.

8.5. CONFIGURE BACK ENDS

The Image service supports several back ends for storing virtual machine images:

- OpenStack Block Storage (cinder)
- A directory on a local file system
- GridFS
- Ceph RBD
- Amazon S3
- Sheepdog
- OpenStack Object Storage (swift)
- VMware ESX

The following tables detail the options available for each.

Table 8.23. Description of cinder configuration options

Configuration option = Default value	Description
[glance_store]	
cinder_api_insecure = <i>False</i>	(Boolean) Allow to perform insecure SSL requests to cinder
cinder_ca_certificates_file = <i>None</i>	(String) Location of ca certificates file to use for cinder client requests.
cinder_catalog_info = <i>volumev2::publicURL</i>	(String) Info to match when looking for cinder in the service catalog. Format is : separated values of the form: <service_type>:<service_name>:<endpoint_type>
cinder_endpoint_template = <i>None</i>	(String) Override service catalog lookup with template for cinder endpoint e.g. http://localhost:8776/v2/%(tenant)s
cinder_http_retries = 3	(Integer) Number of cinderclient retries on failed http calls
cinder_os_region_name = <i>None</i>	(String) Region name of this node. If specified, it will be used to locate OpenStack services for stores.
cinder_state_transition_timeout = 300	(Integer) Time period of time in seconds to wait for a cinder volume transition to complete.
cinder_store_auth_address = <i>None</i>	(String) The address where the Cinder authentication service is listening. If <None>, the cinder endpoint in the service catalog is used.
cinder_store_password = <i>None</i>	(String) Password for the user authenticating against Cinder. If <None>, the current context auth token is used.
cinder_store_project_name = <i>None</i>	(String) Project name where the image is stored in Cinder. If <None>, the project in current context is used.
cinder_store_user_name = <i>None</i>	(String) User name to authenticate against Cinder. If <None>, the user of current context is used.

Table 8.24. Description of filesystem configuration options

Configuration option = Default value	Description
[glance_store]	

Configuration option = Default value	Description
filesystem_store_datadir = <i>/var/lib/glance/images</i>	(String) Directory to which the Filesystem backend store writes images.
filesystem_store_datadirs = <i>None</i>	(Multi-valued) List of directories and its priorities to which the Filesystem backend store writes images.
filesystem_store_file_perm = <i>0</i>	(Integer) The required permission for created image file. In this way the user other service used, e.g. Nova, who consumes the image could be the exclusive member of the group that owns the files created. Assigning it less then or equal to zero means don't change the default permission of the file. This value will be decoded as an octal digit.
filesystem_store_metadata_file = <i>None</i>	(String) The path to a file which contains the metadata to be returned with any location associated with this store. The file must contain a valid JSON object. The object should contain the keys 'id' and 'mountpoint'. The value for both keys should be 'string'.

Table 8.25. Description of http configuration options

Configuration option = Default value	Description
[glance_store]	
http_proxy_information = <i>{}</i>	(Dict) Specify the http/https proxy information that should be used to connect to the remote server. The proxy information should be a key value pair of the scheme and proxy. e.g. http:10.0.0.1:3128. You can specify proxies for multiple schemes by seperating the key value pairs with a comma.e.g. http:10.0.0.1:3128, https:10.0.0.1:1080.
https_ca_certificates_file = <i>None</i>	(String) Specify the path to the CA bundle file to use in verifying the remote server certificate.
https_insecure = <i>True</i>	(Boolean) If true, the remote server certificate is not verified. If false, then the default CA truststore is used for verification. This option is ignored if "https_ca_certificates_file" is set.

Table 8.26. Description of RADOS Block Devices (RBD) configuration options

Configuration option = Default value	Description
[glance_store]	
rados_connect_timeout = 0	(Integer) Timeout value (in seconds) used when connecting to ceph cluster. If value <= 0, no timeout is set and default librados value is used.
rbd_store_ceph_conf = /etc/ceph/ceph.conf	(String) Ceph configuration file path. If <None>, librados will locate the default config. If using cephx authentication, this file should include a reference to the right keyring in a client.<USER> section
rbd_store_chunk_size = 8	(Integer) RADOS images will be chunked into objects of this size (in megabytes). For best performance, this should be a power of two.
rbd_store_pool = images	(String) RADOS pool in which images are stored.
rbd_store_user = None	(String) RADOS user to authenticate as (only applicable if using Cephx. If <None>, a default will be chosen based on the client. section in rbd_store_ceph_conf)

Table 8.27. Description of Sheepdog configuration options

Configuration option = Default value	Description
[glance_store]	
sheepdog_store_address = localhost	(String) IP address of sheep daemon.
sheepdog_store_chunk_size = 64	(Integer) Images will be chunked into objects of this size (in megabytes). For best performance, this should be a power of two.
sheepdog_store_port = 7000	(Integer) Port of sheep daemon.

Table 8.28. Description of swift configuration options

Configuration option = Default value	Description
[DEFAULT]	
default_swift_reference = refl	(String) The reference to the default swift account/backing store parameters to use for adding new images.

Configuration option = Default value	Description
swift_store_auth_address = <i>None</i>	(String) The address where the Swift authentication service is listening.(deprecated)
swift_store_config_file = <i>None</i>	(String) The config file that has the swift account(s)configs.
swift_store_key = <i>None</i>	(String) Auth key for the user authenticating against the Swift authentication service. (deprecated)
swift_store_user = <i>None</i>	(String) The user to authenticate against the Swift authentication service (deprecated)
[glance_store]	
default_swift_reference = <i>refl</i>	(String) The reference to the default swift account/backing store parameters to use for adding new images.
swift_store_admin_tenants =	(List) A list of tenants that will be granted read/write access on all Swift containers created by Glance in multi-tenant mode.
swift_store_auth_address = <i>None</i>	(String) The address where the Swift authentication service is listening. (deprecated - use "auth_address" in swift_store_config_file)
swift_store_auth_insecure = <i>False</i>	(Boolean) If True, swiftclient won't check for a valid SSL certificate when authenticating.
swift_store_auth_version = <i>2</i>	(String) Version of the authentication service to use. Valid versions are 2 and 3 for keystone and 1 (deprecated) for swauth and rackspace. (deprecated - use "auth_version" in swift_store_config_file)
swift_store_cacert = <i>None</i>	(String) A string giving the CA certificate file to use in SSL connections for verifying certs.
swift_store_config_file = <i>None</i>	(String) The config file that has the swift account(s)configs.
swift_store_container = <i>glance</i>	(String) Container within the account that the account should use for storing images in Swift when using single container mode. In multiple container mode, this will be the prefix for all containers.
swift_store_create_container_on_put = <i>False</i>	(Boolean) A boolean value that determines if we create the container if it does not exist.

Configuration option = Default value	Description
swift_store_endpoint = <i>None</i>	(String) If set, the configured endpoint will be used. If <i>None</i> , the storage url from the auth response will be used.
swift_store_endpoint_type = <i>publicURL</i>	(String) A string giving the endpoint type of the swift service to use (<i>publicURL</i> , <i>adminURL</i> or <i>internalURL</i>). This setting is only used if swift_store_auth_version is 2.
swift_store_expire_soon_interval = <i>60</i>	(Integer) The period of time (in seconds) before token expiration when glance_store will try to request new user token. Default value 60 sec means that if token is going to expire in 1 min then glance_store request new user token.
swift_store_key = <i>None</i>	(String) Auth key for the user authenticating against the Swift authentication service. (deprecated - use "key" in swift_store_config_file)
swift_store_large_object_chunk_size = <i>200</i>	(Integer) The amount of data written to a temporary disk buffer during the process of chunking the image file.
swift_store_large_object_size = <i>5120</i>	(Integer) The size, in MB, that Glance will start chunking image files and do a large object manifest in Swift.
swift_store_multi_tenant = <i>False</i>	(Boolean) If set to <i>True</i> , enables multi-tenant storage mode which causes Glance images to be stored in tenant specific Swift accounts.
swift_store_multiple_containers_seed = <i>0</i>	(Integer) When set to 0, a single-tenant store will only use one container to store all images. When set to an integer value between 1 and 32, a single-tenant store will use multiple containers to store images, and this value will determine how many containers are created. Used only when swift_store_multi_tenant is disabled. The total number of containers that will be used is equal to 16^N , so if this config option is set to 2, then $16^2=256$ containers will be used to store images.
swift_store_region = <i>None</i>	(String) The region of the swift endpoint to be used for single tenant. This setting is only necessary if the tenant has multiple swift endpoints.
swift_store_retry_get_count = <i>0</i>	(Integer) The number of times a Swift download will be retried before the request fails.

Configuration option = Default value	Description
swift_store_service_type = <i>object-store</i>	(String) A string giving the service type of the swift service to use. This setting is only used if swift_store_auth_version is 2.
swift_store_ssl_compression = <i>True</i>	(Boolean) If set to False, disables SSL layer compression of https swift requests. Setting to False may improve performance for images which are already in a compressed format, eg qcow2.
swift_store_use_trusts = <i>True</i>	(Boolean) If set to True create a trust for each add/get request to Multi-tenant store in order to prevent authentication token to be expired during uploading/downloading data. If set to False then user token is used for Swift connection (so no overhead on trust creation). Please note that this option is considered only and only if swift_store_multi_tenant =True
swift_store_user = <i>None</i>	(String) The user to authenticate against the Swift authentication service (deprecated - use "user" in swift_store_config_file)

8.6. IMAGE SERVICE SAMPLE CONFIGURATION FILES

You can find the files that are described in this section in the `/etc/glance/` directory.

8.6.1. glance-api.conf

The configuration file for the Image service API is found in the **glance-api.conf** file.

This file must be modified after installation.

```
[DEFAULT]
# Show more verbose log output (sets INFO log level output)
#verbose = False

# Show debugging output in logs (sets DEBUG log level output)
#debug = False

# Maximum image size (in bytes) that may be uploaded through the
# Glance API server. Defaults to 1 TB.
# WARNING: this value should only be increased after careful consideration
# and must be set to a value under 8 EB (9223372036854775808).
#image_size_cap = 1099511627776

# Address to bind the API server
bind_host = 0.0.0.0

# Port the bind the API server to
bind_port = 9292
```

```
# Log to this file. Make sure you do not set the same log file for both the
API
# and registry servers!
#
# If `log_file` is omitted and `use_syslog` is false, then log messages
are
# sent to stdout as a fallback.
log_file = /var/log/glance/api.log

# Backlog requests when creating socket
backlog = 4096

# TCP_KEEPIDLE value in seconds when creating socket.
# Not supported on OS X.
#tcp_keepidle = 600

# Timeout (in seconds) for client connections' socket operations. If an
incoming
# connection is idle for this period it will be closed. A value of "0"
# means wait forever.
#client_socket_timeout = 0

# API to use for accessing data. Default value points to sqlalchemy
# package, it is also possible to use: glance.db.registry.api
# data_api = glance.db.sqlalchemy.api

# The number of child process workers that will be
# created to service API requests. The default will be
# equal to the number of CPUs available. (integer value)
#workers = 4

# Maximum line size of message headers to be accepted.
# max_header_line may need to be increased when using large tokens
# (typically those generated by the Keystone v3 API with big service
# catalogs)
# max_header_line = 16384

# Role used to identify an authenticated user as administrator
#admin_role = admin

# Allow unauthenticated users to access the API with read-only
# privileges. This only applies when using ContextMiddleware.
#allow_anonymous_access = False

# Allow access to version 1 of glance api
#enable_v1_api = True

# Allow access to version 2 of glance api
#enable_v2_api = True

# Return the URL that references where the data is stored on
# the backend storage system. For example, if using the
# file system store a URL of 'file:///path/to/image' will
# be returned to the user in the 'direct_url' meta-data field.
# The default value is false.
```

```

#show_image_direct_url = False

# Send headers containing user and tenant information when making requests
to
# the v1 glance registry. This allows the registry to function as if a
user is
# authenticated without the need to authenticate a user itself using the
# auth_token middleware.
# The default value is false.
#send_identity_headers = False

# Supported values for the 'container_format' image attribute
#container_formats=ami,ari,aki,bare,ovf,ova

# Supported values for the 'disk_format' image attribute
#disk_formats=ami,ari,aki,vhd,vmdk,raw,qcow2,vdi,iso

# Property Protections config file
# This file contains the rules for property protections and the
roles/policies
# associated with it.
# If this config value is not specified, by default, property protections
# won't be enforced.
# If a value is specified and the file is not found, then the glance-api
# service will not start.
#property_protection_file =

# Specify whether 'roles' or 'policies' are used in the
# property_protection_file.
# The default value for property_protection_rule_format is 'roles'.
#property_protection_rule_format = roles

# This value sets what strategy will be used to determine the image
location
# order. Currently two strategies are packaged with Glance
'location_order'
# and 'store_type'.
#location_strategy = location_order

# Public url to use for versions endpoint. The default is None,
# which will use the request's host_url attribute to populate the URL
base.
# If Glance is operating behind a proxy, you will want to change this to
# represent the proxy's URL.
#public_endpoint=<None>

# http_keepalive option. If False, server will return the header
# "Connection: close", If True, server will return "Connection: Keep-
Alive"
# in its responses. In order to close the client socket connection
# explicitly after the response is sent and read successfully by the
client,
# you simply have to set this option to False when you create a wsgi
server.
#http_keepalive = True

```

```

# ===== Syslog Options =====

# Send logs to syslog (/dev/log) instead of to file specified
# by `log_file`
#use_syslog = False

# Facility to use. If unset defaults to LOG_USER.
#syslog_log_facility = LOG_LOCAL0

# ===== SSL Options =====

# Certificate file to use when starting API server securely
#cert_file = /path/to/certfile

# Private key file to use when starting API server securely
#key_file = /path/to/keyfile

# CA certificate file to use to verify connecting clients
#ca_file = /path/to/cafile

# ===== Security Options =====

# AES key for encrypting store 'location' metadata, including
# -- if used -- Swift or S3 credentials
# Should be set to a random string of length 16, 24 or 32 bytes
#metadata_encryption_key = <16, 24 or 32 char registry metadata key>

# Digest algorithm which will be used for digital signature, the default
is
# sha1 in Kilo for a smooth upgrade process, and it will be updated with
# sha256 in next release(L). Use command
# "openssl list-message-digest-algorithms" to get the available algorithms
# supported by the version of OpenSSL on the platform. Examples are
'sha1',
# 'sha256', 'sha512', etc.
#digest_algorithm = sha1

# ===== Registry Options =====

# Address to find the registry server
registry_host = 0.0.0.0

# Port the registry server is listening on
registry_port = 9191

# What protocol to use when connecting to the registry server?
# Set to https for secure HTTP communication
registry_client_protocol = http

# The path to the key file to use in SSL connections to the
# registry server, if any. Alternately, you may set the
# GLANCE_CLIENT_KEY_FILE environ variable to a filepath of the key file
#registry_client_key_file = /path/to/key/file

```

```

# The path to the cert file to use in SSL connections to the
# registry server, if any. Alternately, you may set the
# GLANCE_CLIENT_CERT_FILE environ variable to a filepath of the cert file
#registry_client_cert_file = /path/to/cert/file

# The path to the certifying authority cert file to use in SSL connections
# to the registry server, if any. Alternately, you may set the
# GLANCE_CLIENT_CA_FILE environ variable to a filepath of the CA cert file
#registry_client_ca_file = /path/to/ca/file

# When using SSL in connections to the registry server, do not require
# validation via a certifying authority. This is the registry's equivalent
# of
# specifying --insecure on the command line using glanceclient for the API
# Default: False
#registry_client_insecure = False

# The period of time, in seconds, that the API server will wait for a
# registry
# request to complete. A value of '0' implies no timeout.
# Default: 600
#registry_client_timeout = 600

# Enable DEBUG log messages from sqlalchemy which prints every database
# query and response.
# Default: False
#sqlalchemy_debug = True

# Pass the user's token through for API requests to the registry.
# Default: True
#use_user_token = True

# If 'use_user_token' is not in effect then admin credentials
# can be specified. Requests to the registry on behalf of
# the API will use these credentials.
# Admin user name
#admin_user = None
# Admin password
#admin_password = None
# Admin tenant name
#admin_tenant_name = None
# Keystone endpoint
#auth_url = None
# Keystone region
#auth_region = None
# Auth strategy
#auth_strategy = keystone

# ===== Notification System Options =====

# Driver or drivers to handle sending notifications. Set to
# 'messaging' to send notifications to a message queue.
# notification_driver = noop

# Default publisher_id for outgoing notifications.
# default_publisher_id = image.localhost

```

```

# List of disabled notifications. A notification can be given either as a
# notification type to disable a single event, or as a notification group
# prefix to disable all events within a group.
# Example: if this config option is set to
# ["image.create", "metadef_namespace"], then "image.create" notification
# will
# not be sent after image is created and none of the notifications for
# metadefinition namespaces will be sent.
# disabled_notifications = []

# Messaging driver used for 'messaging' notifications driver
# rpc_backend = 'rabbit'

# Configuration options if sending notifications via rabbitmq (these are
# the defaults)
rabbit_host = localhost
rabbit_port = 5672
rabbit_use_ssl = false
rabbit_userid = guest
rabbit_password = guest
rabbit_virtual_host = /
rabbit_notification_exchange = glance
rabbit_notification_topic = notifications
rabbit_durable_queues = False

# Configuration options if sending notifications via Qpid (these are
# the defaults)
qpid_notification_exchange = glance
qpid_notification_topic = notifications
qpid_hostname = localhost
qpid_port = 5672
qpid_username =
qpid_password =
qpid_sasl_mechanisms =
qpid_reconnect_timeout = 0
qpid_reconnect_limit = 0
qpid_reconnect_interval_min = 0
qpid_reconnect_interval_max = 0
qpid_reconnect_interval = 0
qpid_heartbeat = 5
# Set to 'ssl' to enable SSL
qpid_protocol = tcp
qpid_tcp_nodelay = True

# ===== Delayed Delete Options =====

# Turn on/off delayed delete
delayed_delete = False

# Delayed delete time in seconds
scrub_time = 43200

# Directory that the scrubber will use to remind itself of what to delete
# Make sure this is also set in glance-scrubber.conf
scrubber_datadir = /var/lib/glance/scrubber

```

```

# ===== Quota Options =====

# The maximum number of image members allowed per image
#image_member_quota = 128

# The maximum number of image properties allowed per image
#image_property_quota = 128

# The maximum number of tags allowed per image
#image_tag_quota = 128

# The maximum number of locations allowed per image
#image_location_quota = 10

# Set a system wide quota for every user. This value is the total number
# of bytes that a user can use across all storage systems. A value of
# 0 means unlimited.
#user_storage_quota = 0

# ===== Image Cache Options =====

# Base directory that the Image Cache uses
image_cache_dir = /var/lib/glance/image-cache/

# ===== Policy Options =====

[oslo_policy]
# The JSON file that defines policies.
# Deprecated group/name - [DEFAULT]/policy_file
#policy_file = policy.json

# Default rule. Enforced when a requested rule is not found.
# Deprecated group/name - [DEFAULT]/policy_default_rule
#policy_default_rule = default

# Directories where policy configuration files are stored.
# They can be relative to any directory in the search path
# defined by the config_dir option, or absolute paths.
# The file defined by policy_file must exist for these
# directories to be searched.
# Deprecated group/name - [DEFAULT]/policy_dirs
#policy_dirs = policy.d

# ===== Database Options =====

[database]
# The file name to use with SQLite (string value)
#sqlite_db = oslo.sqlite

# If True, SQLite uses synchronous mode (boolean value)
#sqlite_synchronous = True

# The backend to use for db (string value)
# Deprecated group/name - [DEFAULT]/db_backend
#backend = sqlalchemy

```

```

# The SQLAlchemy connection string used to connect to the
# database (string value)
# Deprecated group/name - [DEFAULT]/sql_connection
# Deprecated group/name - [DATABASE]/sql_connection
# Deprecated group/name - [sql]/connection
#connection = <None>

# The SQL mode to be used for MySQL sessions. This option,
# including the default, overrides any server-set SQL mode. To
# use whatever SQL mode is set by the server configuration,
# set this to no value. Example: mysql_sql_mode= (string
# value)
#mysql_sql_mode = TRADITIONAL

# Timeout before idle sql connections are reaped (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_idle_timeout
# Deprecated group/name - [DATABASE]/sql_idle_timeout
# Deprecated group/name - [sql]/idle_timeout
#idle_timeout = 3600

# Minimum number of SQL connections to keep open in a pool
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_min_pool_size
# Deprecated group/name - [DATABASE]/sql_min_pool_size
#min_pool_size = 1

# Maximum number of SQL connections to keep open in a pool
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_pool_size
# Deprecated group/name - [DATABASE]/sql_max_pool_size
#max_pool_size = <None>

# Maximum db connection retries during startup. (setting -1
# implies an infinite retry count) (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_retries
# Deprecated group/name - [DATABASE]/sql_max_retries
#max_retries = 10

# Interval between retries of opening a sql connection
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_retry_interval
# Deprecated group/name - [DATABASE]/reconnect_interval
#retry_interval = 10

# If set, use this value for max_overflow with sqlalchemy
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_overflow
# Deprecated group/name - [DATABASE]/sqlalchemy_max_overflow
#max_overflow = <None>

# Verbosity of SQL debugging information. 0=None,
# 100=Everything (integer value)
# Deprecated group/name - [DEFAULT]/sql_connection_debug
#connection_debug = 0

```



```

# Add python stack traces to SQL as comment strings (boolean
# value)
# Deprecated group/name - [DEFAULT]/sql_connection_trace
#connection_trace = False

# If set, use this value for pool_timeout with sqlalchemy
# (integer value)
# Deprecated group/name - [DATABASE]/sqlalchemy_pool_timeout
#pool_timeout = <None>

# Enable the experimental use of database reconnect on
# connection lost (boolean value)
#use_db_reconnect = False

# seconds between db connection retries (integer value)
#db_retry_interval = 1

# Whether to increase interval between db connection retries,
# up to db_max_retry_interval (boolean value)
#db_inc_retry_interval = True

# max seconds between db connection retries, if
# db_inc_retry_interval is enabled (integer value)
#db_max_retry_interval = 10

# maximum db connection retries before error is raised.
# (setting -1 implies an infinite retry count) (integer value)
#db_max_retries = 20

[oslo_concurrency]

# Enables or disables inter-process locks. (boolean value)
# Deprecated group/name - [DEFAULT]/disable_process_locking
#disable_process_locking = false

# Directory to use for lock files. For security, the specified
# directory should only be writable by the user running the processes
# that need locking. It could be read from environment variable
# OSLO_LOCK_PATH. This setting needs to be the same for both
# glance-scrubber and glance-api service. Default to a temp directory.
# Deprecated group/name - [DEFAULT]/lock_path (string value)
#lock_path = /tmp

[keystone_authtoken]
identity_uri = http://127.0.0.1:35357
admin_tenant_name = %SERVICE_TENANT_NAME%
admin_user = %SERVICE_USER%
admin_password = %SERVICE_PASSWORD%
revocation_cache_time = 10

[paste_deploy]
# Name of the paste configuration file that defines the available
# pipelines
#config_file = glance-api-paste.ini

```

```

# Partial name of a pipeline in your paste configuration file with the
# service name removed. For example, if your paste section name is
# [pipeline:glance-api-keystone], you would configure the flavor below
# as 'keystone'.
#flavor=

[store_type_location_strategy]
# The scheme list to use to get store preference order. The scheme must be
# registered by one of the stores defined by the 'stores' config option.
# This option will be applied when you using 'store_type' option as image
# location strategy defined by the 'location_strategy' config option.
#store_type_preference =

[profiler]
# If False fully disable profiling feature.
#enabled = False

# If False doesn't trace SQL requests.
#trace_sqlalchemy = False

[task]
# ===== Glance Tasks Options =====

# Specifies how long (in hours) a task is supposed to live in the tasks DB
# after succeeding or failing before getting soft-deleted.
# The default value for task_time_to_live is 48 hours.
# task_time_to_live = 48

# Specifies which task executor to be used to run the task scripts.
# The default value for task_executor is taskflow.
# task_executor = taskflow

# Work dir for asynchronous task operations. The directory set here
# will be used to operate over images - normally before they are
# imported in the destination store. When providing work dir, make sure
# enough space is provided for concurrent tasks to run efficiently
# without running out of space. A rough estimation can be done by
# multiplying the number of `max_workers` - or the N of workers running
# - by an average image size (e.g 500MB). The image size estimation
# should be done based on the average size in your deployment. Note that
# depending on the tasks running you may need to multiply this number by
# some factor depending on what the task does. For example, you may want
# to double the available size if image conversion is enabled. All this
# being said, remember these are just estimations and you should do them
# based on the worst case scenario and be prepared to act in case they
# were wrong.
# work_dir=None

# Specifies the maximum number of eventlet threads which can be spun up by
# the eventlet based task executor to perform execution of Glance tasks.
# DEPRECATED: Use [taskflow_executor]/max_workers instead.
# eventlet_executor_pool_size = 1000

[taskflow_executor]
# The mode in which the engine will run. Can be 'default', 'serial',
# 'parallel' or 'worker-based'

```

```

#engine_mode = serial

# The number of parallel activities executed at the same time by
# the engine. The value can be greater than one when the engine mode is
# 'parallel' or 'worker-based', otherwise this value will be ignored.
#max_workers = 10

[glance_store]
# List of which store classes and store class locations are
# currently known to glance at startup.
# Deprecated group/name - [DEFAULT]/known_stores
# Existing but disabled stores:
#     glance.store.rbd.Store,
#     glance.store.s3.Store,
#     glance.store.swift.Store,
#     glance.store.sheepdog.Store,
#     glance.store.cinder.Store,
#     glance.store.gridfs.Store,
#     glance.store.vmware_datastore.Store,
#stores = glance.store.filesystem.Store,
#     glance.store.http.Store

# Which backend scheme should Glance use by default is not specified
# in a request to add a new image to Glance? Known schemes are determined
# by the stores option.
# Deprecated group/name - [DEFAULT]/default_store
# Default: 'file'
default_store = file

# ===== Filesystem Store Options =====

# Directory that the Filesystem backend store
# writes image data to
filesystem_store_datadir = /var/lib/glance/images/

# A list of directories where image data can be stored.
# This option may be specified multiple times for specifying multiple
# store
# directories. Either one of filesystem_store_datadirs or
# filesystem_store_datadir option is required. A priority number may be
# given
# after each directory entry, separated by a ":".
# When adding an image, the highest priority directory will be selected,
# unless
# there is not enough space available in cases where the image size is
# already
# known. If no priority is given, it is assumed to be zero and the
# directory
# will be considered for selection last. If multiple directories have the
# same
# priority, then the one with the most free space available is selected.
# If same store is specified multiple times then BadStoreConfiguration
# exception will be raised.
#filesystem_store_datadirs = /var/lib/glance/images/:1

# A path to a JSON file that contains metadata describing the storage

```

```

# system. When show_multiple_locations is True the information in this
# file will be returned with any location that is contained in this
# store.
#filesystem_store_metadata_file = None

# ===== Swift Store Options =====

# Version of the authentication service to use
# Valid versions are '2' for keystone and '1' for swauth and rackspace
swift_store_auth_version = 2

# Address where the Swift authentication service lives
# Valid schemes are 'http://' and 'https://'
# If no scheme specified, default to 'https://'
# For swauth, use something like '127.0.0.1:8080/v1.0/'
swift_store_auth_address = 127.0.0.1:5000/v2.0/

# User to authenticate against the Swift authentication service
# If you use Swift authentication service, set it to 'account':'user'
# where 'account' is a Swift storage account and 'user'
# is a user in that account
swift_store_user = jdoe:jdoe

# Auth key for the user authenticating against the
# Swift authentication service
swift_store_key = a86850deb2742ec3cb41518e26aa2d89

# Container within the account that the account should use
# for storing images in Swift
swift_store_container = glance

# Do we create the container if it does not exist?
swift_store_create_container_on_put = False

# What size, in MB, should Glance start chunking image files
# and do a large object manifest in Swift? By default, this is
# the maximum object size in Swift, which is 5GB
swift_store_large_object_size = 5120

# swift_store_config_file = glance-swift.conf
# This file contains references for each of the configured
# Swift accounts/backing stores. If used, this option can prevent
# credentials being stored in the database. Using Swift references
# is disabled if this config is left blank.

# The reference to the default Swift parameters to use for adding new
# images.
# default_swift_reference = 'ref1'

# When doing a large object manifest, what size, in MB, should
# Glance write chunks to Swift? This amount of data is written
# to a temporary disk buffer during the process of chunking
# the image file, and the default is 200MB
swift_store_large_object_chunk_size = 200

# If set, the configured endpoint will be used. If None, the storage URL

```

```

# from the auth response will be used. The location of an object is
# obtained by appending the container and object to the configured URL.
#
# swift_store_endpoint = https://www.example.com/v1/not_a_container
#swift_store_endpoint =

# If set to True enables multi-tenant storage mode which causes Glance
images
# to be stored in tenant specific Swift accounts.
#swift_store_multi_tenant = False

# If set to an integer value between 1 and 32, a single-tenant store will
# use multiple containers to store images. If set to the default value of
0,
# only a single container will be used. Multi-tenant stores are not
affected
# by this option. The max number of containers that will be used to store
# images is approximately 16^N where N is the value of this option.
Discuss
# the impact of this with your swift deployment team, as this option is
only
# beneficial in the largest of deployments where swift rate limiting can
lead
# to unwanted throttling on a single container.
#swift_store_multiple_containers_seed = 0

# A list of swift ACL strings that will be applied as both read and
# write ACLs to the containers created by Glance in multi-tenant
# mode. This grants the specified tenants/users read and write access
# to all newly created image objects. The standard swift ACL string
# formats are allowed, including:
# <tenant_id>:<username>
# <tenant_name>:<username>
# *:<username>
# Multiple ACLs can be combined using a comma separated list, for
# example: swift_store_admin_tenants = service:glance,*:admin
#swift_store_admin_tenants =

# The region of the swift endpoint to be used for single tenant. This
setting
# is only necessary if the tenant has multiple swift endpoints.
#swift_store_region =

# If set to False, disables SSL layer compression of https swift requests.
# Setting to 'False' may improve performance for images which are already
# in a compressed format, eg qcow2. If set to True, enables SSL layer
# compression (provided it is supported by the target swift proxy).
#swift_store_ssl_compression = True

# The number of times a Swift download will be retried before the
# request fails
#swift_store_retry_get_count = 0

# Bypass SSL verification for Swift
#swift_store_auth_insecure = False

```

```

# The path to a CA certificate bundle file to use for SSL verification
when
# communicating with Swift.
#swift_store_cacert =

# ===== S3 Store Options =====

# Address where the S3 authentication service lives
# Valid schemes are 'http://' and 'https://'
# If no scheme specified, default to 'http://'
s3_store_host = s3.amazonaws.com

# User to authenticate against the S3 authentication service
s3_store_access_key = <20-char AWS access key>

# Auth key for the user authenticating against the
# S3 authentication service
s3_store_secret_key = <40-char AWS secret key>

# Container within the account that the account should use
# for storing images in S3. Note that S3 has a flat namespace,
# so you need a unique bucket name for your glance images. An
# easy way to do this is append your AWS access key to "glance".
# S3 buckets in AWS *must* be lowercased, so remember to lowercase
# your AWS access key if you use it in your bucket name below!
s3_store_bucket = <lowercased 20-char aws access key>glance

# Do we create the bucket if it does not exist?
s3_store_create_bucket_on_put = False

# When sending images to S3, the data will first be written to a
# temporary buffer on disk. By default the platform's temporary directory
# will be used. If required, an alternative directory can be specified
# here.
#s3_store_object_buffer_dir = /path/to/dir

# When forming a bucket url, boto will either set the bucket name as the
# subdomain or as the first token of the path. Amazon's S3 service will
# accept it as the subdomain, but Swift's S3 middleware requires it be
# in the path. Set this to 'path' or 'subdomain' - defaults to
# 'subdomain'.
#s3_store_bucket_url_format = subdomain

# Size, in MB, should S3 start chunking image files
# and do a multipart upload in S3. The default is 100MB.
#s3_store_large_object_size = 100

# Multipart upload part size, in MB, should S3 use when uploading
# parts. The size must be greater than or equal to
# 5MB. The default is 10MB.
#s3_store_large_object_chunk_size = 10

# The number of thread pools to perform a multipart upload
# in S3. The default is 10.
#s3_store_thread_pools = 10

```

```
# ===== RBD Store Options =====

# Ceph configuration file path
# If using cephx authentication, this file should
# include a reference to the right keyring
# in a client.<USER> section
#rbd_store_ceph_conf = /etc/ceph/ceph.conf

# RADOS user to authenticate as (only applicable if using cephx)
# If <None>, a default will be chosen based on the client. section
# in rbd_store_ceph_conf
#rbd_store_user = <None>

# RADOS pool in which images are stored
#rbd_store_pool = images

# RADOS images will be chunked into objects of this size (in megabytes).
# For best performance, this should be a power of two
#rbd_store_chunk_size = 8

# ===== Sheepdog Store Options =====

sheepdog_store_address = localhost

sheepdog_store_port = 7000

# Images will be chunked into objects of this size (in megabytes).
# For best performance, this should be a power of two
sheepdog_store_chunk_size = 64

# ===== Cinder Store Options =====

# Info to match when looking for cinder in the service catalog
# Format is : separated values of the form:
# <service_type>:<service_name>:<endpoint_type> (string value)
#cinder_catalog_info = volume:cinder:publicURL

# Override service catalog lookup with template for cinder endpoint
# e.g. http://localhost:8776/v1/%(project_id)s (string value)
#cinder_endpoint_template = <None>

# Region name of this node (string value)
#os_region_name = <None>

# Location of ca certificates file to use for cinder client requests
# (string value)
#cinder_ca_certificates_file = <None>

# Number of cinderclient retries on failed http calls (integer value)
#cinder_http_retries = 3

# Allow to perform insecure SSL requests to cinder (boolean value)
#cinder_api_insecure = False

# ===== VMware Datastore Store Options =====
```

```
# ESX/ESXi or vCenter Server target system.
# The server value can be an IP address or a DNS name
# e.g. 127.0.0.1, 127.0.0.1:443, www.vmware-infra.com
#vmware_server_host = <None>

# Server username (string value)
#vmware_server_username = <None>

# Server password (string value)
#vmware_server_password = <None>

# Inventory path to a datacenter (string value)
# Value optional when vmware_server_ip is an ESX/ESXi host: if specified
# should be `ha-datacenter`.
# Deprecated in favor of vmware_datastores.
#vmware_datacenter_path = <None>

# Datastore associated with the datacenter (string value)
# Deprecated in favor of vmware_datastores.
#vmware_datastore_name = <None>

# A list of datastores where the image can be stored.
# This option may be specified multiple times for specifying multiple
# datastores. Either one of vmware_datastore_name or vmware_datastores is
# required. The datastore name should be specified after its datacenter
# path, separated by ":". An optional weight may be given after the
# datastore
# name, separated again by ":". Thus, the required format becomes
# <datacenter_path>:<datastore_name>:<optional_weight>.
# When adding an image, the datastore with highest weight will be
# selected,
# unless there is not enough free space available in cases where the image
# size
# is already known. If no weight is given, it is assumed to be zero and
# the
# directory will be considered for selection last. If multiple datastores
# have
# the same weight, then the one with the most free space available is
# selected.
#vmware_datastores = <None>

# The number of times we retry on failures
# e.g., socket error, etc (integer value)
#vmware_api_retry_count = 10

# The interval used for polling remote tasks
# invoked on VMware ESX/VC server in seconds (integer value)
#vmware_task_poll_interval = 5

# Absolute path of the folder containing the images in the datastore
# (string value)
#vmware_store_image_dir = /openstack_glance

# Allow to perform insecure SSL requests to the target system (boolean
# value)
#vmware_api_insecure = False
```


8.6.2. glance-registry.conf

Configuration for the Image service's registry, which stores the metadata about images, is found in the `glance-registry.conf` file.

This file must be modified after installation.

```
[DEFAULT]
# Show more verbose log output (sets INFO log level output)
#verbose = False

# Show debugging output in logs (sets DEBUG log level output)
#debug = False

# Address to bind the registry server
bind_host = 0.0.0.0

# Port the bind the registry server to
bind_port = 9191

# Log to this file. Make sure you do not set the same log file for both the
API
# and registry servers!
#
# If `log_file` is omitted and `use_syslog` is false, then log messages
are
# sent to stdout as a fallback.
log_file = /var/log/glance/registry.log

# Backlog requests when creating socket
backlog = 4096

# TCP_KEEPIDLE value in seconds when creating socket.
# Not supported on OS X.
#tcp_keepidle = 600

# Timeout (in seconds) for client connections' socket operations. If an
incoming
# connection is idle for this period it will be closed. A value of "0"
# means wait forever.
#client_socket_timeout = 0

# API to use for accessing data. Default value points to sqlalchemy
# package.
#data_api = glance.db.sqlalchemy.api

# The number of child process workers that will be
# created to service Registry requests. The default will be
# equal to the number of CPUs available. (integer value)
#workers = None

# Enable Registry API versions individually or simultaneously
#enable_v1_registry = True
#enable_v2_registry = True

# Limit the api to return `param_limit_max` items in a call to a
```

```

container. If
# a larger `limit` query param is provided, it will be reduced to this
value.
api_limit_max = 1000

# If a `limit` query param is not provided in an api request, it will
# default to `limit_param_default`
limit_param_default = 25

# Role used to identify an authenticated user as administrator
#admin_role = admin

# Enable DEBUG log messages from sqlalchemy which prints every database
# query and response.
# Default: False
#sqlalchemy_debug = True

# http_keepalive option. If False, server will return the header
# "Connection: close", If True, server will return "Connection: Keep-
Alive"
# in its responses. In order to close the client socket connection
# explicitly after the response is sent and read successfully by the
client,
# you simply have to set this option to False when you create a wsgi
server.
#http_keepalive = True

# ===== Syslog Options =====

# Send logs to syslog (/dev/log) instead of to file specified
# by `log_file`
#use_syslog = False

# Facility to use. If unset defaults to LOG_USER.
#syslog_log_facility = LOG_LOCAL1

# ===== SSL Options =====

# Certificate file to use when starting registry server securely
#cert_file = /path/to/certfile

# Private key file to use when starting registry server securely
#key_file = /path/to/keyfile

# CA certificate file to use to verify connecting clients
#ca_file = /path/to/cafile

# ===== Notification System Options =====

# Driver or drivers to handle sending notifications. Set to
# 'messaging' to send notifications to a message queue.
# notification_driver = noop

# Default publisher_id for outgoing notifications.
# default_publisher_id = image.localhost

```

```

# Messaging driver used for 'messaging' notifications driver
# rpc_backend = 'rabbit'

# Configuration options if sending notifications via rabbitmq (these are
# the defaults)
rabbit_host = localhost
rabbit_port = 5672
rabbit_use_ssl = false
rabbit_userid = guest
rabbit_password = guest
rabbit_virtual_host = /
rabbit_notification_exchange = glance
rabbit_notification_topic = notifications
rabbit_durable_queues = False

# Configuration options if sending notifications via Qpid (these are
# the defaults)
qpid_notification_exchange = glance
qpid_notification_topic = notifications
qpid_hostname = localhost
qpid_port = 5672
qpid_username =
qpid_password =
qpid_sasl_mechanisms =
qpid_reconnect_timeout = 0
qpid_reconnect_limit = 0
qpid_reconnect_interval_min = 0
qpid_reconnect_interval_max = 0
qpid_reconnect_interval = 0
qpid_heartbeat = 5
# Set to 'ssl' to enable SSL
qpid_protocol = tcp
qpid_tcp_nodelay = True

# ===== Policy Options =====

[oslo_policy]
# The JSON file that defines policies.
# Deprecated group/name - [DEFAULT]/policy_file
#policy_file = policy.json

# Default rule. Enforced when a requested rule is not found.
# Deprecated group/name - [DEFAULT]/policy_default_rule
#policy_default_rule = default

# Directories where policy configuration files are stored.
# They can be relative to any directory in the search path
# defined by the config_dir option, or absolute paths.
# The file defined by policy_file must exist for these
# directories to be searched.
# Deprecated group/name - [DEFAULT]/policy_dirs
#policy_dirs = policy.d

# ===== Database Options =====

```

```

[database]
# The file name to use with SQLite (string value)
#sqlite_db = glance.sqlite

# If True, SQLite uses synchronous mode (boolean value)
#sqlite_synchronous = True

# The backend to use for db (string value)
# Deprecated group/name - [DEFAULT]/db_backend
#backend = sqlalchemy

# The SQLAlchemy connection string used to connect to the
# database (string value)
# Deprecated group/name - [DEFAULT]/sql_connection
# Deprecated group/name - [DATABASE]/sql_connection
# Deprecated group/name - [sql]/connection
#connection = <None>

# The SQL mode to be used for MySQL sessions. This option,
# including the default, overrides any server-set SQL mode. To
# use whatever SQL mode is set by the server configuration,
# set this to no value. Example: mysql_sql_mode= (string
# value)
#mysql_sql_mode = TRADITIONAL

# Timeout before idle sql connections are reaped (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_idle_timeout
# Deprecated group/name - [DATABASE]/sql_idle_timeout
# Deprecated group/name - [sql]/idle_timeout
#idle_timeout = 3600

# Minimum number of SQL connections to keep open in a pool
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_min_pool_size
# Deprecated group/name - [DATABASE]/sql_min_pool_size
#min_pool_size = 1

# Maximum number of SQL connections to keep open in a pool
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_pool_size
# Deprecated group/name - [DATABASE]/sql_max_pool_size
#max_pool_size = <None>

# Maximum db connection retries during startup. (setting -1
# implies an infinite retry count) (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_retries
# Deprecated group/name - [DATABASE]/sql_max_retries
#max_retries = 10

# Interval between retries of opening a sql connection
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_retry_interval
# Deprecated group/name - [DATABASE]/reconnect_interval
#retry_interval = 10

```

```

# If set, use this value for max_overflow with sqlalchemy
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_overflow
# Deprecated group/name - [DATABASE]/sqlalchemy_max_overflow
#max_overflow = <None>

# Verbosity of SQL debugging information. 0=None,
# 100=Everything (integer value)
# Deprecated group/name - [DEFAULT]/sql_connection_debug
#connection_debug = 0

# Add python stack traces to SQL as comment strings (boolean
# value)
# Deprecated group/name - [DEFAULT]/sql_connection_trace
#connection_trace = False

# If set, use this value for pool_timeout with sqlalchemy
# (integer value)
# Deprecated group/name - [DATABASE]/sqlalchemy_pool_timeout
#pool_timeout = <None>

# Enable the experimental use of database reconnect on
# connection lost (boolean value)
#use_db_reconnect = False

# seconds between db connection retries (integer value)
#db_retry_interval = 1

# Whether to increase interval between db connection retries,
# up to db_max_retry_interval (boolean value)
#db_inc_retry_interval = True

# max seconds between db connection retries, if
# db_inc_retry_interval is enabled (integer value)
#db_max_retry_interval = 10

# maximum db connection retries before error is raised.
# (setting -1 implies an infinite retry count) (integer value)
#db_max_retries = 20

[keystone_authtoken]
identity_uri = http://127.0.0.1:35357
admin_tenant_name = %SERVICE_TENANT_NAME%
admin_user = %SERVICE_USER%
admin_password = %SERVICE_PASSWORD%

[paste_deploy]
# Name of the paste configuration file that defines the available
# pipelines
#config_file = glance-registry-paste.ini

# Partial name of a pipeline in your paste configuration file with the
# service name removed. For example, if your paste section name is
# [pipeline:glance-registry-keystone], you would configure the flavor
# below
# as 'keystone'.

```

```
#flavor=

[profiler]
# If False fully disable profiling feature.
#enabled = False

# If False doesn't trace SQL requests.
#trace_sqlalchemy = False
```

8.6.3. glance-api-paste.ini

Configuration for the Image service's API middleware pipeline is found in the **glance-api-paste.ini** file.

You should not need to modify this file.

```
# Use this pipeline for no auth or image caching - DEFAULT
[pipeline:glance-api]
pipeline = versionnegotiation osprofiler unauthenticated-context rootapp

# Use this pipeline for image caching and no auth
[pipeline:glance-api-caching]
pipeline = versionnegotiation osprofiler unauthenticated-context cache
rootapp

# Use this pipeline for caching w/ management interface but no auth
[pipeline:glance-api-cachemanagement]
pipeline = versionnegotiation osprofiler unauthenticated-context cache
cachemanage rootapp

# Use this pipeline for keystone auth
[pipeline:glance-api-keystone]
pipeline = versionnegotiation osprofiler authtoken context rootapp

# Use this pipeline for keystone auth with image caching
[pipeline:glance-api-keystone+caching]
pipeline = versionnegotiation osprofiler authtoken context cache rootapp

# Use this pipeline for keystone auth with caching and cache management
[pipeline:glance-api-keystone+cachemanagement]
pipeline = versionnegotiation osprofiler authtoken context cache
cachemanage rootapp

# Use this pipeline for authZ only. This means that the registry will
treat a
# user as authenticated without making requests to keystone to
reauthenticate
# the user.
[pipeline:glance-api-trusted-auth]
pipeline = versionnegotiation osprofiler context rootapp

# Use this pipeline for authZ only. This means that the registry will
treat a
# user as authenticated without making requests to keystone to
reauthenticate
```

```

# the user and uses cache management
[pipeline:glance-api-trusted-auth+cachemanagement]
pipeline = versionnegotiation osprofiler context cache cachemanage rootapp

[composite:rootapp]
paste.composite_factory = glance.api:root_app_factory
/: apiversions
/v1: apiv1app
/v2: apiv2app

[app:apiversions]
paste.app_factory = glance.api.versions:create_resource

[app:apiv1app]
paste.app_factory = glance.api.v1.router:API.factory

[app:apiv2app]
paste.app_factory = glance.api.v2.router:API.factory

[filter:versionnegotiation]
paste.filter_factory =
glance.api.middleware.version_negotiation:VersionNegotiationFilter.factory

[filter:cache]
paste.filter_factory = glance.api.middleware.cache:CacheFilter.factory

[filter:cachemanage]
paste.filter_factory =
glance.api.middleware.cache_manage:CacheManageFilter.factory

[filter:context]
paste.filter_factory =
glance.api.middleware.context:ContextMiddleware.factory

[filter:unauthenticated-context]
paste.filter_factory =
glance.api.middleware.context:UnauthenticatedContextMiddleware.factory

[filter:authtoken]
paste.filter_factory = keystonemiddleware.auth_token:filter_factory
delay_auth_decision = true

[filter:gzip]
paste.filter_factory = glance.api.middleware.gzip:GzipMiddleware.factory

[filter:osprofiler]
paste.filter_factory = osprofiler.web:WsgiMiddleware.factory
hmac_keys = SECRET_KEY
enabled = yes

```

8.6.4. glance-manage.conf

The Image service's custom logging options are found in the `glance-manage.conf` file.

**NOTE**

Options set in **glance-manage.conf** will override options of the same section and name set in **glance-registry.conf** and **glance-api.conf**. Similarly, options in **glance-api.conf** will override options set in **glance-registry.conf**.

[DEFAULT]

```
#
# From glance.manage
#

# Print debugging output (set logging level to DEBUG instead of
# default WARNING level). (boolean value)
#debug = false

# The name of a logging configuration file. This file is appended to
# any existing logging configuration files. For details about logging
# configuration files, see the Python logging module documentation.
# (string value)
# Deprecated group/name - [DEFAULT]/log_config
#log_config_append = <None>

# Format string for %(asctime)s in log records. Default: %(default)s
# . (string value)
#log_date_format = %Y-%m-%d %H:%M:%S

# (Optional) The base directory used for relative --log-file paths.
# (string value)
# Deprecated group/name - [DEFAULT]/logdir
#log_dir = <None>

# (Optional) Name of log file to output to. If no default is set,
# logging will go to stdout. (string value)
# Deprecated group/name - [DEFAULT]/logfile
log_file = /var/log/glance/manage.log

# DEPRECATED. A logging.Formatter log message format string which may
# use any of the available logging.LogRecord attributes. This option
# is deprecated. Please use logging_context_format_string and
# logging_default_format_string instead. (string value)
#log_format = <None>

# Syslog facility to receive log lines. (string value)
#syslog_log_facility = LOG_USER

# Use syslog for logging. Existing syslog format is DEPRECATED during
# I, and will change in J to honor RFC5424. (boolean value)
#use_syslog = false

# (Optional) Enables or disables syslog rfc5424 format for logging. If
# enabled, prefixes the MSG part of the syslog message with APP-NAME
# (RFC5424). The format without the APP-NAME is deprecated in I, and
# will be removed in J. (boolean value)
#use_syslog_rfc_format = false
```



```

# Print more verbose output (set logging level to INFO instead of
# default WARNING level). (boolean value)
#verbose = false

[database]

#
# From oslo.db
#

# The back end to use for the database. (string value)
# Deprecated group/name - [DEFAULT]/db_backend
#backend = sqlalchemy

# The SQLAlchemy connection string to use to connect to the database.
# (string value)
# Deprecated group/name - [DEFAULT]/sql_connection
# Deprecated group/name - [DATABASE]/sql_connection
# Deprecated group/name - [sql]/connection
#connection = <None>

# Verbosity of SQL debugging information: 0=None, 100=Everything.
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_connection_debug
#connection_debug = 0

# Add Python stack traces to SQL as comment strings. (boolean value)
# Deprecated group/name - [DEFAULT]/sql_connection_trace
#connection_trace = false

# If True, increases the interval between database connection retries
# up to db_max_retry_interval. (boolean value)
#db_inc_retry_interval = true

# Maximum database connection retries before error is raised. Set to
# -1 to specify an infinite retry count. (integer value)
#db_max_retries = 20

# If db_inc_retry_interval is set, the maximum seconds between
# database connection retries. (integer value)
#db_max_retry_interval = 10

# Seconds between database connection retries. (integer value)
#db_retry_interval = 1

# Timeout before idle SQL connections are reaped. (integer value)
# Deprecated group/name - [DEFAULT]/sql_idle_timeout
# Deprecated group/name - [DATABASE]/sql_idle_timeout
# Deprecated group/name - [sql]/idle_timeout
#idle_timeout = 3600

# If set, use this value for max_overflow with SQLAlchemy. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_max_overflow

```

```

# Deprecated group/name - [DATABASE]/sqlalchemy_max_overflow
#max_overflow = <None>

# Maximum number of SQL connections to keep open in a pool. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_max_pool_size
# Deprecated group/name - [DATABASE]/sql_max_pool_size
#max_pool_size = <None>

# Maximum number of database connection retries during startup. Set to
# -1 to specify an infinite retry count. (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_retries
# Deprecated group/name - [DATABASE]/sql_max_retries
#max_retries = 10

# Minimum number of SQL connections to keep open in a pool. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_min_pool_size
# Deprecated group/name - [DATABASE]/sql_min_pool_size
#min_pool_size = 1

# The SQL mode to be used for MySQL sessions. This option, including
# the default, overrides any server-set SQL mode. To use whatever SQL
# mode is set by the server configuration, set this to no value.
# Example: mysql_sql_mode= (string value)
#mysql_sql_mode = TRADITIONAL

# If set, use this value for pool_timeout with SQLAlchemy. (integer
# value)
# Deprecated group/name - [DATABASE]/sqlalchemy_pool_timeout
#pool_timeout = <None>

# Interval between retries of opening a SQL connection. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_retry_interval
# Deprecated group/name - [DATABASE]/reconnect_interval
#retry_interval = 10

# The SQLAlchemy connection string to use to connect to the slave
# database. (string value)
#slave_connection = <None>

# The file name to use with SQLite. (string value)
# Deprecated group/name - [DEFAULT]/sqlite_db
#sqlite_db = oslo.sqlite

# If True, SQLite uses synchronous mode. (boolean value)
# Deprecated group/name - [DEFAULT]/sqlite_synchronous
#sqlite_synchronous = true

# Enable the experimental use of database reconnect on connection
# lost. (boolean value)
#use_db_reconnect = false

#
# From oslo.db.concurrency

```

```
#

# Enable the experimental use of thread pooling for all DB API calls
# (boolean value)
# Deprecated group/name - [DEFAULT]/dbapi_use_tpool
#use_tpool = false
```

8.6.5. glance-registry-paste.ini

The Image service's middleware pipeline for its registry is found in the **glance-registry-paste.ini** file.

```
# Use this pipeline for no auth - DEFAULT
[pipeline:glance-registry]
pipeline = osprofiler unauthenticated-context registryapp

# Use this pipeline for keystone auth
[pipeline:glance-registry-keystone]
pipeline = osprofiler authtoken context registryapp

# Use this pipeline for authZ only. This means that the registry will
# treat a
# user as authenticated without making requests to keystone to
# reauthenticate
# the user.
[pipeline:glance-registry-trusted-auth]
pipeline = osprofiler context registryapp

[app:registryapp]
paste.app_factory = glance.registry.api:API.factory

[filter:context]
paste.filter_factory =
glance.api.middleware.context:ContextMiddleware.factory

[filter:unauthenticated-context]
paste.filter_factory =
glance.api.middleware.context:UnauthenticatedContextMiddleware.factory

[filter:authtoken]
paste.filter_factory = keystone.middleware.auth_token:filter_factory

[filter:osprofiler]
paste.filter_factory = osprofiler.web:WsgiMiddleware.factory
hmac_keys = SECRET_KEY
enabled = yes
```

8.6.6. glance-scrubber.conf

glance-scrubber is a utility for the Image service that cleans up images that have been deleted; its configuration is stored in the **glance-scrubber.conf** file.

Multiple instances of **glance-scrubber** can be run in a single deployment, but only one of them can be designated as the **cleanup_scrubber** in the **glance-scrubber.conf** file. The

cleanup_scrubber coordinates other **glance-scrubber** instances by maintaining the master queue of images that need to be removed.

```
[DEFAULT]
# Show more verbose log output (sets INFO log level output)
#verbose = False

# Show debugging output in logs (sets DEBUG log level output)
#debug = False

# Log to this file. Make sure you do not set the same log file for both the
API
# and registry servers!
#
# If `log_file` is omitted and `use_syslog` is false, then log messages
are
# sent to stdout as a fallback.
log_file = /var/log/glance/scrubber.log

# Send logs to syslog (/dev/log) instead of to file specified by
`log_file`
#use_syslog = False

# Should we run our own loop or rely on cron/scheduler to run us
daemon = False

# Loop time between checking for new items to schedule for delete
wakeup_time = 300

# Directory that the scrubber will use to remind itself of what to delete
# Make sure this is also set in glance-api.conf
scrubber_datadir = /var/lib/glance/scrubber

# Only one server in your deployment should be designated the cleanup host
cleanup_scrubber = False

# pending_delete items older than this time are candidates for cleanup
cleanup_scrubber_time = 86400

# Address to find the registry server for cleanups
registry_host = 0.0.0.0

# Port the registry server is listening on
registry_port = 9191

# Auth settings if using Keystone
# auth_url = http://127.0.0.1:5000/v2.0/
# admin_tenant_name = %SERVICE_TENANT_NAME%
# admin_user = %SERVICE_USER%
# admin_password = %SERVICE_PASSWORD%

# API to use for accessing data. Default value points to sqlalchemy
# package, it is also possible to use: glance.db.registry.api
#data_api = glance.db.sqlalchemy.api

# ===== Security Options =====
```

```

# AES key for encrypting store 'location' metadata, including
# -- if used -- Swift or S3 credentials
# Should be set to a random string of length 16, 24 or 32 bytes
#metadata_encryption_key = <16, 24 or 32 char registry metadata key>

# ===== Policy Options =====

# The JSON file that defines policies.
#policy_file = policy.json

# Default rule. Enforced when a requested rule is not found.
#policy_default_rule = default

# Directories where policy configuration files are stored.
# They can be relative to any directory in the search path
# defined by the config_dir option, or absolute paths.
# The file defined by policy_file must exist for these
# directories to be searched.
#policy_dirs = policy.d

# ===== Database Options =====+=====

[database]

# The SQLAlchemy connection string used to connect to the
# database (string value)
#connection=sqlite:///glance/openstack/common/db/$sqlite_db

# The SQLAlchemy connection string used to connect to the
# slave database (string value)
#slave_connection=

# timeout before idle sql connections are reaped (integer
# value)
#idle_timeout=3600

# Minimum number of SQL connections to keep open in a pool
# (integer value)
#min_pool_size=1

# Maximum number of SQL connections to keep open in a pool
# (integer value)
#max_pool_size=<None>

# maximum db connection retries during startup. (setting -1
# implies an infinite retry count) (integer value)
#max_retries=10

# interval between retries of opening a sql connection
# (integer value)
#retry_interval=10

# If set, use this value for max_overflow with sqlalchemy
# (integer value)
#max_overflow=<None>

```

```
# Verbosity of SQL debugging information. 0=None,
# 100=Everything (integer value)
#connection_debug=0

# Add python stack traces to SQL as comment strings (boolean
# value)
#connection_trace=false

# If set, use this value for pool_timeout with sqlalchemy
# (integer value)
#pool_timeout=<None>

[oslo_concurrency]

# Enables or disables inter-process locks. (boolean value)
# Deprecated group/name - [DEFAULT]/disable_process_locking
#disable_process_locking = false

# Directory to use for lock files. For security, the specified
# directory should only be writable by the user running the processes
# that need locking. It could be read from environment variable
# OSLO_LOCK_PATH. This setting needs to be the same for both
# glance-scrubber and glance-api service. Default to a temp directory.
# Deprecated group/name - [DEFAULT]/lock_path (string value)
#lock_path = /tmp
```

8.6.7. policy.json

The `/etc/glance/policy.json` file defines additional access controls that apply to the Image service.

```
{
  "context_is_admin": "role:admin",
  "default": "",

  "add_image": "",
  "delete_image": "",
  "get_image": "",
  "get_images": "",
  "modify_image": "",
  "publicize_image": "role:admin",
  "copy_from": "",

  "download_image": "",
  "upload_image": "",

  "delete_image_location": "",
  "get_image_location": "",
  "set_image_location": "",

  "add_member": "",
  "delete_member": "",
  "get_member": "",
  "get_members": "",
```

```

"modify_member": "",

"manage_image_cache": "role:admin",

"get_task": "",
"get_tasks": "",
"add_task": "",
"modify_task": "",

"deactivate": "",
"reactivate": "",

"get_metadef_namespace": "",
"get_metadef_namespaces": "",
"modify_metadef_namespace": "",
"add_metadef_namespace": "",

"get_metadef_object": "",
"get_metadef_objects": "",
"modify_metadef_object": "",
"add_metadef_object": "",

"list_metadef_resource_types": "",
"get_metadef_resource_type": "",
"add_metadef_resource_type_association": "",

"get_metadef_property": "",
"get_metadef_properties": "",
"modify_metadef_property": "",
"add_metadef_property": "",

"get_metadef_tag": "",
"get_metadef_tags": "",
"modify_metadef_tag": "",
"add_metadef_tag": "",
"add_metadef_tags": ""

}

```

8.7. NEW, UPDATED AND DEPRECATED OPTIONS IN LIBERTY FOR OPENSTACK IMAGE SERVICE

New options

New default values

Deprecated options

Table 8.29. New options

Configuration option = Default value	Description
<i>[profiler] hmac_keys = SECRET_KEY</i>	(StrOpt) Secret key to use to sign Glance API and Glance Registry services tracing messages.

Table 8.30. New default values

Option	New default value	New default value
[DEFAULT] allowed_rpc_exception_modules	<i>glance.common.exception, exceptions</i>	<i>glance.common.exception, builtins, exceptions</i>
[DEFAULT] workers	4	None
[image_format] container_formats	<i>ami, ari, aki, bare, ovf, ova</i>	<i>ami, ari, aki, bare, ovf, ova, docker</i>

Table 8.31. Deprecated options

Configuration option = Default value	Description
[DEFAULT] use_syslog	None

CHAPTER 9. NETWORKING

This chapter explains the OpenStack Networking configuration options.

9.1. NETWORKING CONFIGURATION OPTIONS

The options and descriptions listed in this introduction are auto generated from the code in the Networking service project, which provides software-defined networking between VMs run in Compute. The list contains common options, while the subsections list the options for the various networking plug-ins.

Table 9.1. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
agent_down_time = 75	(Integer) Seconds to regard the agent is down; should be at least twice report_interval, to be sure the agent is down for good.
api_workers = <i>None</i>	(Integer) Number of separate API worker processes for service. If not specified, the default is equal to the number of CPUs available for best performance.
auth_ca_cert = <i>None</i>	(String) Certificate Authority public key (CA cert) file for ssl
auth_strategy = <i>keystone</i>	(String) The type of authentication to use
base_mac = <i>fa:16:3e:00:00:00</i>	(String) The base MAC address Neutron will use for VIFs. The first 3 octets will remain unchanged. If the 4th octet is not 00, it will also be used. The others will be randomly generated.
bgp_drscheduler_driver = <i>neutron.services.bgp.scheduler.bgp_dragent_scheduler.ChanceScheduler</i>	(String) Driver used for scheduling BGP speakers to BGP DrAgent
bind_host = <i>0.0.0.0</i>	(String) The host IP to bind to
bind_port = 9696	(Port number) The port to bind to
core_plugin = <i>None</i>	(String) The core plugin Neutron will use

Configuration option = Default value	Description
default_availability_zones =	(List) Default value of availability zone hints. The availability zone aware schedulers use this when the resources availability_zone_hints is empty. Multiple availability zones can be specified by a comma separated string. This value can be empty. In this case, even if availability_zone_hints for a resource is empty, availability zone is considered for high availability while scheduling the resource.
default_ipv4_subnet_pool = None	(String) DEPRECATED: Default IPv4 subnet pool to be used for automatic subnet CIDR allocation. Specifies by UUID the pool to be used in case where creation of a subnet is being called without a subnet pool ID. If not set then no pool will be used unless passed explicitly to the subnet create. If no pool is used, then a CIDR must be passed to create a subnet and that subnet will not be allocated from any pool; it will be considered part of the tenant's private address space. This option is deprecated for removal in the N release.
default_ipv6_subnet_pool = None	(String) DEPRECATED: Default IPv6 subnet pool to be used for automatic subnet CIDR allocation. Specifies by UUID the pool to be used in case where creation of a subnet is being called without a subnet pool ID. See the description for default_ipv4_subnet_pool for more information. This option is deprecated for removal in the N release.
dhcp_agent_notification = True	(Boolean) Allow sending resource operation notification to DHCP agent
dhcp_agents_per_network = 1	(Integer) Number of DHCP agents scheduled to host a tenant network. If this number is greater than 1, the scheduler automatically assigns multiple DHCP agents for a given tenant network, providing high availability for DHCP service.
dhcp_broadcast_reply = False	(Boolean) Use broadcast in DHCP replies.
dhcp_confs = \$state_path/dhcp	(String) Location to store DHCP server config files.
dhcp_domain = openstacklocal	(String) DEPRECATED: Domain to use for building the hostnames. This option is deprecated. It has been moved to neutron.conf as dns_domain. It will be removed in a future release.
dhcp_lease_duration = 86400	(Integer) DHCP lease duration (in seconds). Use -1 to tell dnsmasq to use infinite lease times.

Configuration option = Default value	Description
dhcp_load_type = <i>networks</i>	(String) Representing the resource type whose load is being reported by the agent. This can be "networks", "subnets" or "ports". When specified (Default is networks), the server will extract particular load sent as part of its agent configuration object from the agent report state, which is the number of resources being consumed, at every report_interval.dhcp_load_type can be used in combination with network_scheduler_driver = neutron.scheduler.dhcp_agent_scheduler.WeightScheduler When the network_scheduler_driver is WeightScheduler, dhcp_load_type can be configured to represent the choice for the resource being balanced. Example: dhcp_load_type=networks
dns_domain = <i>openstacklocal</i>	(String) Domain to use for building the hostnames
enable_new_agents = <i>True</i>	(Boolean) Agent starts with admin_state_up=False when enable_new_agents=False. In the case, user's resources will not be scheduled automatically to the agent until admin changes admin_state_up to True.
enable_services_on_agents_with_admin_state_down = <i>False</i>	(Boolean) Enable services on an agent with admin_state_up False. If this option is False, when admin_state_up of an agent is turned False, services on it will be disabled. Agents with admin_state_up False are not selected for automatic scheduling regardless of this option. But manual scheduling to such agents is available if this option is True.
executor_thread_pool_size = <i>64</i>	(Integer) Size of executor thread pool.
external_dns_driver = <i>None</i>	(String) Driver for external DNS integration.
global_physnet_mtu = <i>1500</i>	(Integer) MTU of the underlying physical network. Neutron uses this value to calculate MTU for all virtual network components. For flat and VLAN networks, neutron uses this value without modification. For overlay networks such as VXLAN, neutron automatically subtracts the overlay protocol overhead from this value. Defaults to 1500, the standard value for Ethernet. If using the ML2 plug-in with overlay/tunnel networks, also configure the ml2 path_mtu option with the same value as the global_physnet_mtu option.
ip_lib_force_root = <i>False</i>	(Boolean) Force ip_lib calls to use the root helper

Configuration option = Default value	Description
ipam_driver = <i>None</i>	(String) Neutron IPAM (IP address management) driver to use. If ipam_driver is not set (default behavior), no IPAM driver is used. In order to use the reference implementation of Neutron IPAM driver, use 'internal'.
mac_generation_retries = 16	(Integer) How many times Neutron will retry MAC generation
max_allowed_address_pair = 10	(Integer) Maximum number of allowed address pairs
max_dns_nameservers = 5	(Integer) Maximum number of DNS nameservers per subnet
max_fixed_ips_per_port = 5	(Integer) DEPRECATED: Maximum number of fixed ips per port. This option is deprecated and will be removed in the N release.
max_rtr_adv_interval = 100	(Integer) MaxRtrAdvInterval setting for radvd.conf
max_subnet_host_routes = 20	(Integer) Maximum number of host routes per subnet
memcached_servers = <i>None</i>	(List) Memcached servers or None for in process cache.
min_rtr_adv_interval = 30	(Integer) MinRtrAdvInterval setting for radvd.conf
periodic_fuzzy_delay = 5	(Integer) Range of seconds to randomly delay when starting the periodic task scheduler to reduce stampeding. (Disable by setting to 0)
periodic_interval = 40	(Integer) Seconds between running periodic tasks
report_interval = 300	(Integer) Interval between two metering reports
state_path = <i>/var/lib/neutron</i>	(String) Where to store Neutron state files. This directory must be writable by the agent.
vlan_transparent = <i>False</i>	(Boolean) If True, then allow plugins that support it to create VLAN transparent networks.
web_framework = <i>legacy</i>	(String) This will choose the web framework in which to run the Neutron API server. 'pecan' is a new experiemental rewrite of the API server.
[AGENT]	

Configuration option = Default value	Description
check_child_processes_action = <i>respawn</i>	(String) Action to be executed when a child process dies
check_child_processes_interval = <i>60</i>	(Integer) Interval between checks of child process liveness (seconds), use 0 to disable
log_agent_heartbeats = <i>False</i>	(Boolean) Log agent heartbeats
polling_interval = <i>2</i>	(Integer) The number of seconds the agent will wait between polling for local device changes.
root_helper = <i>sudo</i>	(String) Root helper application. Use 'sudo neutron-rootwrap /etc/neutron/rootwrap.conf' to use the real root filter facility. Change to 'sudo' to skip the filtering and just run the command directly.
root_helper_daemon = <i>None</i>	(String) Root helper daemon application to use when possible.
[keystone_authtoken]	
memcached_servers = <i>None</i>	(List) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.
[qos]	
notification_drivers = <i>message_queue</i>	(List) Drivers list to use to send the update notification
[service_providers]	
service_provider = <i>[]</i>	(Multi-valued) Defines providers for advanced services using the format: <service_type>:<name>:<driver>[:default]

9.1.1. Networking plug-ins

OpenStack Networking introduces the concept of a plug-in, which is a back-end implementation of the OpenStack Networking API. A plug-in can use a variety of technologies to implement the logical API requests. Some OpenStack Networking plug-ins might use basic Linux VLANs and IP tables, while others might use more advanced technologies, such as L2-in-L3 tunneling or OpenFlow. These sections detail the configuration options for the various plug-ins.

**NOTE**

The following plugins have been removed in Kilo:

- Ryu plugin. The Ryu team recommends that you migrate to the ML2 plugin with the ofagent mechanism driver. However, note that the functionality is not the same. There is no upgrade procedure currently available.
- Mellanox plugin.

9.1.1.1. BaGpipe configuration options**Table 9.2. Description of BaGpipe BGP configuration options**

Configuration option = Default value	Description
[BAGPIPE]	
bagpipe_bgp_ip = <i>127.0.0.1</i>	(StrOpt) BGP component REST service IP address.
bagpipe_bgp_port = <i>8082</i>	(IntOpt) BGP component REST service IP port.
mpls_bridge = <i>br-mpls</i>	(StrOpt) OVS MPLS bridge to use.
mpls_from_tun_peer_patch_port = <i>patch-from-tun</i>	(StrOpt) OVS Peer patch port in MPLS bridge to tunnel bridge (traffic from tunnel bridge).
mpls_to_tun_peer_patch_port = <i>patch-to-tun</i>	(StrOpt) OVS Peer patch port in MPLS bridge to tunnel bridge(traffic to tunnel bridge).
ping_interval = <i>10</i>	(IntOpt) The number of seconds the BGP component client will wait between polling for restart detection.
tun_from_mpls_peer_patch_port = <i>patch-from-mpls</i>	(StrOpt) OVS Peer patch port in tunnel bridge to MPLS bridge (traffic from MPLS bridge).
tun_to_mpls_peer_patch_port = <i>patch-to-mpls</i>	(StrOpt) OVS Peer patch port in tunnel bridge to MPLS bridge (traffic to MPLS bridge).

9.1.1.2. BigSwitch configuration options**Table 9.3. Description of BigSwitch configuration options**

Configuration option = Default value	Description
[NOVA]	
node_override_vif_802.1qbg =	(ListOpt) Nova compute nodes to manually set VIF type to 802.1qbg

Configuration option = Default value	Description
node_override_vif_802.1qbh =	(ListOpt) Nova compute nodes to manually set VIF type to 802.1qbh
node_override_vif_binding_failed =	(ListOpt) Nova compute nodes to manually set VIF type to binding_failed
node_override_vif_bridge =	(ListOpt) Nova compute nodes to manually set VIF type to bridge
node_override_vif_distributed =	(ListOpt) Nova compute nodes to manually set VIF type to distributed
node_override_vif_dvs =	(ListOpt) Nova compute nodes to manually set VIF type to dvs
node_override_vif_hw_web =	(ListOpt) Nova compute nodes to manually set VIF type to hw_web
node_override_vif_hyperv =	(ListOpt) Nova compute nodes to manually set VIF type to hyperv
node_override_vif_ib_hostdev =	(ListOpt) Nova compute nodes to manually set VIF type to ib_hostdev
node_override_vif_iovisor =	(ListOpt) Nova compute nodes to manually set VIF type to iovisor
node_override_vif_ivs =	(ListOpt) Nova compute nodes to manually set VIF type to ivs
node_override_vif_midonet =	(ListOpt) Nova compute nodes to manually set VIF type to midonet
node_override_vif_other =	(ListOpt) Nova compute nodes to manually set VIF type to other
node_override_vif_ovs =	(ListOpt) Nova compute nodes to manually set VIF type to ovs
node_override_vif_unbound =	(ListOpt) Nova compute nodes to manually set VIF type to unbound
node_override_vif_vhostuser =	(ListOpt) Nova compute nodes to manually set VIF type to vhostuser
node_override_vif_vrouter =	(ListOpt) Nova compute nodes to manually set VIF type to vrouter

Configuration option = Default value	Description
vif_type = <i>ivs</i>	(StrOpt) Virtual interface type to configure on Nova compute nodes
vif_types = <i>unbound, binding_failed, distributed, ovs, bridge, other, ivs, iovisor, vhostuser, dvs, 802.1qbg, 802.1qbh, hyperv, midonet, ib_hostdev, hw_web, vrouter</i>	(ListOpt) List of allowed vif_type values.
[RESTPROXY]	
add_meta_server_route = <i>True</i>	(BoolOpt) Determines whether a route to the metadata server should be injected into the VM
auto_sync_on_failure = <i>True</i>	(BoolOpt) If neutron fails to create a resource because the back end controller doesn't know of a dependency, the plugin automatically triggers a full data synchronization to the controller.
cache_connections = <i>True</i>	(BoolOpt) Re-use HTTP/HTTPS connections to the controller.
consistency_interval = <i>60</i>	(IntOpt) Time between verifications that the backend controller database is consistent with Neutron. (0 to disable)
neutron_id = <i>neutron-images</i>	(StrOpt) User-defined identifier for this neutron deployment
no_ssl_validation = <i>False</i>	(BoolOpt) Disables SSL certificate validation for controllers
server_auth = <i>None</i>	(StrOpt) The username and password for authenticating against the Big Switch or Floodlight controller.
server_ssl = <i>True</i>	(BoolOpt) If True, use SSL when connecting to the Big Switch or Floodlight controller.
server_timeout = <i>10</i>	(IntOpt) Maximum number of seconds to wait for proxy request to connect and complete.
servers = <i>localhost:8800</i>	(ListOpt) A comma separated list of Big Switch/Floodlight servers, and port numbers. The plugin proxies the requests to the Big Switch/Floodlight server, which performs the networking configuration. Only one server is needed per deployment, but you can deploy multiple servers for failover.

Configuration option = Default value	Description
ssl_cert_directory = <i>/etc/neutron/plugins/bigswitch/ssl</i>	(StrOpt) Directory containing ca_certs and host_certs certificate directories.
ssl_sticky = <i>True</i>	(BoolOpt) Trust and store the first certificate received for each controller address and use it to validate future connections to that address.
sync_data = <i>False</i>	(BoolOpt) Sync data on connect
thread_pool_size = 4	(IntOpt) Maximum number of threads to spawn to handle large volumes of port creations.
[RESTPROXYAGENT]	
integration_bridge = <i>br-int</i>	(StrOpt) Name of integration bridge on Compute nodes used for security group insertion.
polling_interval = 5	(IntOpt) Seconds between agent checks for port changes
virtual_switch_type = <i>ivs</i>	(StrOpt) Virtual switch type.
[ROUTER]	
max_router_rules = 200	(IntOpt) Maximum number of router rules
tenant_default_router_rule = <i>['*:any:any:permit']</i>	(MultiStrOpt) The default router rules installed in new tenant routers. Repeat the config option for each rule. Format is <tenant>:<source>:<destination>:<action> Use an * to specify default for all tenants.

9.1.1.3. Brocade configuration options

Table 9.4. Description of Brocade configuration options

Configuration option = Default value	Description
[PHYSICAL_INTERFACE]	
physical_interface = <i>eth0</i>	(StrOpt) The network interface to use when creating a port
[SWITCH]	
address =	(StrOpt) The address of the host to SSH to

Configuration option = Default value	Description
ostype = <i>NOS</i>	(StrOpt) Currently unused
password =	(StrOpt) The SSH password to use
username =	(StrOpt) The SSH username to use

9.1.1.4. Brocade MLX L3 plug-in

Configure switch names to be used as group names as described below

Table 9.5. Description of Brocade MLX L3 plug-in configuration options

Configuration option = Default value	Description
[L3_BROCADE_MLX_EXAMPLE]	
address =	(StrOpt) The IP address of the MLX switch
password = <i>password</i>	(StrOpt) The SSH password of the switch
physical_networks =	(StrOpt) Allowed physical networks where VLAN can be configured on this switch
ports =	(StrOpt) Ports to be tagged in the VLAN being configured on the switch
username = <i>admin</i>	(StrOpt) The SSH username for the switch
[l3_brocade_mlx]	
switch_names =	(StrOpt) Switches connected to the Compute nodes

9.1.1.5. Brocade Vyatta layer 3 plug-in

The Brocade Vyatta Layer 3 plug-in configures Vyatta vRouter. More information about the plug-in is available at: [Brocade_Vyatta_L3_Plugin](#).

Use the following options to configure the Brocade Vyatta Layer 3 plug-in.

Table 9.6. Description of Brocade Vyatta L3 plug-in configuration options

Configuration option = Default value	Description
[VROUTER]	

Configuration option = Default value	Description
flavor = 2	(StrOpt) Nova VM flavor for instances of Vyatta vRouter.
image_id = None	(StrOpt) Nova image id for instances of Vyatta vRouter.
keystone_url = None	(StrOpt) Keystone URL.
management_network_id = None	(StrOpt) Vyatta vRouter management network id.
nova_poll_interval = 5	(IntOpt) Number of seconds between consecutive Nova queries when waiting for router instance status change.
nova_spawn_timeout = 300	(IntOpt) Number of seconds to wait for Nova to activate instance before setting resource to error state.
tenant_admin_name = None	(StrOpt) Name of tenant admin user.
tenant_admin_password = None	(StrOpt) Tenant admin password.
tenant_id = None	(StrOpt) UUID of tenant that holds Vyatta vRouter instances.
vrouter_boot_timeout = 300	(IntOpt) Number of seconds to wait for Vyatta vRouter to boot before setting resource to error state.
vrouter_credentials = vyatta:vyatta	(StrOpt) Vyatta vRouter login credentials
vrouter_poll_interval = 5	(IntOpt) Number of seconds between consecutive Vyatta vRouter queries when waiting for router instance boot.

9.1.1.6. CISCO configuration options

Table 9.7. Description of Cisco configuration options

Configuration option = Default value	Description
[cfg_agent]	
device_connection_timeout = 30	(IntOpt) Time in seconds for connecting to a hosting device

Configuration option = Default value	Description
fw_svc_helper_class = <i>neutron_fwaas.services.firewall.drivers.cisco.csr_firewall_svc_helper.CsrFirewallServiceHelper</i>	(StrOpt) Path of the firewall service helper class.
hosting_device_dead_timeout = 300	(IntOpt) The time in seconds until a backlogged hosting device is presumed dead. This value should be set up high enough to recover from a period of connectivity loss or high load when the device may not be responding.
routing_svc_helper_class = <i>networking_cisco.plugins.cisco.cfg_agent.service_helpers.routing_svc_helper.RoutingServiceHelper</i>	(StrOpt) Path of the routing service helper class.
rpc_loop_interval = 10	(IntOpt) Interval when the process_services() loop executes in seconds. This is when the config agent lets each service helper process its neutron resources.
[cisco_csr_ipsec]	
status_check_interval = 60	(IntOpt) Status check interval for Cisco CSR IPSec connections
[general]	
backlog_processing_interval = 10	(IntOpt) Time in seconds between renewed scheduling attempts of non-scheduled routers.
cfg_agent_down_time = 60	(IntOpt) Seconds of no status update until a cfg agent is considered down.
default_security_group = <i>mgmt_sec_grp</i>	(StrOpt) Default security group applied on management port. Default value is mgmt_sec_grp.
ensure_nova_running = <i>True</i>	(BoolOpt) Ensure that nova is running before attempting to create a VM.
l3_admin_tenant = <i>L3AdminTenant</i>	(StrOpt) Name of the L3 admin tenant.
management_network = <i>osn_mgmt_nw</i>	(StrOpt) Name of management network for device configuration. Default value is osn_mgmt_nw
service_vm_config_path = <i>/opt/stack/data/neutron/cisco/config_drive</i>	(StrOpt) Path to config drive files for service VM instances.
templates_path = <i>/opt/stack/data/neutron/cisco/templates</i>	(StrOpt) Path to templates for hosting devices.

Configuration option = Default value	Description
[hosting_devices]	
csr1kv_booting_time = 420	(IntOpt) Booting time in seconds before a CSR1kv becomes operational.
csr1kv_cfgagent_router_driver = <i>networking_cisco.plugins.cisco.cfg_agent.device_drivers.csr1kv.csr1kv_routing_driver.CSR1kvRoutingDriver</i>	(StrOpt) Config agent driver for CSR1kv.
csr1kv_configdrive_template = <i>csr1kv_cfg_template</i>	(StrOpt) CSR1kv configdrive template file.
csr1kv_device_driver = <i>networking_cisco.plugins.cisco.l3.hosting_device_drivers.csr1kv_hd_driver.CSR1kvHostingDeviceDriver</i>	(StrOpt) Hosting device driver for CSR1kv.
csr1kv_flavor = 621	(StrOpt) UUID of Nova flavor for CSR1kv.
csr1kv_image = <i>csr1kv_openstack_img</i>	(StrOpt) Name of Glance image for CSR1kv.
csr1kv_password = <i>cisco</i>	(StrOpt) Password to use for CSR1kv configurations.
csr1kv_plugging_driver = <i>networking_cisco.plugins.cisco.l3.plugging_drivers.n1kv_trunking_driver.N1kvTrunkingPlugDriver</i>	(StrOpt) Plugging driver for CSR1kv.
csr1kv_username = <i>stack</i>	(StrOpt) Username to use for CSR1kv configurations.
[ml2_cisco_n1kv]	
max_vsm_retries = 2	(IntOpt) Maximum number of retry attempts for VSM REST API.
[n1kv]	
management_port_profile = <i>osn_mgmt_pp</i>	(StrOpt) Name of N1kv port profile for management ports.
t1_network_profile = <i>osn_t1_np</i>	(StrOpt) Name of N1kv network profile for T1 networks (for example, trunk networks for VXLAN segmented traffic).

Configuration option = Default value	Description
t1_port_profile = <i>osn_t1_pp</i>	(StrOpt) Name of N1kv port profile for T1 ports (for example, ports carrying traffic from VXLAN segmented networks).
t2_network_profile = <i>osn_t2_np</i>	(StrOpt) Name of N1kv network profile for T2 networks (for example, trunk networks for VLAN segmented traffic).
t2_port_profile = <i>osn_t2_pp</i>	(StrOpt) Name of N1kv port profile for T2 ports (for example, ports carrying traffic from VLAN segmented networks).

9.1.1.7. Fujitsu CFAB configuration options

Table 9.8. Description of FUJITSU Converged Fabric Switch configuration options

Configuration option = Default value	Description
[fujitsu_cfab]	
address =	(StrOpt) The address of the C-Fabric to telnet to.
password = <i>admin</i>	(StrOpt) The C-Fabric password to use.
physical_networks =	(ListOpt) List of <physical_network>:<vfab_id> tuples specifying physical_network names and corresponding vfab ids.
pprofile_prefix =	(StrOpt) The prefix string for pprofile name.
save_config = <i>True</i>	(BoolOpt) Whether to save configuration.
share_pprofile = <i>False</i>	(BoolOpt) Whether to share a C-Fabric pprofile among Neutron ports using the same VLAN ID.
username = <i>admin</i>	(StrOpt) The C-Fabric username to use.

9.1.1.8. Fujitsu ISM configuration options

Table 9.9. Description of FUJITSU Software ServerView Infrastructure Manager configuration options

Configuration option = Default value	Description
[fujitsu_ism]	

Configuration option = Default value	Description
address = <i>furukawa-ism</i>	(StrOpt) The IP address or hostname of the ISM.
certificate_authority = <i>/etc/neutron/plugins/ml2/fujitsu/server.crt</i>	(StrOpt) The certification authority for ISM.
password = <i>admin</i>	(StrOpt) The ISM password to use.
port = <i>25566</i>	(StrOpt) The port number of the ISM.
timeout = <i>30</i>	(StrOpt) The API timeout value for ISM.
username = <i>admin</i>	(StrOpt) The ISM username to use.

9.1.1.9. CloudBase Hyper-V Agent configuration options

Table 9.10. Description of HyperV agent configuration options

Configuration option = Default value	Description
[AGENT]	
enable_metrics_collection = <i>False</i>	(BoolOpt) Enables metrics collections for switch ports by using Hyper-V's metric APIs. Collected data can be retrieved by other apps and services, e.g. ceilometer. Requires Hyper-V / Windows Server 2012 and above
local_network_vswitch = <i>private</i>	(StrOpt) Private vswitch name used for local networks
metrics_max_retries = <i>100</i>	(IntOpt) Specifies the maximum number of retries to enable Hyper-V's port metrics collection. The agent will try to enable the feature once every <i>polling_interval</i> period for at most <i>metrics_max_retries</i> , or until it succeeds.
neutron_metadata_address = <i>169.254.169.254</i>	(StrOpt) Specifies the address which will serve the metadata for the instance.
physical_network_vswitch_mappings =	(ListOpt) List of <physical_network>:<vswitch> where the physical networks can be expressed with wildcards, e.g. <i>.*:external</i>
polling_interval = <i>2</i>	(IntOpt) The number of seconds the agent will wait between polling for local device changes.
[NVGRE]	

Configuration option = Default value	Description
enable_support = <i>False</i>	(BoolOpt) Enables Hyper-V NVGRE. Requires Windows Server 2012 or above.
provider_tunnel_ip = <i>None</i>	(StrOpt) Specifies the tunnel IP which will be used and reported by this host for NVGRE networks.
provider_vlan_id = <i>0</i>	(IntOpt) Specifies the VLAN ID of the physical network, required for setting the NVGRE Provider Address.
[hyperv]	
force_hyperv_utils_v1 = <i>False</i>	(BoolOpt) Force V1 WMI utility classes
[neutron]	
admin_auth_url = <i>http://localhost:5000/v2.0</i>	(StrOpt) auth url for connecting to neutron in admin context
admin_password = <i>None</i>	(StrOpt) password for connecting to neutron in admin context
admin_tenant_name = <i>None</i>	(StrOpt) tenant name for connecting to neutron in admin context
admin_username = <i>None</i>	(StrOpt) username for connecting to neutron in admin context
auth_strategy = <i>keystone</i>	(StrOpt) auth strategy for connecting to neutron in admin context
url = <i>http://127.0.0.1:9696</i>	(StrOpt) URL for connecting to neutron
url_timeout = <i>30</i>	(IntOpt) timeout value for connecting to neutron in seconds

9.1.1.10. Embrane configuration options

Table 9.11. Description of Embrane configuration options

Configuration option = Default value	Description
[heleos]	
admin_username = <i>admin</i>	(StrOpt) ESM admin username.

Configuration option = Default value	Description
async_requests = <i>True</i>	(BoolOpt) Define if the requests have run asynchronously or not
dummy_utif_id = <i>None</i>	(StrOpt) Dummy user traffic Security Zone id
esm_mgmt = <i>None</i>	(StrOpt) ESM management root address
inband_id = <i>None</i>	(StrOpt) In band Security Zone id
mgmt_id = <i>None</i>	(StrOpt) Management Security Zone id
oob_id = <i>None</i>	(StrOpt) Out of band Security Zone id
resource_pool_id = <i>default</i>	(StrOpt) Shared resource pool id
router_image = <i>None</i>	(StrOpt) Router image id (Embrane FW/VPN)

9.1.1.11. IBM SDN-VE configuration options

Table 9.12. Description of SDN-VE configuration options

Configuration option = Default value	Description
[SDNVE]	
base_url = <i>/one/nb/v2/</i>	(StrOpt) Base URL for SDN-VE controller REST API.
controller_ips = <i>127.0.0.1</i>	(ListOpt) List of IP addresses of SDN-VE controller(s).
default_tenant_type = <i>OVERLAY</i>	(StrOpt) Tenant type: OVERLAY (default) or OF.
format = <i>json</i>	(StrOpt) SDN-VE request/response format.
info = <i>sdnve_info_string</i>	(StrOpt) SDN-VE RPC subject.
integration_bridge = <i>None</i>	(StrOpt) Integration bridge to use.
interface_mappings =	(ListOpt) List of <physical_network_name>: <interface_name> mappings.
of_signature = <i>SDNVE-OF</i>	(StrOpt) The string in tenant description that indicates the tenant is a OF tenant.

Configuration option = Default value	Description
out_of_band = <i>True</i>	(BoolOpt) Indicating if controller is out of band or not.
overlay_signature = <i>SDNVE-OVERLAY</i>	(StrOpt) The string in tenant description that indicates the tenant is a OVERLAY tenant.
password = <i>admin</i>	(StrOpt) SDN-VE administrator password.
port = <i>8443</i>	(StrOpt) SDN-VE controller port number.
reset_bridge = <i>True</i>	(BoolOpt) Whether to reset the integration bridge before use.
use_fake_controller = <i>False</i>	(BoolOpt) Whether to use a fake controller.
userid = <i>admin</i>	(StrOpt) SDN-VE administrator user ID.
[SDNVE_AGENT]	
polling_interval = <i>2</i>	(IntOpt) Agent polling interval if necessary.
rpc = <i>True</i>	(BoolOpt) Whether to use rpc.

9.1.1.12. Layer 2 Gateway configuration options

Table 9.13. Description of L2 agent extension configuration options

Configuration option = Default value	Description
[agent]	
extensions =	(List) Extensions list to use

9.1.1.13. Layer 2 Gateway configuration options

Table 9.14. Description of Layer 2 Gateway configuration options

Configuration option = Default value	Description
[DEFAULT]	
default_device_name = <i>Switch1</i>	(StrOpt) default_device_name of the l2 gateway

Configuration option = Default value	Description
<code>default_l2_gw_service_uuid = None</code>	(StrOpt) Unique identifier of the NSX L2 Gateway service which will be used by default for network gateways
<code>default_l3_gw_service_uuid = None</code>	(StrOpt) Unique identifier of the NSX L3 Gateway service which will be used for implementing routers and floating IPs
<code>l2gw_callback_class = networking_l2gw.services.l2gateway.ovsdb.data.L2GatewayOVSDBCallbacks</code>	(StrOpt) L2 gateway plugin callback class where the RPCs from the agent are going to get invoked
<code>quota_l2_gateway = 5</code>	(IntOpt) Number of L2 gateways allowed per tenant, -1 for unlimited
[ovsdb]	
<code>enable_manager = False</code>	(BoolOpt) Set to 'True' if ovsdb Manager manages the client
<code>l2_gw_agent_ca_cert_base_path = None</code>	(StrOpt) Trusted issuer CA cert
<code>l2_gw_agent_cert_base_path = None</code>	(StrOpt) L2 gateway agent public certificate
<code>l2_gw_agent_priv_key_base_path = None</code>	(StrOpt) L2 gateway agent private key
<code>max_connection_retries = 10</code>	(IntOpt) Maximum number of retries to open a socket with the OVSDb server
<code>ovsdb_hosts = host1:127.0.0.1:6632</code>	(StrOpt) OVSDb server name:host/IP:port
<code>periodic_interval = 20</code>	(IntOpt) Seconds between periodic task runs

9.1.1.14. Linux bridge Agent configuration options

Table 9.15. Description of Linux Bridge agent configuration options

Configuration option = Default value	Description
[AGENT]	
<code>quitting_rpc_timeout = 10</code>	(Integer) Set new timeout in seconds for new rpc calls after agent receives SIGTERM. If value is set to 0, rpc timeout won't be changed
[LINUX_BRIDGE]	

Configuration option = Default value	Description
bridge_mappings =	(List) List of <physical_network>:<physical_bridge>
physical_interface_mappings =	(List) Comma-separated list of <physical_network>:<physical_interface> tuples mapping physical network names to the agent's node-specific physical network interfaces to be used for flat and VLAN networks. All physical networks listed in <code>network_vlan_ranges</code> on the server should have mappings to appropriate interfaces on each agent.
[VXLAN]	
arp_responder = <i>False</i>	(Boolean) Enable local ARP responder which provides local responses instead of performing ARP broadcast into the overlay. Enabling local ARP responder is not fully compatible with the allowed-address-pairs extension.
enable_vxlan = <i>True</i>	(Boolean) Enable VXLAN on the agent. Can be enabled when agent is managed by ml2 plugin using linuxbridge mechanism driver
l2_population = <i>False</i>	(Boolean) Extension to use alongside ml2 plugin's l2population mechanism driver. It enables the plugin to populate VXLAN forwarding table.
local_ip = <i>None</i>	(Unknown) Local IP address of the VXLAN endpoints.
tos = <i>None</i>	(Integer) TOS for vxlan interface protocol packets.
ttl = <i>None</i>	(Integer) TTL for vxlan interface protocol packets.
vxlan_group = <i>224.0.0.1</i>	(String) Multicast group(s) for vxlan interface. A range of group addresses may be specified by using CIDR notation. Specifying a range allows different VNIs to use different group addresses, reducing or eliminating spurious broadcast traffic to the tunnel endpoints. To reserve a unique group for each possible (24-bit) VNI, use a /8 such as 239.0.0.0/8. This setting must be the same on all the agents.

9.1.1.15. Modular Layer 2 (ml2) configuration options

The Modular Layer 2 (ml2) plug-in has two components: network types and mechanisms. You can configure these components separately. This section describes these configuration options.



CONFIGURE MTU FOR VXLAN TUNNELLING

Specific MTU configuration is necessary for VXLAN to function as expected:

- One option is to increase the MTU value of the physical interface and physical switch fabric by at least 50 bytes. For example, increase the MTU value to 1550. This value enables an automatic 50-byte MTU difference between the physical interface (1500) and the VXLAN interface (automatically $1500 - 50 = 1450$). An MTU value of 1450 causes issues when virtual machine taps are configured at an MTU value of 1500.
- Another option is to decrease the virtual Ethernet devices' MTU. Set the `network_device_mtu` option to 1450 in the `neutron.conf` file, and set all guest virtual machines' MTU to the same value by using a DHCP option. For information about how to use this option, see [Configure OVS plug-in](#).

Table 9.16. Description of ML2 configuration options

Configuration option = Default value	Description
[ml2]	
<code>extension_drivers =</code>	(List) An ordered list of extension driver entrypoints to be loaded from the <code>neutron.ml2.extension_drivers</code> namespace. For example: <code>extension_drivers = port_security,qos</code>
<code>external_network_type = None</code>	(String) Default network type for external networks when no provider attributes are specified. By default it is <code>None</code> , which means that if provider attributes are not specified while creating external networks then they will have the same type as tenant networks. Allowed values for <code>external_network_type</code> config option depend on the network type values configured in <code>type_drivers</code> config option.
<code>mechanism_drivers =</code>	(List) An ordered list of networking mechanism driver entrypoints to be loaded from the <code>neutron.ml2.mechanism_drivers</code> namespace.
<code>path_mtu = 1500</code>	(Integer) Maximum size of an IP packet (MTU) that can traverse the underlying physical network infrastructure without fragmentation for overlay/tunnel networks. In most cases, use the same value as the <code>global_physnet_mtu</code> option.
<code>physical_network_mtus =</code>	(List) A list of mappings of physical networks to MTU values. The format of the mapping is <code><physnet>: <mtu val></code> . This mapping allows specifying a physical network MTU value that differs from the default <code>global_physnet_mtu</code> value.

Configuration option = Default value	Description
tenant_network_types = <i>local</i>	(List) Ordered list of network_types to allocate as tenant networks. The default value 'local' is useful for single-box testing but provides no connectivity between hosts.
type_drivers = <i>local, flat, vlan, gre, vxlan, geneve</i>	(List) List of network type driver entrypoints to be loaded from the neutron.ml2.type_drivers namespace.

9.1.1.15.1. Modular Layer 2 (ml2) Flat Type configuration options

Table 9.17. Description of ML2 Flat mechanism driver configuration options

Configuration option = Default value	Description
[ml2_type_flat]	
flat_networks = *	(List) List of physical_network names with which flat networks can be created. Use default '*' to allow flat networks with arbitrary physical_network names. Use an empty list to disable flat networks.

9.1.1.15.2. Modular Layer 2 (ml2) GRE Type configuration options

Table 9.18. Description of ML2 GRE configuration options

Configuration option = Default value	Description
[ml2_type_gre]	
tunnel_id_ranges =	(List) Comma-separated list of <tun_min>: <tun_max> tuples enumerating ranges of GRE tunnel IDs that are available for tenant network allocation

9.1.1.15.3. Modular Layer 2 (ml2) VLAN Type configuration options

Table 9.19. Description of ML2 VLAN configuration options

Configuration option = Default value	Description
[ml2_type_vlan]	

Configuration option = Default value	Description
network_vlan_ranges =	(List) List of <physical_network>:<vlan_min>:<vlan_max> or <physical_network> specifying physical_network names usable for VLAN provider and tenant networks, as well as ranges of VLAN tags on each available for allocation to tenant networks.

9.1.15.4. Modular Layer 2 (ml2) VXLAN Type configuration options

Table 9.20. Description of ML2 VXLN configuration options

Configuration option = Default value	Description
[ml2_type_vxlan]	
vni_ranges =	(List) Comma-separated list of <vni_min>:<vni_max> tuples enumerating ranges of VXLAN VNI IDs that are available for tenant network allocation
vxlan_group = None	(String) Multicast group for VXLAN. When configured, will enable sending all broadcast traffic to this multicast group. When left unconfigured, will disable multicast VXLAN mode.

9.1.15.5. Modular Layer 2 (ml2) Arista Mechanism configuration options

Table 9.21. Description of ML2 Arista mechanism driver configuration options

Configuration option = Default value	Description
[ml2_arista]	
eapi_host =	(StrOpt) Arista EOS IP address. This is required field. If not set, all communications to Arista EOS will fail.
eapi_password =	(StrOpt) Password for Arista EOS. This is required field. If not set, all communications to Arista EOS will fail.
eapi_username =	(StrOpt) Username for Arista EOS. This is required field. If not set, all communications to Arista EOS will fail.

Configuration option = Default value	Description
region_name = <i>RegionOne</i>	(StrOpt) Defines Region Name that is assigned to this OpenStack Controller. This is useful when multiple OpenStack/neutron controllers are managing the same Arista HW clusters. Note that this name must match with the region name registered (or known) to keystone service. Authentication with keystone is performed by EOS. This is an optional field. If not set, a value of 'RegionOne' is assumed.
sync_interval = <i>180</i>	(IntOpt) Sync interval in seconds between neutron plugin and EOS. This interval defines how often the synchronization is performed. This is an optional field. If not set, a value of '180' seconds is assumed.
use_fqdn = <i>True</i>	(BoolOpt) Defines if hostnames are sent to Arista EOS as FQDNs ("node1.domain.com") or as short names ("node1"). This is optional field. If not set, a value of 'True' is assumed.

Table 9.22. Description of Arista layer-3 service plug-in configuration options

Configuration option = Default value	Description
[l3_arista]	
l3_sync_interval = <i>180</i>	(IntOpt) Sync interval in seconds between L3 Service plugin and EOS. This interval defines how often the synchronization is performed. This is an optional field. If not set, a value of 180 seconds is assumed
mlag_config = <i>False</i>	(BoolOpt) This flag is used indicate if Arista Switches are configured in MLAG mode. If yes, all L3 config is pushed to both the switches automatically. If this flag is set to True, ensure to specify IP addresses of both switches. This is optional. If not set, a value of "False" is assumed.
primary_l3_host =	(StrOpt) Arista EOS IP address. This is required field. If not set, all communications to Arista EOS will fail
primary_l3_host_password =	(StrOpt) Password for Arista EOS. This is required field. If not set, all communications to Arista EOS will fail

Configuration option = Default value	Description
primary_13_host_username =	(StrOpt) Username for Arista EOS. This is required field. If not set, all communications to Arista EOS will fail
secondary_13_host =	(StrOpt) Arista EOS IP address for second Switch MLAGed with the first one. This an optional field, however, if mlag_config flag is set, then this is required. If not set, all communications to Arista EOS will fail
use_vrf = False	(BoolOpt) A "True" value for this flag indicates to create a router in VRF. If not set, all routers are created in default VRF. This is optional. If not set, a value of "False" is assumed.

9.1.1.15.6. Modular Layer 2 (ml2) BaGpipe Mechanism configuration options

Table 9.23. Description of ML2 BaGpipe BGP driver configuration options

Configuration option = Default value	Description
[ml2_bagpipe]	
as_number = 64512	(IntOpt) Autonomous System number
[ml2_type_route_target]	
rt_asn = 64512	(IntOpt) Route Target Autonomous System number.
rt_nn_ranges =	(ListOpt) Comma-separated list of <rt_nn_min>: <rt_nn_max> tuples enumerating ranges of Route Target number that are available for tenant network allocation

9.1.1.15.7. Modular Layer 2 (ml2) BigSwitch Mechanism configuration options

Table 9.24. Description of ML2 BigSwitch mechanism driver configuration options

Configuration option = Default value	Description
[NOVA]	
node_override_vif_802.1qbg =	(ListOpt) Nova compute nodes to manually set VIF type to 802.1qbg
node_override_vif_802.1qbh =	(ListOpt) Nova compute nodes to manually set VIF type to 802.1qbh

Configuration option = Default value	Description
node_override_vif_binding_failed =	(ListOpt) Nova compute nodes to manually set VIF type to binding_failed
node_override_vif_bridge =	(ListOpt) Nova compute nodes to manually set VIF type to bridge
node_override_vif_distributed =	(ListOpt) Nova compute nodes to manually set VIF type to distributed
node_override_vif_dvs =	(ListOpt) Nova compute nodes to manually set VIF type to dvs
node_override_vif_hw_web =	(ListOpt) Nova compute nodes to manually set VIF type to hw_web
node_override_vif_hyperv =	(ListOpt) Nova compute nodes to manually set VIF type to hyperv
node_override_vif_ib_hostdev =	(ListOpt) Nova compute nodes to manually set VIF type to ib_hostdev
node_override_vif_iovisor =	(ListOpt) Nova compute nodes to manually set VIF type to iovisor
node_override_vif_ivs =	(ListOpt) Nova compute nodes to manually set VIF type to ivs
node_override_vif_midonet =	(ListOpt) Nova compute nodes to manually set VIF type to midonet
node_override_vif_other =	(ListOpt) Nova compute nodes to manually set VIF type to other
node_override_vif_ovs =	(ListOpt) Nova compute nodes to manually set VIF type to ovs
node_override_vif_unbound =	(ListOpt) Nova compute nodes to manually set VIF type to unbound
node_override_vif_vhostuser =	(ListOpt) Nova compute nodes to manually set VIF type to vhostuser
node_override_vif_vrouter =	(ListOpt) Nova compute nodes to manually set VIF type to vrouter
vif_type = <i>ivs</i>	(StrOpt) Virtual interface type to configure on Nova compute nodes

Configuration option = Default value	Description
vif_types = <i>unbound, binding_failed, distributed, ovs, bridge, other, ivs, iovisor, vhostuser, dvs, 802.1qbg, 802.1qbh, hyperv, midonet, ib_hostdev, hw_web, vrouter</i>	(ListOpt) List of allowed vif_type values.
[RESTPROXY]	
add_meta_server_route = <i>True</i>	(BoolOpt) Determines if a route to the metadata server should be injected into the VM.
auto_sync_on_failure = <i>True</i>	(BoolOpt) If neutron fails to create a resource because the back end controller doesn't know of a dependency, the plugin automatically triggers a full data synchronization to the controller.
cache_connections = <i>True</i>	(BoolOpt) Re-use HTTP/HTTPS connections to the controller.
consistency_interval = <i>60</i>	(IntOpt) Time between verifications that the backend controller database is consistent with Neutron. (0 to disable)
neutron_id = <i>neutron-ubuntu1404-master</i>	(StrOpt) User defined identifier for this Neutron deployment
no_ssl_validation = <i>False</i>	(BoolOpt) Disables SSL certificate validation for controllers
server_auth = <i>None</i>	(StrOpt) The username and password for authenticating against the Big Switch or Floodlight controller.
server_ssl = <i>True</i>	(BoolOpt) If True, Use SSL when connecting to the Big Switch or Floodlight controller.
server_timeout = <i>10</i>	(IntOpt) Maximum number of seconds to wait for proxy request to connect and complete.
servers = <i>localhost:8800</i>	(ListOpt) A comma separated list of Big Switch or Floodlight servers and port numbers. The plugin proxies the requests to the Big Switch/Floodlight server, which performs the networking configuration. Only one server is needed per deployment, but you can deploy multiple servers for failover.
ssl_cert_directory = <i>/etc/neutron/plugins/bigswitch/ssl</i>	(StrOpt) Directory containing ca_certs and host_certs certificate directories.

Configuration option = Default value	Description
ssl_sticky = <i>True</i>	(BoolOpt) Trust and store the first certificate received for each controller address and use it to validate future connections to that address.
sync_data = <i>False</i>	(BoolOpt) Sync data on connect
thread_pool_size = 4	(IntOpt) Maximum number of threads to spawn to handle large volumes of port creations.
[RESTPROXYAGENT]	
integration_bridge = <i>br-int</i>	(StrOpt) Name of integration bridge on compute nodes used for security group insertion.
polling_interval = 5	(IntOpt) Seconds between agent checks for port changes
virtual_switch_type = <i>ivs</i>	(StrOpt) Virtual switch type.
[ROUTER]	
max_router_rules = 200	(IntOpt) Maximum number of router rules
tenant_default_router_rule = <i>['*:any:any:permit']</i>	(MultiStrOpt) The default router rules installed in new tenant routers. Repeat the config option for each rule. Format is <tenant>:<source>:<destination>:<action> Use an * to specify default for all tenants.

9.1.15.8. Modular Layer 2 (ml2) Brocade Mechanism configuration options

Table 9.25. Description of ML2 Brocade mechanism driver configuration options

Configuration option = Default value	Description
[ML2_BROCADE_MLX_EXAMPLE]	
address =	(StrOpt) The address of the host to SSH to
ostype = <i>NI</i>	(StrOpt) OS type of the device.
password = <i>password</i>	(StrOpt) The SSH password to use
physical_networks =	(StrOpt) Allowed physical networks
ports =	(StrOpt) Ports

Configuration option = Default value	Description
transport = <i>SSH</i>	(StrOpt) Protocol used to communicate with the switch
username = <i>admin</i>	(StrOpt) The SSH username to use
[ml2_brocade]	
address =	(StrOpt) The address of the host to SSH to
ostype = <i>NOS</i>	(StrOpt) OS Type of the switch
osversion = <i>4.0.0</i>	(StrOpt) OS Version number
password = <i>password</i>	(StrOpt) The SSH password to use
physical_networks =	(StrOpt) Allowed physical networks
rbridge_id = <i>1</i>	(StrOpt) Rbridge id of provider edge router(s)
username = <i>admin</i>	(StrOpt) The SSH username to use

9.1.15.9. Modular Layer 3 (ml2) Brocade MLX ICX Mechanism configuration options

Configure switch names to be used as group names as described below

Table 9.26. Description of ML2 Brocade MLX ICX mechanism driver configuration options

Configuration option = Default value	Description
[ml2_brocade_fi_ni]	
switch_names =	(StrOpt) Switches connected to the compute nodes

9.1.15.10. Modular Layer 2 (ml2) Cisco Mechanism configuration options

Table 9.27. Description of ML2 Cisco mechanism driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
apic_system_id = <i>openstack</i>	(StrOpt) Prefix for APIC domain/names/profiles created
[ml2_cisco]	

Configuration option = Default value	Description
host_key_checks = <i>False</i>	(BoolOpt) Enable strict host key checks when connecting to Nexus switches
managed_physical_network = <i>None</i>	(StrOpt) The physical network managed by the switches.
never_cache_ssh_connection = <i>False</i>	(BoolOpt) Prevent caching SSH connections to Nexus device
persistent_switch_config = <i>False</i>	(BoolOpt) To make Nexus configuration persistent
provider_vlan_auto_create = <i>True</i>	(BoolOpt) Provider VLANs are automatically created as needed on the Nexus switch
provider_vlan_auto_trunk = <i>True</i>	(BoolOpt) Provider VLANs are automatically trunked as needed on the ports of the Nexus switch
provider_vlan_name_prefix = <i>p-</i>	(StrOpt) VLAN Name prefix for provider VLANs
svi_round_robin = <i>False</i>	(BoolOpt) Distribute SVI interfaces over all switches
switch_heartbeat_time = <i>0</i>	(IntOpt) Periodic time to check switch connection. (0=disabled)
vlan_name_prefix = <i>q-</i>	(StrOpt) VLAN Name prefix
vxlan_global_config = <i>False</i>	(BoolOpt) Create and delete Nexus switch VXLAN global settings; feature nv overlay, feature vn-segment-vlan-based, interface nve + source-interface loopback
[ml2_cisco_apic]	
apic_agent_poll_interval = <i>2</i>	(FloatOpt) Interval between agent poll for topology (in sec)
apic_agent_report_interval = <i>30</i>	(FloatOpt) Interval between agent status updates (in sec)
apic_app_profile_name = <i>\${apic_system_id}_app</i>	(StrOpt) Name for the app profile used for Openstack
apic_domain_name = <i>\${apic_system_id}</i>	(StrOpt) Name for the domain created on APIC
apic_entity_profile = <i>\${apic_system_id}_entity_profile</i>	(StrOpt) Name of the entity profile to be created

Configuration option = Default value	Description
apic_function_profile = <i>\${apic_system_id}_function_profile</i>	(StrOpt) Name of the function profile to be created
apic_host_uplink_ports =	(ListOpt) The uplink ports to check for ACI connectivity
apic_hosts =	(ListOpt) An ordered list of host names or IP addresses of the APIC controller(s).
apic_lacp_profile = <i>\${apic_system_id}_lacp_profile</i>	(StrOpt) Name of the LACP profile to be created
apic_name_mapping = <i>use_name</i>	(StrOpt) Name mapping strategy to use: <i>use_uuid</i> <i>use_name</i>
apic_node_profile = <i>\${apic_system_id}_node_profile</i>	(StrOpt) Name of the node profile to be created
apic_password = <i>None</i>	(StrOpt) Password for the APIC controller
apic_sync_interval = <i>0</i>	(IntOpt) Synchronization interval in seconds
apic_use_ssl = <i>True</i>	(BoolOpt) Use SSL to connect to the APIC controller
apic_username = <i>None</i>	(StrOpt) Username for the APIC controller
apic_vlan_ns_name = <i>\${apic_system_id}_vlan_ns</i>	(StrOpt) Name of the VLAN namespace to be used for Openstack
apic_vlan_range = <i>2:4093</i>	(StrOpt) Range of VLANs to be used for Openstack
apic_vpc_pairs =	(ListOpt) The switch pairs for VPC connectivity
[ml2_cisco_n1kv]	
default_policy_profile = <i>default-pp</i>	(StrOpt) Cisco Nexus1000V default policy profile.
http_pool_size = <i>4</i>	(IntOpt) Number of threads to use to make HTTP requests.
http_timeout = <i>15</i>	(IntOpt) HTTP timeout, in seconds, for connections to the Cisco Nexus1000V VSMs.
n1kv_vsm_ips = <i>None</i>	(ListOpt) Comma-separated IP Addresses of the Cisco Nexus1000V VSMs.

Configuration option = Default value	Description
password = <i>None</i>	(StrOpt) Password for all configured Cisco Nexus1000V VSMs.
poll_duration = 60	(IntOpt) Cisco Nexus1000V policy profile polling duration in seconds.
restrict_network_profiles = <i>False</i>	(BoolOpt) Restrict the visibility of network profiles to the tenants.
restrict_policy_profiles = <i>False</i>	(BoolOpt) Restrict the visibility of policy profiles to the tenants.
sync_interval = 300	(IntOpt) Time interval between consecutive neutron-VSM syncs.
username = <i>None</i>	(StrOpt) Username for all configured Cisco Nexus1000V VSMs.
[ml2_cisco_ucsm]	
supported_pci_devs = <i>1137:0071, 8086:10c9</i>	(ListOpt) List of comma separated vendor_id:product_id of SR_IOV capable devices supported by this MD. This MD supports both VM-FEX and SR-IOV devices.
ucsm_host_list = <i>None</i>	(ListOpt) List of comma separated Host:Service Profile tuples providing the Service Profile associated with each host to be supported by this MD.
ucsm_ip = <i>None</i>	(StrOpt) Cisco UCS Manager IP address. This is a required field to communicate with a Cisco UCS Manager.
ucsm_password = <i>None</i>	(StrOpt) Password for UCS Manager. This is a required field to communicate with a Cisco UCS Manager.
ucsm_username = <i>None</i>	(StrOpt) Username for UCS Manager. This is a required field to communicate with a Cisco UCS Manager.
[ml2_type_nexus_vxlan]	
mcast_ranges =	(ListOpt) List of multicast groups to be used for global VNIDs in the format - a:b,c,e:f.

Configuration option = Default value	Description
vni_ranges =	(ListOpt) List of global VNID ranges in the format - a:b, c:d. Multiple ranges can be separated by a comma

9.1.15.11. Modular Layer 2 (ml2) Freescale SDN Mechanism configuration options

Table 9.28. Description of ML2 Freescale SDN mechanism driver configuration options

Configuration option = Default value	Description
[ml2_fsldsn]	
crd_api_insecure = <i>False</i>	(BoolOpt) If set, ignore any SSL validation issues.
crd_auth_strategy = <i>keystone</i>	(StrOpt) Auth strategy for connecting to neutron in admin context.
crd_auth_url = <i>http://127.0.0.1:5000/v2.0/</i>	(StrOpt) CRD Auth URL.
crd_ca_certificates_file = <i>None</i>	(StrOpt) Location of ca certificates file to use for CRD client requests.
crd_password = <i>password</i>	(StrOpt) CRD Service Password.
crd_region_name = <i>RegionOne</i>	(StrOpt) Region name for connecting to CRD Service in admin context.
crd_tenant_name = <i>service</i>	(StrOpt) CRD Tenant Name.
crd_url = <i>http://127.0.0.1:9797</i>	(StrOpt) URL for connecting to CRD service.
crd_url_timeout = <i>30</i>	(IntOpt) Timeout value for connecting to CRD service in seconds.
crd_user_name = <i>crd</i>	(StrOpt) CRD service Username.

9.1.15.12. Modular Layer 2 (ml2) Geneve Mechanism configuration options

Table 9.29. Description of ML2 Geneve type driver configuration options

Configuration option = Default value	Description
[ml2_type_geneve]	

Configuration option = Default value	Description
max_header_size = 50	(Integer) Geneve encapsulation header size is dynamic, this value is used to calculate the maximum MTU for the driver. This is the sum of the sizes of the outer ETH + IP + UDP + GENEVE header sizes. The default size for this field is 50, which is the size of the Geneve header without any additional option headers.
vni_ranges =	(List) Comma-separated list of <vni_min>: <vni_max> tuples enumerating ranges of Geneve VNI IDs that are available for tenant network allocation

9.1.15.13. Modular Layer 2 (ml2) OpenDaylight Mechanism configuration options

Use of VLANs with the OpenDaylight mechanism driver requires OpenDaylight Helium or newer to be installed.

Table 9.30. Description of ML2 OpenDaylight mechanism driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
backdoor_port = None	(StrOpt) Enable eventlet backdoor. Acceptable values are 0, <port>, and <start>:<end>, where 0 results in listening on a random tcp port number; <port> results in listening on the specified port number (and not enabling backdoor if that port is in use); and <start>:<end> results in listening on the smallest unused port number within the specified range of port numbers. The chosen port is displayed in the service's log file.
policy_default_rule = default	(StrOpt) Default rule. Enforced when a requested rule is not found.
policy_dirs = ['policy.d']	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the config_dir option, or absolute paths. The file defined by policy_file must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = policy.json	(StrOpt) The JSON file that defines policies.
run_external_periodic_tasks = True	(BoolOpt) Some periodic tasks can be run in a separate process. Should we run them here?

Configuration option = Default value	Description
[ml2_odl]	
password = <i>None</i>	(StrOpt) HTTP password for authentication
session_timeout = 30	(IntOpt) Tomcat session timeout in minutes.
timeout = 10	(IntOpt) HTTP timeout in seconds.
url = <i>None</i>	(StrOpt) HTTP URL of OpenDaylight REST interface.
username = <i>None</i>	(StrOpt) HTTP username for authentication

9.1.15.14. Modular Layer 2 (ml2) OpenFlow Agent (ofagent) Mechanism configuration options

Table 9.31. Description of ML2 ofagent mechanism driver configuration options

Configuration option = Default value	Description
[AGENT]	
dont_fragment = <i>True</i>	(Boolean) Set or un-set the don't fragment (DF) bit on outgoing IP packet carrying GRE/VXLAN tunnel.

9.1.15.15. Modular Layer 2 (ml2) L2 Population Mechanism configuration options

Table 9.32. Description of ML2 L2 population configuration options

Configuration option = Default value	Description
[l2pop]	
agent_boot_time = 180	(Integer) Delay within which agent is expected to update existing ports when it restarts

9.1.15.16. Modular Layer 2 (ml2) Tail-f NCS Mechanism configuration options

Table 9.33. Description of ML2 NCS mechanism driver configuration options

Configuration option = Default value	Description
[ml2_ncs]	
password = <i>None</i>	(StrOpt) HTTP password for authentication
timeout = 10	(IntOpt) HTTP timeout in seconds.

Configuration option = Default value	Description
url = <i>None</i>	(StrOpt) HTTP URL of Tail-f NCS REST interface.
username = <i>None</i>	(StrOpt) HTTP username for authentication

9.1.1.15.17. Modular Layer 2 (ml2) SR-IOV Mechanism configuration options

Table 9.34. Description of ML2 ML2 SR-IOV driver configuration options

Configuration option = Default value	Description
[ml2_sriov]	
supported_pci_vendor_devs = <i>15b3:1004, 8086:10ca</i>	(List) Comma-separated list of supported PCI vendor devices, as defined by vendor_id:product_id according to the PCI ID Repository. Default enables support for Intel and Mellanox SR-IOV capable NICs.

9.1.1.16. MidoNet configuration options

Table 9.35. Description of Midonet configuration options

Configuration option = Default value	Description
[MIDONET]	
client = <i>midonet.neutron.client.api.MidonetApiClient</i>	(StrOpt) MidoNet client used to access MidoNet data storage.
cluster_ip = <i>localhost</i>	(StrOpt) IP that the cluster service can be reached on
cluster_port = <i>8088</i>	(StrOpt) Port that the cluster service can be reached on
midonet_uri = <i>http://localhost:8080/midonet-api</i>	(StrOpt) MidoNet API server URI.
password = <i>passw0rd</i>	(StrOpt) MidoNet admin password.
project_id = <i>77777777-7777-7777-7777-777777777777</i>	(StrOpt) ID of the project that MidoNet admin user belongs to.
tunnel_protocol = <i>vxlan</i>	(StrOpt) Tunnel protocol used by Midonet
username = <i>admin</i>	(StrOpt) MidoNet admin username.

9.1.1.17. NEC configuration options

Table 9.36. Description of Nec configuration options

Configuration option = Default value	Description
[OFC]	
api_max_attempts = 3	(IntOpt) Maximum attempts per OFC API request. NEC plugin retries API request to OFC when OFC returns ServiceUnavailable (503). The value must be greater than 0.
cert_file = <i>None</i>	(StrOpt) Location of certificate file.
driver = <i>trema</i>	(StrOpt) Driver to use.
enable_packet_filter = <i>True</i>	(BoolOpt) Enable packet filter.
host = <i>127.0.0.1</i>	(StrOpt) Host to connect to.
insecure_ssl = <i>False</i>	(BoolOpt) Disable SSL certificate verification.
key_file = <i>None</i>	(StrOpt) Location of key file.
path_prefix =	(StrOpt) Base URL of OFC REST API. It is prepended to each API request.
port = <i>8888</i>	(StrOpt) Port to connect to.
support_packet_filter_on_ofc_router = <i>True</i>	(BoolOpt) Support packet filter on OFC router interface.
use_ssl = <i>False</i>	(BoolOpt) Use SSL to connect.
[PROVIDER]	
default_router_provider = <i>l3-agent</i>	(StrOpt) Default router provider to use.
router_providers = <i>l3-agent, openflow</i>	(ListOpt) List of enabled router providers.
[fwaas]	
driver =	(StrOpt) Name of the FWaaS Driver

9.1.1.18. One Convergence NVSD configuration options

Table 9.37. Description of NVSD driver configuration options

Configuration option = Default value	Description
[AGENT]	
integration_bridge = <i>br-int</i>	(StrOpt) Integration bridge
[nvsd]	
nvsd_ip = <i>127.0.0.1</i>	(StrOpt) NVSD Controller IP address
nvsd_passwd = <i>oc123</i>	(StrOpt) NVSD Controller password
nvsd_port = <i>8082</i>	(IntOpt) NVSD Controller Port number
nvsd_retries = <i>0</i>	(IntOpt) Number of login retries to NVSD controller
nvsd_user = <i>ocplugin</i>	(StrOpt) NVSD Controller username
request_timeout = <i>30</i>	(IntOpt) NVSD controller REST API request timeout in seconds

9.1.1.19. Open Networking Operating System (ONOS) configuration options

Table 9.38. Description of Open Networking Operating System (ONOS) configuration options

Configuration option = Default value	Description
[onos]	
password =	(StrOpt) Password for authentication.
url_path =	(StrOpt) ONOS ReST interface URL
username =	(StrOpt) Username for authentication.

9.1.1.20. OpenContrail configuration options

Table 9.39. Description of OpenContrail configuration options

Configuration option = Default value	Description
[CONTRAIL]	
api_server_ip = <i>127.0.0.1</i>	(StrOpt) IP address to connect to the OpenContrail controller.

Configuration option = Default value	Description
api_server_port = 8082	(IntOpt) Port to connect to the OpenContrail controller.

9.1.1.21. Open vSwitch Agent configuration options

Table 9.40. Description of Open vSwitch agent configuration options

Configuration option = Default value	Description
[DEFAULT]	
ovs_integration_bridge = <i>br-int</i>	(String) Name of Open vSwitch bridge to use
ovs_use_veth = <i>False</i>	(Boolean) Uses veth for an OVS interface or not. Support kernels with limited namespace support (e.g. RHEL 6.5) so long as ovs_use_veth is set to True.
ovs_vsctl_timeout = 10	(Integer) Timeout in seconds for ovs-vsctl commands. If the timeout expires, ovs commands will fail with ALARMCLOCK error.
[AGENT]	
arp_responder = <i>False</i>	(Boolean) Enable local ARP responder if it is supported. Requires OVS 2.1 and ML2 I2population driver. Allows the switch (when supporting an overlay) to respond to an ARP request locally without performing a costly ARP broadcast into the overlay.
dont_fragment = <i>True</i>	(Boolean) Set or un-set the don't fragment (DF) bit on outgoing IP packet carrying GRE/VXLAN tunnel.
drop_flows_on_start = <i>False</i>	(Boolean) Reset flow table on start. Setting this to True will cause brief traffic interruption.
enable_distributed_routing = <i>False</i>	(Boolean) Make the I2 agent run in DVR mode.
l2_population = <i>False</i>	(Boolean) Use ML2 I2population mechanism driver to learn remote MAC and IPs and improve tunnel scalability.
minimize_polling = <i>True</i>	(Boolean) Minimize polling by monitoring ovsdb for interface changes.

Configuration option = Default value	Description
ovsdb_monitor_respawn_interval = 30	(Integer) The number of seconds to wait before respawning the ovsdb monitor after losing communication with it.
prevent_arp_spoofing = <i>True</i>	(Boolean) DEPRECATED: Enable suppression of ARP responses that don't match an IP address that belongs to the port from which they originate. Note: This prevents the VMs attached to this agent from spoofing, it doesn't protect them from other devices which have the capability to spoof (e.g. bare metal or VMs attached to agents without this flag set to True). Spoofing rules will not be added to any ports that have port security disabled. For LinuxBridge, this requires ebttables. For OVS, it requires a version that supports matching ARP headers. This option will be removed in Newton so the only way to disable protection will be via the port security extension.
quitting_rpc_timeout = 10	(Integer) Set new timeout in seconds for new rpc calls after agent receives SIGTERM. If value is set to 0, rpc timeout won't be changed
tunnel_csum = <i>False</i>	(Boolean) Set or un-set the tunnel header checksum on outgoing IP packet carrying GRE/VXLAN tunnel.
tunnel_types =	(List) Network types supported by the agent (gre and/or vxlan).
veth_mtu = 9000	(Integer) MTU size of veth interfaces
vxlan_udp_port = 4789	(Port number) The UDP port to use for VXLAN tunnels.
[OVS]	
bridge_mappings =	(List) Comma-separated list of <physical_network>: <bridge> tuples mapping physical network names to the agent's node-specific Open vSwitch bridge names to be used for flat and VLAN networks. The length of bridge names should be no more than 11. Each bridge must exist, and should have a physical network interface configured as a port. All physical networks configured on the server should have mappings to appropriate bridges on each agent. Note: If you remove a bridge from this mapping, make sure to disconnect it from the integration bridge as it won't be managed by the agent anymore. Deprecated for ofagent.

Configuration option = Default value	Description
datapath_type = <i>system</i>	(String) OVS datapath to use. 'system' is the default value and corresponds to the kernel datapath. To enable the userspace datapath set this value to 'netdev'.
int_peer_patch_port = <i>patch-tun</i>	(String) Peer patch port in integration bridge for tunnel bridge.
integration_bridge = <i>br-int</i>	(String) Integration bridge to use. Do not change this parameter unless you have a good reason to. This is the name of the OVS integration bridge. There is one per hypervisor. The integration bridge acts as a virtual 'patch bay'. All VM VIFs are attached to this bridge and then 'patched' according to their network connectivity.
local_ip = <i>None</i>	(Unknown) Local IP address of tunnel endpoint.
of_connect_timeout = 30	(Integer) Timeout in seconds to wait for the local switch connecting the controller. Used only for 'native' driver.
of_interface = <i>ovs-ofctl</i>	(String) OpenFlow interface to use.
of_listen_address = <i>127.0.0.1</i>	(Unknown) Address to listen on for OpenFlow connections. Used only for 'native' driver.
of_listen_port = 6633	(Port number) Port to listen on for OpenFlow connections. Used only for 'native' driver.
of_request_timeout = 10	(Integer) Timeout in seconds to wait for a single OpenFlow request. Used only for 'native' driver.
ovsdb_connection = <i>tcp:127.0.0.1:6640</i>	(String) The connection string for the native OVSDb backend. Requires the native ovsdb_interface to be enabled.
ovsdb_interface = <i>vsctl</i>	(String) The interface for interacting with the OVSDb
tun_peer_patch_port = <i>patch-int</i>	(String) Peer patch port in tunnel bridge for integration bridge.
tunnel_bridge = <i>br-tun</i>	(String) Tunnel bridge to use.

Configuration option = Default value	Description
use_veth_interconnection = <i>False</i>	(Boolean) Use veths instead of patch ports to interconnect the integration bridge to physical networks. Support kernel without Open vSwitch patch port support so long as it is set to True.
vhostuser_socket_dir = <i>/var/run/openvswitch</i>	(String) OVS vhost-user socket directory.

9.1.1.22. Virtual Network for Open vSwitch options

Table 9.41. Description of Virtual Network for Open vSwitch configuration options

Configuration option = Default value	Description
[ovn]	
neutron_sync_mode = <i>log</i>	(StrOpt) The synchronization mode of OVN with Neutron DB. Available options are: 'off' - synchronization is off. 'log' - during neutron-server startup, check to see if OVN is in sync with the neutron database. Log warnings for any inconsistencies found so that an admin can investigate. 'repair' - during neutron-server startup, automatically create resources found in Neutron but not in OVN. Also remove resources from OVN that are no longer in neutron.
ovsdb_connection = <i>tcp:127.0.0.1:6640</i>	(StrOpt) The connection string for the native OVSDB backend.
ovsdb_connection_timeout = <i>60</i>	(IntOpt) Timeout in seconds for the OVSDB connection transaction.

9.1.1.23. IPv6 Prefix Delegation configuration options

Table 9.42. Description of IPv6 Prefix Delegation driver configuration options

Configuration option = Default value	Description
[DEFAULT]	
pd_confs = <i>\$state_path/pd</i>	(String) Location to store IPv6 PD files.
pd_dhcp_driver = <i>dibbler</i>	(String) Service to handle DHCPv6 Prefix delegation.

Configuration option = Default value	Description
vendor_pen = 8888	(String) A decimal value as Vendor's Registered Private Enterprise Number as required by RFC3315 DUID-EN.

9.1.1.24. PLUMgrid configuration options

Table 9.43. Description of PLUMgrid configuration options

Configuration option = Default value	Description
[plumgriddirector]	
director_server = <i>localhost</i>	(StrOpt) PLUMgrid Director server to connect to
director_server_port = 8080	(IntOpt) PLUMgrid Director server port to connect to
distributed_locking = <i>True</i>	(BoolOpt) Distributed locking is enabled or disabled
driver = <i>networking_plumgrid.neutron.plugins.drivers.plumlib.Plumlib</i>	(StrOpt) PLUMgrid Driver
password = <i>password</i>	(StrOpt) PLUMgrid Director admin password
servertimeout = 5	(IntOpt) PLUMgrid Director server timeout
username = <i>username</i>	(StrOpt) PLUMgrid Director admin username

9.1.1.25. SR-IOV configuration options

Table 9.44. Description of SR-IOV configuration options

Configuration option = Default value	Description
[SRIOV_NIC]	
exclude_devices =	(ListOpt) List of <network_device>: <excluded_devices> mapping network_device to the agent's node-specific list of virtual functions that should not be used for virtual networking. excluded_devices is a semicolon separated list of virtual functions (BDF format).to exclude from network_device. The network_device in the mapping should appear in the physical_device_mappings list.

Configuration option = Default value	Description
physical_device_mappings =	(ListOpt) List of <physical_network>: <network_device> mapping physical network names to the agent's node-specific physical network device of SR-IOV physical function to be used for VLAN networks. All physical networks listed in network_vlan_ranges on the server should have mappings to appropriate interfaces on each agent

9.1.1.26. VMware vSphere configuration options

Table 9.45. Description of VMware configuration options

Configuration option = Default value	Description
[DEFAULT]	
default_interface_name = <i>FortyGigE1/0/1</i>	(StrOpt) default_interface_name of the I2 gateway
[OVSVAPP]	
agent_driver = <i>networking_vsphere.agent.ovsvapp_agent.OVSvAppL2Agent</i>	(StrOpt) OVSvApp Agent implementation.
bridge_mappings =	(ListOpt) Bridge mappings.
dont_fragment = <i>True</i>	(IntOpt) Do not fragment.
enable_ovsvapp_monitor = <i>True</i>	(BoolOpt) To monitor the OVSvApp Agents.
integration_bridge = <i>br-int</i>	(StrOpt) Integration Bridge.
local_ip =	(StrOpt) Local IP address of VXLAN tunnel endpoint.
monitoring_ip =	(StrOpt) IP address for monitoring OVS Status.
network_manager = <i>networking_vsphere.drivers.manager.VcenterManager</i>	(StrOpt) Driver Manager implementation for NetworkDriver.
polling_interval = <i>2</i>	(IntOpt) The number of seconds the agent will wait between polling for local device changes.
report_interval = <i>30</i>	(IntOpt) Seconds between nodes reporting state to server.
tenant_network_type = <i>vlan</i>	(StrOpt) Network type for tenant networks

Configuration option = Default value	Description
tunnel_bridge = <i>br-tun</i>	(StrOpt) Tunnel Bridge for tunneling.
tunnel_csum = <i>False</i>	(BoolOpt) Set or un-set the tunnel header checksum on outgoing IP packet carrying GRE/VXLAN tunnel.
tunnel_types = <i>vxlan</i>	(ListOpt) Tunnel network types supported by the OVSvApp Agent.
veth_mtu = <i>1500</i>	(IntOpt) MTU size of veth interfaces.
vxlan_udp_port = <i>4789</i>	(IntOpt) The UDP port to use for VXLAN tunnels.
[VMWARE]	
cert_check = <i>False</i>	(BoolOpt) Enable SSL certificate check for vCenter.
cert_path = <i>None</i>	(StrOpt) Certificate chain path containing cacert of vCenters.
cluster_dvs_mapping = <i>[""]</i>	(MultiStrOpt) vCenter cluster to DVS mapping.
esx_hostname = <i>None</i>	(StrOpt) ESX host name where this OVSvApp is hosted.
esx_maintenance_mode = <i>True</i>	(BoolOpt) Set host into maintenance mode.
https_port = <i>443</i>	(IntOpt) Customized https_port for vCenter communication.
vcenter_api_retry_count = <i>5</i>	(StrOpt) Number of retries while connecting to vcenter server.
vcenter_id = <i>None</i>	(StrOpt) Unique ID of the vCenter Server on which this OVSvApp is hosted
vcenter_ip = <i>None</i>	(StrOpt) vCenter server IP.
vcenter_password = <i>None</i>	(StrOpt) vCenter server password.
vcenter_username = <i>None</i>	(StrOpt) vCenter server user name.
wsdl_location = <i>None</i>	(StrOpt) vCenter server wsdl location.
[vmware]	

Configuration option = Default value	Description
console_delay_seconds = <i>None</i>	(IntOpt) Set this value if affected by an increased network latency causing repeated characters when typing in a remote console.
maximum_objects = <i>100</i>	(IntOpt) The maximum number of ObjectContent data objects that should be returned in a single result. A positive value will cause the operation to suspend the retrieval when the count of objects reaches the specified maximum. The server may still limit the count to something less than the configured value. Any remaining objects may be retrieved with additional requests.
serial_port_proxy_uri = <i>None</i>	(StrOpt) Identifies a proxy service that provides network access to the serial_port_service_uri. This option is ignored if serial_port_service_uri is not specified.
serial_port_service_uri = <i>None</i>	(StrOpt) Identifies the remote system that serial port traffic will be sent to. If this is not set, no serial ports will be added to the created VMs.

9.1.1.27. VMware NSX configuration options

Table 9.46. Description of VMware NSX configuration options

Configuration option = Default value	Description
[DEFAULT]	
conn_idle_timeout = <i>900</i>	(IntOpt) Reconnect connection to nsx if not used within this amount of time.
default_service_cluster_uuid = <i>None</i>	(StrOpt) Unique identifier of the Service Cluster which will be used by logical services like dhcp and metadata
default_tz_uuid = <i>None</i>	(StrOpt) This is uuid of the default NSX Transport zone that will be used for creating tunneled isolated "Neutron" networks. It needs to be created in NSX before starting Neutron with the nsx plugin.
http_timeout = <i>75</i>	(IntOpt) Time before aborting a request
nsx_controllers = <i>None</i>	(ListOpt) Lists the NSX controllers in this cluster

Configuration option = Default value	Description
nsx_default_interface_name = <i>breth0</i>	(StrOpt) Name of the interface on a L2 Gateway transport nodewhich should be used by default when setting up a network connection
nsx_l2gw_driver = <i>None</i>	(StrOpt) Class path for the L2 gateway backend driver
nsx_password = <i>admin</i>	(StrOpt) Password for NSX controllers in this cluster
nsx_user = <i>admin</i>	(StrOpt) User name for NSX controllers in this cluster
redirects = <i>2</i>	(IntOpt) Number of times a redirect should be followed
retries = <i>2</i>	(IntOpt) Number of time a request should be retried
[NSX]	
agent_mode = <i>agent</i>	(StrOpt) The mode used to implement DHCP/metadata services.
concurrent_connections = <i>10</i>	(IntOpt) Maximum concurrent connections to each NSX controller.
default_transport_type = <i>stt</i>	(StrOpt) The default network tranport type to use (stt, gre, bridge, ipsec_gre, or ipsec_stt)
max_lp_per_bridged_ls = <i>5000</i>	(IntOpt) Maximum number of ports of a logical switch on a bridged transport zone (default 5000)
max_lp_per_overlay_ls = <i>256</i>	(IntOpt) Maximum number of ports of a logical switch on an overlay transport zone (default 256)
metadata_mode = <i>access_network</i>	(StrOpt) If set to <i>access_network</i> this enables a dedicated connection to the metadata proxy for metadata server access via Neutron router. If set to <i>dhcp_host_route</i> this enables host route injection via the dhcp agent. This option is only useful if running on a host that does not support namespaces otherwise <i>access_network</i> should be used.
nsx_gen_timeout = <i>-1</i>	(IntOpt) Number of seconds a generation id should be valid for (default -1 meaning do not time out)

Configuration option = Default value	Description
replication_mode = <i>service</i>	(StrOpt) The default option leverages service nodes to perform packet replication though one could set to this to 'source' to perform replication locally. This is useful if one does not want to deploy a service node(s). It must be set to 'service' for leveraging distributed routers.
[NSX_DHCP]	
default_lease_time = <i>43200</i>	(IntOpt) Default DHCP lease time
domain_name = <i>openstacklocal</i>	(StrOpt) Domain to use for building the hostnames
extra_domain_name_servers =	(ListOpt) Comma separated list of additional domain name servers
[NSX_LSN]	
sync_on_missing_data = <i>False</i>	(BoolOpt) Pull LSN information from NSX in case it is missing from the local data store. This is useful to rebuild the local store in case of server recovery.
[NSX_METADATA]	
metadata_server_address = <i>127.0.0.1</i>	(StrOpt) IP address used by Metadata server.
metadata_server_port = <i>8775</i>	(IntOpt) TCP Port used by Metadata server.
metadata_shared_secret =	(StrOpt) Shared secret to sign instance-id request
[NSX_SYNC]	
always_read_status = <i>False</i>	(BoolOpt) Always read operational status from backend on show operations. Enabling this option might slow down the system.
max_random_sync_delay = <i>0</i>	(IntOpt) Maximum value for the additional random delay in seconds between runs of the state synchronization task
min_chunk_size = <i>500</i>	(IntOpt) Minimum number of resources to be retrieved from NSX during state synchronization
min_sync_req_delay = <i>1</i>	(IntOpt) Minimum delay, in seconds, between two state synchronization queries to NSX. It must not exceed <code>state_sync_interval</code>

Configuration option = Default value	Description
state_sync_interval = 10	(IntOpt) Interval in seconds between runs of the state synchronization task. Set it to 0 to disable it
[nsx_v3]	
ca_file = None	(StrOpt) Specify a CA bundle file to use in verifying the NSX Manager server certificate.
default_bridge_cluster_uuid = None	(StrOpt) Default bridge cluster identifier for L2 gateway. This needs to be created in NSX before using the L2 gateway service plugin.
default_edge_cluster_uuid = None	(StrOpt) Default edge cluster identifier
default_overlay_tz_uuid = None	(StrOpt) This is the UUID of the default NSX overlay transport zone that will be used for creating tunneled isolated Neutron networks. It needs to be created in NSX before starting Neutron with the NSX plugin.
default_tier0_router_uuid = None	(StrOpt) Default tier0 router identifier
default_vlan_tz_uuid = None	(StrOpt) This is the UUID of the default NSX VLAN transport zone that will be used for bridging between Neutron networks. It needs to be created in NSX before starting Neutron with the NSX plugin.
insecure = True	(BoolOpt) If true, the NSX Manager server certificate is not verified. If false, then the default CA truststore is used for verification. This option is ignored if "ca_file" is set.
nsx_manager = None	(StrOpt) IP address of the NSX manager
nsx_password = default	(StrOpt) Password for the NSX manager
nsx_user = admin	(StrOpt) User name for the NSX manager
retries = 10	(IntOpt) Maximum number of times to retry API request
[nsxv]	
backup_edge_pool = service:large:4:10, service:compact:4:10, vdr:large:4:10	(ListOpt) Defines edge pool using the format: <edge_type>:[edge_size]:<min_edges>:<max_edges>. edge_type: service,vdr. edge_size: compact, large, xlarge, quadlarge and default is large.

Configuration option = Default value	Description
ca_file = <i>None</i>	(StrOpt) Specify a CA bundle file to use in verifying the NSXv server certificate.
cluster_moid =	(ListOpt) Parameter listing the IDs of the clusters which are used by OpenStack.
datacenter_moid = <i>None</i>	(StrOpt) Optional parameter identifying the ID of datacenter to deploy NSX Edges
datastore_id = <i>None</i>	(StrOpt) Optional parameter identifying the ID of datastore to deploy NSX Edges
deployment_container_id = <i>None</i>	(StrOpt) Optional parameter identifying the ID of datastore to deploy NSX Edges
dhcp_lease_time = <i>86400</i>	(IntOpt) DHCP default lease time.
dvs_id = <i>None</i>	(StrOpt) DVS ID for VLANs
edge_appliance_password = <i>None</i>	(StrOpt) Password to configure for Edge appliance login
edge_appliance_user = <i>None</i>	(StrOpt) Username to configure for Edge appliance login
edge_ha = <i>False</i>	(BoolOpt) Enable HA for NSX Edges
exclusive_router_appliance_size = <i>compact</i>	(StrOpt) Edge appliance size to be used for creating exclusive router. Valid values: ['compact', 'large', 'xlarge', 'quadlarge']. This edge_appliance_size will be picked up if --router-size parameter is not specified while doing neutron router-create
external_network = <i>None</i>	(StrOpt) Network ID for physical network connectivity
insecure = <i>True</i>	(BoolOpt) If true, the NSXv server certificate is not verified. If false, then the default CA truststore is used for verification. This option is ignored if "ca_file" is set.
locking_coordinator_url = <i>None</i>	(StrOpt) A URL to a locking mechanism coordinator

Configuration option = Default value	Description
manager_uri = <i>None</i>	(StrOpt) uri for vsm
maximum_tunnels_per_vnic = 20	(IntOpt) Maximum number of sub interfaces supported per vnic in edge.
metadata_initializer = <i>True</i>	(BoolOpt) If True, the server instance will attempt to initialize the metadata infrastructure
metadata_shared_secret = <i>None</i>	(StrOpt) Shared secret to sign metadata requests
mgt_net_default_gateway = <i>None</i>	(StrOpt) Management network default gateway for metadata proxy
mgt_net_moid = <i>None</i>	(StrOpt) Network ID for management network connectivity
mgt_net_proxy_ips = <i>None</i>	(ListOpt) Management network IP address for metadata proxy
mgt_net_proxy_netmask = <i>None</i>	(StrOpt) Management network netmask for metadata proxy
nova_metadata_ips = <i>None</i>	(ListOpt) IP addresses used by Nova metadata service
nova_metadata_port = 8775	(IntOpt) TCP Port used by Nova metadata server
password = <i>default</i>	(StrOpt) Password for vsm
resource_pool_id = <i>None</i>	(StrOpt) Optional parameter identifying the ID of resource to deploy NSX Edges
retries = 10	(IntOpt) Maximum number of API retries on endpoint.
spoofguard_enabled = <i>True</i>	(BoolOpt) If True then plugin will use NSXV spoofguard component for port-security feature.
task_status_check_interval = 2000	(IntOpt) Task status check interval
tenant_router_types = <i>shared, distributed, exclusive</i>	(ListOpt) Ordered list of router_types to allocate as tenant routers.
user = <i>admin</i>	(StrOpt) User name for vsm

Configuration option = Default value	Description
vdn_scope_id = <i>None</i>	(StrOpt) Network scope ID for VXLAN virtual wires

9.1.1.28. VMware DVS configuration options

Table 9.47. Description of VMware DVS configuration options

Configuration option = Default value	Description
[dvs]	
api_retry_count = 10	(IntOpt) The number of times we retry on failures, e.g., socket error, etc.
ca_file = <i>None</i>	(StrOpt) Specify a CA bundle file to use in verifying the vCenter server certificate.
dvs_name = <i>None</i>	(StrOpt) The name of the preconfigured DVS.
host_ip = <i>None</i>	(StrOpt) Hostname or IP address for connection to VMware vCenter host.
host_password = <i>None</i>	(StrOpt) Password for connection to VMware vCenter host.
host_port = 443	(IntOpt) Port for connection to VMware vCenter host.
host_username = <i>None</i>	(StrOpt) Username for connection to VMware vCenter host.
insecure = <i>False</i>	(BoolOpt) If true, the vCenter server certificate is not verified. If false, then the default CA truststore is used for verification. This option is ignored if "ca_file" is set.
task_poll_interval = 0.5	(FloatOpt) The interval used for polling of remote tasks.

9.1.2. Configure the Oslo RPC messaging system

OpenStack projects use an open standard for messaging middleware known as AMQP. This messaging middleware enables the OpenStack services that run on multiple servers to talk to each other. OpenStack Oslo RPC supports two implementations of AMQP: **RabbitMQ** and **Qpid**.

9.1.2.1. Configure RabbitMQ

OpenStack Oslo RPC uses **RabbitMQ** by default. Use these options to configure the **RabbitMQ**

message system. The `rpc_backend` option is optional as long as **RabbitMQ** is the default messaging system. However, if it is included the configuration, you must set it to `neutron.openstack.common.rpc.impl_kombu`.

```
rpc_backend=neutron.openstack.common.rpc.impl_kombu
```

Use these options to configure the **RabbitMQ** messaging system. You can configure messaging communication for different installation scenarios, tune retries for RabbitMQ, and define the size of the RPC thread pool. To monitor notifications through RabbitMQ, you must set the `notification_driver` option to `neutron.openstack.common.notifier.rpc_notifier` in the `neutron.conf` file:

Table 9.48. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
<code>amqp_auto_delete = False</code>	(Boolean) Auto-delete queues in AMQP.
<code>amqp_durable_queues = False</code>	(Boolean) Use durable queues in AMQP.
<code>channel_max = None</code>	(Integer) Maximum number of channels to allow
<code>default_notification_exchange = \${control_exchange}_notification</code>	(String) Exchange name for for sending notifications
<code>default_notification_retry_attempts = -1</code>	(Integer) Reconnecting retry count in case of connectivity problem during sending notification, -1 means infinite retry.
<code>default_rpc_exchange = \${control_exchange}_rpc</code>	(String) Exchange name for sending RPC messages
<code>default_rpc_retry_attempts = -1</code>	(Integer) Reconnecting retry count in case of connectivity problem during sending RPC message, -1 means infinite retry. If actual retry attempts in not 0 the rpc request could be processed more then one time
<code>fake_rabbit = False</code>	(Boolean) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
<code>frame_max = None</code>	(Integer) The maximum byte size for an AMQP frame
<code>heartbeat_interval = 1</code>	(Integer) How often to send heartbeats for consumer's connections

Configuration option = Default value	Description
heartbeat_rate = 2	(Integer) How often times during the heartbeat_timeout_threshold we check the heartbeat.
heartbeat_timeout_threshold = 60	(Integer) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disable the heartbeat). EXPERIMENTAL
host_connection_reconnect_delay = 0.25	(Floating point) Set delay for reconnection to some host which has connection error
kombu_compression = None	(String) EXPERIMENTAL: Possible values are: gzip, bz2. If not set compression will not be used. This option may not be available in future versions.
kombu_failover_strategy = round-robin	(String) Determines how the next RabbitMQ node is chosen in case the one we are currently connected to becomes unavailable. Takes effect only if more than one RabbitMQ node is provided in config.
kombu_missing_consumer_retry_timeout = 60	(Integer) How long to wait a missing client before abandoning to send it its replies. This value should not be longer than rpc_response_timeout.
kombu_reconnect_delay = 1.0	(Floating point) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_ssl_ca_certs =	(String) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(String) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(String) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(String) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
notification_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to notification listener.
notification_persistence = False	(Boolean) Persist notification messages.
notification_retry_delay = 0.25	(Floating point) Reconnecting retry delay in case of connectivity problem during sending notification message

Configuration option = Default value	Description
pool_max_overflow = 0	(Integer) Maximum number of connections to create above <code>pool_max_size</code> .
pool_max_size = 10	(Integer) Maximum number of connections to keep queued.
pool_recycle = 600	(Integer) Lifetime of a connection (since creation) in seconds or None for no recycling. Expired connections are closed on acquire.
pool_stale = 60	(Integer) Threshold at which inactive (since release) connections are considered stale in seconds or None for no staleness. Stale connections are closed on acquire.
pool_timeout = 30	(Integer) Default number of seconds to wait for a connections to available
rabbit_ha_queues = <i>False</i>	(Boolean) Try to use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database. In RabbitMQ 3.0, queue mirroring is no longer controlled by the x-ha-policy argument when declaring a queue. If you just want to make sure that all queues (except those with auto-generated names) are mirrored across all nodes, run: <code>"rabbitmqctl set_policy HA '^(?!amq\.).*' '{"ha-mode": "all"}' "</code>
rabbit_host = <i>localhost</i>	(String) The RabbitMQ broker address where a single node is used.
rabbit_hosts = <i>\$rabbit_host:\$rabbit_port</i>	(List) RabbitMQ HA cluster host:port pairs.
rabbit_interval_max = 30	(Integer) Maximum interval of RabbitMQ connection retries. Default is 30 seconds.
rabbit_login_method = <i>AMQPLAIN</i>	(String) The RabbitMQ login method.
rabbit_max_retries = 0	(Integer) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = <i>guest</i>	(String) The RabbitMQ password.
rabbit_port = 5672	(Port number) The RabbitMQ broker port where a single node is used.
rabbit_qos_prefetch_count = 0	(Integer) Specifies the number of messages to prefetch. Setting to zero allows unlimited messages.

Configuration option = Default value	Description
rabbit_retry_backoff = 2	(Integer) How long to backoff for between retries when connecting to RabbitMQ.
rabbit_retry_interval = 1	(Integer) How frequently to retry connecting with RabbitMQ.
rabbit_transient_queues_ttl = 1800	(Integer) Positive integer representing duration in seconds for queue TTL (x-expires). Queues which are unused for the duration of the TTL are automatically deleted. The parameter affects only reply and fanout queues.
rabbit_use_ssl = False	(Boolean) Connect over SSL for RabbitMQ.
rabbit_userid = guest	(String) The RabbitMQ userid.
rabbit_virtual_host = /	(String) The RabbitMQ virtual host.
rpc_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc listener.
rpc_queue_expiration = 60	(Integer) Time to live for rpc queues without consumers in seconds.
rpc_reply_exchange = <i>#{control_exchange}_rpc_reply</i>	(String) Exchange name for receiving RPC replies
rpc_reply_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc reply listener.
rpc_reply_retry_attempts = -1	(Integer) Reconnecting retry count in case of connectivity problem during sending reply. -1 means infinite retry during rpc_timeout
rpc_reply_retry_delay = 0.25	(Floating point) Reconnecting retry delay in case of connectivity problem during sending reply.
rpc_retry_delay = 0.25	(Floating point) Reconnecting retry delay in case of connectivity problem during sending RPC message
socket_timeout = 0.25	(Floating point) Set socket timeout in seconds for connection's socket
ssl = None	(Boolean) Enable SSL
ssl_options = None	(Dict) Arguments passed to ssl.wrap_socket

Configuration option = Default value	Description
<code>tcp_user_timeout = 0.25</code>	(Floating point) Set TCP_USER_TIMEOUT in seconds for connection's socket

9.1.2.2. Configure Qpid

Use these options to configure the **Qpid** messaging system for OpenStack Oslo RPC. **Qpid** is not the default messaging system, so you must enable it by setting the `rpc_backend` option in the `neutron.conf` file:

```
rpc_backend=neutron.openstack.common.rpc.impl_qpid
```

This critical option points the compute nodes to the **Qpid** broker (server). Set the `qpid_hostname` option to the host name where the broker runs in the `neutron.conf` file.



NOTE

The `--qpid_hostname` parameter accepts a host name or IP address value.

```
qpid_hostname=hostname.example.com
```

If the **Qpid** broker listens on a port other than the AMQP default of **5672**, you must set the `qpid_port` option to that value:

```
qpid_port=12345
```

If you configure the **Qpid** broker to require authentication, you must add a user name and password to the configuration:

```
qpid_username=username
qpid_password=password
```

By default, TCP is used as the transport. To enable SSL, set the `qpid_protocol` option:

```
qpid_protocol=ssl
```

Use these additional options to configure the Qpid messaging driver for OpenStack Oslo RPC. These options are used infrequently.

Table 9.49. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
<code>amqp_auto_delete = False</code>	(BoolOpt) Auto-delete queues in AMQP.
<code>amqp_durable_queues = False</code>	(BoolOpt) Use durable queues in AMQP.

Configuration option = Default value	Description
<code>qpid_heartbeat = 60</code>	(IntOpt) Seconds between connection keepalive heartbeats.
<code>qpid_hostname = localhost</code>	(StrOpt) Qpid broker hostname.
<code>qpid_hosts = \$qpid_hostname:\$qpid_port</code>	(ListOpt) Qpid HA cluster host:port pairs.
<code>qpid_password =</code>	(StrOpt) Password for Qpid connection.
<code>qpid_port = 5672</code>	(IntOpt) Qpid broker port.
<code>qpid_protocol = tcp</code>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
<code>qpid_receiver_capacity = 1</code>	(IntOpt) The number of prefetched messages held by receiver.
<code>qpid_sasl_mechanisms =</code>	(StrOpt) Space separated list of SASL mechanisms to use for auth.
<code>qpid_tcp_nodelay = True</code>	(BoolOpt) Whether to disable the Nagle algorithm.
<code>qpid_topology_version = 1</code>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
<code>qpid_username =</code>	(StrOpt) Username for Qpid connection.
<code>send_single_reply = False</code>	(BoolOpt) Send a single AMQP reply to call message. The current behavior since oslo-incubator is to send two AMQP replies - first one with the payload, a second one to ensure the other has finished to send the payload. We are going to remove it in the N release, but we must keep backward compatible at the same time. This option provides such compatibility - it defaults to False in Liberty and can be turned on for early adopters with new installations or for testing. <i>This option will be removed in the Mitaka release.</i>

9.1.2.3. Configure messaging

Use these common options to configure the **RabbitMQ**, and **Qpid** messaging drivers:

Table 9.50. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
rpc_backend = <i>rabbit</i>	(String) The messaging driver to use, defaults to rabbit. Other drivers include amqp and zmq.
rpc_cast_timeout = <i>-1</i>	(Integer) Seconds to wait before a cast expires (TTL). The default value of -1 specifies an infinite linger period. The value of 0 specifies no linger period. Pending messages shall be discarded immediately when the socket is closed. Only supported by impl_zmq.
rpc_conn_pool_size = <i>30</i>	(Integer) Size of RPC connection pool.
rpc_poll_timeout = <i>1</i>	(Integer) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.
rpc_response_timeout = <i>60</i>	(Integer) Seconds to wait for a response from a call.
rpc_state_report_workers = <i>1</i>	(Integer) Number of RPC worker processes dedicated to state reports queue
rpc_workers = <i>1</i>	(Integer) Number of RPC worker processes for service
[oslo_concurrency]	
disable_process_locking = <i>False</i>	(Boolean) Enables or disables inter-process locks.
lock_path = <i>None</i>	(String) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging]	
event_stream_topic = <i>neutron_lbaas_event</i>	(String) topic name for receiving events from a queue
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(Boolean) Accept clients using either SSL or plain TCP

Configuration option = Default value	Description
broadcast_prefix = <i>broadcast</i>	(String) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(String) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(String) address prefix when sending to any server in group
idle_timeout = <i>0</i>	(Integer) Timeout for inactive connections (in seconds)
password =	(String) Password for message broker authentication
sasl_config_dir =	(String) Path to directory that contains the SASL configuration
sasl_config_name =	(String) Name of configuration file (without .conf suffix)
sasl_mechanisms =	(String) Space separated list of acceptable SASL mechanisms
server_request_prefix = <i>exclusive</i>	(String) address prefix used when sending to a specific server
ssl_ca_file =	(String) CA certificate PEM file to verify server certificate
ssl_cert_file =	(String) Identifying certificate PEM file to present to clients
ssl_key_file =	(String) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(String) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>	(Boolean) Debug: dump AMQP frames to stdout
username =	(String) User name for message broker authentication
[oslo_messaging_notifications]	

Configuration option = Default value	Description
driver = []	(Multi-valued) The Drivers(s) to handle sending notifications. Possible values are messaging, messagingv2, routing, log, test, noop
topics = <i>notifications</i>	(List) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(String) A URL representing the messaging driver to use for notifications. If not set, we fall back to the same configuration used for RPC.

Table 9.51. Description of Redis configuration options

Configuration option = Default value	Description
[matchmaker_redis]	
check_timeout = 20000	(Integer) Time in ms to wait before the transaction is killed.
host = 127.0.0.1	(String) Host to locate redis.
password =	(String) Password for Redis server (optional).
port = 6379	(Port number) Use this port to connect to redis host.
sentinel_group_name = <i>oslo-messaging-zeromq</i>	(String) Redis replica set name.
sentinel_hosts =	(List) List of Redis Sentinel hosts (fault tolerance mode) e.g. [host:port, host1:port ...]
socket_timeout = 1000	(Integer) Timeout in ms on blocking socket operations
wait_timeout = 500	(Integer) Time in ms to wait between connection attempts.

Table 9.52. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>neutron</i>	(String) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the transport_url option.

Configuration option = Default value	Description
transport_url = <i>None</i>	(String) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the <code>rpc_backend</code> option and driver specific configuration.

9.1.3. Agent

Use the following options to alter agent-related settings.

Table 9.53. Description of agent configuration options

Configuration option = Default value	Description
[DEFAULT]	
external_pids = <i>\$state_path/external/pids</i>	(String) Location to store child pid files
network_device_mtu = <i>None</i>	(Integer) DEPRECATED: MTU setting for device. This option will be removed in Newton. Please use the system-wide <code>segment_mtu</code> setting which the agents will take into account when wiring VIFs.
prefix_delegation_driver = <i>dibbler</i>	(String) Driver used for ipv6 prefix delegation. This needs to be an entry point defined in the <code>neutron.agent.linux.pd_drivers</code> namespace. See <code>setup.cfg</code> for entry points included with the neutron source.
[AGENT]	
agent_type = <i>Open vSwitch agent</i>	(String) DEPRECATED: Selects the Agent Type reported
availability_zone = <i>nova</i>	(String) Availability zone of this node

9.1.4. API

Use the following options to alter API-related settings.

Table 9.54. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
allow_bulk = <i>True</i>	(Boolean) Allow the usage of the bulk API

Configuration option = Default value	Description
allow_pagination = <i>False</i>	(Boolean) Allow the usage of the pagination
allow_sorting = <i>False</i>	(Boolean) Allow the usage of the sorting
api_extensions_path =	(String) The path for API extensions. Note that this can be a colon-separated list of paths. For example: <code>api_extensions_path = extensions:/path/to/more/exts:/even/more/exts</code> . The <code>__path__</code> of <code>neutron.extensions</code> is appended to this, so if your extensions are in there you don't need to specify them here.
api_paste_config = <i>api-paste.ini</i>	(String) File name for the paste.deploy config for api service
backlog = <i>4096</i>	(Integer) Number of backlog requests to configure the socket with
client_socket_timeout = <i>900</i>	(Integer) Timeout for client connections' socket operations. If an incoming connection is idle for this number of seconds it will be closed. A value of '0' means wait forever.
max_header_line = <i>16384</i>	(Integer) Maximum line size of message headers to be accepted. <code>max_header_line</code> may need to be increased when using large tokens (typically those generated when keystone is configured to use PKI tokens with big service catalogs).
pagination_max_limit = <i>-1</i>	(String) The maximum number of items returned in a single response, value was 'infinite' or negative integer means no limit
retry_until_window = <i>30</i>	(Integer) Number of seconds to keep retrying to listen
service_plugins =	(List) The service plugins Neutron will use
tcp_keepidle = <i>600</i>	(Integer) Sets the value of <code>TCP_KEEPIDLE</code> in seconds for each server socket. Not supported on OS X.
wsgi_default_pool_size = <i>100</i>	(Integer) Size of the pool of greenthreads used by wsgi
wsgi_keep_alive = <i>True</i>	(Boolean) If False, closes the client socket connection explicitly.

Configuration option = Default value	Description
[oslo_middleware]	
max_request_body_size = <i>114688</i>	(Integer) The maximum body size for each request, in bytes.
secure_proxy_ssl_header = <i>X-Forwarded-Proto</i>	(String) DEPRECATED: The HTTP Header that will be used to determine what the original request protocol scheme was, even if it was hidden by an SSL termination proxy.
[oslo_policy]	
policy_default_rule = <i>default</i>	(String) Default rule. Enforced when a requested rule is not found.
policy_dirs = <i>['policy.d']</i>	(Multi-valued) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the <code>config_dir</code> option, or absolute paths. The file defined by <code>policy_file</code> must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(String) The JSON file that defines policies.

9.1.5. Token authentication

Use the following options to alter token authentication settings.

Table 9.55. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(String) Service user password.
admin_tenant_name = <i>admin</i>	(String) Service tenant name.
admin_token = <i>None</i>	(String) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(String) Service username.

Configuration option = Default value	Description
auth_admin_prefix =	(String) Prefix to prepend at the beginning of the path. Deprecated, use identity_uri.
auth_host = 127.0.0.1	(String) Host providing the admin Identity API endpoint. Deprecated, use identity_uri.
auth_port = 35357	(Integer) Port of the admin Identity API endpoint. Deprecated, use identity_uri.
auth_protocol = https	(String) Protocol of the admin Identity API endpoint. Deprecated, use identity_uri.
auth_section = None	(Unknown) Config Section from which to load plugin specific options
auth_type = None	(Unknown) Authentication type to load
auth_uri = None	(String) Complete public Identity API endpoint.
auth_version = None	(String) API version of the admin Identity API endpoint.
cache = None	(String) Env key for the swift cache.
cafile = None	(String) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = None	(String) Required if identity server requires client certificate
check_revocations_for_cached = False	(Boolean) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.
delay_auth_decision = False	(Boolean) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.

Configuration option = Default value	Description
enforce_token_bind = <i>permissive</i>	(String) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(List) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard <code>hashlib.new()</code> . The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(Integer) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(Integer) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(String) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. <code>https://localhost:35357/</code>
include_service_catalog = <i>True</i>	(Boolean) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(Boolean) Verify HTTPS connections.
keyfile = <i>None</i>	(String) Required if identity server requires client certificate
memcache_pool_conn_get_timeout = <i>10</i>	(Integer) (Optional) Number of seconds that an operation will wait to get a memcached client connection from the pool.
memcache_pool_dead_retry = <i>300</i>	(Integer) (Optional) Number of seconds memcached server is considered dead before it is tried again.

Configuration option = Default value	Description
<code>memcache_pool_maxsize = 10</code>	(Integer) (Optional) Maximum total number of open connections to every memcached server.
<code>memcache_pool_socket_timeout = 3</code>	(Integer) (Optional) Socket timeout in seconds for communicating with a memcached server.
<code>memcache_pool_unused_timeout = 60</code>	(Integer) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
<code>memcache_secret_key = None</code>	(String) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
<code>memcache_security_strategy = None</code>	(String) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.
<code>memcache_use_advanced_pool = False</code>	(Boolean) (Optional) Use the advanced (eventlet safe) memcached client pool. The advanced pool will only work under python 2.x.
<code>region_name = None</code>	(String) The region in which the identity server can be found.
<code>revocation_cache_time = 10</code>	(Integer) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
<code>signing_dir = None</code>	(String) Directory used to cache files related to PKI tokens.
<code>token_cache_time = 300</code>	(Integer) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

9.1.6. Compute

Use the following options to alter Compute-related settings.

Table 9.56. Description of Compute configuration options

Configuration option = Default value	Description
[DEFAULT]	
notify_nova_on_port_data_changes = <i>True</i>	(Boolean) Send notification to nova when port data (fixed_ips/floatingip) changes so nova can update its cache.
notify_nova_on_port_status_changes = <i>True</i>	(Boolean) Send notification to nova when port status changes
nova_client_cert =	(String) Client certificate for nova metadata api server.
nova_client_priv_key =	(String) Private key of client certificate.
send_events_interval = 2	(Integer) Number of seconds between sending events to nova if there are any events to send.

9.1.7. CORS

Use the following options to alter CORS-related settings.

Table 9.57. Description of CORS configuration options

Configuration option = Default value	Description
[cors]	
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = 3600	(Integer) Maximum cache age of CORS preflight requests.
[cors.subdomain]	

Configuration option = Default value	Description
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(Integer) Maximum cache age of CORS preflight requests.

9.1.8. Database

Use the following options to alter Database-related settings.

Table 9.58. Description of database configuration options

Configuration option = Default value	Description
[database]	
backend = <i>sqlalchemy</i>	(String) The back end to use for the database.
connection = <i>None</i>	(String) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(Integer) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(Boolean) Add Python stack traces to SQL as comment strings.
db_inc_retry_interval = <i>True</i>	(Boolean) If True, increases the interval between retries of a database operation up to <code>db_max_retry_interval</code> .
db_max_retries = <i>20</i>	(Integer) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.

Configuration option = Default value	Description
db_max_retry_interval = 10	(Integer) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.
db_retry_interval = 1	(Integer) Seconds between retries of a database transaction.
idle_timeout = 3600	(Integer) Timeout before idle SQL connections are reaped.
max_overflow = 50	(Integer) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = None	(Integer) Maximum number of SQL connections to keep open in a pool.
max_retries = 10	(Integer) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = 1	(Integer) Minimum number of SQL connections to keep open in a pool.
mysql_sql_mode = TRADITIONAL	(String) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode=
pool_timeout = None	(Integer) If set, use this value for pool_timeout with SQLAlchemy.
retry_interval = 10	(Integer) Interval between retries of opening a SQL connection.
slave_connection = None	(String) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = oslo.sqlite	(String) The file name to use with SQLite.
sqlite_synchronous = True	(Boolean) If True, SQLite uses synchronous mode.
use_db_reconnect = False	(Boolean) Enable the experimental use of database reconnect on connection lost.

9.1.9. Designate

Use the following options to alter Designate-related settings.

Table 9.59. Description of designate configuration options

Configuration option = Default value	Description
[designate]	
admin_auth_url = <i>None</i>	(String) Authorization URL for connecting to designate in admin context
admin_password = <i>None</i>	(String) Password for connecting to designate in admin context
admin_tenant_id = <i>None</i>	(String) Tenant id for connecting to designate in admin context
admin_tenant_name = <i>None</i>	(String) Tenant name for connecting to designate in admin context
admin_username = <i>None</i>	(String) Username for connecting to designate in admin context
allow_reverse_dns_lookup = <i>True</i>	(Boolean) Allow the creation of PTR records
ipv4_ptr_zone_prefix_size = <i>24</i>	(Integer) Number of bits in an ipv4 PTR zone that will be considered network prefix. It has to align to byte boundary. Minimum value is 8. Maximum value is 24. As a consequence, range of values is 8, 16 and 24
ipv6_ptr_zone_prefix_size = <i>120</i>	(Integer) Number of bits in an ipv6 PTR zone that will be considered network prefix. It has to align to nyble boundary. Minimum value is 4. Maximum value is 124. As a consequence, range of values is 4, 8, 12, 16,..., 124
ptr_zone_email =	(String) The email address to be used when creating PTR zones. If not specified, the email address will be admin@<dns_domain>
url = <i>None</i>	(String) URL for connecting to designate

9.1.10. DHCP agent

Use the following options to alter Database-related settings.

Table 9.60. Description of DHCP agent configuration options

Configuration option = Default value	Description
[DEFAULT]	
advertise_mtu = <i>True</i>	(Boolean) If True, advertise network MTU values if core plugin calculates them. MTU is advertised to running instances via DHCP and RA MTU options.
dhcp_driver = <i>neutron.agent.linux.dhcp.Dnsmasq</i>	(String) The driver used to manage the DHCP server.
dnsmasq_base_log_dir = <i>None</i>	(String) Base log dir for dnsmasq logging. The log contains DHCP and DNS log information and is useful for debugging issues with either DHCP or DNS. If this section is null, disable dnsmasq log.
dnsmasq_config_file =	(String) Override the default dnsmasq settings with this file.
dnsmasq_dns_servers = <i>None</i>	(List) Comma-separated list of the DNS servers which will be used as forwarders.
dnsmasq_lease_max = <i>16777216</i>	(Integer) Limit number of leases to prevent a denial-of-service.
dnsmasq_local_resolv = <i>False</i>	(Boolean) Enables the dnsmasq service to provide name resolution for instances via DNS resolvers on the host running the DHCP agent. Effectively removes the '--no-resolv' option from the dnsmasq process arguments. Adding custom DNS resolvers to the 'dnsmasq_dns_servers' option disables this feature.
enable_isolated_metadata = <i>False</i>	(Boolean) The DHCP server can assist with providing metadata support on isolated networks. Setting this value to True will cause the DHCP server to append specific host routes to the DHCP request. The metadata service will only be activated when the subnet does not contain any router port. The guest instance must be configured to request host routes via DHCP (Option 121). This option doesn't have any effect when <code>force_metadata</code> is set to True.
enable_metadata_network = <i>False</i>	(Boolean) Allows for serving metadata requests coming from a dedicated metadata access network whose CIDR is 169.254.169.254/16 (or larger prefix), and is connected to a Neutron router from which the VMs send metadata:1 request. In this case DHCP Option 121 will not be injected in VMs, as they will be able to reach 169.254.169.254 through a router. This option requires <code>enable_isolated_metadata</code> = True.

Configuration option = Default value	Description
force_metadata = <i>False</i>	(Boolean) In some cases the Neutron router is not present to provide the metadata IP but the DHCP server can be used to provide this info. Setting this value will force the DHCP server to append specific host routes to the DHCP request. If this option is set, then the metadata service will be activated for all the networks.
host = <i>example.domain</i>	(String) Hostname to be used by the Neutron server, agents and services running on this machine. All the agents and services running on this machine must use the same host value.
interface_driver = <i>None</i>	(String) The driver used to manage the virtual interface.
num_sync_threads = 4	(Integer) Number of threads to use during sync process. Should not exceed connection pool size configured on server.
resync_interval = 5	(Integer) The DHCP agent will resync its state with Neutron to recover from any transient notification or RPC errors. The interval is number of seconds between attempts.

9.1.11. Distributed virtual router

Use the following options to alter DVR-related settings.

Table 9.61. Description of DVR configuration options

Configuration option = Default value	Description
[DEFAULT]	
dvr_base_mac = <i>fa:16:3f:00:00:00</i>	(String) The base mac address used for unique DVR instances by Neutron. The first 3 octets will remain unchanged. If the 4th octet is not 00, it will also be used. The others will be randomly generated. The 'dvr_base_mac' <i>*must*</i> be different from 'base_mac' to avoid mixing them up with MAC's allocated for tenant ports. A 4 octet example would be dvr_base_mac = fa:16:3f:4f:00:00. The default is 3 octet
router_distributed = <i>False</i>	(Boolean) System-wide flag to determine the type of router that tenants can create. Only admin can override.

9.1.12. Firewall-as-a-Service driver

Use the following options in the `fwaas_driver.ini` file for the FWaaS driver.

Table 9.62. Description of Firewall-as-a-Service configuration options

Configuration option = Default value	Description
[fwaas]	
driver =	(String) Name of the FWaaS Driver
enabled = <i>False</i>	(Boolean) Enable FWaaS

Table 9.63. Description of FWaaS NGFW plug-in configuration options

Configuration option = Default value	Description
[ngfw]	
smc_api_auth_key =	(String) Authentication key to SMC API
smc_api_version =	(String) version of SMC API
smc_url =	(String) URL to contact SMC server

Table 9.64. Description of FWaaS vArmour plug-in configuration options

Configuration option = Default value	Description
[vArmour]	
director = <i>localhost</i>	(String) vArmour director ip
director_port = <i>443</i>	(String) vArmour director port
password = <i>varmour</i>	(String) vArmour director password
username = <i>varmour</i>	(String) vArmour director username

9.1.13. Load-Balancer-as-a-Service configuration options

Use the following options in the `neutron_lbaas.conf` file for the LBaaS agent.

Table 9.65. Description of Load-Balancer-as-a-Service configuration options

Configuration option = Default value	Description
[certificates]	
barbican_auth = <i>barbican_acl_auth</i>	(String) Name of the Barbican authentication method to use
cert_manager_type = <i>barbican</i>	(String) Certificate Manager plugin. Defaults to barbican.
storage_path = <i>/var/lib/neutron-lbaas/certificates/</i>	(String) Absolute path to the certificate storage directory. Defaults to env[OS_LBAAS_TLS_STORAGE].

Use the following options in the `neutron_lbaas.conf` file for the LBaaS agent.

Table 9.66. Description of LBaaS service authentication configuration options

Configuration option = Default value	Description
[service_auth]	
admin_password = <i>password</i>	(String) The service admin password
admin_project_domain = <i>admin</i>	(String) The admin project domain name
admin_tenant_name = <i>admin</i>	(String) The service admin tenant name
admin_user = <i>admin</i>	(String) The service admin user name
admin_user_domain = <i>admin</i>	(String) The admin user domain name
auth_url = <i>http://127.0.0.1:5000/v2.0</i>	(String) Authentication endpoint
auth_version = <i>2</i>	(String) The auth version used to authenticate
endpoint_type = <i>public</i>	(String) The endpoint_type to be used
region = <i>RegionOne</i>	(String) The deployment region
service_name = <i>lbaas</i>	(String) The name of the service

Use the following options in the `lbaas_agent.ini` file for the LBaaS agent.

Table 9.67. Description of LBaaS agent configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(Boolean) If set to true, the logging level will be set to DEBUG instead of the default INFO level.
device_driver = <i>['neutron_lbaas.drivers.haproxy.namespace_driver.HaproxyNSDriver']</i>	(Multi-valued) Drivers used to manage loadbalancing devices
interface_driver = <i>None</i>	(String) The driver used to manage the virtual interface.
periodic_interval = <i>40</i>	(Integer) Seconds between running periodic tasks
[haproxy]	
loadbalancer_state_path = <i>\$state_path/lbaas</i>	(String) Location to store config and state files
send_gratuitous_arp = <i>3</i>	(Integer) When delete and re-add the same vip, send this many gratuitous ARPs to flush the ARP cache in the Router. Set it below or equal to 0 to disable this feature.
user_group = <i>nogroup</i>	(String) The user group

Use the following options in the `services_lbaas.conf` file for the LBaaS agent.

Table 9.68. Description of LBaaS Embrane, Radware, NetScaler, HAproxy, octavia plug-in configuration options

Configuration option = Default value	Description
[DEFAULT]	
loadbalancer_pool_scheduler_driver = <i>neutron_lbaas.services.loadbalancer.agent_scheduler.ChanceScheduler</i>	(String) Driver to use for scheduling pool to a default loadbalancer agent
loadbalancer_scheduler_driver = <i>neutron_lbaas.agent_scheduler.ChanceScheduler</i>	(String) Driver to use for scheduling to a default loadbalancer agent
[haproxy]	

Configuration option = Default value	Description
jinja_config_template = <i>/usr/lib/python/site-packages/neutron-lbaas/neutron_lbaas/services/loadbalancer/drivers/haproxy/templates/haproxy.loadbalancer.j2</i>	(String) Jinja template file for haproxy configuration
[netscaler_driver]	
is_synchronous = <i>True</i>	(String) Setting for option to enable synchronous operationsNetScaler Control Center Server.
netscaler_ncc_cleanup_mode = <i>None</i>	(String) Setting to enable/disable cleanup mode for NetScaler Control Center Server
netscaler_ncc_password = <i>None</i>	(String) Password to login to the NetScaler Control Center Server.
netscaler_ncc_uri = <i>None</i>	(String) The URL to reach the NetScaler Control Center Server.
netscaler_ncc_username = <i>None</i>	(String) Username to login to the NetScaler Control Center Server.
netscaler_status_collection = <i>True,300</i>	(String) Setting for member status collection fromNetScaler Control Center Server.
periodic_task_interval = <i>2</i>	(String) Setting for periodic task collection interval fromNetScaler Control Center Server..
[octavia]	
allocates_vip = <i>False</i>	(Boolean) True if Octavia will be responsible for allocating the VIP. False if neutron-lbaas will allocate it and pass to Octavia.
base_url = <i>http://127.0.0.1:9876</i>	(String) URL of Octavia controller root
request_poll_interval = <i>3</i>	(Integer) Interval in seconds to poll octavia when an entity is created, updated, or deleted.
request_poll_timeout = <i>100</i>	(Integer) Time to stop polling octavia when a status of an entity does not change.
[radware]	
actions_to_skip = <i>setup_l2_l3</i>	(List) List of actions that are not pushed to the completion queue.

Configuration option = Default value	Description
ha_secondary_address = <i>None</i>	(String) IP address of secondary vDirect server.
l2_l3_ctor_params = <i>{'ha_network_name': 'HA-Network', 'service': '_REPLACE_', 'ha_ip_pool_name': 'default', 'twoleg_enabled': '_REPLACE_', 'allocate_ha_ips': True, 'allocate_ha_vrrp': True}</i>	(Dict) Parameter for l2_l3 workflow constructor.
l2_l3_setup_params = <i>{'data_ip_address': '192.168.200.99', 'data_port': 1, 'gateway': '192.168.200.1', 'ha_port': 2, 'data_ip_mask': '255.255.255.0'}</i>	(Dict) Parameter for l2_l3 workflow setup.
l2_l3_workflow_name = <i>openstack_l2_l3</i>	(String) Name of l2_l3 workflow. Default: openstack_l2_l3.
l4_action_name = <i>BaseCreate</i>	(String) Name of the l4 workflow action. Default: BaseCreate.
l4_workflow_name = <i>openstack_l4</i>	(String) Name of l4 workflow. Default: openstack_l4.
service_adc_type = <i>VA</i>	(String) Service ADC type. Default: VA.
service_adc_version =	(String) Service ADC version.
service_cache = <i>20</i>	(Integer) Size of service cache. Default: 20.
service_compression_throughput = <i>100</i>	(Integer) Service compression throughput. Default: 100.
service_ha_pair = <i>False</i>	(Boolean) Enables or disables the Service HA pair. Default: False.
service_isl_vlan = <i>-1</i>	(Integer) A required VLAN for the interswitch link to use.
service_resource_pool_ids =	(List) Resource pool IDs.
service_session_mirroring_enabled = <i>False</i>	(Boolean) Enable or disable Alteon interswitch link for stateful session failover. Default: False.
service_ssl_throughput = <i>100</i>	(Integer) Service SSL throughput. Default: 100.
service_throughput = <i>1000</i>	(Integer) Service throughput. Default: 1000.
vdirect_address = <i>None</i>	(String) IP address of vDirect server.

Configuration option = Default value	Description
vdirect_password = <i>radware</i>	(String) vDirect user password.
vdirect_user = <i>vDirect</i>	(String) vDirect user name.
[radwarev2]	
child_workflow_template_names = <i>manage_l3</i>	(List) Name of child workflow templates used. Default: <i>manage_l3</i>
ha_secondary_address = <i>None</i>	(String) IP address of secondary vDirect server.
service_adc_type = <i>VA</i>	(String) Service ADC type. Default: <i>VA</i> .
service_adc_version =	(String) Service ADC version.
service_cache = <i>20</i>	(Integer) Size of service cache. Default: <i>20</i> .
service_compression_throughput = <i>100</i>	(Integer) Service compression throughput. Default: <i>100</i> .
service_ha_pair = <i>False</i>	(Boolean) Enables or disables the Service HA pair. Default: <i>False</i> .
service_isl_vlan = <i>-1</i>	(Integer) A required VLAN for the interswitch link to use.
service_resource_pool_ids =	(List) Resource pool IDs.
service_session_mirroring_enabled = <i>False</i>	(Boolean) Enable or disable Alteon interswitch link for stateful session failover. Default: <i>False</i> .
service_ssl_throughput = <i>100</i>	(Integer) Service SSL throughput. Default: <i>100</i> .
service_throughput = <i>1000</i>	(Integer) Service throughput. Default: <i>1000</i> .
stats_action_name = <i>stats</i>	(String) Name of the workflow action for statistics. Default: <i>stats</i> .
vdirect_address = <i>None</i>	(String) IP address of vDirect server.
vdirect_password = <i>radware</i>	(String) vDirect user password.
vdirect_user = <i>vDirect</i>	(String) vDirect user name.
workflow_action_name = <i>apply</i>	(String) Name of the workflow action. Default: <i>apply</i> .

Configuration option = Default value	Description
workflow_params = {'data_ip_address': '192.168.200.99', 'ha_network_name': 'HA-Network', 'ha_port': 2, 'allocate_ha_ips': True, 'ha_ip_pool_name': 'default', 'allocate_ha_vrrp': True, 'data_port': 1, 'gateway': '192.168.200.1', 'twoleg_enabled': '_REPLACE_', 'data_ip_mask': '255.255.255.0'}	(Dict) Parameter for l2_l3 workflow constructor.
workflow_template_name = <i>os_lb_v2</i>	(String) Name of the workflow template. Default: <i>os_lb_v2</i> .
[radwarev2_debug]	
configure_l3 = <i>True</i>	(Boolean) Configure ADC with L3 parameters?
configure_l4 = <i>True</i>	(Boolean) Configure ADC with L4 parameters?
provision_service = <i>True</i>	(Boolean) Provision ADC service?

Use the following options in the `/etc/octavia/octavia.conf` file for octavia config.

Table 9.69. Description of Octavia configuration options

Configuration option = Default value	Description
[DEFAULT]	
verbose = <i>False</i>	(BoolOpt) Print more verbose output (set logging level to INFO instead of default WARNING level).
debug = <i>False</i>	(BoolOpt) Print more verbose output (set logging level to INFO instead of default WARNING level).
bind_host = <i>0.0.0.0</i>	(StrOpt) The host IP to bind to the api service to .
bind_port = <i>9876</i>	(IntOpt) The port to bind to the api service to.
api_handler = <i>simulated_handler</i>	(StrOpt) The handler that the API communicates with.
octavia_plugins = <i>hot_plug_plugin</i>	(StrOpt) Name of the controller plugin to use.
os_region_name =	(StrOpt) Region in Identity service catalog to use for communication with the OpenStack services.

Configuration option = Default value	Description
host =	(StrOpt) Hostname to be used by the host machine for services running on it. The default value is the hostname of the host machine.
[database]	
connection = <i>mysql+pymysql://root:pass@127.0.0.1:3306/octavia</i>	(StrOpt) The SQLAlchemy connection string used to connect to the database.
[health_manager]	
bind_ip = 0.0.0.0	(StrOpt) IP address the controller will listen on for heart beats from the amphora.
bind_port = 5555	(IntOpt) Port number the controller will listen on for heart beats from the amphora.
controller_ip_port_list =	(StrOpt) List of controller ip and port pairs for the heartbeat receivers. Example [127.0.0.1:5555, 127.0.0.1:5555].
failover_threads = 10	(IntOpt) Number of threads performing amphora failovers.
status_update_threads = 50	(IntOpt) Number of threads performing amphora status update.
heartbeat_interval = 10	(IntOpt) Sleep time between sending heartbeats from the amphora.
heartbeat_key =	(StrOpt) key used to authenticate the heartbeat message sent by the amphora.
heartbeat_timeout = 60	(IntOpt) Interval, in seconds, to wait before failing over an amphora.
health_check_interval = 3	(IntOpt) Sleep time between health checks in seconds.
sock_rlimit = 0	(IntOpt) sets the value of the heartbeat recv buffer.
[keystone_auth_token]	
auth_uri = https://localhost:5000/v3	(StrOpt) Complete public Identity API endpoint.
admin_user = octavia	(StrOpt) Keystone account username.

Configuration option = Default value	Description
admin_password = <i>password</i>	(StrOpt) Keystone account password.
admin_tenant_name = <i>service</i>	(StrOpt) Keystone service account tenant name to validate user tokens.
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
[keystone_authtoken_v3]	
admin_user_domain = <i>default</i>	(StrOpt) Admin user keystone authentication domain.
admin_project_domain = <i>default</i>	(StrOpt) Admin project keystone authentication domain.
[certificates]	
cert_generator_class = <i>octavia.certificates.generator.LocalCertGenerator</i>	(StrOpt) Class name which generate certificates.
cert_manager_class = <i>octavia.certificates.manager.LocalCertManager</i>	(StrOpt) Class name of certificate manager.
ca_certificate = <i>/etc/ssl/certs/ssl-cert-snakeoil.pem</i>	(StrOpt) Absolute path to the CA Certificate for signing. Defaults to <code>env[OS_OCTAVIA_TLS_CA_CERT]</code> .Local Cert generator only.
ca_private_key = <i>/etc/ssl/private/ssl-cert-snakeoil.key</i>	(StrOpt) Absolute path to the Private Key for signing. Defaults to <code>env[OS_OCTAVIA_TLS_CA_KEY]</code> .Local Cert generator only.
ca_private_key_passphrase =	(StrOpt) Passphrase for the Private Key. Defaults to <code>env[OS_OCTAVIA_CA_KEY_PASS]</code> or None.Local Cert generator only.
signing_digest = <i>sha256</i>	(StrOpt) Certificate signing digest. Defaults to <code>env[OS_OCTAVIA_CA_SIGNING_DIGEST]</code> or <code>sha256</code> .Local Cert generator only.
storage_path = <i>/var/lib/octavia/certificates/</i>	(StrOpt) Absolute path to the certificate storage directory. Defaults to <code>env[OS_OCTAVIA_TLS_STORAGE]</code> .Local Cert manager only.
[octavia_network]	

Configuration option = Default value	Description
lb_network_name =	(StrOpt) Network to communicate with amphora.
max_retries = 15	(IntOpt) The maximum attempts to retry an action with the networking service.
retry_interval = 1	(IntOpt) Seconds to wait before retrying an action with the networking service.
[haproxy_amphora]	
base_path = <i>/var/lib/octavia</i>	(StrOpt) Base directory for amphora files on amphora.
base_cert_dir = <i>/var/lib/octavia/certs</i>	(StrOpt) Base directory for cert storage on amphora.
haproxy_template = <i>/var/lib/octavia/custom_template</i>	(StrOpt) Custom haproxy template.
base_log_dir = <i>/logs</i>	(StrOpt) Base director for log on amphora.
connection_max_retries = 300	(IntOpt) Retry threshold for connecting to amphorae.
connection_retry_interval = 5	(IntOpt) Retry threshold for connecting to amphorae.
cert_manager = <i>barbican_cert_manager</i>	(StrOpt) Name of the cert manager to use.
username = <i>ubuntu</i>	(StrOpt) Name of user for access to amphora,ssh driver only.
key_path = <i>/opt/stack/.ssh/id_rsa</i>	(StrOpt) Local absolute path to the private key loaded on amphora at boot,ssh driver only.
bind_host = 0.0.0.0	(StrOpt) The host IP to bind to amphora hose/REST driver only.
bind_port = 9191	(IntOpt) The port to bind to.REST driver only.
haproxy_cmd = <i>/usr/sbin/haproxy</i>	(StrOpt) The full path to haproxy.
respawn_count = 2	(IntOpt) The respawn count for haproxy's upstart script.

Configuration option = Default value	Description
respawn_interval = 2	(IntOpt) The respawn interval for haproxy's upstart script.
haproxy_cert_dir = /tmp	(StrOpt) The directory to store haproxy cert files in.
[controller_worker]	
amp_active_retries = 10	(IntOpt) Retry attempts to wait for Amphora to become active.
amp_active_wait_sec = 10	(IntOpt) Seconds to wait for an Amphora to become active.
amp_flavor_id =	(StrOpt) Nova instance flavor id for the Amphora.
amp_image_id =	(StrOpt) Glance image id for the Amphora image to boot.
amp_ssh_key_name =	(StrOpt) SSH key name used to boot the Amphora.REST driver/or debugging.
amp_network =	(StrOpt) Network to attach to the Amphora.
amp_secgroup_list =	(StrOpt) List of security groups to attach to the Amphora.
client_ca = /etc/octavia/certs/ca_01.pem	(StrOpt) Client CA for the amphora agent to use.REST driver only.
amphora_driver = <i>amphora_noop_driver</i>	(StrOpt) Name of the amphora driver to use.
compute_driver = <i>compute_noop_driver</i>	(StrOpt) Name of the compute driver to use.
network_driver = <i>network_noop_driver</i>	(StrOpt) Name of the network driver to use.
cert_generator = <i>local_cert_generator</i>	(StrOpt) Name of the cert generator to use.
[task_flow]	
engine = <i>serial</i>	(StrOpt) TaskFlow engine to use.
max_workers = 5	(IntOpt) The maximum number of workers.
[oslo_messaging_rabbit]	
rabbit_userid = <i>octavia</i>	(StrOpt) RabbitMQ username.

Configuration option = Default value	Description
rabbit_password = <i>password</i>	(StrOpt) RabbitMQ password.
rabbit_port = 5672	(IntOpt) RabbitMQ port.
rabbit_hosts = <i>localhost:5672</i>	(StrOpt) RabbitMQ host.
[oslo_messaging]	
rpc_thread_pool_size = 2	(IntOpt) Queue Consumer Thread Pool Size.
topic = <i>octavia_prov</i>	(StrOpt) Topic (i.e. Queue) Name.
[house_keeping]	
spare_check_interval = 30	(IntOpt) Interval in seconds to initiate spare amphora checks.
spare_amphora_pool_size = 0	(IntOpt) Number of spare amphorae.
cleanup_interval = 30	(IntOpt) Cleanup interval for Deleted amphora.
amphora_expiry_age = 604800	(IntOpt) Amphora expiry age in seconds. Default is 1 week.

9.1.14. VPN-as-a-Service configuration options

Use the following options in the `vpnaas_agent.ini` file for the VPNaaS agent.

Table 9.70. Description of VPN-as-a-Service configuration options

Configuration option = Default value	Description
[vpnagent]	
vpn_device_driver = [<i>'neutron_vpnaas.services.vpn.device_drivers.ipsec.OpenSwanDriver,</i> <i>neutron_vpnaas.services.vpn.device_drivers.cisco_ipsec.CiscoCsrIPSecDriver,</i> <i>neutron_vpnaas.services.vpn.device_drivers.vyatta_ipsec.VyattaIPSecDriver,</i> <i>neutron_vpnaas.services.vpn.device_drivers.strongswan_ipsec.StrongSwanDriver,</i> <i>neutron_vpnaas.services.vpn.device_drivers.fedora_strongswan_ipsec.FedoraStrongSwanDriver,</i> <i>neutron_vpnaas.services.vpn.device_drivers.libreswan_ipsec.LibreswanDriver'</i>]	(Multi-valued) The vpn device drivers Neutron will use

Table 9.71. Description of VPNaaS IPsec plug-in configuration options

Configuration option = Default value	Description
[cisco_csr_ipsec]	
status_check_interval = 60	(Integer) Status check interval for Cisco CSR IPsec connections
[ipsec]	
config_base_dir = <i>\$state_path/ipsec</i>	(String) Location to store ipsec server config files
enable_detailed_logging = <i>False</i>	(Boolean) Enable detail logging for ipsec pluto process. If the flag set to True, the detailed logging will be written into config_base_dir/<pid>/log. Note: This setting applies to OpenSwan and LibreSwan only. StrongSwan logs to syslog.
ipsec_status_check_interval = 60	(Integer) Interval for checking ipsec status
[pluto]	
shutdown_check_back_off = 1.5	(Floating point) A factor to increase the retry interval for each retry
shutdown_check_retries = 5	(Integer) The maximum number of retries for checking for pluto daemon shutdown
shutdown_check_timeout = 1	(Integer) Initial interval in seconds for checking if pluto daemon is shutdown

Table 9.72. Description of VPNaaS Openswan plug-in configuration options

Configuration option = Default value	Description
[openswan]	
ipsec_config_template = <i>/usr/lib/python/site-packages/neutron-vpnaas/neutron_vpnaas/services/vpn/device_drivers/template/openswan/ipsec.conf.template</i>	(String) Template file for ipsec configuration
ipsec_secret_template = <i>/usr/lib/python/site-packages/neutron-vpnaas/neutron_vpnaas/services/vpn/device_drivers/template/openswan/ipsec.secret.template</i>	(String) Template file for ipsec secret configuration

Table 9.73. Description of VPNaaS strongSwan plug-in configuration options

Configuration option = Default value	Description
[strongswan]	
default_config_area = <i>/etc/strongswan.d</i>	(String) The area where default StrongSwan configuration files are located.
ipsec_config_template = <i>/usr/lib/python/site-packages/neutron-vpnaas/neutron_vpnaas/services/vpn/device_drivers/template/strongswan/ipsec.conf.template</i>	(String) Template file for ipsec configuration.
ipsec_secret_template = <i>/usr/lib/python/site-packages/neutron-vpnaas/neutron_vpnaas/services/vpn/device_drivers/template/strongswan/ipsec.secret.template</i>	(String) Template file for ipsec secret configuration.
strongswan_config_template = <i>/usr/lib/python/site-packages/neutron-vpnaas/neutron_vpnaas/services/vpn/device_drivers/template/strongswan/strongswan.conf.template</i>	(String) Template file for strongswan configuration.

9.1.15. IPv6 router advertisement

Use the following options to alter IPv6 RA settings.

Table 9.74. Description of IPv6 router advertisement configuration options

Configuration option = Default value	Description
[DEFAULT]	
ra_confs = <i>\$state_path/ra</i>	(String) Location to store IPv6 RA config files

9.1.16. L3 agent

Use the following options in the `l3_agent.ini` file for the L3 agent.

Table 9.75. Description of L3 agent configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
agent_mode = <i>legacy</i>	(String) The working mode for the agent. Allowed modes are: 'legacy' - this preserves the existing behavior where the L3 agent is deployed on a centralized networking node to provide L3 services like DNAT, and SNAT. Use this mode if you do not want to adopt DVR. 'dvr' - this mode enables DVR functionality and must be used for an L3 agent that runs on a compute host. 'dvr_snat' - this enables centralized SNAT support in conjunction with DVR. This mode must be used for an L3 agent running on a centralized node (or in single-host deployments, e.g. devstack)
allow_automatic_dhcp_failover = <i>True</i>	(Boolean) Automatically remove networks from offline DHCP agents.
allow_automatic_l3agent_failover = <i>False</i>	(Boolean) Automatically reschedule routers from offline L3 agents to online L3 agents.
enable_metadata_proxy = <i>True</i>	(Boolean) Allow running metadata proxy.
enable_snat_by_default = <i>True</i>	(Boolean) Define the default value of enable_snat if not provided in external_gateway_info.
external_ingress_mark = <i>0x2</i>	(String) Iptables mangle mark used to mark ingress from external network. This mark will be masked with 0xffff so that only the lower 16 bits will be used.
external_network_bridge = <i>br-ex</i>	(String) DEPRECATED: Name of bridge used for external network traffic. This should be set to an empty value for the Linux Bridge. When this parameter is set, each L3 agent can be associated with no more than one external network. This option is deprecated and will be removed in the M release.
gateway_external_network_id =	(String) When external_network_bridge is set, each L3 agent can be associated with no more than one external network. This value should be set to the UUID of that external network. To allow L3 agent support multiple external networks, both the external_network_bridge and gateway_external_network_id must be left empty.
ha_confs_path = <i>\$state_path/ha_confs</i>	(String) Location to store keepalived/contrackd config files
ha_vrrp_advert_int = <i>2</i>	(Integer) The advertisement interval in seconds
ha_vrrp_auth_password = <i>None</i>	(String) VRRP authentication password

Configuration option = Default value	Description
ha_vrrp_auth_type = <i>PASS</i>	(String) VRRP authentication type
handle_internal_only_routers = <i>True</i>	(Boolean) Indicates that this L3 agent should also handle routers that do not have an external network gateway configured. This option should be True only for a single agent in a Neutron deployment, and may be False for all agents if all routers must have an external network gateway.
host = <i>example.domain</i>	(String) Hostname to be used by the Neutron server, agents and services running on this machine. All the agents and services running on this machine must use the same host value.
interface_driver = <i>None</i>	(String) The driver used to manage the virtual interface.
ipv6_gateway =	(String) With IPv6, the network used for the external gateway does not need to have an associated subnet, since the automatically assigned link-local address (LLA) can be used. However, an IPv6 gateway address is needed for use as the next-hop for the default route. If no IPv6 gateway address is configured here, (and only then) the neutron router will be configured to get its default route from router advertisements (RAs) from the upstream router; in which case the upstream router must also be configured to send these RAs. The <code>ipv6_gateway</code> , when configured, should be the LLA of the interface on the upstream router. If a next-hop using a global unique address (GUA) is desired, it needs to be done via a subnet allocated to the network and not through this parameter.
ipv6_pd_enabled = <i>False</i>	(Boolean) Enables IPv6 Prefix Delegation for automatic subnet CIDR allocation. Set to True to enable IPv6 Prefix Delegation for subnet allocation in a PD-capable environment. Users making subnet creation requests for IPv6 subnets without providing a CIDR or subnetpool ID will be given a CIDR via the Prefix Delegation mechanism. Note that enabling PD will override the behavior of the default IPv6 subnetpool.
l3_ha = <i>False</i>	(Boolean) Enable HA mode for virtual routers.
l3_ha_net_cidr = <i>169.254.192.0/18</i>	(String) Subnet used for the l3 HA admin network.
l3_ha_network_physical_name =	(String) The physical network name with which the HA network can be created.

Configuration option = Default value	Description
l3_ha_network_type =	(String) The network type to use when creating the HA network for an HA router. By default or if empty, the first 'tenant_network_types' is used. This is helpful when the VRRP traffic should use a specific network which is not the default one.
max_l3_agents_per_router = 3	(Integer) Maximum number of L3 agents which a HA router will be scheduled on. If it is set to 0 then the router will be scheduled on every agent.
min_l3_agents_per_router = 2	(Integer) Minimum number of L3 agents which a HA router will be scheduled on. If it is set to 0 then the router will be scheduled on every agent.
router_id =	(String) DEPRECATED: If non-empty, the l3 agent can only configure a router that has the matching router ID.
send_arp_for_ha = 3	(Integer) Send this many gratuitous ARPs for HA setup, if less than or equal to 0, the feature is disabled
[AGENT]	
comment_iptables_rules = True	(Boolean) Add comments to iptables rules. Set to false to disallow the addition of comments to generated iptables rules that describe each rule's purpose. System must support the iptables comments module for addition of comments.
use_helper_for_ns_read = True	(Boolean) Use the root helper when listing the namespaces on a system. This may not be required depending on the security configuration. If the root helper is not required, set this to False for a performance improvement.

9.1.17. Logging

Use the following options to alter logging settings.

Table 9.76. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = False	(Boolean) If set to true, the logging level will be set to DEBUG instead of the default INFO level.

Configuration option = Default value	Description
default_log_levels = <i>amqp=WARN, amqplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystonemiddleware=WARN, routes.middleware=WARN, stevedore=WARN, taskflow=WARN, keystoneauth=WARN, oslo.cache=INFO, dogpile.core.dogpile=INFO</i>	(List) List of package logging levels in logger=LEVEL pairs. This option is ignored if log_config_append is set.
fatal_deprecations = <i>False</i>	(Boolean) Enables or disables fatal status of deprecations.
instance_format = <i>"[instance: %(uuid)s] "</i>	(String) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(String) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(String) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, logging_context_format_string).
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(String) Defines the format string for <i>%(asctime)s</i> in log records. Default: <i>%(default)s</i> . This option is ignored if log_config_append is set.
log_dir = <i>None</i>	(String) (Optional) The base directory used for relative log_file paths. This option is ignored if log_config_append is set.
log_file = <i>None</i>	(String) (Optional) Name of log file to send logging output to. If no default is set, logging will go to stderr as defined by use_stderr. This option is ignored if log_config_append is set.
logging_context_format_string = <i>%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%(message)s</i>	(String) Format string to use for log messages with context.

Configuration option = Default value	Description
logging_debug_format_suffix = % (funcName)s %(pathname)s: %(lineno)d	(String) Additional data to append to log message when logging level for the message is DEBUG.
logging_default_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [-] %(instance)s%(message)s	(String) Format string to use for log messages when context is undefined.
logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d ERROR %(name)s % (instance)s	(String) Prefix each line of exception output with this format.
logging_user_identity_format = %(user)s %(tenant)s %(domain)s %(user_domain)s % (project_domain)s	(String) Defines the format string for % (user_identity)s that is used in logging_context_format_string.
publish_errors = <i>False</i>	(Boolean) Enables or disables publication of error events.
syslog_log_facility = <i>LOG_USER</i>	(String) Syslog facility to receive log lines. This option is ignored if log_config_append is set.
use_ssl = <i>False</i>	(Boolean) Enable SSL on the API server
use_stderr = <i>True</i>	(Boolean) Log output to standard error. This option is ignored if log_config_append is set.
use_syslog = <i>False</i>	(Boolean) Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.
verbose = <i>True</i>	(Boolean) DEPRECATED: If set to false, the logging level will be set to WARNING instead of the default INFO level.
watch_log_file = <i>False</i>	(Boolean) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log_file option is specified and Linux platform is used. This option is ignored if log_config_append is set.
wsgi_log_format = %(client_ip)s "% (request_line)s" status: %(status_code)s len: % (body_length)s time: %(wall_seconds).7f	(String) A python format string that is used as the template to generate log lines. The following values can be formatted into it: client_ip, date_time, request_line, status_code, body_length, wall_seconds.

Configuration option = Default value	Description
[oslo_versionedobjects]	
fatal_exception_format_errors = False	(Boolean) Make exception message format errors fatal

9.1.18. Metadata Agent

Use the following options in the `metadata_agent.ini` file for the Metadata agent.

Table 9.77. Description of metadata configuration options

Configuration option = Default value	Description
[DEFAULT]	
metadata_access_mark = 0x1	(String) Iptables mangle mark used to mark metadata valid requests. This mark will be masked with 0xffff so that only the lower 16 bits will be used.
metadata_backlog = 4096	(Integer) Number of backlog requests to configure the metadata server socket with
metadata_port = 9697	(Port number) TCP Port used by Neutron metadata namespace proxy.
metadata_proxy_group =	(String) Group (gid or name) running metadata proxy after its initialization (if empty: agent effective group).
metadata_proxy_shared_secret =	(String) When proxying metadata requests, Neutron signs the Instance-ID header with a shared secret to prevent spoofing. You may select any string for a secret, but it must match here and in the configuration used by the Nova Metadata Server. NOTE: Nova uses the same config key, but in [neutron] section.
metadata_proxy_socket = \$state_path/metadata_proxy	(String) Location for Metadata Proxy UNIX domain socket.

Configuration option = Default value	Description
metadata_proxy_socket_mode = <i>deduce</i>	(String) Metadata Proxy UNIX domain socket mode, 4 values allowed: 'deduce': deduce mode from metadata_proxy_user/group values, 'user': set metadata proxy socket mode to 0o644, to use when metadata_proxy_user is agent effective user or root, 'group': set metadata proxy socket mode to 0o664, to use when metadata_proxy_group is agent effective group or root, 'all': set metadata proxy socket mode to 0o666, to use otherwise.
metadata_proxy_user =	(String) User (uid or name) running metadata proxy after its initialization (if empty: agent effective user).
metadata_proxy_watch_log = <i>None</i>	(Boolean) Enable/Disable log watch by metadata proxy. It should be disabled when metadata_proxy_user/group is not allowed to read/write its log file and copytruncate logrotate option must be used if logrotate is enabled on metadata proxy log files. Option default value is deduced from metadata_proxy_user: watch log is enabled if metadata_proxy_user is agent effective user id/name.
metadata_workers = 2	(Integer) Number of separate worker processes for metadata server (defaults to half of the number of CPUs)
nova_metadata_insecure = <i>False</i>	(Boolean) Allow to perform insecure SSL (https) requests to nova metadata
nova_metadata_ip = <i>127.0.0.1</i>	(String) IP address used by Nova metadata server.
nova_metadata_port = 8775	(Port number) TCP Port used by Nova metadata server.
nova_metadata_protocol = <i>http</i>	(String) Protocol to access nova metadata, http or https



NOTE

Previously, neutron metadata agent connected to a neutron server via REST API using a neutron client. This is ineffective because keystone is then fully involved into the authentication process and gets overloaded.

The neutron metadata agent has been reworked to use RPC by default to connect to a server since Kilo release. This is a typical way of interacting between neutron server and its agents. If neutron server does not support metadata RPC then neutron client will be used.

**WARNING**

Do not run the `neutron-ns-metadata-proxy` proxy namespace as root on a node with the L3 agent running. In OpenStack Kilo and newer, you can change the permissions of `neutron-ns-metadata-proxy` after the proxy installation using the `metadata_proxy_user` and `metadata_proxy_group` options.

9.1.19. Metering Agent

Use the following options in the `metering_agent.ini` file for the Metering agent.

Table 9.78. Description of metering agent configuration options

Configuration option = Default value	Description
[DEFAULT]	
driver = <i>neutron.services.metering.drivers.noop.noop_driver.NoopMeteringDriver</i>	(String) Metering driver
measure_interval = 30	(Integer) Interval between two metering measures
[AGENT]	
report_interval = 30	(Floating point) Seconds between nodes reporting state to server; should be less than <code>agent_down_time</code> , best if it is half or less than <code>agent_down_time</code> .

9.1.20. ML2 MacVTap

Use the following options to alter ML2 MacVTap-related settings.

Table 9.79. Description of ML2 MacVTap mechanism driver configuration options

Configuration option = Default value	Description
[AGENT]	
quitting_rpc_timeout = 10	(Integer) Set new timeout in seconds for new rpc calls after agent receives SIGTERM. If value is set to 0, rpc timeout won't be changed
[macvtap]	

Configuration option = Default value	Description
physical_interface_mappings =	(List) Comma-separated list of <physical_network>: <physical_interface> tuples mapping physical network names to the agent's node-specific physical network interfaces to be used for flat and VLAN networks. All physical networks listed in network_vlan_ranges on the server should have mappings to appropriate interfaces on each agent.

9.1.21. Nova

Use the following options in the `neutron.conf` file to change nova-related settings.

Table 9.80. Description of nova configuration options

Configuration option = Default value	Description
[nova]	
auth_section = <i>None</i>	(Unknown) Config Section from which to load plugin specific options
auth_type = <i>None</i>	(Unknown) Authentication type to load
cafile = <i>None</i>	(String) PEM encoded Certificate Authority to use when verifying HTTPs connections.
certfile = <i>None</i>	(String) PEM encoded client certificate cert file
endpoint_type = <i>public</i>	(String) Type of the nova endpoint to use. This endpoint will be looked up in the keystone catalog and should be one of public, internal or admin.
insecure = <i>False</i>	(Boolean) Verify HTTPS connections.
keyfile = <i>None</i>	(String) PEM encoded client certificate key file
region_name = <i>None</i>	(String) Name of nova region to use. Useful if keystone manages more than one region.
timeout = <i>None</i>	(Integer) Timeout value for http requests

9.1.22. Policy

Use the following options in the `neutron.conf` file to change policy settings.

Table 9.81. Description of policy configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>allow_overlapping_ips = False</code>	(Boolean) Allow overlapping IP support in Neutron. Attention: the following parameter MUST be set to False if Neutron is being used in conjunction with Nova security groups.

9.1.23. Quotas

Use the following options in the `neutron.conf` file for the quota system.

Table 9.82. Description of quotas configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>max_routes = 30</code>	(Integer) Maximum number of routes per router
[QUOTAS]	
<code>default_quota = -1</code>	(Integer) Default number of resource allowed per tenant. A negative value means unlimited.
<code>quota_driver = neutron.db.quota.driver.DbQuotaDriver</code>	(String) Default driver to use for quota checks
<code>quota_firewall = 10</code>	(Integer) Number of firewalls allowed per tenant. A negative value means unlimited.
<code>quota_firewall_policy = 10</code>	(Integer) Number of firewall policies allowed per tenant. A negative value means unlimited.
<code>quota_firewall_rule = 100</code>	(Integer) Number of firewall rules allowed per tenant. A negative value means unlimited.
<code>quota_floatingip = 50</code>	(Integer) Number of floating IPs allowed per tenant. A negative value means unlimited.
<code>quota_health_monitor = -1</code>	(Integer) Number of health monitors allowed per tenant. A negative value means unlimited.
<code>quota_healthmonitor = -1</code>	(Integer) Number of health monitors allowed per tenant. A negative value means unlimited.

Configuration option = Default value	Description
quota_items = <i>network, subnet, port</i>	(List) DEPRECATED: Resource name(s) that are supported in quota features. This option is now deprecated for removal.
quota_listener = <i>-1</i>	(Integer) Number of Loadbalancer Listeners allowed per tenant. A negative value means unlimited.
quota_loadbalancer = <i>10</i>	(Integer) Number of LoadBalancers allowed per tenant. A negative value means unlimited.
quota_member = <i>-1</i>	(Integer) Number of pool members allowed per tenant. A negative value means unlimited.
quota_network = <i>10</i>	(Integer) Number of networks allowed per tenant. A negative value means unlimited.
quota_pool = <i>10</i>	(Integer) Number of pools allowed per tenant. A negative value means unlimited.
quota_port = <i>50</i>	(Integer) Number of ports allowed per tenant. A negative value means unlimited.
quota_rbac_policy = <i>10</i>	(Integer) Default number of RBAC entries allowed per tenant. A negative value means unlimited.
quota_router = <i>10</i>	(Integer) Number of routers allowed per tenant. A negative value means unlimited.
quota_security_group = <i>10</i>	(Integer) Number of security groups allowed per tenant. A negative value means unlimited.
quota_security_group_rule = <i>100</i>	(Integer) Number of security rules allowed per tenant. A negative value means unlimited.
quota_subnet = <i>10</i>	(Integer) Number of subnets allowed per tenant, A negative value means unlimited.
quota_vip = <i>10</i>	(Integer) Number of vips allowed per tenant. A negative value means unlimited.
track_quota_usage = <i>True</i>	(Boolean) Keep in track in the database of current resourcequota usage. Plugins which do not leverage the neutron database should set this flag to False

9.1.24. Scheduler

Use the following options in the `neutron.conf` file to change scheduler settings.

Table 9.83. Description of scheduler configuration options

Configuration option = Default value	Description
[DEFAULT]	
network_auto_schedule = <i>True</i>	(Boolean) Allow auto scheduling networks to DHCP agent.
network_scheduler_driver = <i>neutron.scheduler.dhcp_agent_scheduler.WeightScheduler</i>	(String) Driver to use for scheduling network to DHCP agent
router_auto_schedule = <i>True</i>	(Boolean) Allow auto scheduling of routers to L3 agent.
router_scheduler_driver = <i>neutron.scheduler.l3_agent_scheduler.LeastRoutersScheduler</i>	(String) Driver to use for scheduling router to a default L3 agent

9.1.25. Security Groups

Use the following options in the configuration file for your driver to change security group settings.

Table 9.84. Description of security groups configuration options

Configuration option = Default value	Description
[SECURITYGROUP]	
enable_ipset = <i>True</i>	(Boolean) Use ipset to speed-up the iptables based security groups. Enabling ipset support requires that ipset is installed on L2 agent node.
enable_security_group = <i>True</i>	(Boolean) Controls whether the neutron security group API is enabled in the server. It should be false when using no security groups or using the nova security group API.
firewall_driver = <i>None</i>	(String) Driver for security groups firewall in the L2 agent

**NOTE**

Now Networking uses iptables to achieve security group functions. In L2 agent with `enable_ipset` option enabled, it makes use of IPset to improve security group's performance, as it represents a hash set which is insensitive to the number of elements.

When a port is created, L2 agent will add an additional IPset chain to its iptables chain, if the security group that this port belongs to has rules between other security group, the member of that security group will be added to the ipset chain.

If a member of a security group is changed, it used to reload iptables rules which is expensive. However, when IPset option is enabled on L2 agent, it does not need to reload iptables if only members of security group were changed, it should just update an IPset.

**NOTE**

A single default security group has been introduced in order to avoid race conditions when creating a tenant's default security group. The race conditions are caused by the uniqueness check of a new security group name. A table `default_security_group` implements such a group. It has `tenant_id` field as a primary key and `security_group_id`, which is an identifier of a default security group. The migration that introduces this table has a sanity check that verifies if a default security group is not duplicated in any tenant.

9.1.26. SSL and Certification Authority

Use the following options in the `neutron.conf` file to enable SSL.

Table 9.85. Description of CA and SSL configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>ssl_ca_file = None</code>	(StrOpt) CA certificate file to use to verify connecting clients
<code>ssl_cert_file = None</code>	(StrOpt) Certificate file to use when starting the server securely
<code>ssl_key_file = None</code>	(StrOpt) Private key file to use when starting the server securely

9.1.27. Misc

Table 9.86. Description of BGP configuration options

Configuration option = Default value	Description
[BGP]	

Configuration option = Default value	Description
bgp_router_id = <i>None</i>	(String) 32-bit BGP identifier, typically an IPv4 address owned by the system running the BGP DrAgent.
bgp_speaker_driver = <i>None</i>	(String) BGP speaker driver class to be instantiated.

Table 9.87. Description of QoS configuration options

Configuration option = Default value	Description
[QoS]	
kernel_hz = 250	(Integer) Value of host kernel tick rate (hz) for calculating minimum burst value in bandwidth limit rules for a port with QoS. See kernel configuration file for HZ value and tc-tbf manual for more information.
tbf_latency = 50	(Integer) Value of latency (ms) for calculating size of queue for a port with QoS. See tc-tbf manual for more information.

9.2. LOG FILES USED BY NETWORKING

The corresponding log file of each Networking service is stored in the `/var/log/neutron/` directory of the host on which each service runs.

Table 9.88. Log files used by Networking services

Log file	Service/interface
dhcp-agent.log	neutron-dhcp-agent
l3-agent.log	neutron-l3-agent
lbaas-agent.log	neutron-lbaas-agent ^[a]
linuxbridge-agent.log	neutron-linuxbridge-agent
metadata-agent.log	neutron-metadata-agent
metering-agent.log	neutron-metering-agent
openvswitch-agent.log	neutron-openvswitch-agent

Log file	Service/interface
server.log	neutron-server
[a] The neutron-lbaas-agent service only runs when Load-Balancer-as-a-Service is enabled.	

9.3. NETWORKING SAMPLE CONFIGURATION FILES

All the files in this section can be found in `/etc/neutron/`.

9.3.1. neutron.conf

Use the **neutron.conf** file to configure the majority of the OpenStack Networking options.

```
[DEFAULT]
# Print more verbose output (set logging level to INFO instead of default
WARNING level).
# verbose = False

# =====Start Global Config Option for Distributed L3
Router=====
# Setting the "router_distributed" flag to "True" will default to the
creation
# of distributed tenant routers. The admin can override this flag by
specifying
# the type of the router on the create request (admin-only attribute).
Default
# value is "False" to support legacy mode (centralized) routers.
#
# router_distributed = False
#
# =====End Global Config Option for Distributed L3
Router=====

# Print debugging output (set logging level to DEBUG instead of default
WARNING level).
# debug = False

# Where to store Neutron state files. This directory must be writable by
the
# user executing the agent.
# state_path = /var/lib/neutron

# log_format = %(asctime)s %(levelname)s [%s] %(message)s
# log_date_format = %Y-%m-%d %H:%M:%S

# use_syslog                                -> syslog
# log_file and log_dir                      -> log_dir/log_file
# (not log_file) and log_dir                -> log_dir/{binary_name}.log
# use_stderr                               -> stderr
# (not user_stderr) and (not log_file)     -> stdout
# publish_errors                           -> notification system
```

```

# use_syslog = False
# syslog_log_facility = LOG_USER

# use_stderr = True
# log_file =
# log_dir =

# publish_errors = False

# Address to bind the API server to
# bind_host = 0.0.0.0

# Port to bind the API server to
# bind_port = 9696

# Path to the extensions. Note that this can be a colon-separated list of
# paths. For example:
# api_extensions_path =
# extensions:/path/to/more/extensions:/even/more/extensions
# The __path__ of neutron.extensions is appended to this, so if your
# extensions are in there you don't need to specify them here
# api_extensions_path =

# (StrOpt) Neutron core plugin entrypoint to be loaded from the
# neutron.core_plugins namespace. See setup.cfg for the entrypoint names
# of the
# plugins included in the neutron source distribution. For compatibility
# with
# previous versions, the class name of a plugin can be specified instead
# of its
# entrypoint name.
#
# core_plugin =
# Example: core_plugin = ml2

# (ListOpt) List of service plugin entrypoints to be loaded from the
# neutron.service_plugins namespace. See setup.cfg for the entrypoint
# names of
# the plugins included in the neutron source distribution. For
# compatibility
# with previous versions, the class name of a plugin can be specified
# instead
# of its entrypoint name.
#
# service_plugins =
# Example: service_plugins = router,firewall,lbaas,vpnaas,metering

# Paste configuration file
# api_paste_config = api-paste.ini

# (StrOpt) Hostname to be used by the neutron server, agents and services
# running on this machine. All the agents and services running on this
# machine
# must use the same host value.
# The default value is hostname of the machine.

```

```
#
# host =

# The strategy to be used for auth.
# Supported values are 'keystone'(default), 'noauth'.
# auth_strategy = keystone

# Base MAC address. The first 3 octets will remain unchanged. If the
# 4th octet is not 00, it will also be used. The others will be
# randomly generated.
# 3 octet
# base_mac = fa:16:3e:00:00:00
# 4 octet
# base_mac = fa:16:3e:4f:00:00

# DVR Base MAC address. The first 3 octets will remain unchanged. If the
# 4th octet is not 00, it will also be used. The others will be randomly
# generated. The 'dvr_base_mac' *must* be different from 'base_mac' to
# avoid mixing them up with MAC's allocated for tenant ports.
# A 4 octet example would be dvr_base_mac = fa:16:3f:4f:00:00
# The default is 3 octet
# dvr_base_mac = fa:16:3f:00:00:00

# Maximum amount of retries to generate a unique MAC address
# mac_generation_retries = 16

# DHCP Lease duration (in seconds). Use -1 to
# tell dnsmasq to use infinite lease times.
# dhcp_lease_duration = 86400

# Allow sending resource operation notification to DHCP agent
# dhcp_agent_notification = True

# Enable or disable bulk create/update/delete operations
# allow_bulk = True
# Enable or disable pagination
# allow_pagination = False
# Enable or disable sorting
# allow_sorting = False
# Enable or disable overlapping IPs for subnets
# Attention: the following parameter MUST be set to False if Neutron is
# being used in conjunction with nova security groups
# allow_overlapping_ips = False
# Ensure that configured gateway is on subnet. For IPv6, validate only if
# gateway is not a link local address. Deprecated, to be removed during
# the
# K release, at which point the check will be mandatory.
# force_gateway_on_subnet = True

# Default maximum number of items returned in a single response,
# value == infinite and value < 0 means no max limit, and value must
# be greater than 0. If the number of items requested is greater than
# pagination_max_limit, server will just return pagination_max_limit
# of number of items.
# pagination_max_limit = -1
```



```

# Maximum number of DNS nameservers per subnet
# max_dns_nameservers = 5

# Maximum number of host routes per subnet
# max_subnet_host_routes = 20

# Maximum number of fixed ips per port
# max_fixed_ips_per_port = 5

# Maximum number of routes per router
# max_routes = 30

# Default Subnet Pool to be used for IPv4 subnet-allocation.
# Specifies by UUID the pool to be used in case of subnet-create being
called
# without a subnet-pool ID. The default of None means that no pool will
be
# used unless passed explicitly to subnet create. If no pool is used,
then a
# CIDR must be passed to create a subnet and that subnet will not be
allocated
# from any pool; it will be considered part of the tenant's private
address
# space.
# default_ipv4_subnet_pool =

# Default Subnet Pool to be used for IPv6 subnet-allocation.
# Specifies by UUID the pool to be used in case of subnet-create being
called without a subnet-pool ID. Set to "prefix_delegation"
# to enable IPv6 Prefix Delegation in a PD-capable environment.
# See the description for default_ipv4_subnet_pool for more information.
# default_ipv6_subnet_pool =

# ===== items for MTU selection and advertisement =====
# Advertise MTU. If True, effort is made to advertise MTU
# settings to VMs via network methods (ie. DHCP and RA MTU options)
# when the network's preferred MTU is known.
# advertise_mtu = False
# ===== end of items for MTU selection and advertisement =====

# ===== items for agent management extension =====
# Seconds to regard the agent as down; should be at least twice
# report_interval, to be sure the agent is down for good
# agent_down_time = 75
# ===== end of items for agent management extension =====

# ===== items for agent scheduler extension =====
# Driver to use for scheduling network to DHCP agent
# network_scheduler_driver =
neutron.scheduler.dhcp_agent_scheduler.ChanceScheduler
# Driver to use for scheduling router to a default L3 agent
# router_scheduler_driver =
neutron.scheduler.l3_agent_scheduler.ChanceScheduler
# Driver to use for scheduling a loadbalancer pool to an lbaas agent
# loadbalancer_pool_scheduler_driver =
neutron.services.loadbalancer.agent_scheduler.ChanceScheduler

```

```

# (StrOpt) Representing the resource type whose load is being reported by
# the agent.
# This can be 'networks', 'subnets' or 'ports'. When specified (Default is
# networks),
# the server will extract particular load sent as part of its agent
# configuration object
# from the agent report state, which is the number of resources being
# consumed, at
# every report_interval.
# dhcp_load_type can be used in combination with network_scheduler_driver
# =
# neutron.scheduler.dhcp_agent_scheduler.WeightScheduler
# When the network_scheduler_driver is WeightScheduler, dhcp_load_type can
# be configured to represent the choice for the resource being balanced.
# Example: dhcp_load_type = networks
# Values:
#   networks - number of networks hosted on the agent
#   subnets - number of subnets associated with the networks hosted on
# the agent
#   ports    - number of ports associated with the networks hosted on the
# agent
# dhcp_load_type = networks

# Allow auto scheduling networks to DHCP agent. It will schedule non-
# hosted
# networks to first DHCP agent which sends get_active_networks message to
# neutron server
# network_auto_schedule = True

# Allow auto scheduling routers to L3 agent. It will schedule non-hosted
# routers to first L3 agent which sends sync_routers message to neutron
# server
# router_auto_schedule = True

# Allow automatic rescheduling of routers from dead L3 agents with
# admin_state_up set to True to alive agents.
# allow_automatic_l3agent_failover = False

# Allow automatic removal of networks from dead DHCP agents with
# admin_state_up set to True.
# Networks could then be rescheduled if network_auto_schedule is True
# allow_automatic_dhcp_failover = True

# Number of DHCP agents scheduled to host a network. This enables
# redundant
# DHCP agents for configured networks.
# dhcp_agents_per_network = 1

# Enable services on agents with admin_state_up False.
# If this option is False, when admin_state_up of an agent is turned to
# False, services on it will be disabled. If this option is True, services
# on agents with admin_state_up False keep available and manual scheduling
# to such agents is available. Agents with admin_state_up False are not
# selected for automatic scheduling regardless of this option.
# enable_services_on_agents_with_admin_state_down = False

```

```

# ===== end of items for agent scheduler extension =====

# ===== items for l3 extension =====
# Enable high availability for virtual routers.
# l3_ha = False
#
# Maximum number of l3 agents which a HA router will be scheduled on. If
it
# is set to 0 the router will be scheduled on every agent.
# max_l3_agents_per_router = 3
#
# Minimum number of l3 agents which a HA router will be scheduled on. The
# default value is 2.
# min_l3_agents_per_router = 2
#
# CIDR of the administrative network if HA mode is enabled
# l3_ha_net_cidr = 169.254.192.0/18
# ===== end of items for l3 extension =====

# ===== items for metadata proxy configuration =====
# User (uid or name) running metadata proxy after its initialization
# (if empty: agent effective user)
# metadata_proxy_user =

# Group (gid or name) running metadata proxy after its initialization
# (if empty: agent effective group)
# metadata_proxy_group =

# Enable/Disable log watch by metadata proxy, it should be disabled when
# metadata_proxy_user/group is not allowed to read/write its log file and
# 'copytruncate' logrotate option must be used if logrotate is enabled on
# metadata proxy log files. Option default value is deduced from
# metadata_proxy_user: watch log is enabled if metadata_proxy_user is
agent
# effective user id/name.
# metadata_proxy_watch_log =

# Location of Metadata Proxy UNIX domain socket
# metadata_proxy_socket = $state_path/metadata_proxy
# ===== end of items for metadata proxy configuration =====

# ===== items for VLAN trunking networks =====
# Setting this flag to True will allow plugins that support it to
# create VLAN transparent networks. This flag has no effect for
# plugins that do not support VLAN transparent networks.
# vlan_transparent = False
# ===== end of items for VLAN trunking networks =====

# ===== WSGI parameters related to the API server =====
# Number of separate worker processes to spawn. The default, 0, runs the
# worker thread in the current process. Greater than 0 launches that
number of
# child processes as workers. The parent process manages them.
# api_workers = 0

```

```
# Number of separate RPC worker processes to spawn. The default, 0, runs
the
# worker thread in the current process. Greater than 0 launches that
number of
# child processes as RPC workers. The parent process manages them.
# This feature is experimental until issues are addressed and testing has
been
# enabled for various plugins for compatibility.
# rpc_workers = 0

# Timeout for client connections socket operations. If an
# incoming connection is idle for this number of seconds it
# will be closed. A value of '0' means wait forever. (integer
# value)
# client_socket_timeout = 900

# wsgi keepalive option. Determines if connections are allowed to be held
open
# by clients after a request is fulfilled. A value of False will ensure
that
# the socket connection will be explicitly closed once a response has been
# sent to the client.
# wsgi_keep_alive = True

# Sets the value of TCP_KEEPIDLE in seconds to use for each server socket
when
# starting API server. Not supported on OS X.
# tcp_keepidle = 600

# Number of seconds to keep retrying to listen
# retry_until_window = 30

# Number of backlog requests to configure the socket with.
# backlog = 4096

# Max header line to accommodate large tokens
# max_header_line = 16384

# Enable SSL on the API server
# use_ssl = False

# Certificate file to use when starting API server securely
# ssl_cert_file = /path/to/certfile

# Private key file to use when starting API server securely
# ssl_key_file = /path/to/keyfile

# CA certificate file to use when starting API server securely to
# verify connecting clients. This is an optional parameter only required
if
# API clients need to authenticate to the API server using SSL
certificates
# signed by a trusted CA
# ssl_ca_file = /path/to/cafile
# ===== end of WSGI parameters related to the API server =====
```

```

# ===== neutron nova interactions =====
# Send notification to nova when port status is active.
# notify_nova_on_port_status_changes = True

# Send notifications to nova when port data (fixed_ips/floatingips) change
# so nova can update it's cache.
# notify_nova_on_port_data_changes = True

# URL for connection to nova (Only supports one nova region currently).
# nova_url = http://127.0.0.1:8774/v2

# Name of nova region to use. Useful if keystone manages more than one
region
# nova_region_name =

# Username for connection to nova in admin context
# nova_admin_username =

# The uuid of the admin nova tenant
# nova_admin_tenant_id =

# The name of the admin nova tenant. If the uuid of the admin nova tenant
# is set, this is optional. Useful for cases where the uuid of the admin
# nova tenant is not available when configuration is being done.
# nova_admin_tenant_name =

# Password for connection to nova in admin context.
# nova_admin_password =

# Authorization URL for connection to nova in admin context.
# nova_admin_auth_url =

# CA file for novaclient to verify server certificates
# nova_ca_certificates_file =

# Boolean to control ignoring SSL errors on the nova url
# nova_api_insecure = False

# Number of seconds between sending events to nova if there are any events
to send
# send_events_interval = 2

# ===== end of neutron nova interactions =====

#
# Options defined in oslo.messaging
#

# Use durable queues in amqp. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
# amqp_durable_queues=false

# Auto-delete queues in amqp. (boolean value)
# amqp_auto_delete=false

# Size of RPC connection pool. (integer value)

```

```
# rpc_conn_pool_size=30

# Qpid broker hostname. (string value)
# qpid_hostname=localhost

# Qpid broker port. (integer value)
# qpid_port=5672

# Qpid HA cluster host:port pairs. (list value)
# qpid_hosts=$qpid_hostname:$qpid_port

# Username for Qpid connection. (string value)
# qpid_username=

# Password for Qpid connection. (string value)
# qpid_password=

# Space separated list of SASL mechanisms to use for auth.
# (string value)
# qpid_sasl_mechanisms=

# Seconds between connection keepalive heartbeats. (integer
# value)
# qpid_heartbeat=60

# Transport to use, either 'tcp' or 'ssl'. (string value)
# qpid_protocol=tcp

# Whether to disable the Nagle algorithm. (boolean value)
# qpid_tcp_nodelay=true

# The qpid topology version to use. Version 1 is what was
# originally used by impl_qpid. Version 2 includes some
# backwards-incompatible changes that allow broker federation
# to work. Users should update to version 2 when they are
# able to take everything down, as it requires a clean break.
# (integer value)
# qpid_topology_version=1

# SSL version to use (valid only if SSL enabled). valid values
# are TLSv1, SSLv23 and SSLv3. SSLv2 may be available on some
# distributions. (string value)
# kombu_ssl_version=

# SSL key file (valid only if SSL enabled). (string value)
# kombu_ssl_keyfile=

# SSL cert file (valid only if SSL enabled). (string value)
# kombu_ssl_certfile=

# SSL certification authority file (valid only if SSL
# enabled). (string value)
# kombu_ssl_ca_certs=

# How long to wait before reconnecting in response to an AMQP
# consumer cancel notification. (floating point value)
```

```

# kombu_reconnect_delay=1.0

# The RabbitMQ broker address where a single node is used.
# (string value)
# rabbit_host=localhost

# The RabbitMQ broker port where a single node is used.
# (integer value)
# rabbit_port=5672

# RabbitMQ HA cluster host:port pairs. (list value)
# rabbit_hosts=$rabbit_host:$rabbit_port

# Connect over SSL for RabbitMQ. (boolean value)
# rabbit_use_ssl=false

# The RabbitMQ userid. (string value)
# rabbit_userid=guest

# The RabbitMQ password. (string value)
# rabbit_password=guest

# the RabbitMQ login method (string value)
# rabbit_login_method=AMQPLAIN

# The RabbitMQ virtual host. (string value)
# rabbit_virtual_host=/

# How frequently to retry connecting with RabbitMQ. (integer
# value)
# rabbit_retry_interval=1

# How long to backoff for between retries when connecting to
# RabbitMQ. (integer value)
# rabbit_retry_backoff=2

# Maximum number of RabbitMQ connection retries. Default is 0
# (infinite retry count). (integer value)
# rabbit_max_retries=0

# Use HA queues in RabbitMQ (x-ha-policy: all). If you change
# this option, you must wipe the RabbitMQ database. (boolean
# value)
# rabbit_ha_queues=false

# If passed, use a fake RabbitMQ provider. (boolean value)
# fake_rabbit=false

# ZeroMQ bind address. Should be a wildcard (*), an ethernet
# interface, or IP. The "host" option should point or resolve
# to this address. (string value)
# rpc_zmq_bind_address=*

# MatchMaker driver. (string value)
#
rpc_zmq_matchmaker=oslo.messaging._drivers.matchmaker.MatchMakerLocalhost

```

```
# ZeroMQ receiver listening port. (integer value)
# rpc_zmq_port=9501

# Number of ZeroMQ contexts, defaults to 1. (integer value)
# rpc_zmq_contexts=1

# Maximum number of ingress messages to locally buffer per
# topic. Default is unlimited. (integer value)
# rpc_zmq_topic_backlog=

# Directory for holding IPC sockets. (string value)
# rpc_zmq_ipc_dir=/var/run/openstack

# Name of this node. Must be a valid hostname, FQDN, or IP
# address. Must match "host" option, if running Nova. (string
# value)
# rpc_zmq_host=oslo

# Seconds to wait before a cast expires (TTL). Only supported
# by impl_zmq. (integer value)
# rpc_cast_timeout=30

# Heartbeat frequency. (integer value)
# matchmaker_heartbeat_freq=300

# Heartbeat time-to-live. (integer value)
# matchmaker_heartbeat_ttl=600

# Size of RPC greenthread pool. (integer value)
# rpc_thread_pool_size=64

# Driver or drivers to handle sending notifications. (multi
# valued)
# notification_driver=

# AMQP topic used for OpenStack notifications. (list value)
# Deprecated group/name - [rpc_notifier2]/topics
# notification_topics=notifications

# Seconds to wait for a response from a call. (integer value)
# rpc_response_timeout=60

# A URL representing the messaging driver to use and its full
# configuration. If not set, we fall back to the rpc_backend
# option and driver specific configuration. (string value)
# transport_url=

# The messaging driver to use, defaults to rabbit. Other
# drivers include qpid and zmq. (string value)
# rpc_backend=rabbit

# The default exchange under which topics are scoped. May be
# overridden by an exchange name specified in the
# transport_url option. (string value)
# control_exchange=openstack
```



```

[matchmaker_redis]

#
# Options defined in oslo.messaging
#

# Host to locate redis. (string value)
# host=127.0.0.1

# Use this port to connect to redis host. (integer value)
# port=6379

# Password for Redis server (optional). (string value)
# password=

[matchmaker_ring]

#
# Options defined in oslo.messaging
#

# Matchmaker ring file (JSON). (string value)
# Deprecated group/name - [DEFAULT]/matchmaker_ringfile
# ringfile=/etc/oslo/matchmaker_ring.json

[quotas]
# Default driver to use for quota checks
# quota_driver = neutron.db.quota_db.DbQuotaDriver

# Resource name(s) that are supported in quota features
# quota_items = network,subnet,port

# Default number of resource allowed per tenant. A negative value means
# unlimited.
# default_quota = -1

# Number of networks allowed per tenant. A negative value means unlimited.
# quota_network = 10

# Number of subnets allowed per tenant. A negative value means unlimited.
# quota_subnet = 10

# Number of ports allowed per tenant. A negative value means unlimited.
# quota_port = 50

# Number of security groups allowed per tenant. A negative value means
# unlimited.
# quota_security_group = 10

# Number of security group rules allowed per tenant. A negative value
# means
# unlimited.
# quota_security_group_rule = 100

```

```
# Number of vips allowed per tenant. A negative value means unlimited.
# quota_vip = 10

# Number of pools allowed per tenant. A negative value means unlimited.
# quota_pool = 10

# Number of pool members allowed per tenant. A negative value means
unlimited.
# The default is unlimited because a member is not a real resource
consumer
# on Openstack. However, on back-end, a member is a resource consumer
# and that is the reason why quota is possible.
# quota_member = -1

# Number of health monitors allowed per tenant. A negative value means
# unlimited.
# The default is unlimited because a health monitor is not a real resource
# consumer on Openstack. However, on back-end, a member is a resource
consumer
# and that is the reason why quota is possible.
# quota_health_monitor = -1

# Number of loadbalancers allowed per tenant. A negative value means
unlimited.
# quota_loadbalancer = 10

# Number of listeners allowed per tenant. A negative value means
unlimited.
# quota_listener = -1

# Number of v2 health monitors allowed per tenant. A negative value means
# unlimited. These health monitors exist under the lbaas v2 API
# quota_healthmonitor = -1

# Number of routers allowed per tenant. A negative value means unlimited.
# quota_router = 10

# Number of floating IPs allowed per tenant. A negative value means
unlimited.
# quota_floatingip = 50

# Number of firewalls allowed per tenant. A negative value means
unlimited.
# quota_firewall = 1

# Number of firewall policies allowed per tenant. A negative value means
# unlimited.
# quota_firewall_policy = 1

# Number of firewall rules allowed per tenant. A negative value means
# unlimited.
# quota_firewall_rule = 100

[agent]
# Use "sudo neutron-rootwrap /etc/neutron/rootwrap.conf" to use the real
```

```

# root filter facility.
# Change to "sudo" to skip the filtering and just run the command directly
# root_helper = sudo

# Set to true to add comments to generated iptables rules that describe
# each rule's purpose. (System must support the iptables comments module.)
# comment_iptables_rules = True

# Root helper daemon application to use when possible.
# root_helper_daemon =

# Use the root helper when listing the namespaces on a system. This may
# not
# be required depending on the security configuration. If the root helper
# is
# not required, set this to False for a performance improvement.
# use_helper_for_ns_read = True

# The interval to check external processes for failure in seconds
# (0=disabled)
# check_child_processes_interval = 60

# Action to take when an external process spawned by an agent dies
# Values:
#   respawn - Respawns the external process
#   exit - Exits the agent
# check_child_processes_action = respawn

# ===== items for agent management extension =====
# seconds between nodes reporting state to server; should be less than
# agent_down_time, best if it is half or less than agent_down_time
# report_interval = 30

# ===== end of items for agent management extension =====

[keystone_authtoken]
auth_uri = http://127.0.0.1:35357/v2.0/
identity_uri = http://127.0.0.1:5000
admin_tenant_name = %SERVICE_TENANT_NAME%
admin_user = %SERVICE_USER%
admin_password = %SERVICE_PASSWORD%

[database]
# This line MUST be changed to actually run the plugin.
# Example:
# connection = mysql://root:pass@127.0.0.1:3306/neutron
# Replace 127.0.0.1 above with the IP address of the database used by the
# main neutron server. (Leave it as is if the database runs on this host.)
# connection = sqlite://
# NOTE: In deployment the [database] section and its connection attribute
# may
# be set in the corresponding core plugin '.ini' file. However, it is
# suggested
# to put the [database] section and its connection attribute in this
# configuration file.

```

```
# Database engine for which script will be generated when using offline
# migration
# engine =

# The SQLAlchemy connection string used to connect to the slave database
# slave_connection =

# Database reconnection retry times - in event connectivity is lost
# set to -1 implies an infinite retry count
# max_retries = 10

# Database reconnection interval in seconds - if the initial connection to
the
# database fails
# retry_interval = 10

# Minimum number of SQL connections to keep open in a pool
# min_pool_size = 1

# Maximum number of SQL connections to keep open in a pool
# max_pool_size = 10

# Timeout in seconds before idle sql connections are reaped
# idle_timeout = 3600

# If set, use this value for max_overflow with sqlalchemy
# max_overflow = 20

# Verbosity of SQL debugging information. 0=None, 100=Everything
# connection_debug = 0

# Add python stack traces to SQL as comment strings
# connection_trace = False

# If set, use this value for pool_timeout with sqlalchemy
# pool_timeout = 10

[nova]
# Name of the plugin to load
# auth_plugin =

# Config Section from which to load plugin specific options
# auth_section =

# PEM encoded Certificate Authority to use when verifying HTTPS
connections.
# cafile =

# PEM encoded client certificate cert file
# certfile =

# Verify HTTPS connections.
# insecure = False

# PEM encoded client certificate key file
# keyfile =
```

```

# Name of nova region to use. Useful if keystone manages more than one
region.
# region_name =

# Timeout value for http requests
# timeout =

[oslo_concurrency]

# Directory to use for lock files. For security, the specified directory
should
# only be writable by the user running the processes that need locking.
# Defaults to environment variable OSLO_LOCK_PATH. If external locks are
used,
# a lock path must be set.
lock_path = $state_path/lock

# Enables or disables inter-process locks.
# disable_process_locking = False

[oslo_policy]

# The JSON file that defines policies.
# policy_file = policy.json

# Default rule. Enforced when a requested rule is not found.
# policy_default_rule = default

# Directories where policy configuration files are stored.
# They can be relative to any directory in the search path defined by the
# config_dir option, or absolute paths. The file defined by policy_file
# must exist for these directories to be searched. Missing or empty
# directories are ignored.
# policy_dirs = policy.d

[oslo_messaging_amqp]

#
# From oslo.messaging
#

# Address prefix used when sending to a specific server (string value)
# Deprecated group/name - [amqp1]/server_request_prefix
# server_request_prefix = exclusive

# Address prefix used when broadcasting to all servers (string value)
# Deprecated group/name - [amqp1]/broadcast_prefix
# broadcast_prefix = broadcast

# Address prefix when sending to any server in group (string value)
# Deprecated group/name - [amqp1]/group_request_prefix
# group_request_prefix = unicast

# Name for the AMQP container (string value)
# Deprecated group/name - [amqp1]/container_name

```

```

# container_name =

# Timeout for inactive connections (in seconds) (integer value)
# Deprecated group/name - [amqp1]/idle_timeout
# idle_timeout = 0

# Debug: dump AMQP frames to stdout (boolean value)
# Deprecated group/name - [amqp1]/trace
# trace = false

# CA certificate PEM file for verifying server certificate (string value)
# Deprecated group/name - [amqp1]/ssl_ca_file
# ssl_ca_file =

# Identifying certificate PEM file to present to clients (string value)
# Deprecated group/name - [amqp1]/ssl_cert_file
# ssl_cert_file =

# Private key PEM file used to sign cert_file certificate (string value)
# Deprecated group/name - [amqp1]/ssl_key_file
# ssl_key_file =

# Password for decrypting ssl_key_file (if encrypted) (string value)
# Deprecated group/name - [amqp1]/ssl_key_password
# ssl_key_password =

# Accept clients using either SSL or plain TCP (boolean value)
# Deprecated group/name - [amqp1]/allow_insecure_clients
# allow_insecure_clients = false

[oslo_messaging_qpid]

#
# From oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
# amqp_durable_queues = false

# Auto-delete queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/amqp_auto_delete
# amqp_auto_delete = false

# Size of RPC connection pool. (integer value)
# Deprecated group/name - [DEFAULT]/rpc_conn_pool_size
# rpc_conn_pool_size = 30

# Qpid broker hostname. (string value)
# Deprecated group/name - [DEFAULT]/qpid_hostname
# qpid_hostname = localhost

# Qpid broker port. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_port
# qpid_port = 5672

```

```

# Qpid HA cluster host:port pairs. (list value)
# Deprecated group/name - [DEFAULT]/qpid_hosts
# qpid_hosts = $qpid_hostname:$qpid_port

# Username for Qpid connection. (string value)
# Deprecated group/name - [DEFAULT]/qpid_username
# qpid_username =

# Password for Qpid connection. (string value)
# Deprecated group/name - [DEFAULT]/qpid_password
# qpid_password =

# Space separated list of SASL mechanisms to use for auth. (string value)
# Deprecated group/name - [DEFAULT]/qpid_sasl_mechanisms
# qpid_sasl_mechanisms =

# Seconds between connection keepalive heartbeats. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_heartbeat
# qpid_heartbeat = 60

# Transport to use, either 'tcp' or 'ssl'. (string value)
# Deprecated group/name - [DEFAULT]/qpid_protocol
# qpid_protocol = tcp

# Whether to disable the Nagle algorithm. (boolean value)
# Deprecated group/name - [DEFAULT]/qpid_tcp_nodelay
# qpid_tcp_nodelay = true

# The number of prefetched messages held by receiver. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_receiver_capacity
# qpid_receiver_capacity = 1

# The qpid topology version to use. Version 1 is what was originally used
by
# impl_qpid. Version 2 includes some backwards-incompatible changes that
allow
# broker federation to work. Users should update to version 2 when they
are
# able to take everything down, as it requires a clean break. (integer
value)
# Deprecated group/name - [DEFAULT]/qpid_topology_version
# qpid_topology_version = 1

[oslo_messaging_rabbit]

#
# From oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
# amqp_durable_queues = false

# Auto-delete queues in AMQP. (boolean value)

```

```
# Deprecated group/name - [DEFAULT]/amqp_auto_delete
# amqp_auto_delete = false

# Size of RPC connection pool. (integer value)
# Deprecated group/name - [DEFAULT]/rpc_conn_pool_size
# rpc_conn_pool_size = 30

# SSL version to use (valid only if SSL enabled). Valid values are TLSv1
and
# SSLv23, SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some
# distributions. (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_version
# kombu_ssl_version =

# SSL key file (valid only if SSL enabled). (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_keyfile
# kombu_ssl_keyfile =

# SSL cert file (valid only if SSL enabled). (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_certfile
# kombu_ssl_certfile =

# SSL certification authority file (valid only if SSL enabled). (string
value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_ca_certs
# kombu_ssl_ca_certs =

# How long to wait before reconnecting in response to an AMQP consumer
cancel
# notification. (floating point value)
# Deprecated group/name - [DEFAULT]/kombu_reconnect_delay
# kombu_reconnect_delay = 1.0

# The RabbitMQ broker address where a single node is used. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_host
# rabbit_host = localhost

# The RabbitMQ broker port where a single node is used. (integer value)
# Deprecated group/name - [DEFAULT]/rabbit_port
# rabbit_port = 5672

# RabbitMQ HA cluster host:port pairs. (list value)
# Deprecated group/name - [DEFAULT]/rabbit_hosts
# rabbit_hosts = $rabbit_host:$rabbit_port

# Connect over SSL for RabbitMQ. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_use_ssl
# rabbit_use_ssl = false

# The RabbitMQ userid. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_userid
# rabbit_userid = guest

# The RabbitMQ password. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_password
# rabbit_password = guest
```



```

# The RabbitMQ login method. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_login_method
# rabbit_login_method = AMQPPLAIN

# The RabbitMQ virtual host. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_virtual_host
# rabbit_virtual_host = /

# How frequently to retry connecting with RabbitMQ. (integer value)
# rabbit_retry_interval = 1

# How long to backoff for between retries when connecting to RabbitMQ.
(integer
# value)
# Deprecated group/name - [DEFAULT]/rabbit_retry_backoff
# rabbit_retry_backoff = 2

# Maximum number of RabbitMQ connection retries. Default is 0 (infinite
retry
# count). (integer value)
# Deprecated group/name - [DEFAULT]/rabbit_max_retries
# rabbit_max_retries = 0

# Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option,
you
# must wipe the RabbitMQ database. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_ha_queues
# rabbit_ha_queues = false

# Deprecated, use rpc_backend=kombu+memory or rpc_backend=fake (boolean
value)
# Deprecated group/name - [DEFAULT]/fake_rabbit
# fake_rabbit = false

```

9.3.2. api-paste.ini

Use the `api-paste.ini` to configure the OpenStack Networking API.

```

[composite:neutron]
use = egg:Paste#urlmap
/: neutronversions
/v2.0: neutronapi_v2_0

[composite:neutronapi_v2_0]
use = call:neutron.auth:pipeline_factory
noauth = request_id catch_errors extensions neutronapiapp_v2_0
keystone = request_id catch_errors authtoken keystonecontext extensions
neutronapiapp_v2_0

[filter:request_id]
paste.filter_factory = oslo.middleware:RequestId.factory

```

```
[filter:catch_errors]
paste.filter_factory = oslo.middleware:CatchErrors.factory

[filter:keystonecontext]
paste.filter_factory = neutron.auth:NeutronKeystoneContext.factory

[filter:authtoken]
paste.filter_factory = keystone.middleware.auth_token:filter_factory

[filter:extensions]
paste.filter_factory =
neutron.api.extensions:plugin_aware_extension_middleware_factory

[app:neutronversions]
paste.app_factory = neutron.api.versions:Versions.factory

[app:neutronapiapp_v2_0]
paste.app_factory = neutron.api.v2.router:APIRouter.factory
```

9.3.3. policy.json

Use the **policy.json** file to define additional access controls that apply to the OpenStack Networking service.

```
{
  "context_is_admin": "role:admin",
  "admin_or_owner": "rule:context_is_admin or tenant_id:%(tenant_id)s",
  "context_is_advsvc": "role:advsvc",
  "admin_or_network_owner": "rule:context_is_admin or tenant_id:%
(network:tenant_id)s",
  "admin_only": "rule:context_is_admin",
  "regular_user": "",
  "shared": "field:networks:shared=True",
  "shared_firewalls": "field:firewalls:shared=True",
  "shared_firewall_policies": "field:firewall_policies:shared=True",
  "shared_subnetpools": "field:subnetpools:shared=True",
  "external": "field:networks:router:external=True",
  "default": "rule:admin_or_owner",

  "create_subnet": "rule:admin_or_network_owner",
  "get_subnet": "rule:admin_or_owner or rule:shared",
  "update_subnet": "rule:admin_or_network_owner",
  "delete_subnet": "rule:admin_or_network_owner",

  "create_subnetpool": "",
  "create_subnetpool:shared": "rule:admin_only",
  "get_subnetpool": "rule:admin_or_owner or rule:shared_subnetpools",
  "update_subnetpool": "rule:admin_or_owner",
  "delete_subnetpool": "rule:admin_or_owner",

  "create_network": "",
```

```

    "get_network": "rule:admin_or_owner or rule:shared or rule:external or
rule:context_is_advsvc",
    "get_network:router:external": "rule:regular_user",
    "get_network:segments": "rule:admin_only",
    "get_network:provider:network_type": "rule:admin_only",
    "get_network:provider:physical_network": "rule:admin_only",
    "get_network:provider:segmentation_id": "rule:admin_only",
    "get_network:queue_id": "rule:admin_only",
    "create_network:shared": "rule:admin_only",
    "create_network:router:external": "rule:admin_only",
    "create_network:segments": "rule:admin_only",
    "create_network:provider:network_type": "rule:admin_only",
    "create_network:provider:physical_network": "rule:admin_only",
    "create_network:provider:segmentation_id": "rule:admin_only",
    "update_network": "rule:admin_or_owner",
    "update_network:segments": "rule:admin_only",
    "update_network:shared": "rule:admin_only",
    "update_network:provider:network_type": "rule:admin_only",
    "update_network:provider:physical_network": "rule:admin_only",
    "update_network:provider:segmentation_id": "rule:admin_only",
    "update_network:router:external": "rule:admin_only",
    "delete_network": "rule:admin_or_owner",

    "create_port": "",
    "create_port:mac_address": "rule:admin_or_network_owner or
rule:context_is_advsvc",
    "create_port:fixed_ips": "rule:admin_or_network_owner or
rule:context_is_advsvc",
    "create_port:port_security_enabled": "rule:admin_or_network_owner or
rule:context_is_advsvc",
    "create_port:binding:host_id": "rule:admin_only",
    "create_port:binding:profile": "rule:admin_only",
    "create_port:mac_learning_enabled": "rule:admin_or_network_owner or
rule:context_is_advsvc",
    "create_port:allowed_address_pairs": "rule:admin_or_network_owner",
    "get_port": "rule:admin_or_owner or rule:context_is_advsvc",
    "get_port:queue_id": "rule:admin_only",
    "get_port:binding:vif_type": "rule:admin_only",
    "get_port:binding:vif_details": "rule:admin_only",
    "get_port:binding:host_id": "rule:admin_only",
    "get_port:binding:profile": "rule:admin_only",
    "update_port": "rule:admin_or_owner or rule:context_is_advsvc",
    "update_port:mac_address": "rule:admin_only or
rule:context_is_advsvc",
    "update_port:fixed_ips": "rule:admin_or_network_owner or
rule:context_is_advsvc",
    "update_port:port_security_enabled": "rule:admin_or_network_owner or
rule:context_is_advsvc",
    "update_port:binding:host_id": "rule:admin_only",
    "update_port:binding:profile": "rule:admin_only",
    "update_port:mac_learning_enabled": "rule:admin_or_network_owner or
rule:context_is_advsvc",
    "update_port:allowed_address_pairs": "rule:admin_or_network_owner",
    "delete_port": "rule:admin_or_owner or rule:context_is_advsvc",

    "get_router:ha": "rule:admin_only",

```

```

"create_router": "rule:regular_user",
"create_router:external_gateway_info:enable_snat": "rule:admin_only",
"create_router:distributed": "rule:admin_only",
"create_router:ha": "rule:admin_only",
"get_router": "rule:admin_or_owner",
"get_router:distributed": "rule:admin_only",
"update_router:external_gateway_info:enable_snat": "rule:admin_only",
"update_router:distributed": "rule:admin_only",
"update_router:ha": "rule:admin_only",
"delete_router": "rule:admin_or_owner",

"add_router_interface": "rule:admin_or_owner",
"remove_router_interface": "rule:admin_or_owner",

"create_router:external_gateway_info:external_fixed_ips":
"rule:admin_only",
  "update_router:external_gateway_info:external_fixed_ips":
"rule:admin_only",

"create_firewall": "",
"get_firewall": "rule:admin_or_owner",
"create_firewall:shared": "rule:admin_only",
"get_firewall:shared": "rule:admin_only",
"update_firewall": "rule:admin_or_owner",
"update_firewall:shared": "rule:admin_only",
"delete_firewall": "rule:admin_or_owner",

"create_firewall_policy": "",
"get_firewall_policy": "rule:admin_or_owner or
rule:shared_firewall_policies",
"create_firewall_policy:shared": "rule:admin_or_owner",
"update_firewall_policy": "rule:admin_or_owner",
"delete_firewall_policy": "rule:admin_or_owner",

"create_firewall_rule": "",
"get_firewall_rule": "rule:admin_or_owner or rule:shared_firewalls",
"update_firewall_rule": "rule:admin_or_owner",
"delete_firewall_rule": "rule:admin_or_owner",

"create_qos_queue": "rule:admin_only",
"get_qos_queue": "rule:admin_only",

"update_agent": "rule:admin_only",
"delete_agent": "rule:admin_only",
"get_agent": "rule:admin_only",

"create_dhcp-network": "rule:admin_only",
"delete_dhcp-network": "rule:admin_only",
"get_dhcp-networks": "rule:admin_only",
"create_l3-router": "rule:admin_only",
"delete_l3-router": "rule:admin_only",
"get_l3-routers": "rule:admin_only",
"get_dhcp-agents": "rule:admin_only",
"get_l3-agents": "rule:admin_only",
"get_loadbalancer-agent": "rule:admin_only",
"get_loadbalancer-pools": "rule:admin_only",

```

```

"get_agent-loadbalancers": "rule:admin_only",
"get_loadbalancer-hosting-agent": "rule:admin_only",

"create_floatingip": "rule:regular_user",
"create_floatingip:floating_ip_address": "rule:admin_only",
"update_floatingip": "rule:admin_or_owner",
"delete_floatingip": "rule:admin_or_owner",
"get_floatingip": "rule:admin_or_owner",

"create_network_profile": "rule:admin_only",
"update_network_profile": "rule:admin_only",
"delete_network_profile": "rule:admin_only",
"get_network_profiles": "",
"get_network_profile": "",
"update_policy_profiles": "rule:admin_only",
"get_policy_profiles": "",
"get_policy_profile": "",

"create_metering_label": "rule:admin_only",
"delete_metering_label": "rule:admin_only",
"get_metering_label": "rule:admin_only",

"create_metering_label_rule": "rule:admin_only",
"delete_metering_label_rule": "rule:admin_only",
"get_metering_label_rule": "rule:admin_only",

"get_service_provider": "rule:regular_user",
"get_lsn": "rule:admin_only",
"create_lsn": "rule:admin_only"
}

```

9.3.4. rootwrap.conf

Use the `rootwrap.conf` file to define configuration values used by the `rootwrap` script when the OpenStack Networking service must escalate its privileges to those of the root user.

```

# Configuration for neutron-rootwrap
# This file should be owned by (and only-writeable by) the root user

[DEFAULT]
# List of directories to load filter definitions from (separated by ',').
# These directories MUST all be only writeable by root !
filters_path=/etc/neutron/rootwrap.d,/usr/share/neutron/rootwrap

# List of directories to search executables in, in case filters do not
# explicitly specify a full path (separated by ',')
# If not specified, defaults to system PATH environment variable.
# These directories MUST all be only writeable by root !
exec_dirs=/sbin,/usr/sbin,/bin,/usr/bin

# Enable logging to syslog
# Default value is False

```

```

use_syslog=False

# Which syslog facility to use.
# Valid values include auth, authpriv, syslog, local0, local1...
# Default value is 'syslog'
syslog_log_facility=syslog

# Which messages to log.
# INFO means log all usage
# ERROR means only log unsuccessful attempts
syslog_log_level=ERROR

[xenapi]
# XenAPI configuration is only required by the L2 agent if it is to
# target a XenServer/XCP compute host's dom0.
xenapi_connection_url=<None>
xenapi_connection_username=root
xenapi_connection_password=<None>

```

9.3.5. Configuration files for plug-in agents

Each plug-in agent that runs on an OpenStack Networking node, to perform local networking configuration for the node's VMs and networking services, has its own configuration file.

9.3.5.1. dhcp_agent.ini

```

[DEFAULT]
# Show debugging output in log (sets DEBUG log level output)
# debug = False

# The DHCP agent will resync its state with Neutron to recover from any
# transient notification or rpc errors. The interval is number of
# seconds between attempts.
# resync_interval = 5

# The DHCP agent requires an interface driver be set. Choose the one that
# best
# matches your plugin.
# interface_driver =

# Example of interface_driver option for OVS based plugins(OVS, Ryu, NEC,
# NVP,
# BigSwitch/Floodlight)
# interface_driver = neutron.agent.linux.interface.OVSInterfaceDriver

# Name of Open vSwitch bridge to use
# ovs_integration_bridge = br-int

# Use veth for an OVS interface or not.
# Support kernels with limited namespace support
# (e.g. RHEL 6.5) so long as ovs_use_veth is set to True.
# ovs_use_veth = False

```

```

# Example of interface_driver option for LinuxBridge
# interface_driver = neutron.agent.linux.interface.BridgeInterfaceDriver

# The agent can use other DHCP drivers. Dnsmasq is the simplest and
requires
# no additional setup of the DHCP server.
# dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq

# Allow overlapping IP (Must have kernel build with CONFIG_NET_NS=y and
# iproute2 package that supports namespaces). This option is deprecated
and
# will be removed in a future release, at which point the old behavior of
# use_namespaces = True will be enforced.
# use_namespaces = True

# The DHCP server can assist with providing metadata support on isolated
# networks. Setting this value to True will cause the DHCP server to
append
# specific host routes to the DHCP request. The metadata service will only
# be activated when the subnet does not contain any router port. The guest
# instance must be configured to request host routes via DHCP (Option
121).
# enable_isolated_metadata = False

# Allows for serving metadata requests coming from a dedicated metadata
# access network whose cidr is 169.254.169.254/16 (or larger prefix), and
# is connected to a Neutron router from which the VMs send metadata
# request. In this case DHCP Option 121 will not be injected in VMs, as
# they will be able to reach 169.254.169.254 through a router.
# This option requires enable_isolated_metadata = True
# enable_metadata_network = False

# Number of threads to use during sync process. Should not exceed
connection
# pool size configured on server.
# num_sync_threads = 4

# Location to store DHCP server config files
# dhcp_confs = $state_path/dhcp

# Domain to use for building the hostnames
# dhcp_domain = openstacklocal

# Override the default dnsmasq settings with this file
# dnsmasq_config_file =

# Comma-separated list of DNS servers which will be used by dnsmasq
# as forwarders.
# dnsmasq_dns_servers =

# Limit number of leases to prevent a denial-of-service.
# dnsmasq_lease_max = 16777216

# Location to DHCP lease relay UNIX domain socket
# dhcp_lease_relay_socket = $state_path/dhcp/lease_relay

```

```
# Use broadcast in DHCP replies
# dhcp_broadcast_reply = False

# dhcp_delete_namespaces, which is false by default, can be set to True if
# namespaces can be deleted cleanly on the host running the dhcp agent.
# Do not enable this until you understand the problem with the Linux
# iproute
# utility mentioned in https://bugs.launchpad.net/neutron/+bug/1052535 and
# you are sure that your version of iproute does not suffer from the
# problem.
# If True, namespaces will be deleted when a dhcp server is disabled.
# dhcp_delete_namespaces = False

# Timeout for ovs-vsctl commands.
# If the timeout expires, ovs commands will fail with ALARMCLOCK error.
# ovs_vsctl_timeout = 10
```

9.3.5.2. l3_agent.ini

```
[DEFAULT]
# Show debugging output in log (sets DEBUG log level output)
# debug = False

# L3 requires that an interface driver be set. Choose the one that best
# matches your plugin.
# interface_driver =

# Example of interface_driver option for OVS based plugins (OVS, Ryu, NEC)
# that supports L3 agent
# interface_driver = neutron.agent.linux.interface.OVSInterfaceDriver

# Use veth for an OVS interface or not.
# Support kernels with limited namespace support
# (e.g. RHEL 6.5) so long as ovs_use_veth is set to True.
# ovs_use_veth = False

# Example of interface_driver option for LinuxBridge
# interface_driver = neutron.agent.linux.interface.BridgeInterfaceDriver

# Allow overlapping IP (Must have kernel build with CONFIG_NET_NS=y and
# iproute2 package that supports namespaces). This option is deprecated
# and
# will be removed in a future release, at which point the old behavior of
# use_namespaces = True will be enforced.
# use_namespaces = True

# If use_namespaces is set as False then the agent can only configure one
# router.

# This is done by setting the specific router_id.
# router_id =
```



```

# When external_network_bridge is set, each L3 agent can be associated
# with no more than one external network. This value should be set to the
# UUID
# of that external network. To allow L3 agent support multiple external
# networks, both the external_network_bridge and
# gateway_external_network_id
# must be left empty.
# gateway_external_network_id =

# With IPv6, the network used for the external gateway does not need
# to have an associated subnet, since the automatically assigned
# link-local address (LLA) can be used. However, an IPv6 gateway address
# is needed for use as the next-hop for the default route. If no IPv6
# gateway address is configured here, (and only then) the neutron router
# will be configured to get its default route from router advertisements
# (RAs)
# from the upstream router; in which case the upstream router must also be
# configured to send these RAs.
# The ipv6_gateway, when configured, should be the LLA of the interface
# on the upstream router. If a next-hop using a global unique address
# (GUA)
# is desired, it needs to be done via a subnet allocated to the network
# and not through this parameter.
# ipv6_gateway =

# Indicates that this L3 agent should also handle routers that do not have
# an external network gateway configured. This option should be True only
# for a single agent in a Neutron deployment, and may be False for all
# agents
# if all routers must have an external network gateway
# handle_internal_only_routers = True

# Name of bridge used for external network traffic. This should be set to
# empty value for the linux bridge. when this parameter is set, each L3
# agent
# can be associated with no more than one external network.
# external_network_bridge = br-ex

# TCP Port used by Neutron metadata server
# metadata_port = 9697

# Send this many gratuitous ARPs for HA setup. Set it below or equal to 0
# to disable this feature.
# send_arp_for_ha = 3

# seconds between re-sync routers' data if needed
# periodic_interval = 40

# seconds to start to sync routers' data after
# starting agent
# periodic_fuzzy_delay = 5

# enable_metadata_proxy, which is true by default, can be set to False
# if the Nova metadata server is not available
# enable_metadata_proxy = True

```

```

# Iptables mangle mark used to mark metadata valid requests
# metadata_access_mark = 0x1

# Iptables mangle mark used to mark ingress from external network
# external_ingress_mark = 0x2

# router_delete_namespaces, which is false by default, can be set to True
if
# namespaces can be deleted cleanly on the host running the L3 agent.
# Do not enable this until you understand the problem with the Linux
iproute
# utility mentioned in https://bugs.launchpad.net/neutron/+bug/1052535 and
# you are sure that your version of iproute does not suffer from the
problem.
# If True, namespaces will be deleted when a router is destroyed.
# router_delete_namespaces = False

# Timeout for ovs-vsctl commands.
# If the timeout expires, ovs commands will fail with ALARMCLOCK error.
# ovs_vsctl_timeout = 10

# The working mode for the agent. Allowed values are:
# - legacy: this preserves the existing behavior where the L3 agent is
#   deployed on a centralized networking node to provide L3 services
#   like DNAT, and SNAT. Use this mode if you do not want to adopt DVR.
# - dvr: this mode enables DVR functionality, and must be used for an L3
#   agent that runs on a compute host.
# - dvr_snat: this enables centralized SNAT support in conjunction with
#   DVR. This mode must be used for an L3 agent running on a centralized
#   node (or in single-host deployments, e.g. devstack).
# agent_mode = legacy

# Location to store keepalived and all HA configurations
# ha_confs_path = $state_path/ha_confs

# VRRP authentication type AH/PASS
# ha_vrrp_auth_type = PASS

# VRRP authentication password
# ha_vrrp_auth_password =

# The advertisement interval in seconds
# ha_vrrp_advert_int = 2

```

9.3.5.3. metadata_agent.ini

```

[DEFAULT]
# Show debugging output in log (sets DEBUG log level output)
# debug = True

# The Neutron user information for accessing the Neutron API.

```

```

auth_url = http://localhost:5000/v2.0
auth_region = RegionOne
# Turn off verification of the certificate for ssl
# auth_insecure = False
# Certificate Authority public key (CA cert) file for ssl
# auth_ca_cert =
admin_tenant_name = %SERVICE_TENANT_NAME%
admin_user = %SERVICE_USER%
admin_password = %SERVICE_PASSWORD%

# Network service endpoint type to pull from the keystone catalog
# endpoint_type = adminURL

# IP address used by Nova metadata server
# nova_metadata_ip = 127.0.0.1

# TCP Port used by Nova metadata server
# nova_metadata_port = 8775

# Which protocol to use for requests to Nova metadata server, http or
https
# nova_metadata_protocol = http

# Whether insecure SSL connection should be accepted for Nova metadata
server
# requests
# nova_metadata_insecure = False

# Client certificate for nova api, needed when nova api requires client
# certificates
# nova_client_cert =

# Private key for nova client certificate
# nova_client_priv_key =

# When proxying metadata requests, Neutron signs the Instance-ID header
with a
# shared secret to prevent spoofing. You may select any string for a
secret,
# but it must match here and in the configuration used by the Nova
Metadata
# Server. NOTE: Nova uses the same config key, but in [neutron] section.
# metadata_proxy_shared_secret =

# Location of Metadata Proxy UNIX domain socket
# metadata_proxy_socket = $state_path/metadata_proxy

# Metadata Proxy UNIX domain socket mode, 3 values allowed:
# 'deduce': deduce mode from metadata_proxy_user/group values,
# 'user': set metadata proxy socket mode to 0o644, to use when
# metadata_proxy_user is agent effective user or root,
# 'group': set metadata proxy socket mode to 0o664, to use when
# metadata_proxy_group is agent effective group,
# 'all': set metadata proxy socket mode to 0o666, to use otherwise.
# metadata_proxy_socket_mode = deduce

```

```
# Number of separate worker processes for metadata server. Defaults to
# half the number of CPU cores
# metadata_workers =

# Number of backlog requests to configure the metadata server socket with
# metadata_backlog = 4096

# URL to connect to the cache backend.
# default_ttl=0 parameter will cause cache entries to never expire.
# Otherwise default_ttl specifies time in seconds a cache entry is valid
# for.
# No cache is used in case no value is passed.
# cache_url = memory:///default_ttl=5
```

9.4. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR OPENSTACK NETWORKING

Table 9.89. New options

Option = default value	(Type) Help string
[DEFAULT] bgp_drscheduler_driver = neutron.services.bgp.scheduler.bgp_dragent_scheduler.ChanceScheduler	(StrOpt) Driver used for scheduling BGP speakers to BGP DrAgent
[DEFAULT] default_availability_zones =	(ListOpt) Default value of availability zone hints. The availability zone aware schedulers use this when the resources availability_zone_hints is empty. Multiple availability zones can be specified by a comma separated string. This value can be empty. In this case, even if availability_zone_hints for a resource is empty, availability zone is considered for high availability while scheduling the resource.
[DEFAULT] dnsmasq_local_resolv = False	(BoolOpt) Enables the dnsmasq service to provide name resolution for instances via DNS resolvers on the host running the DHCP agent. Effectively removes the '--no-resolv' option from the dnsmasq process arguments. Adding custom DNS resolvers to the 'dnsmasq_dns_servers' option disables this feature.
[DEFAULT] external_dns_driver = None	(StrOpt) Driver for external DNS integration.

Option = default value	(Type) Help string
[DEFAULT] global_physnet_mtu = 1500	(IntOpt) MTU of the underlying physical network. Neutron uses this value to calculate MTU for all virtual network components. For flat and VLAN networks, neutron uses this value without modification. For overlay networks such as VXLAN, neutron automatically subtracts the overlay protocol overhead from this value. Defaults to 1500, the standard value for Ethernet.
[DEFAULT] ipv6_pd_enabled = False	(BoolOpt) Enables IPv6 Prefix Delegation for automatic subnet CIDR allocation. Set to True to enable IPv6 Prefix Delegation for subnet allocation in a PD-capable environment. Users making subnet creation requests for IPv6 subnets without providing a CIDR or subnetpool ID will be given a CIDR via the Prefix Delegation mechanism. Note that enabling PD will override the behavior of the default IPv6 subnetpool.
[DEFAULT] max_rtr_adv_interval = 100	(IntOpt) MaxRtrAdvInterval setting for radvd.conf
[DEFAULT] min_rtr_adv_interval = 30	(IntOpt) MinRtrAdvInterval setting for radvd.conf
[DEFAULT] rpc_state_report_workers = 1	(IntOpt) Number of RPC worker processes dedicated to state reports queue
[DEFAULT] web_framework = legacy	(StrOpt) This will choose the web framework in which to run the Neutron API server. 'pecan' is a new experiemental rewrite of the API server.
[DEFAULT] wsgi_default_pool_size = 100	(IntOpt) Size of the pool of greenthreads used by wsgi
[DEFAULT] wsgi_log_format = %(client_ip)s "%(request_line)s" status: %(status_code)s len: %(body_length)s time: %(wall_seconds).7f	(StrOpt) A python format string that is used as the template to generate log lines. The following values can beformatted into it: client_ip, date_time, request_line, status_code, body_length, wall_seconds.
[AGENT] availability_zone = nova	(StrOpt) Availability zone of this node
[BGP] bgp_router_id = None	(StrOpt) 32-bit BGP identifier, typically an IPv4 address owned by the system running the BGP DrAgent.
[BGP] bgp_speaker_driver = None	(StrOpt) BGP speaker driver class to be instantiated.

Option = default value	(Type) Help string
[OVS] ovsdb_connection = tcp:127.0.0.1:6640	(StrOpt) The connection string for the native OVSDb backend. Requires the native ovsdb_interface to be enabled.
[OVS] vhostuser_socket_dir = /var/run/openvswitch	(StrOpt) OVS vhost-user socket directory.
[QOS] kernel_hz = 250	(IntOpt) Value of host kernel tick rate (hz) for calculating minimum burst value in bandwidth limit rules for a port with QoS. See kernel configuration file for HZ value and tc-tbf manual for more information.
[QOS] tbf_latency = 50	(IntOpt) Value of latency (ms) for calculating size of queue for a port with QoS. See tc-tbf manual for more information.
[VXLAN] arp_responder = False	(BoolOpt) Enable local ARP responder which provides local responses instead of performing ARP broadcast into the overlay. Enabling local ARP responder is not fully compatible with the allowed-address-pairs extension.
[designate] admin_auth_url = None	(StrOpt) Authorization URL for connecting to designate in admin context
[designate] admin_password = None	(StrOpt) Password for connecting to designate in admin context
[designate] admin_tenant_id = None	(StrOpt) Tenant id for connecting to designate in admin context
[designate] admin_tenant_name = None	(StrOpt) Tenant name for connecting to designate in admin context
[designate] admin_username = None	(StrOpt) Username for connecting to designate in admin context
[designate] allow_reverse_dns_lookup = True	(BoolOpt) Allow the creation of PTR records
[designate] ipv4_ptr_zone_prefix_size = 24	(IntOpt) Number of bits in an ipv4 PTR zone that will be considered network prefix. It has to align to byte boundary. Minimum value is 8. Maximum value is 24. As a consequence, range of values is 8, 16 and 24

Option = default value	(Type) Help string
[designate] ipv6_ptr_zone_prefix_size = 120	(IntOpt) Number of bits in an ipv6 PTR zone that will be considered network prefix. It has to align to nyble boundary. Minimum value is 4. Maximum value is 124. As a consequence, range of values is 4, 8, 12, 16,..., 124
[designate] ptr_zone_email =	(StrOpt) The email address to be used when creating PTR zones. If not specified, the email address will be admin@<dns_domain>
[designate] url = None	(StrOpt) URL for connecting to designate
[macvtap] physical_interface_mappings =	(ListOpt) Comma-separated list of <physical_network>:<physical_interface> tuples mapping physical network names to the agent's node-specific physical network interfaces to be used for flat and VLAN networks. All physical networks listed in network_vlan_ranges on the server should have mappings to appropriate interfaces on each agent.
[nova] auth_type = None	(Opt) Authentication type to load
[nova] endpoint_type = public	(StrOpt) Type of the nova endpoint to use. This endpoint will be looked up in the keystone catalog and should be one of public, internal or admin.

Table 9.90. New default values

Option	Previous default value	New default value
[DEFAULT] advertise_mtu	False	True
[DEFAULT] host	localhost	example.domain
[AGENT] veth_mtu	None	9000
[ml2] path_mtu	0	1500
[ml2_type_flat] flat_networks		*

Table 9.91. Description of configuration options for [account-auditor] in account-server.conf

Deprecated option	New Option
[DEFAULT] use_syslog	None
[m12] segment_mtu	[DEFAULT] global_physnet_mtu

CHAPTER 10. OBJECT STORAGE

OpenStack Object Storage uses multiple configuration files for multiple services and background daemons, and `paste.deploy` to manage server configurations. Default configuration options appear in the `[DEFAULT]` section. You can override the default values by setting values in the other sections.

10.1. INTRODUCTION TO OBJECT STORAGE

Object Storage is a robust, highly scalable and fault tolerant storage platform for unstructured data such as objects. Objects are stored bits, accessed through a RESTful, HTTP-based interface. You cannot access data at the block or file level. Object Storage is commonly used to archive and back up data, with use cases in virtual machine image, photo, video and music storage.

Object Storage provides a high degree of availability, throughput, and performance with its scale out architecture. Each object is replicated across multiple servers, residing within the same data center or across data centers, which mitigates the risk of network and hardware failure. In the event of hardware failure, Object Storage will automatically copy objects to a new location to ensure that there are always three copies available. Object Storage is an eventually consistent distributed storage platform; it sacrifices consistency for maximum availability and partition tolerance. Object Storage enables you to create a reliable platform by using commodity hardware and inexpensive storage.

10.2. OBJECT STORAGE GENERAL SERVICE CONFIGURATION

Most Object Storage services fall into two categories: Object Storage WSGI servers and background daemons.

Object Storage uses `paste.deploy` to manage server configurations. Read more at <http://pythonpaste.org/deploy/>.

Default configuration options are set in the `[DEFAULT]` section, and any options specified there can be overridden in any of the other sections when the syntax `set option_name = value` is in place.

Configuration for servers and daemons can be expressed together in the same file for each type of server, or separately. If a required section for the service trying to start is missing, there will be an error. Sections not used by the service are ignored.

Consider the example of an Object Storage node. By convention configuration for the `object-server`, `object-updater`, `object-replicator`, and `object-auditor` exist in a single file `/etc/swift/object-server.conf`:

```
[DEFAULT]

[pipeline:main]
pipeline = object-server

[app:object-server]
use = egg:swift#object

[object-replicator]
reclaim_age = 259200

[object-updater]

[object-auditor]
```

Object Storage services expect a configuration path as the first argument:

```
$ swift-object-auditor
Usage: swift-object-auditor CONFIG [options]

Error: missing config path argument
```

If you omit the `object-auditor` section, this file cannot be used as the configuration path when starting the `swift-object-auditor` daemon:

```
$ swift-object-auditor /etc/swift/object-server.conf
Unable to find object-auditor config section in /etc/swift/object-
server.conf
```

If the configuration path is a directory instead of a file, all of the files in the directory with the file extension ".conf" will be combined to generate the configuration object which is delivered to the Object Storage service. This is referred to generally as "directory-based configuration".

Directory-based configuration leverages ConfigParser's native multi-file support. Files ending in ".conf" in the given directory are parsed in lexicographical order. File names starting with '.' are ignored. A mixture of file and directory configuration paths is not supported - if the configuration path is a file, only that file will be parsed.

The Object Storage service management tool `swift-init` has adopted the convention of looking for `/etc/swift/{type}-server.conf.d/` if the file `/etc/swift/{type}-server.conf` file does not exist.

When using directory-based configuration, if the same option under the same section appears more than once in different files, the last value parsed is said to override previous occurrences. You can ensure proper override precedence by prefixing the files in the configuration directory with numerical values, as in the following example file layout:

```
/etc/swift/
  default.base
  object-server.conf.d/
    000_default.conf -> ../default.base
    001_default-override.conf
    010_server.conf
    020_replicator.conf
    030_updater.conf
    040_auditor.conf
```

You can inspect the resulting combined configuration object using the `swift-config` command-line tool.

All the services of an Object Store deployment share a common configuration in the `[swift-hash]` section of the `/etc/swift/swift.conf` file. The `swift_hash_path_suffix` and `swift_hash_path_prefix` values must be identical on all the nodes.

Table 10.1. Description of configuration options for `[swift-hash]` in `swift.conf`

Configuration option = Default value	Description
--------------------------------------	-------------

Configuration option = Default value	Description
swift_hash_path_prefix = <i>changeme</i>	A prefix used by hash_path to offer a bit more security when generating hashes for paths. It simply appends this value to all paths; if someone knows this suffix, it's easier for them to guess the hash a path will end up with. New installations are advised to set this parameter to a random secret, which would not be disclosed outside the organization. The same secret needs to be used by all swift servers of the same cluster. Existing installations should set this parameter to an empty string.
swift_hash_path_suffix = <i>changeme</i>	A suffix used by hash_path to offer a bit more security when generating hashes for paths. It simply appends this value to all paths; if someone knows this suffix, it's easier for them to guess the hash a path will end up with. New installations are advised to set this parameter to a random secret, which would not be disclosed outside the organization. The same secret needs to be used by all swift servers of the same cluster. Existing installations should set this parameter to an empty string.

10.3. OBJECT SERVER CONFIGURATION

Find an example object server configuration at `etc/object-server.conf-sample` in the source code repository.

The available configuration options are:

Table 10.2. Description of configuration options for [DEFAULT] in `object-server.conf`

Configuration option = Default value	Description
backlog = 4096	Maximum number of allowed pending TCP connections
bind_ip = 0.0.0.0	IP Address for server to bind to
bind_port = 6000	Port for server to bind to
bind_timeout = 30	Seconds to attempt bind before giving up
client_timeout = 60	Timeout to read one chunk from a client external services
conn_timeout = 0.5	Connection timeout to external services

Configuration option = Default value	Description
container_update_timeout = <i>1.0</i>	Time to wait while sending a container update on object update. object server. For most cases, this should be
devices = <i>/srv/node</i>	Parent directory of where devices are mounted
disable_fallocate = <i>false</i>	Disable "fast fail" fallocate checks if the underlying filesystem does not support it.
disk_chunk_size = <i>65536</i>	Size of chunks to read/write to disk
eventlet_debug = <i>false</i>	If true, turn on debug logging for eventlet
expiring_objects_account_name = <i>expiring_objects</i>	Account name for the expiring objects
expiring_objects_container_divisor = <i>86400</i>	Divisor for the expiring objects container
fallocate_reserve = <i>0</i>	You can set fallocate_reserve to the number of bytes you'd like fallocate to reserve, whether there is space for the given file size or not. This is useful for systems that behave badly when they completely run out of space; you can make the services pretend they're out of space early. server. For most cases, this should be
log_address = <i>/dev/log</i>	Location where syslog sends the logs to
log_custom_handlers =	Comma-separated list of functions to call to setup custom log handlers.
log_facility = <i>LOG_LOCAL0</i>	Syslog log facility
log_level = <i>INFO</i>	Logging level
log_max_line_length = <i>0</i>	Caps the length of log lines to the value given; no limit if set to 0, the default.
log_name = <i>swift</i>	Label used when logging
log_statsd_default_sample_rate = <i>1.0</i>	Defines the probability of sending a sample for any given event or timing measurement.
log_statsd_host = <i>localhost</i>	If not set, the StatsD feature is disabled.

Configuration option = Default value	Description
<code>log_statsd_metric_prefix =</code>	Value will be prepended to every metric sent to the StatsD server.
<code>log_statsd_port = 8125</code>	Port value for the StatsD server.
<code>log_statsd_sample_rate_factor = 1.0</code>	Not recommended to set this to a value less than 1.0, if frequency of logging is too high, tune the <code>log_statsd_default_sample_rate</code> instead.
<code>log_udp_host =</code>	If not set, the UDP receiver for syslog is disabled.
<code>log_udp_port = 514</code>	Port value for UDP receiver, if enabled.
<code>max_clients = 1024</code>	Maximum number of clients one worker can process simultaneously Lowering the number of clients handled per worker, and raising the number of workers can lessen the impact that a CPU intensive, or blocking, request can have on other requests served by the same worker. If the maximum number of clients is set to one, then a given worker will not perform another call while processing, allowing other workers a chance to process it.
<code>mount_check = true</code>	Whether or not check if the devices are mounted to prevent accidentally writing to the root device
<code>network_chunk_size = 65536</code>	Size of chunks to read/write over the network
<code>node_timeout = 3</code>	Request timeout to external services
<code>servers_per_port = 0</code>	If each disk in each storage policy ring has unique port numbers for its "ip" value, you can use this setting to have each object-server worker only service requests for the single disk matching the port in the ring. The value of this setting determines how many worker processes run for each port (disk) in the
<code>swift_dir = /etc/swift</code>	Swift configuration directory
<code>user = swift</code>	User to run as
<code>workers = auto</code>	a much higher value, one can reduce the impact of slow file system operations in one request from negatively impacting other requests.

Table 10.3. Description of configuration options for `[app-object-server]` in `object-server.conf`

Configuration option = Default value	Description
allowed_headers = ``Content-Disposition, Content-Encoding, X-Delete-At, X-Object-Manifest, X-Static-Large-Object``	Comma-separated list of headers that can be set in metadata of an object
auto_create_account_prefix = .	Prefix to use when automatically creating accounts
keep_cache_private = <i>false</i>	Allow non-public objects to stay in kernel's buffer cache
keep_cache_size = 5242880	Largest object size to keep in buffer cache
max_upload_time = 86400	Maximum time allowed to upload an object
mb_per_sync = 512	On PUT requests, sync file every n MB
replication_concurrency = 4	Set to restrict the number of concurrent incoming REPLICATION requests; set to 0 for unlimited
replication_failure_ratio = 1.0	If the value of failures / successes of REPLICATION subrequests exceeds this ratio, the overall REPLICATION request will be aborted
replication_failure_threshold = 100	The number of subrequest failures before the replication_failure_ratio is checked
replication_lock_timeout = 15	Number of seconds to wait for an existing replication device lock before giving up.
replication_one_per_device = <i>True</i>	Restricts incoming REPLICATION requests to one per device, replication_currency above allowing. This can help control I/O to each device, but you may wish to set this to False to allow multiple REPLICATION requests (up to the above replication_concurrency setting) per device.
replication_server = <i>false</i>	If defined, tells server how to handle replication verbs in requests. When set to True (or 1), only replication verbs will be accepted. When set to False, replication verbs will be rejected. When undefined, server will accept any verb in the request.
set_log_address = /dev/log	Location where syslog sends the logs to
set_log_facility = LOG_LOCAL0	Syslog log facility
set_log_level = INFO	Log level

Configuration option = Default value	Description
set log_name = object-server	Label to use when logging
set log_requests = true	Whether or not to log requests
slow = 0	If > 0, Minimum time in seconds for a PUT or DELETE request to complete
splice = no	Use splice() for zero-copy object GETs. This requires Linux kernel version 3.0 or greater. When you set "splice = yes" but the kernel does not support it, error messages will appear in the object server logs at startup, but your object servers should continue to function.
threads_per_disk = 0	Size of the per-disk thread pool used for performing disk I/O. The default of 0 means to not use a per-disk thread pool. It is recommended to keep this value small, as large values can result in high read latencies due to large queue depths. A good starting point is 4 threads per disk.
use = egg:swift#object	Entry point of paste.deploy in the server

Table 10.4. Description of configuration options for [pipeline-main] in object-server.conf

Configuration option = Default value	Description
pipeline = healthcheck recon object-server	Pipeline to use for processing operations.

Table 10.5. Description of configuration options for [object-replicator] in object-server.conf

Configuration option = Default value	Description
concurrency = 1	Number of replication workers to spawn
daemonize = on	Whether or not to run replication as a daemon
handoff_delete = auto	By default handoff partitions will be removed when it has successfully replicated to all the canonical nodes. If set to an integer n, it will remove the partition if it is successfully replicated to n nodes. The default setting should not be changed, except for extremem situations. This uses what's set here, or what's set in the DEFAULT section, or 10 (though other sections use 3 as the final default).

Configuration option = Default value	Description
handoffs_first = <i>False</i>	If set to True, partitions that are not supposed to be on the node will be replicated first. The default setting should not be changed, except for extreme situations.
http_timeout = 60	Maximum duration for an HTTP request
interval = 30	Minimum time for a pass to take
lockup_timeout = 1800	Attempts to kill all workers if nothing replications for lockup_timeout seconds
log_address = <i>/dev/log</i>	Location where syslog sends the logs to
log_facility = <i>LOG_LOCAL0</i>	Syslog log facility
log_level = <i>INFO</i>	Logging level
log_name = <i>object-replicator</i>	Label used when logging
node_timeout = ``<whatever's in the DEFAULT section or 10>``	Request timeout to external services
reclaim_age = 604800	Time elapsed in seconds before an object can be reclaimed
recon_cache_path = <i>/var/cache/swift</i>	Directory where stats for a few items will be stored
ring_check_interval = 15	How often (in seconds) to check the ring
rsync_bwlimit = 0	bandwidth limit for rsync in kB/s. 0 means unlimited
rsync_compress = <i>no</i>	Allows rsync to compress data which is transmitted to the destination node during sync. However, this applies only when the destination node is in a different region than the local one. ... note:: Objects that are already compressed (for example: .tar.gz, .mp3) might slow down the syncing process.
rsync_error_log_line_length = 0	Limits the length of the rsync error log lines. 0 will log the entire line.
rsync_io_timeout = 30	Passed to rsync for a max duration (seconds) of an I/O op

Configuration option = Default value	Description
rsync_module = ``{replication_ip}::object``	Format of the rsync module where the replicator will send data. The configuration value can include some variables that will be extracted from the ring. Variables must follow the format {NAME} where NAME is one of: ip, port, replication_ip, replication_port, region, zone, device, meta. See etc/rsyncd.conf-sample for some examples. uses what's set here, or what's set in the DEFAULT section, or 10 (though other sections use 3 as the final default).
rsync_timeout = 900	Max duration (seconds) of a partition rsync
run_pause = 30	Time in seconds to wait between replication passes
stats_interval = 300	Interval in seconds between logging replication statistics
sync_method = <i>rsync</i>	default is rsync, alternative is ssync

Table 10.6. Description of configuration options for [object-updater] in object-server.conf

Configuration option = Default value	Description
concurrency = 1	Number of replication workers to spawn
interval = 300	Minimum time for a pass to take
log_address = /dev/log	Location where syslog sends the logs to
log_facility = LOG_LOCAL0	Syslog log facility
log_level = INFO	Logging level
log_name = object-updater	Label used when logging
node_timeout = ``<whatever's in the DEFAULT section or 10>``	Request timeout to external services
recon_cache_path = /var/cache/swift	Directory where stats for a few items will be stored
slowdown = 0.01	Time in seconds to wait between objects

Table 10.7. Description of configuration options for [object-auditor] in object-server.conf

Configuration option = Default value	Description
bytes_per_second = 10000000	Maximum bytes audited per second. Should be tuned according to individual system specs. 0 is unlimited. mounted to prevent accidentally writing to the root device process simultaneously (it will actually accept(2) N + 1). Setting this to one (1) will only handle one request at a time, without accepting another request concurrently. By increasing the number of workers to a much higher value, one can reduce the impact of slow file system operations in one request from negatively impacting other requests. underlying filesystem does not support it. to setup custom log handlers. bytes you'd like fallocation to reserve, whether there is space for the given file size or not. This is useful for systems that behave badly when they completely run out of space; you can make the services pretend they're out of space early. container server. For most cases, this should be
concurrency = 1	Number of replication workers to spawn
disk_chunk_size = 65536	Size of chunks to read/write to disk
files_per_second = 20	Maximum files audited per second. Should be tuned according to individual system specs. 0 is unlimited.
log_address = /dev/log	Location where syslog sends the logs to
log_facility = LOG_LOCAL0	Syslog log facility
log_level = INFO	Logging level
log_name = object-auditor	Label used when logging
log_time = 3600	Frequency of status logs in seconds.
object_size_stats =	Takes a comma-separated list of ints. When set, the object auditor will increment a counter for every object whose size is greater or equal to the given breaking points and reports the result after a full scan.
recon_cache_path = /var/cache/swift	Directory where stats for a few items will be stored
zero_byte_files_per_second = 50	Maximum zero byte files audited per second.

Table 10.8. Description of configuration options for `[filter-healthcheck]` in `object-server.conf`

Configuration option = Default value	Description
disable_path =	An optional filesystem path, which if present, will cause the healthcheck URL to return "503 Service Unavailable" with a body of "DISABLED BY FILE"
use = <i>egg:swift#healthcheck</i>	Entry point of paste.deploy in the server

Table 10.9. Description of configuration options for `[filter-recon]` in `object-server.conf`

Configuration option = Default value	Description
recon_cache_path = <i>/var/cache/swift</i>	Directory where stats for a few items will be stored
recon_lock_path = <i>/var/lock</i>	Directory where lock files will be stored
use = <i>egg:swift#recon</i>	Entry point of paste.deploy in the server

Table 10.10. Description of configuration options for `[filter-xprofile]` in `object-server.conf`

Configuration option = Default value	Description
dump_interval = <i>5.0</i>	the profile data will be dumped to local disk based on above naming rule in this interval (seconds).
dump_timestamp = <i>false</i>	Be careful, this option will enable the profiler to dump data into the file with a time stamp which means that there will be lots of files piled up in the directory.
flush_at_shutdown = <i>false</i>	Clears the data when the wsgi server shutdowns.
log_filename_prefix = <i>/tmp/log/swift/profile/default.profile</i>	This prefix is used to combine the process ID and timestamp to name the profile data file. Make sure the executing user has permission to write into this path. Any missing path segments will be created, if necessary. When you enable profiling in more than one type of daemon, you must override it with a unique value like: <i>/var/log/swift/profile/object.profile</i>
path = <i>/__profile__</i>	This is the path of the URL to access the mini web UI.

Configuration option = Default value	Description
profile_module = <i>eventlet.green.profile</i>	This option enables you to switch profilers which inherit from the Python standard profiler. Currently, the supported value can be 'cProfile', 'eventlet.green.profile', etc.
unwind = <i>false</i>	unwind the iterator of applications
use = <i>egg:swift#xprofile</i>	Entry point of paste.deploy in the server

10.3.1. Sample object server configuration file

```
[DEFAULT]
# bind_ip = 0.0.0.0
bind_port = 6000
# bind_timeout = 30
# backlog = 4096
# user = swift
# swift_dir = /etc/swift
# devices = /srv/node
# mount_check = true
# disable_fallocate = false
# expiring_objects_container_divisor = 86400
# expiring_objects_account_name = expiring_objects
#
# Use an integer to override the number of pre-forked processes that will
# accept connections.
# workers = auto
#
# Maximum concurrent requests per worker
# max_clients = 1024
#
# You can specify default log routing here if you want:
# log_name = swift
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
# The following caps the length of log lines to the value given; no limit
if
# set to 0, the default.
# log_max_line_length = 0
#
# comma separated list of functions to call to setup custom log handlers.
# functions get passed: conf, name, log_to_console, log_route, fmt,
logger,
# adapted_logger
# log_custom_handlers =
#
# If set, log_udp_host will override log_address
# log_udp_host =
# log_udp_port = 514
#
```

```

# You can enable StatsD logging here:
# log_statsd_host = localhost
# log_statsd_port = 8125
# log_statsd_default_sample_rate = 1.0
# log_statsd_sample_rate_factor = 1.0
# log_statsd_metric_prefix =
#
# eventlet_debug = false
#
# You can set fallocate_reserve to the number of bytes you'd like
fallocate to
# reserve, whether there is space for the given file size or not.
# fallocate_reserve = 0
#
# Time to wait while attempting to connect to another backend node.
# conn_timeout = 0.5
# Time to wait while sending each chunk of data to another backend node.
# node_timeout = 3
# Time to wait while receiving each chunk of data from a client or another
# backend node.
# client_timeout = 60
#
# network_chunk_size = 65536
# disk_chunk_size = 65536

[pipeline:main]
pipeline = healthcheck recon object-server

[app:object-server]
use = egg:swift#object
# You can override the default log routing for this app here:
# set log_name = object-server
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_requests = true
# set log_address = /dev/log
#
# max_upload_time = 86400
# slow = 0
#
# Objects smaller than this are not evicted from the buffercache once read
# keep_cache_size = 5242880
#
# If true, objects for authenticated GET requests may be kept in buffer
cache
# if small enough
# keep_cache_private = false
#
# on PUTs, sync data every n MB
# mb_per_sync = 512
#
# Comma separated list of headers that can be set in metadata on an
object.
# This list is in addition to X-Object-Meta-* headers and cannot include
# Content-Type, etag, Content-Length, or deleted
# allowed_headers = Content-Disposition, Content-Encoding, X-Delete-At, X-

```

```

Object-Manifest, X-Static-Large-Object
#
# auto_create_account_prefix = .
#
# A value of 0 means "don't use thread pools". A reasonable starting point
is
# 4.
# threads_per_disk = 0
#
# Configure parameter for creating specific server
# To handle all verbs, including replication verbs, do not specify
# "replication_server" (this is the default). To only handle replication,
# set to a True value (e.g. "True" or "1"). To handle only non-replication
# verbs, set to "False". Unless you have a separate replication network,
you
# should not specify any value for "replication_server".
# replication_server = false
#
# Set to restrict the number of concurrent incoming REPLICATION requests
# Set to 0 for unlimited
# Note that REPLICATION is currently an async only item
# replication_concurrency = 4
#
# Restricts incoming REPLICATION requests to one per device,
# replication_concurrency above allowing. This can help control I/O to each
# device, but you may wish to set this to False to allow multiple
REPLICATION
# requests (up to the above replication_concurrency setting) per device.
# replication_one_per_device = True
#
# Number of seconds to wait for an existing replication device lock before
# giving up.
# replication_lock_timeout = 15
#
# These next two settings control when the REPLICATION subrequest handler
will
# abort an incoming REPLICATION attempt. An abort will occur if there are
at
# least threshold number of failures and the value of failures / successes
# exceeds the ratio. The defaults of 100 and 1.0 means that at least 100
# failures have to occur and there have to be more failures than successes
for
# an abort to occur.
# replication_failure_threshold = 100
# replication_failure_ratio = 1.0
#
# Use splice() for zero-copy object GETs. This requires Linux kernel
# version 3.0 or greater. If you set "splice = yes" but the kernel
# does not support it, error messages will appear in the object server
# logs at startup, but your object servers should continue to function.
#
# splice = no

[filter:healthcheck]
use = egg:swift#healthcheck
# An optional filesystem path, which if present, will cause the

```

```

healthcheck
# URL to return "503 Service Unavailable" with a body of "DISABLED BY
FILE"
# disable_path =

[filter:recon]
use = egg:swift#recon
#recon_cache_path = /var/cache/swift
#recon_lock_path = /var/lock

[object-replicator]
# You can override the default log routing for this app here (don't use
set!):
# log_name = object-replicator
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# vm_test_mode = no
# daemonize = on
# run_pause = 30
# concurrency = 1
# stats_interval = 300
#
# The sync method to use; default is rsync but you can use ssync to try
the
# EXPERIMENTAL all-swift-code-no-rsync-callouts method. Once ssync is
verified
# as having performance comparable to, or better than, rsync, we plan to
# deprecate rsync so we can move on with more features for replication.
# sync_method = rsync
#
# max duration of a partition rsync
# rsync_timeout = 900
#
# bandwidth limit for rsync in kB/s. 0 means unlimited
# rsync_bwlimit = 0
#
# passed to rsync for io op timeout
# rsync_io_timeout = 30
#
# node_timeout = <whatever's in the DEFAULT section or 10>
# max duration of an http request; this is for REPLICATE finalization
calls and
# so should be longer than node_timeout
# http_timeout = 60
#
# attempts to kill all workers if nothing replicates for lockup_timeout
seconds
# lockup_timeout = 1800
#
# The replicator also performs reclamation
# reclaim_age = 604800
#
# ring_check_interval = 15
# recon_cache_path = /var/cache/swift

```

```

#
# limits how long rsync error log lines are
# 0 means to log the entire line
# rsync_error_log_line_length = 0
#
# handoffs_first and handoff_delete are options for a special case
# such as disk full in the cluster. These two options SHOULD NOT BE
# CHANGED, except for such an extreme situations. (e.g. disks filled up
# or are about to fill up. Anyway, DO NOT let your drives fill up)
# handoffs_first is the flag to replicate handoffs prior to canonical
# partitions. It allows to force syncing and deleting handoffs quickly.
# If set to a True value(e.g. "True" or "1"), partitions
# that are not supposed to be on the node will be replicated first.
# handoffs_first = False
#
# handoff_delete is the number of replicas which are ensured in swift.
# If the number less than the number of replicas is set, object-replicator
# could delete local handoffs even if all replicas are not ensured in the
# cluster. Object-replicator would remove local handoff partition
directories
# after syncing partition when the number of successful responses is
greater
# than or equal to this number. By default(auto), handoff partitions will
be
# removed when it has successfully replicated to all the canonical nodes.
# handoff_delete = auto

[object-reconstructor]
# You can override the default log routing for this app here (don't use
set!):
# Unless otherwise noted, each setting below has the same meaning as
described
# in the [object-replicator] section, however these settings apply to the
EC
# reconstructor
#
# log_name = object-reconstructor
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# daemonize = on
# run_pause = 30
# concurrency = 1
# stats_interval = 300
# node_timeout = 10
# http_timeout = 60
# lockup_timeout = 1800
# reclaim_age = 604800
# ring_check_interval = 15
# recon_cache_path = /var/cache/swift
# handoffs_first = False

[object-updater]
# You can override the default log routing for this app here (don't use
set!):

```



```

# log_name = object-updater
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# interval = 300
# concurrency = 1
# node_timeout = <whatever's in the DEFAULT section or 10>
# slowdown will sleep that amount between objects
# slowdown = 0.01
#
# recon_cache_path = /var/cache/swift

[object-auditor]
# You can override the default log routing for this app here (don't use
# set!):
# log_name = object-auditor
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# You can set the disk chunk size that the auditor uses making it larger
# if
# you like for more efficient local auditing of larger objects
# disk_chunk_size = 65536
# files_per_second = 20
# concurrency = 1
# bytes_per_second = 10000000
# log_time = 3600
# zero_byte_files_per_second = 50
# recon_cache_path = /var/cache/swift

# Takes a comma separated list of ints. If set, the object auditor will
# increment a counter for every object whose size is <= to the given break
# points and report the result after a full scan.
# object_size_stats =

# Note: Put it at the beginning of the pipeline to profile all
# middleware. But
# it is safer to put this after healthcheck.
[filter:xprofile]
use = egg:swift#xprofile
# This option enable you to switch profilers which should inherit from
# python
# standard profiler. Currently the supported value can be 'cProfile',
# 'eventlet.green.profile' etc.
# profile_module = eventlet.green.profile
#
# This prefix will be used to combine process ID and timestamp to name the
# profile data file. Make sure the executing user has permission to write
# into this path (missing path segments will be created, if necessary).
# If you enable profiling in more than one type of daemon, you must
# override
# it with an unique value like: /var/log/swift/profile/object.profile
# log_filename_prefix = /tmp/log/swift/profile/default.profile
#

```

```
# the profile data will be dumped to local disk based on above naming rule
# in this interval.
# dump_interval = 5.0
#
# Be careful, this option will enable profiler to dump data into the file
# with
# time stamp which means there will be lots of files piled up in the
# directory.
# dump_timestamp = false
#
# This is the path of the URL to access the mini web UI.
# path = /__profile__
#
# Clear the data when the wsgi server shutdown.
# flush_at_shutdown = false
#
# unwind the iterator of applications
# unwind = false
```

10.4. OBJECT EXPIRER CONFIGURATION

Find an example object expirer configuration at `etc/object-expirer.conf-sample` in the source code repository.

The available configuration options are:

Table 10.11. Description of configuration options for [DEFAULT] in `object-expirer.conf`

Configuration option = Default value	Description
<code>log_address = /dev/log</code>	Location where syslog sends the logs to
<code>log_custom_handlers =</code>	Comma-separated list of functions to call to setup custom log handlers.
<code>log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>log_level = INFO</code>	Logging level
<code>log_max_line_length = 0</code>	Caps the length of log lines to the value given; no limit if set to 0, the default.
<code>log_name = swift</code>	Label used when logging
<code>log_statsd_default_sample_rate = 1.0</code>	Defines the probability of sending a sample for any given event or timing measurement.
<code>log_statsd_host = localhost</code>	If not set, the StatsD feature is disabled.
<code>log_statsd_metric_prefix =</code>	Value will be prepended to every metric sent to the StatsD server.

Configuration option = Default value	Description
<code>log_statsd_port = 8125</code>	Port value for the StatsD server.
<code>log_statsd_sample_rate_factor = 1.0</code>	Not recommended to set this to a value less than 1.0, if frequency of logging is too high, tune the <code>log_statsd_default_sample_rate</code> instead.
<code>log_udp_host =</code>	If not set, the UDP receiver for syslog is disabled.
<code>log_udp_port = 514</code>	Port value for UDP receiver, if enabled.
<code>swift_dir = /etc/swift</code>	Swift configuration directory
<code>user = swift</code>	User to run as

Table 10.12. Description of configuration options for `[app-proxy-server]` in `object-expirer.conf`

Configuration option = Default value	Description
<code>use = egg:swift#proxy</code>	Entry point of paste.deploy in the server

Table 10.13. Description of configuration options for `[filter-cache]` in `object-expirer.conf`

Configuration option = Default value	Description
<code>use = egg:swift#memcache</code>	Entry point of paste.deploy in the server

Table 10.14. Description of configuration options for `[filter-catch_errors]` in `object-expirer.conf`

Configuration option = Default value	Description
<code>use = egg:swift#catch_errors</code>	Entry point of paste.deploy in the server

Table 10.15. Description of configuration options for `[filter-proxy-logging]` in `object-expirer.conf`

Configuration option = Default value	Description
<code>access_log_address = /dev/log</code>	Location where syslog sends the logs to. If not set, logging directives from [DEFAULT] without "access__" will be used.

Configuration option = Default value	Description
access_log_facility = <i>LOG_LOCAL0</i>	Syslog facility to receive log lines. If not set, logging directives from [DEFAULT] without "access_" will be used.
access_log_headers = <i>false</i>	Header to receive log lines. If not set, logging directives from [DEFAULT] without "access_" will be used.
access_log_headers_only =	If access_log_headers is True and access_log_headers_only is set only these headers are logged. Multiple headers can be defined as comma separated list like this: access_log_headers_only = Host, X-Object-Meta-Mtime
access_log_level = <i>INFO</i>	Syslog logging level to receive log lines. If not set, logging directives from [DEFAULT] without "access_" will be used.
access_log_name = <i>swift</i>	Label used when logging. If not set, logging directives from [DEFAULT] without "access_" will be used.
access_log_statsd_default_sample_rate = <i>1.0</i>	Defines the probability of sending a sample for any given event or timing measurement. If not set, logging directives from [DEFAULT] without "access_" will be used.
access_log_statsd_host = <i>localhost</i>	You can use log_statsd_* from [DEFAULT], or override them here. StatsD server. IPv4/IPv6 addresses and hostnames are supported. If a hostname resolves to an IPv4 and IPv6 address, the IPv4 address will be used.
access_log_statsd_metric_prefix =	Value will be prepended to every metric sent to the StatsD server. If not set, logging directives from [DEFAULT] without "access_" will be used.
access_log_statsd_port = <i>8125</i>	Port value for the StatsD server. If not set, logging directives from [DEFAULT] without "access_" will be used.
access_log_statsd_sample_rate_factor = <i>1.0</i>	Not recommended to set this to a value less than 1.0, if frequency of logging is too high, tune the log_statsd_default_sample_rate instead. If not set, logging directives from [DEFAULT] without "access_" will be used.

Configuration option = Default value	Description
access_log_udp_host =	If not set, the UDP receiver for syslog is disabled. If not set, logging directives from [DEFAULT] without "access__" will be used.
access_log_udp_port = 514	Port value for UDP receiver, if enabled. If not set, logging directives from [DEFAULT] without "access__" will be used.
log_statsd_valid_http_methods = <i>GET,HEAD,POST,PUT,DELETE,COPY,OPTIONS</i>	What HTTP methods are allowed for StatsD logging (comma-sep). request methods not in this list will have "BAD_METHOD" for the <verb> portion of the metric.
reveal_sensitive_prefix = 16	By default, the X-Auth-Token is logged. To obscure the value, set reveal_sensitive_prefix to the number of characters to log. For example, if set to 12, only the first 12 characters of the token appear in the log. An unauthorized access of the log file won't allow unauthorized usage of the token. However, the first 12 or so characters is unique enough that you can trace/debug token usage. Set to 0 to suppress the token completely (replaced by '...' in the log). .. note:: reveal_sensitive_prefix will not affect the value logged with access_log_headers=True.
use = egg:swift#proxy_logging	Entry point of paste.deploy in the server

Table 10.16. Description of configuration options for [object-expirer] in object-expirer.conf

Configuration option = Default value	Description
auto_create_account_prefix = .	Prefix to use when automatically creating accounts
concurrency = 1	Number of replication workers to spawn
expiring_objects_account_name = <i>expiring_objects</i>	Account name for expiring objects.
interval = 300	Minimum time for a pass to take
process = 0	(it will actually accept(2) N + 1). Setting this to one (1) will only handle one request at a time, without accepting another request concurrently.

Configuration option = Default value	Description
processes = 0	for each port (disk) in the ring. If you have 24 disks per server, and this setting is 4, then each storage node will have $1 + (24 * 4) = 97$ total object-server processes running. This gives complete I/O isolation, drastically reducing the impact of slow disks on storage node performance. The object-replicator and object-reconstructor need to see this setting too, so it must be in the [DEFAULT] section.
reclaim_age = 604800	Time elapsed in seconds before an object can be reclaimed
recon_cache_path = /var/cache/swift	Directory where stats for a few items will be stored
report_interval = 300	Interval in seconds between reports.

Table 10.17. Description of configuration options for [pipeline-main] in object-expirer.conf

Configuration option = Default value	Description
pipeline = catch_errors proxy-logging cache proxy-server	Pipeline to use for processing operations.

10.4.1. Sample object expirer configuration file

```
[DEFAULT]
# swift_dir = /etc/swift
# user = swift
# You can specify default log routing here if you want:
# log_name = swift
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
# The following caps the length of log lines to the value given; no limit
if
# set to 0, the default.
# log_max_line_length = 0
#
# comma separated list of functions to call to setup custom log handlers.
# functions get passed: conf, name, log_to_console, log_route, fmt,
logger,
# adapted_logger
# log_custom_handlers =
#
# If set, log_udp_host will override log_address
# log_udp_host =
# log_udp_port = 514
#
# You can enable StatsD logging here:
```

```

# log_statsd_host = localhost
# log_statsd_port = 8125
# log_statsd_default_sample_rate = 1.0
# log_statsd_sample_rate_factor = 1.0
# log_statsd_metric_prefix =

[object-expirer]
# interval = 300
# auto_create_account_prefix = .
# expiring_objects_account_name = expiring_objects
# report_interval = 300
# concurrency is the level of concurrency o use to do the work, this value
# must be set to at least 1
# concurrency = 1
# processes is how many parts to divide the work into, one part per
process
#   that will be doing the work
# processes set 0 means that a single process will be doing all the work
# processes can also be specified on the command line and will override
the
#   config value
# processes = 0
# process is which of the parts a particular process will work on
# process can also be specified on the command line and will override the
config
#   value
# process is "zero based", if you want to use 3 processes, you should run
# processes with process set to 0, 1, and 2
# process = 0
# The expirer will re-attempt expiring if the source object is not
available
# up to reclaim_age seconds before it gives up and deletes the entry in
the
# queue.
# reclaim_age = 604800
# recon_cache_path = /var/cache/swift

[pipeline:main]
pipeline = catch_errors proxy-logging cache proxy-server

[app:proxy-server]
use = egg:swift#proxy
# See proxy-server.conf-sample for options

[filter:cache]
use = egg:swift#memcache
# See proxy-server.conf-sample for options

[filter:catch_errors]
use = egg:swift#catch_errors
# See proxy-server.conf-sample for options

[filter:proxy-logging]
use = egg:swift#proxy_logging
# If not set, logging directives from [DEFAULT] without "access_" will be
used

```

```

# access_log_name = swift
# access_log_facility = LOG_LOCAL0
# access_log_level = INFO
# access_log_address = /dev/log
#
# If set, access_log_udp_host will override access_log_address
# access_log_udp_host =
# access_log_udp_port = 514
#
# You can use log_statsd_* from [DEFAULT] or override them here:
# access_log_statsd_host = localhost
# access_log_statsd_port = 8125
# access_log_statsd_default_sample_rate = 1.0
# access_log_statsd_sample_rate_factor = 1.0
# access_log_statsd_metric_prefix =
# access_log_headers = false
#
# If access_log_headers is True and access_log_headers_only is set only
# these headers are logged. Multiple headers can be defined as comma
# separated
# list like this: access_log_headers_only = Host, X-Object-Meta-Mtime
# access_log_headers_only =
#
# By default, the X-Auth-Token is logged. To obscure the value,
# set reveal_sensitive_prefix to the number of characters to log.
# For example, if set to 12, only the first 12 characters of the
# token appear in the log. An unauthorized access of the log file
# won't allow unauthorized usage of the token. However, the first
# 12 or so characters is unique enough that you can trace/debug
# token usage. Set to 0 to suppress the token completely (replaced
# by '...' in the log).
# Note: reveal_sensitive_prefix will not affect the value
# logged with access_log_headers=True.
# reveal_sensitive_prefix = 16
#
# What HTTP methods are allowed for StatsD logging (comma-sep); request
# methods
# not in this list will have "BAD_METHOD" for the <verb> portion of the
# metric.
# log_statsd_valid_http_methods = GET,HEAD,POST,PUT,DELETE,COPY,OPTIONS

```

10.5. CONTAINER SERVER CONFIGURATION

Find an example container server configuration at `etc/container-server.conf-sample` in the source code repository.

The available configuration options are:

Table 10.18. Description of configuration options for [DEFAULT] in `container-server.conf`

Configuration option = Default value	Description
<code>allowed_sync_hosts = 127.0.0.1</code>	The list of hosts that are allowed to send syncs to.

Configuration option = Default value	Description
backlog = 4096	Maximum number of allowed pending TCP connections
bind_ip = 0.0.0.0	IP Address for server to bind to
bind_port = 6001	Port for server to bind to
bind_timeout = 30	Seconds to attempt bind before giving up
db_preallocation = off	If you don't mind the extra disk space usage in overhead, you can turn this on to preallocate disk space with SQLite databases to decrease fragmentation. underlying filesystem does not support it. to setup custom log handlers. bytes you'd like fallocate to reserve, whether there is space for the given file size or not. This is useful for systems that behave badly when they completely run out of space; you can make the services pretend they're out of space early. server. For most cases, this should be
devices = /srv/node	Parent directory of where devices are mounted
disable_fallocate = false	Disable "fast fail" fallocate checks if the underlying filesystem does not support it.
eventlet_debug = false	If true, turn on debug logging for eventlet
fallocate_reserve = 0	You can set fallocate_reserve to the number of bytes you'd like fallocate to reserve, whether there is space for the given file size or not. This is useful for systems that behave badly when they completely run out of space; you can make the services pretend they're out of space early. server. For most cases, this should be
log_address = /dev/log	Location where syslog sends the logs to
log_custom_handlers =	Comma-separated list of functions to call to setup custom log handlers.
log_facility = LOG_LOCAL0	Syslog log facility
log_level = INFO	Logging level
log_max_line_length = 0	Caps the length of log lines to the value given; no limit if set to 0, the default.

Configuration option = Default value	Description
<code>log_name = swift</code>	Label used when logging
<code>log_statsd_default_sample_rate = 1.0</code>	Defines the probability of sending a sample for any given event or timing measurement.
<code>log_statsd_host = localhost</code>	If not set, the StatsD feature is disabled.
<code>log_statsd_metric_prefix =</code>	Value will be prepended to every metric sent to the StatsD server.
<code>log_statsd_port = 8125</code>	Port value for the StatsD server.
<code>log_statsd_sample_rate_factor = 1.0</code>	Not recommended to set this to a value less than 1.0, if frequency of logging is too high, tune the <code>log_statsd_default_sample_rate</code> instead.
<code>log_udp_host =</code>	If not set, the UDP receiver for syslog is disabled.
<code>log_udp_port = 514</code>	Port value for UDP receiver, if enabled.
<code>max_clients = 1024</code>	Maximum number of clients one worker can process simultaneously Lowering the number of clients handled per worker, and raising the number of workers can lessen the impact that a CPU intensive, or blocking, request can have on other requests served by the same worker. If the maximum number of clients is set to one, then a given worker will not perform another call while processing, allowing other workers a chance to process it.
<code>mount_check = true</code>	Whether or not check if the devices are mounted to prevent accidentally writing to the root device
<code>swift_dir = /etc/swift</code>	Swift configuration directory
<code>user = swift</code>	User to run as
<code>workers = auto</code>	a much higher value, one can reduce the impact of slow file system operations in one request from negatively impacting other requests.

Table 10.19. Description of configuration options for [app-container-server] in container-server.conf

Configuration option = Default value	Description
<code>allow_versions = false</code>	Enable/Disable object versioning feature

Configuration option = Default value	Description
<code>auto_create_account_prefix = .</code>	Prefix to use when automatically creating accounts
<code>conn_timeout = 0.5</code>	Connection timeout to external services
<code>node_timeout = 3</code>	Request timeout to external services
<code>replication_server = false</code>	If defined, tells server how to handle replication verbs in requests. When set to True (or 1), only replication verbs will be accepted. When set to False, replication verbs will be rejected. When undefined, server will accept any verb in the request.
<code>set log_address = /dev/log</code>	Location where syslog sends the logs to
<code>set log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>set log_level = INFO</code>	Log level
<code>set log_name = container-server</code>	Label to use when logging
<code>set log_requests = true</code>	Whether or not to log requests
<code>use = egg:swift#container</code>	Entry point of paste.deploy in the server

Table 10.20. Description of configuration options for `[pipeline-main]` in `container-server.conf`

Configuration option = Default value	Description
<code>pipeline = healthcheck recon</code> <code>container-server</code>	Pipeline to use for processing operations.

Table 10.21. Description of configuration options for `[container-replicator]` in `container-server.conf`

Configuration option = Default value	Description
<code>concurrency = 8</code>	Number of replication workers to spawn
<code>conn_timeout = 0.5</code>	Connection timeout to external services
<code>interval = 30</code>	Minimum time for a pass to take
<code>log_address = /dev/log</code>	Location where syslog sends the logs to

Configuration option = Default value	Description
log_facility = <i>LOG_LOCAL0</i>	Syslog log facility
log_level = <i>INFO</i>	Logging level
log_name = <i>container-replicator</i>	Label used when logging
max_diffs = <i>100</i>	Caps how long the replicator spends trying to sync a database per pass
node_timeout = <i>10</i>	Request timeout to external services
per_diff = <i>1000</i>	Limit number of items to get per diff
reclaim_age = <i>604800</i>	Time elapsed in seconds before an object can be reclaimed
recon_cache_path = <i>/var/cache/swift</i>	Directory where stats for a few items will be stored
rsync_compress = <i>no</i>	Allow rsync to compress data which is transmitted to destination node during sync. However, this is applicable only when destination node is in a different region than the local one.
rsync_module = <i>``{replication_ip}::container``</i>	Format of the rsync module where the replicator will send data. The configuration value can include some variables that will be extracted from the ring. Variables must follow the format {NAME} where NAME is one of: ip, port, replication_ip, replication_port, region, zone, device, meta. See etc/rsyncd.conf-sample for some examples. uses what's set here, or what's set in the DEFAULT section, or 10 (though other sections use 3 as the final default).
run_pause = <i>30</i>	Time in seconds to wait between replication passes

Table 10.22. Description of configuration options for [container-updater] in container-server.conf

Configuration option = Default value	Description
account_suppression_time = <i>60</i>	Seconds to suppress updating an account that has generated an error (timeout, not yet found, etc.)
concurrency = <i>4</i>	Number of replication workers to spawn
conn_timeout = <i>0.5</i>	Connection timeout to external services

Configuration option = Default value	Description
interval = 300	Minimum time for a pass to take
log_address = /dev/log	Location where syslog sends the logs to
log_facility = LOG_LOCAL0	Syslog log facility
log_level = INFO	Logging level
log_name = container-updater	Label used when logging
node_timeout = 3	Request timeout to external services
recon_cache_path = /var/cache/swift	Directory where stats for a few items will be stored
slowdown = 0.01	Time in seconds to wait between objects

Table 10.23. Description of configuration options for [container-auditor] in container-server.conf

Configuration option = Default value	Description
containers_per_second = 200	Maximum containers audited per second. Should be tuned according to individual system specs. 0 is unlimited. mounted to prevent accidentally writing to the root device process simultaneously (it will actually accept(2) N + 1). Setting this to one (1) will only handle one request at a time, without accepting another request concurrently. By increasing the number of workers to a much higher value, one can reduce the impact of slow file system operations in one request from negatively impacting other requests.
interval = 1800	Minimum time for a pass to take
log_address = /dev/log	Location where syslog sends the logs to
log_facility = LOG_LOCAL0	Syslog log facility
log_level = INFO	Logging level
log_name = container-auditor	Label used when logging
recon_cache_path = /var/cache/swift	Directory where stats for a few items will be stored

Table 10.24. Description of configuration options for [container-sync] in container-server.conf

Configuration option = Default value	Description
conn_timeout = 5	Connection timeout to external services
container_time = 60	Maximum amount of time to spend syncing each container
internal_client_conf_path = /etc/swift/internal-client.conf	Internal client config file path
interval = 300	Minimum time for a pass to take
log_address = /dev/log	Location where syslog sends the logs to
log_facility = LOG_LOCAL0	Syslog log facility
log_level = INFO	Logging level
log_name = container-sync	Label used when logging
request_tries = 3	Server errors from requests will be retried by default
sync_proxy = http://10.1.1.1:8888,http://10.1.1.2:8888	If you need to use an HTTP proxy, set it here. Defaults to no proxy.

Table 10.25. Description of configuration options for [filter-healthcheck] in container-server.conf

Configuration option = Default value	Description
disable_path =	An optional filesystem path, which if present, will cause the healthcheck URL to return "503 Service Unavailable" with a body of "DISABLED BY FILE"
use = egg:swift#healthcheck	Entry point of paste.deploy in the server

Table 10.26. Description of configuration options for [filter-recon] in container-server.conf

Configuration option = Default value	Description
recon_cache_path = /var/cache/swift	Directory where stats for a few items will be stored
use = egg:swift#recon	Entry point of paste.deploy in the server

Table 10.27. Description of configuration options for `[filter-xprofile]` in `container-server.conf`

Configuration option = Default value	Description
<code>dump_interval = 5.0</code>	the profile data will be dumped to local disk based on above naming rule in this interval (seconds).
<code>dump_timestamp = false</code>	Be careful, this option will enable the profiler to dump data into the file with a time stamp which means that there will be lots of files piled up in the directory.
<code>flush_at_shutdown = false</code>	Clears the data when the wsgi server shutdowns.
<code>log_filename_prefix = /tmp/log/swift/profile/default.profile</code>	This prefix is used to combine the process ID and timestamp to name the profile data file. Make sure the executing user has permission to write into this path. Any missing path segments will be created, if necessary. When you enable profiling in more than one type of daemon, you must override it with a unique value like: <code>/var/log/swift/profile/object.profile</code>
<code>path = /__profile__</code>	This is the path of the URL to access the mini web UI.
<code>profile_module = eventlet.green.profile</code>	This option enables you to switch profilers which inherit from the Python standard profiler. Currently, the supported value can be 'cProfile', 'eventlet.green.profile', etc.
<code>unwind = false</code>	unwind the iterator of applications
<code>use = egg:swift#xprofile</code>	Entry point of paste.deploy in the server

10.5.1. Sample container server configuration file

```
[DEFAULT]
# bind_ip = 0.0.0.0
bind_port = 6001
# bind_timeout = 30
# backlog = 4096
# user = swift
# swift_dir = /etc/swift
# devices = /srv/node
# mount_check = true
# disable_fallocate = false
#
# Use an integer to override the number of pre-forked processes that will
# accept connections.
# workers = auto
```

```

#
# Maximum concurrent requests per worker
# max_clients = 1024
#
# This is a comma separated list of hosts allowed in the X-Container-Sync-
To
# field for containers. This is the old-style of using container sync. It
is
# strongly recommended to use the new style of a separate
# container-sync-realms.conf -- see container-sync-realms.conf-sample
# allowed_sync_hosts = 127.0.0.1
#
# You can specify default log routing here if you want:
# log_name = swift
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
# The following caps the length of log lines to the value given; no limit
if
# set to 0, the default.
# log_max_line_length = 0
#
# comma-separated list of functions to call to setup custom log handlers.
# functions get passed: conf, name, log_to_console, log_route, fmt,
logger,
# adapted_logger
# log_custom_handlers =
#
# If set, log_udp_host will override log_address
# log_udp_host =
# log_udp_port = 514
#
# You can enable StatsD logging here:
# log_statsd_host = localhost
# log_statsd_port = 8125
# log_statsd_default_sample_rate = 1.0
# log_statsd_sample_rate_factor = 1.0
# log_statsd_metric_prefix =
#
# If you don't mind the extra disk space usage in overhead, you can turn
this
# on to preallocate disk space with SQLite databases to decrease
fragmentation.
# db_preallocation = off
#
# eventlet_debug = false
#
# You can set fallocate_reserve to the number of bytes you'd like
fallocate to
# reserve, whether there is space for the given file size or not.
# fallocate_reserve = 0

[pipeline:main]
pipeline = healthcheck recon container-server

[app:container-server]

```



```

use = egg:swift#container
# You can override the default log routing for this app here:
# set log_name = container-server
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_requests = true
# set log_address = /dev/log
#
# node_timeout = 3
# conn_timeout = 0.5
# allow_versions = false
# auto_create_account_prefix = .
#
# Configure parameter for creating specific server
# To handle all verbs, including replication verbs, do not specify
# "replication_server" (this is the default). To only handle replication,
# set to a True value (e.g. "True" or "1"). To handle only non-replication
# verbs, set to "False". Unless you have a separate replication network,
you
# should not specify any value for "replication_server".
# replication_server = false

[filter:healthcheck]
use = egg:swift#healthcheck
# An optional filesystem path, which if present, will cause the
healthcheck
# URL to return "503 Service Unavailable" with a body of "DISABLED BY
FILE"
# disable_path =

[filter:recon]
use = egg:swift#recon
#recon_cache_path = /var/cache/swift

[container-replicator]
# You can override the default log routing for this app here (don't use
set!):
# log_name = container-replicator
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# vm_test_mode = no
# per_diff = 1000
# max_diffs = 100
# concurrency = 8
# interval = 30
# node_timeout = 10
# conn_timeout = 0.5
#
# The replicator also performs reclamation
# reclaim_age = 604800
#
# Time in seconds to wait between replication passes
# Note: if the parameter 'interval' is defined then it will be used in
place

```

```
# of run_pause.
# run_pause = 30
#
# recon_cache_path = /var/cache/swift

[container-updater]
# You can override the default log routing for this app here (don't use
set!):
# log_name = container-updater
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# interval = 300
# concurrency = 4
# node_timeout = 3
# conn_timeout = 0.5
#
# slowdown will sleep that amount between containers
# slowdown = 0.01
#
# Seconds to suppress updating an account that has generated an error
# account_suppression_time = 60
#
# recon_cache_path = /var/cache/swift

[container-auditor]
# You can override the default log routing for this app here (don't use
set!):
# log_name = container-auditor
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# Will audit each container at most once per interval
# interval = 1800
#
# containers_per_second = 200
# recon_cache_path = /var/cache/swift

[container-sync]
# You can override the default log routing for this app here (don't use
set!):
# log_name = container-sync
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# If you need to use an HTTP Proxy, set it here; defaults to no proxy.
# You can also set this to a comma separated list of HTTP Proxies and they
will
# be randomly used (simple load balancing).
# sync_proxy = http://10.1.1.1:8888,http://10.1.1.2:8888
#
# Will sync each container at most once per interval
# interval = 300
```

```

#
# Maximum amount of time to spend syncing each container per pass
# container_time = 60
#
# Maximum amount of time in seconds for the connection attempt
# conn_timeout = 5
# Server errors from requests will be retried by default
# request_tries = 3
#
# Internal client config file path
# internal_client_conf_path = /etc/swift/internal-client.conf

# Note: Put it at the beginning of the pipeline to profile all middleware.
# But
# it is safer to put this after healthcheck.
[filter:xprofile]
use = egg:swift#xprofile
# This option enable you to switch profilers which should inherit from
python
# standard profiler. Currently the supported value can be 'cProfile',
# 'eventlet.green.profile' etc.
# profile_module = eventlet.green.profile
#
# This prefix will be used to combine process ID and timestamp to name the
# profile data file. Make sure the executing user has permission to write
# into this path (missing path segments will be created, if necessary).
# If you enable profiling in more than one type of daemon, you must
override
# it with an unique value like: /var/log/swift/profile/container.profile
# log_filename_prefix = /tmp/log/swift/profile/default.profile
#
# the profile data will be dumped to local disk based on above naming rule
# in this interval.
# dump_interval = 5.0
#
# Be careful, this option will enable profiler to dump data into the file
with
# time stamp which means there will be lots of files piled up in the
directory.
# dump_timestamp = false
#
# This is the path of the URL to access the mini web UI.
# path = /__profile__
#
# Clear the data when the wsgi server shutdown.
# flush_at_shutdown = false
#
# unwind the iterator of applications
# unwind = false

```

10.6. CONTAINER SYNC REALMS CONFIGURATION

Find an example container sync realms configuration at `etc/container-sync-realms.conf-sample` in the source code repository.

The available configuration options are:

Table 10.28. Description of configuration options for [DEFAULT] in container-sync-realms.conf

Configuration option = Default value	Description
<code>mtime_check_interval = 300</code>	The number of seconds between checking the modified time of this config file for changes and therefore reloading it.

Table 10.29. Description of configuration options for [realm1] in container-sync-realms.conf

Configuration option = Default value	Description
<code>cluster_clustername1 = https://host1/v1/</code>	Any values in the realm section whose names begin with <code>cluster_</code> will indicate the name and endpoint of a cluster and will be used by external users in their containers' X-Container-Sync-To metadata header values with the format <code>"realm_name/cluster_name/container_name"</code> . Realm and cluster names are considered case insensitive.
<code>cluster_clustername2 = https://host2/v1/</code>	Any values in the realm section whose names begin with <code>cluster_</code> will indicate the name and endpoint of a cluster and will be used by external users in their containers' X-Container-Sync-To metadata header values with the format <code>"realm_name/cluster_name/container_name"</code> . Realm and cluster names are considered case insensitive.
<code>key = realm1key</code>	The key is the overall cluster-to-cluster key used in combination with the external users' key that they set on their containers' X-Container-Sync-Key metadata header values. These keys will be used to sign each request the container sync daemon makes and used to validate each incoming container sync request.
<code>key2 = realm1key2</code>	The key2 is optional and is an additional key incoming requests will be checked against. This is so you can rotate keys if you wish; you move the existing key to key2 and make a new key value.

Table 10.30. Description of configuration options for [realm2] in container-sync-realms.conf

Configuration option = Default value	Description
cluster_clustername3 = <i>https://host3/v1/</i>	Any values in the realm section whose names begin with cluster_ will indicate the name and endpoint of a cluster and will be used by external users in their containers' X-Container-Sync-To metadata header values with the format "realm_name/cluster_name/container_name". Realm and cluster names are considered case insensitive.
cluster_clustername4 = <i>https://host4/v1/</i>	Any values in the realm section whose names begin with cluster_ will indicate the name and endpoint of a cluster and will be used by external users in their containers' X-Container-Sync-To metadata header values with the format "realm_name/cluster_name/container_name". Realm and cluster names are considered case insensitive.
key = <i>realm2key</i>	The key is the overall cluster-to-cluster key used in combination with the external users' key that they set on their containers' X-Container-Sync-Key metadata header values. These keys will be used to sign each request the container sync daemon makes and used to validate each incoming container sync request.
key2 = <i>realm2key2</i>	The key2 is optional and is an additional key incoming requests will be checked against. This is so you can rotate keys if you wish; you move the existing key to key2 and make a new key value.

10.6.1. Sample container sync realms configuration file

```
# [DEFAULT]
# The number of seconds between checking the modified time of this config
# file
# for changes and therefore reloading it.
# mtime_check_interval = 300

# [realm1]
# key = realm1key
# key2 = realm1key2
# cluster_name1 = https://host1/v1/
# cluster_name2 = https://host2/v1/
#
# [realm2]
# key = realm2key
# key2 = realm2key2
# cluster_name3 = https://host3/v1/
# cluster_name4 = https://host4/v1/
```

```

# Each section name is the name of a sync realm. A sync realm is a set of
# clusters that have agreed to allow container syncing with each other.
Realm
# names will be considered case insensitive.
#
# The key is the overall cluster-to-cluster key used in combination with
the
# external users' key that they set on their containers' X-Container-Sync-
Key
# metadata header values. These keys will be used to sign each request the
# container sync daemon makes and used to validate each incoming container
sync
# request.
#
# The key2 is optional and is an additional key incoming requests will be
# checked against. This is so you can rotate keys if you wish; you move
the
# existing key to key2 and make a new key value.
#
# Any values in the realm section whose names begin with cluster_ will
indicate
# the name and endpoint of a cluster and will be used by external users in
# their containers' X-Container-Sync-To metadata header values with the
format
# "realm_name/cluster_name/container_name". Realm and cluster names are
# considered case insensitive.
#
# The endpoint is what the container sync daemon will use when sending out
# requests to that cluster. Keep in mind this endpoint must be reachable
by all
# container servers, since that is where the container sync daemon runs.
Note
# the the endpoint ends with /v1/ and that the container sync daemon will
then
# add the account/container/obj name after that.
#
# Distribute this container-sync-realms.conf file to all your proxy
servers
# and container servers.

```

10.7. CONTAINER RECONCILER CONFIGURATION

Find an example container sync realms configuration at `etc/container-reconciler.conf-sample` in the source code repository.

The available configuration options are:

Table 10.31. Description of configuration options for [DEFAULT] in `container-reconciler.conf`

Configuration option = Default value	Description
<code>log_address = /dev/log</code>	Location where syslog sends the logs to

Configuration option = Default value	Description
<code>log_custom_handlers =</code>	Comma-separated list of functions to call to setup custom log handlers.
<code>log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>log_level = INFO</code>	Logging level
<code>log_name = swift</code>	Label used when logging
<code>log_statsd_default_sample_rate = 1.0</code>	Defines the probability of sending a sample for any given event or timing measurement.
<code>log_statsd_host = localhost</code>	If not set, the StatsD feature is disabled.
<code>log_statsd_metric_prefix =</code>	Value will be prepended to every metric sent to the StatsD server.
<code>log_statsd_port = 8125</code>	Port value for the StatsD server.
<code>log_statsd_sample_rate_factor = 1.0</code>	Not recommended to set this to a value less than 1.0, if frequency of logging is too high, tune the <code>log_statsd_default_sample_rate</code> instead.
<code>log_udp_host =</code>	If not set, the UDP receiver for syslog is disabled.
<code>log_udp_port = 514</code>	Port value for UDP receiver, if enabled.
<code>swift_dir = /etc/swift</code>	Swift configuration directory
<code>user = swift</code>	User to run as

Table 10.32. Description of configuration options for `[app-proxy-server]` in `container-reconciler.conf`

Configuration option = Default value	Description
<code>use = egg:swift#proxy</code>	Entry point of paste.deploy in the server

Table 10.33. Description of configuration options for `[container-reconciler]` in `container-reconciler.conf`

Configuration option = Default value	Description
<code>interval = 30</code>	Minimum time for a pass to take

Configuration option = Default value	Description
reclaim_age = 604800	Time elapsed in seconds before an object can be reclaimed
request_tries = 3	No help text available for this option.

Table 10.34. Description of configuration options for `[filter-cache]` in `container-reconciler.conf`

Configuration option = Default value	Description
use = <i>egg:swift#memcache</i>	Entry point of paste.deploy in the server

Table 10.35. Description of configuration options for `[filter-catch_errors]` in `container-reconciler.conf`

Configuration option = Default value	Description
use = <i>egg:swift#catch_errors</i>	Entry point of paste.deploy in the server

Table 10.36. Description of configuration options for `[filter-proxy-logging]` in `container-reconciler.conf`

Configuration option = Default value	Description
use = <i>egg:swift#proxy_logging</i>	Entry point of paste.deploy in the server

Table 10.37. Description of configuration options for `[pipeline-main]` in `container-reconciler.conf`

Configuration option = Default value	Description
pipeline = <i>catch_errors proxy-logging cache proxy-server</i>	No help text available for this option.

10.7.1. Sample container sync reconciler configuration file

```
[DEFAULT]
# swift_dir = /etc/swift
# user = swift
# You can specify default log routing here if you want:
# log_name = swift
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
```



```

# comma separated list of functions to call to setup custom log handlers.
# functions get passed: conf, name, log_to_console, log_route, fmt,
logger,
# adapted_logger
# log_custom_handlers =
#
# If set, log_udp_host will override log_address
# log_udp_host =
# log_udp_port = 514
#
# You can enable StatsD logging here:
# log_statsd_host = localhost
# log_statsd_port = 8125
# log_statsd_default_sample_rate = 1.0
# log_statsd_sample_rate_factor = 1.0
# log_statsd_metric_prefix =

[container-reconciler]
# The reconciler will re-attempt reconciliation if the source object is
not
# available up to reclaim_age seconds before it gives up and deletes the
entry
# in the queue.
# reclaim_age = 604800
# The cycle time of the daemon
# interval = 30
# Server errors from requests will be retried by default
# request_tries = 3

[pipeline:main]
pipeline = catch_errors proxy-logging cache proxy-server

[app:proxy-server]
use = egg:swift#proxy
# See proxy-server.conf-sample for options

[filter:cache]
use = egg:swift#memcache
# See proxy-server.conf-sample for options

[filter:proxy-logging]
use = egg:swift#proxy_logging

[filter:catch_errors]
use = egg:swift#catch_errors
# See proxy-server.conf-sample for options

```

10.8. ACCOUNT SERVER CONFIGURATION

Find an example account server configuration at `etc/account-server.conf-sample` in the source code repository.

The available configuration options are:

Table 10.38. Description of configuration options for [DEFAULT] in `account-server.conf`

Configuration option = Default value	Description
backlog = 4096	Maximum number of allowed pending TCP connections
bind_ip = 0.0.0.0	IP Address for server to bind to
bind_port = 6002	Port for server to bind to
bind_timeout = 30	Seconds to attempt bind before giving up
db_preallocation = off	If you don't mind the extra disk space usage in overhead, you can turn this on to preallocate disk space with SQLite databases to decrease fragmentation. underlying filesystem does not support it. to setup custom log handlers. bytes you'd like fallocate to reserve, whether there is space for the given file size or not. This is useful for systems that behave badly when they completely run out of space; you can make the services pretend they're out of space early. server. For most cases, this should be
devices = /srv/node	Parent directory of where devices are mounted
disable_fallocate = false	Disable "fast fail" fallocate checks if the underlying filesystem does not support it.
eventlet_debug = false	If true, turn on debug logging for eventlet
fallocate_reserve = 0	You can set fallocate_reserve to the number of bytes you'd like fallocate to reserve, whether there is space for the given file size or not. This is useful for systems that behave badly when they completely run out of space; you can make the services pretend they're out of space early. server. For most cases, this should be
log_address = /dev/log	Location where syslog sends the logs to
log_custom_handlers =	Comma-separated list of functions to call to setup custom log handlers.
log_facility = LOG_LOCAL0	Syslog log facility
log_level = INFO	Logging level
log_max_line_length = 0	Caps the length of log lines to the value given; no limit if set to 0, the default.

Configuration option = Default value	Description
<code>log_name = swift</code>	Label used when logging
<code>log_statsd_default_sample_rate = 1.0</code>	Defines the probability of sending a sample for any given event or timing measurement.
<code>log_statsd_host = localhost</code>	If not set, the StatsD feature is disabled.
<code>log_statsd_metric_prefix =</code>	Value will be prepended to every metric sent to the StatsD server.
<code>log_statsd_port = 8125</code>	Port value for the StatsD server.
<code>log_statsd_sample_rate_factor = 1.0</code>	Not recommended to set this to a value less than 1.0, if frequency of logging is too high, tune the <code>log_statsd_default_sample_rate</code> instead.
<code>log_udp_host =</code>	If not set, the UDP receiver for syslog is disabled.
<code>log_udp_port = 514</code>	Port value for UDP receiver, if enabled.
<code>max_clients = 1024</code>	Maximum number of clients one worker can process simultaneously Lowering the number of clients handled per worker, and raising the number of workers can lessen the impact that a CPU intensive, or blocking, request can have on other requests served by the same worker. If the maximum number of clients is set to one, then a given worker will not perform another call while processing, allowing other workers a chance to process it.
<code>mount_check = true</code>	Whether or not check if the devices are mounted to prevent accidentally writing to the root device
<code>swift_dir = /etc/swift</code>	Swift configuration directory
<code>user = swift</code>	User to run as
<code>workers = auto</code>	a much higher value, one can reduce the impact of slow file system operations in one request from negatively impacting other requests.

Table 10.39. Description of configuration options for [app-account-server] in account-server.conf

Configuration option = Default value	Description
<code>auto_create_account_prefix =.</code>	Prefix to use when automatically creating accounts

Configuration option = Default value	Description
replication_server = <i>false</i>	If defined, tells server how to handle replication verbs in requests. When set to True (or 1), only replication verbs will be accepted. When set to False, replication verbs will be rejected. When undefined, server will accept any verb in the request.
set log_address = <i>/dev/log</i>	Location where syslog sends the logs to
set log_facility = <i>LOG_LOCAL0</i>	Syslog log facility
set log_level = <i>INFO</i>	Log level
set log_name = <i>account-server</i>	Label to use when logging
set log_requests = <i>true</i>	Whether or not to log requests
use = <i>egg:swift#account</i>	Entry point of paste.deploy in the server

Table 10.40. Description of configuration options for [pipeline-main] in account-server.conf

Configuration option = Default value	Description
pipeline = <i>healthcheck recon account-server</i>	No help text available for this option.

Table 10.41. Description of configuration options for [account-replicator] in account-server.conf

Configuration option = Default value	Description
concurrency = <i>8</i>	Number of replication workers to spawn
conn_timeout = <i>0.5</i>	Connection timeout to external services
interval = <i>30</i>	Minimum time for a pass to take
log_address = <i>/dev/log</i>	Location where syslog sends the logs to
log_facility = <i>LOG_LOCAL0</i>	Syslog log facility
log_level = <i>INFO</i>	Logging level
log_name = <i>account-replicator</i>	Label used when logging

Configuration option = Default value	Description
<code>max_diffs = 100</code>	Caps how long the replicator spends trying to sync a database per pass
<code>node_timeout = 10</code>	Request timeout to external services
<code>per_diff = 1000</code>	Limit number of items to get per diff
<code>reclaim_age = 604800</code>	Time elapsed in seconds before an object can be reclaimed
<code>recon_cache_path = /var/cache/swift</code>	Directory where stats for a few items will be stored
<code>rsync_compress = no</code>	No help text available for this option.
<code>rsync_module = ``{replication_ip}::account``</code>	Format of the rsync module where the replicator will send data. The configuration value can include some variables that will be extracted from the ring. Variables must follow the format {NAME} where NAME is one of: ip, port, replication_ip, replication_port, region, zone, device, meta. See etc/rsyncd.conf-sample for some examples. uses what's set here, or what's set in the DEFAULT section, or 10 (though other sections use 3 as the final default).
<code>run_pause = 30</code>	Time in seconds to wait between replication passes

Table 10.42. Description of configuration options for `[account-auditor]` in `account-server.conf`

Configuration option = Default value	Description
<code>accounts_per_second = 200</code>	Maximum accounts audited per second. Should be tuned according to individual system specs. 0 is unlimited.
<code>interval = 1800</code>	Minimum time for a pass to take
<code>log_address = /dev/log</code>	Location where syslog sends the logs to
<code>log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>log_level = INFO</code>	Logging level
<code>log_name = account-auditor</code>	Label used when logging

Configuration option = Default value	Description
recon_cache_path = <code>/var/cache/swift</code>	Directory where stats for a few items will be stored

Table 10.43. Description of configuration options for [account - reaper] in `account-server.conf`

Configuration option = Default value	Description
concurrency = 25	Number of replication workers to spawn
conn_timeout = 0.5	Connection timeout to external services
delay_reaping = 0	Normally, the reaper begins deleting account information for deleted accounts immediately; you can set this to delay its work however. The value is in seconds, 2592000 = 30 days, for example. bind to giving up worker can process simultaneously (it will actually accept(2) N + 1). Setting this to one (1) will only handle one request at a time, without accepting another request concurrently. By increasing the number of workers to a much higher value, one can reduce the impact of slow file system operations in one request from negatively impacting other requests.
interval = 3600	Minimum time for a pass to take
log_address = <code>/dev/log</code>	Location where syslog sends the logs to
log_facility = <code>LOG_LOCAL0</code>	Syslog log facility
log_level = <code>INFO</code>	Logging level
log_name = <code>account - reaper</code>	Label used when logging
node_timeout = 10	Request timeout to external services
reap_warn_after = 2592000	No help text available for this option.

Table 10.44. Description of configuration options for [filter-healthcheck] in `account-server.conf`

Configuration option = Default value	Description
disable_path =	No help text available for this option.
use = <code>egg:swift#healthcheck</code>	Entry point of paste.deploy in the server

Table 10.45. Description of configuration options for `[filter-recon]` in `account-server.conf`

Configuration option = Default value	Description
recon_cache_path = <code>/var/cache/swift</code>	Directory where stats for a few items will be stored
use = <code>egg:swift#recon</code>	Entry point of paste.deploy in the server

Table 10.46. Description of configuration options for `[filter-xprofile]` in `account-server.conf`

Configuration option = Default value	Description
dump_interval = <code>5.0</code>	No help text available for this option.
dump_timestamp = <code>false</code>	No help text available for this option.
flush_at_shutdown = <code>false</code>	No help text available for this option.
log_filename_prefix = <code>/tmp/log/swift/profile/default.profile</code>	No help text available for this option.
path = <code>/__profile__</code>	No help text available for this option.
profile_module = <code>eventlet.green.profile</code>	No help text available for this option.
unwind = <code>false</code>	No help text available for this option.
use = <code>egg:swift#xprofile</code>	Entry point of paste.deploy in the server

10.8.1. Sample account server configuration file

```
[DEFAULT]
# bind_ip = 0.0.0.0
bind_port = 6002
# bind_timeout = 30
# backlog = 4096
# user = swift
# swift_dir = /etc/swift
# devices = /srv/node
# mount_check = true
# disable_fallocate = false
#
# Use an integer to override the number of pre-forked processes that will
# accept connections.
# workers = auto
#
# Maximum concurrent requests per worker
# max_clients = 1024
#
```

```

# You can specify default log routing here if you want:
# log_name = swift
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
# The following caps the length of log lines to the value given; no limit
if
# set to 0, the default.
# log_max_line_length = 0
#
# comma separated list of functions to call to setup custom log handlers.
# functions get passed: conf, name, log_to_console, log_route, fmt,
logger,
# adapted_logger
# log_custom_handlers =
#
# If set, log_udp_host will override log_address
# log_udp_host =
# log_udp_port = 514
#
# You can enable StatsD logging here:
# log_statsd_host = localhost
# log_statsd_port = 8125
# log_statsd_default_sample_rate = 1.0
# log_statsd_sample_rate_factor = 1.0
# log_statsd_metric_prefix =
#
# If you don't mind the extra disk space usage in overhead, you can turn
this
# on to preallocate disk space with SQLite databases to decrease
fragmentation.
# db_preallocation = off
#
# eventlet_debug = false
#
# You can set fallocate_reserve to the number of bytes you'd like
fallocate to
# reserve, whether there is space for the given file size or not.
# fallocate_reserve = 0

[pipeline:main]
pipeline = healthcheck recon account-server

[app:account-server]
use = egg:swift#account
# You can override the default log routing for this app here:
# set log_name = account-server
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_requests = true
# set log_address = /dev/log
#
# auto_create_account_prefix = .
#
# Configure parameter for creating specific server
# To handle all verbs, including replication verbs, do not specify

```



```

# "replication_server" (this is the default). To only handle replication,
# set to a True value (e.g. "True" or "1"). To handle only non-replication
# verbs, set to "False". Unless you have a separate replication network,
# you
# should not specify any value for "replication_server".
# replication_server = false

[filter:healthcheck]
use = egg:swift#healthcheck
# An optional filesystem path, which if present, will cause the
# healthcheck
# URL to return "503 Service Unavailable" with a body of "DISABLED BY
# FILE"
# disable_path =

[filter:recon]
use = egg:swift#recon
# recon_cache_path = /var/cache/swift

[account-replicator]
# You can override the default log routing for this app here (don't use
# set!):
# log_name = account-replicator
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# vm_test_mode = no
# per_diff = 1000
# max_diffs = 100
# concurrency = 8
# interval = 30
#
# How long without an error before a node's error count is reset. This
# will
# also be how long before a node is reenabled after suppression is
# triggered.
# error_suppression_interval = 60
#
# How many errors can accumulate before a node is temporarily ignored.
# error_suppression_limit = 10
#
# node_timeout = 10
# conn_timeout = 0.5
#
# The replicator also performs reclamation
# reclaim_age = 604800
#
# Time in seconds to wait between replication passes
# Note: if the parameter 'interval' is defined then it will be used in
# place
# of run_pause.
# run_pause = 30
#
# recon_cache_path = /var/cache/swift

```

```

[account-auditor]
# You can override the default log routing for this app here (don't use
set!):
# log_name = account-auditor
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# Will audit each account at most once per interval
# interval = 1800
#
# log_facility = LOG_LOCAL0
# log_level = INFO
# accounts_per_second = 200
# recon_cache_path = /var/cache/swift

[account-reaper]
# You can override the default log routing for this app here (don't use
set!):
# log_name = account-reaper
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_address = /dev/log
#
# concurrency = 25
# interval = 3600
# node_timeout = 10
# conn_timeout = 0.5
#
# Normally, the reaper begins deleting account information for deleted
accounts
# immediately; you can set this to delay its work however. The value is in
# seconds; 2592000 = 30 days for example.
# delay_reaping = 0
#
# If the account fails to be reaped due to a persistent error, the
# account reaper will log a message such as:
#     Account <name> has not been reaped since <date>
# You can search logs for this message if space is not being reclaimed
# after you delete account(s).
# Default is 2592000 seconds (30 days). This is in addition to any time
# requested by delay_reaping.
# reap_warn_after = 2592000

# Note: Put it at the beginning of the pipeline to profile all middleware.
But
# it is safer to put this after healthcheck.
[filter:xprofile]
use = egg:swift#xprofile
# This option enable you to switch profilers which should inherit from
python
# standard profiler. Currently the supported value can be 'cProfile',
# 'eventlet.green.profile' etc.
# profile_module = eventlet.green.profile
#
# This prefix will be used to combine process ID and timestamp to name the

```

```

# profile data file. Make sure the executing user has permission to write
# into this path (missing path segments will be created, if necessary).
# If you enable profiling in more than one type of daemon, you must
# override
# it with an unique value like: /var/log/swift/profile/account.profile
# log_filename_prefix = /tmp/log/swift/profile/default.profile
#
# the profile data will be dumped to local disk based on above naming rule
# in this interval.
# dump_interval = 5.0
#
# Be careful, this option will enable profiler to dump data into the file
# with
# time stamp which means there will be lots of files piled up in the
# directory.
# dump_timestamp = false
#
# This is the path of the URL to access the mini web UI.
# path = /__profile__
#
# Clear the data when the wsgi server shutdown.
# flush_at_shutdown = false
#
# unwind the iterator of applications
# unwind = false

```

10.9. PROXY SERVER CONFIGURATION

Find an example proxy server configuration at `etc/proxy-server.conf-sample` in the source code repository.

The available configuration options are:

Table 10.47. Description of configuration options for [DEFAULT] in `proxy-server.conf`

Configuration option = Default value	Description
admin_key = <i>secret_admin_key</i>	to use for admin calls that are HMAC signed. Default is empty, which will disable admin calls to <code>/info</code> . the proxy server. For most cases, this should be
backlog = 4096	Maximum number of allowed pending TCP connections
bind_ip = 0.0.0.0	IP Address for server to bind to
bind_port = 8080	Port for server to bind to
bind_timeout = 30	Seconds to attempt bind before giving up
cert_file = <i>/etc/swift/proxy.crt</i>	to the <code>ssl.crt</code> . This should be enabled for testing purposes only.

Configuration option = Default value	Description
client_timeout = 60	Timeout to read one chunk from a client external services
cors_allow_origin =	is a list of hosts that are included with any CORS request by default and returned with the Access-Control-Allow-Origin header in addition to what the container has set. to call to setup custom log handlers. for eventlet the proxy server. For most cases, this should be
disallowed_sections = <i>swift.valid_api_versions, container_quotas, tempurl</i>	No help text available for this option.
eventlet_debug = false	If true, turn on debug logging for eventlet
expiring_objects_account_name = <i>expiring_objects</i>	No help text available for this option.
expiring_objects_container_divisor = 86400	No help text available for this option.
expose_info = true	Enables exposing configuration settings via HTTP GET /info.
key_file = /etc/swift/proxy.key	to the ssl .key. This should be enabled for testing purposes only.
log_address = /dev/log	Location where syslog sends the logs to
log_custom_handlers =	Comma-separated list of functions to call to setup custom log handlers.
log_facility = LOG_LOCAL0	Syslog log facility
log_headers = false	No help text available for this option.
log_level = INFO	Logging level
log_max_line_length = 0	Caps the length of log lines to the value given; no limit if set to 0, the default.
log_name = swift	Label used when logging
log_statsd_default_sample_rate = 1.0	Defines the probability of sending a sample for any given event or timing measurement.
log_statsd_host = localhost	If not set, the StatsD feature is disabled.

Configuration option = Default value	Description
<code>log_statsd_metric_prefix =</code>	Value will be prepended to every metric sent to the StatsD server.
<code>log_statsd_port = 8125</code>	Port value for the StatsD server.
<code>log_statsd_sample_rate_factor = 1.0</code>	Not recommended to set this to a value less than 1.0, if frequency of logging is too high, tune the <code>log_statsd_default_sample_rate</code> instead.
<code>log_udp_host =</code>	If not set, the UDP receiver for syslog is disabled.
<code>log_udp_port = 514</code>	Port value for UDP receiver, if enabled.
<code>max_clients = 1024</code>	Maximum number of clients one worker can process simultaneously Lowering the number of clients handled per worker, and raising the number of workers can lessen the impact that a CPU intensive, or blocking, request can have on other requests served by the same worker. If the maximum number of clients is set to one, then a given worker will not perform another call while processing, allowing other workers a chance to process it.
<code>strict_cors_mode = True</code>	No help text available for this option.
<code>swift_dir = /etc/swift</code>	Swift configuration directory
<code>trans_id_suffix =</code>	No help text available for this option.
<code>user = swift</code>	User to run as
<code>workers = auto</code>	a much higher value, one can reduce the impact of slow file system operations in one request from negatively impacting other requests.

Table 10.48. Description of configuration options for `[app-proxy-server]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>account_autocreate = false</code>	If set to 'true' authorized accounts that do not yet exist within the Swift cluster will be automatically created.
<code>allow_account_management = false</code>	Whether account PUTs and DELETEs are even callable

Configuration option = Default value	Description
auto_create_account_prefix = .	Prefix to use when automatically creating accounts
client_chunk_size = 65536	Chunk size to read from clients
conn_timeout = 0.5	Connection timeout to external services
deny_host_headers =	No help text available for this option.
error_suppression_interval = 60	Time in seconds that must elapse since the last error for a node to be considered no longer error limited
error_suppression_limit = 10	Error count to consider a node error limited
log_handoffs = true	No help text available for this option.
max_containers_per_account = 0	If set to a positive value, trying to create a container when the account already has at least this maximum containers will result in a 403 Forbidden. Note: This is a soft limit, meaning a user might exceed the cap for recheck_account_existence before the 403s kick in.
max_containers_whitelist =	is a comma separated list of account names that ignore the max_containers_per_account cap.
max_large_object_get_time = 86400	No help text available for this option.
node_timeout = 10	Request timeout to external services
object_chunk_size = 65536	Chunk size to read from object servers
object_post_as_copy = true	Set object_post_as_copy = false to turn on fast posts where only the metadata changes are stored anew and the original data file is kept in place. This makes for quicker posts; but since the container metadata isn't updated in this mode, features like container sync won't be able to sync posts.
post_quorum_timeout = 0.5	No help text available for this option.
put_queue_depth = 10	No help text available for this option.
read_affinity = ``r1z1=100, r1z2=200, r2=300``	No help text available for this option.
recheck_account_existence = 60	Cache timeout in seconds to send memcached for account existence

Configuration option = Default value	Description
recheck_container_existence = 60	Cache timeout in seconds to send memcached for container existence
recoverable_node_timeout = <i>node_timeout</i>	Request timeout to external services for requests that, on failure, can be recovered from. For example, object GET. from a client external services
request_node_count = 2 * <i>replicas</i>	<i>replicas</i> Set to the number of nodes to contact for a normal request. You can use '* replicas' at the end to have it use the number given times the number of replicas for the ring being used for the request. conf file for values will only be shown to the list of <i>swift_owners</i> . The exact default definition of a <i>swift_owner</i> is headers> up to the auth system in use, but usually indicates administrative responsibilities. paste.deploy to use for auth. To use tempauth set to:
set log_address = /dev/log	Location where syslog sends the logs to
set log_facility = LOG_LOCAL0	Syslog log facility
set log_level = INFO	Log level
set log_name = proxy-server	Label to use when logging
sorting_method = <i>shuffle</i>	No help text available for this option.
swift_owner_headers = ``x-container-read, x-container-write, x-container-sync-key, x-container-sync-to, x-account-meta-temp-url-key, x-account-meta-temp-url-key-2, x-container-meta-temp-url-key, x-container-meta-temp-url-key-2, x-account-access-control``	These are the headers whose conf file for values will only be shown to the list of <i>swift_owners</i> . The exact default definition of a <i>swift_owner</i> is headers> up to the auth system in use, but usually indicates administrative responsibilities. paste.deploy to use for auth. To use tempauth set to:
timing_expiry = 300	No help text available for this option.
use = egg:swift#proxy	Entry point of paste.deploy in the server

Configuration option = Default value	Description
<code>write_affinity = r1, r2</code>	This setting lets you trade data distribution for throughput. It makes the proxy server prefer local back-end servers for object PUT requests over non-local ones. Note that only object PUT requests are affected by the <code>write_affinity</code> setting; POST, GET, HEAD, DELETE, OPTIONS, and account/container PUT requests are not affected. The format is <code>r<N></code> for region N or <code>r<N>z<M></code> for region N, zone M. If this is set, then when handling an object PUT request, some number (see the <code>write_affinity_node_count</code> setting) of local backend servers will be tried before any nonlocal ones. Example: try to write to regions 1 and 2 before writing to any other nodes: <code>write_affinity = r1, r2</code>
<code>write_affinity_node_count = 2 * replicas</code>	This setting is only useful in conjunction with <code>write_affinity</code> ; it governs how many local object servers will be tried before falling back to non-local ones. You can use <code>'* replicas'</code> at the end to have it use the number given times the number of replicas for the ring being used for the request: <code>write_affinity_node_count = 2 * replicas</code>

Table 10.49. Description of configuration options for `[pipeline-main]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>pipeline = catch_errors gatekeeper healthcheck proxy-logging cache container_sync bulk tempurl ratelimit tempauth container-quotas account-quotas slo dlo versioned_writes proxy-logging proxy-server</code>	No help text available for this option.

Table 10.50. Description of configuration options for `[filter-account-quotas]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>use = egg:swift#account_quotas</code>	Entry point of paste.deploy in the server

Table 10.51. Description of configuration options for `[filter-authtoken]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>admin_password = password</code>	No help text available for this option.
<code>admin_tenant_name = service</code>	No help text available for this option.
<code>admin_user = swift</code>	No help text available for this option.
<code>auth_uri = http://keystonehost:5000/</code>	No help text available for this option.
<code>cache = swift.cache</code>	No help text available for this option.
<code>delay_auth_decision = False</code>	No help text available for this option.
<code>identity_uri = http://keystonehost:35357/</code>	No help text available for this option.
<code>include_service_catalog = False</code>	No help text available for this option.

Table 10.52. Description of configuration options for `[filter-cache]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>memcache_max_connections = 2</code>	Max number of connections to each memcached server per worker services
<code>memcache_serialization_support = 2</code>	Sets how memcache values are serialized and deserialized
<code>memcache_servers = 127.0.0.1:11211</code>	Comma-separated list of memcached servers ip:port services
<code>set log_address = /dev/log</code>	Location where syslog sends the logs to
<code>set log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>set log_headers = false</code>	If True, log headers in each request
<code>set log_level = INFO</code>	Log level
<code>set log_name = cache</code>	Label to use when logging
<code>use = egg:swift#memcache</code>	Entry point of paste.deploy in the server

Table 10.53. Description of configuration options for `[filter-catch_errors]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>set log_address = /dev/log</code>	Location where syslog sends the logs to
<code>set log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>set log_headers = false</code>	If True, log headers in each request
<code>set log_level = INFO</code>	Log level
<code>set log_name = catch_errors</code>	Label to use when logging
<code>use = egg:swift#catch_errors</code>	Entry point of paste.deploy in the server

Table 10.54. Description of configuration options for `[filter-container_sync]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>allow_full_urls = true</code>	No help text available for this option.
<code>current = //REALM/CLUSTER</code>	No help text available for this option.
<code>use = egg:swift#container_sync</code>	Entry point of paste.deploy in the server

Table 10.55. Description of configuration options for `[filter-dlo]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>max_get_time = 86400</code>	No help text available for this option.
<code>rate_limit_after_segment = 10</code>	Rate limit the download of large object segments after this segment is downloaded.
<code>rate_limit_segments_per_sec = 1</code>	Rate limit large object downloads at this rate. contact for a normal request. You can use '*' replicas' at the end to have it use the number given times the number of replicas for the ring being used for the request. paste.deploy to use for auth. To use tempauth set to:
<code>use = egg:swift#dlo</code>	Entry point of paste.deploy in the server

Table 10.56. Description of configuration options for `[filter-gatekeeper]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>set log_address = /dev/log</code>	Location where syslog sends the logs to
<code>set log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>set log_headers = false</code>	If True, log headers in each request
<code>set log_level = INFO</code>	Log level
<code>set log_name = gatekeeper</code>	Label to use when logging
<code>use = egg:swift#gatekeeper</code>	Entry point of paste.deploy in the server

Table 10.57. Description of configuration options for `[filter-healthcheck]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>disable_path =</code>	No help text available for this option.
<code>use = egg:swift#healthcheck</code>	Entry point of paste.deploy in the server

Table 10.58. Description of configuration options for `[filter-keystoneauth]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>allow_names_in_acls = true</code>	The backwards compatible behavior can be disabled by setting this option to False.
<code>allow_overrides = true</code>	This option allows middleware higher in the WSGI pipeline to override auth processing, useful for middleware such as <code>tempurl</code> and <code>formpost</code> . If you know you are not going to use such middleware and you want a bit of extra security, you can set this to False.
<code>default_domain_id = default</code>	Name of the default domain. It is identified by its UUID, which by default has the value "default".
<code>is_admin = false</code>	If this option is set to True, it allows to give a user whose username is the same as the project name and who has any role in the project access rights elevated to be the same as if the user had one of the <code>operator_roles</code> . Note that the condition compares names rather than UUIDs. This option is deprecated. It is False by default.

Configuration option = Default value	Description
operator_roles = <i>admin, swiftoperator</i>	Operator role defines the user which is allowed to manage a tenant and create containers or give ACL to others. This parameter may be prefixed with an appropriate prefix.
reseller_admin_role = <i>ResellerAdmin</i>	The reseller admin role gives the ability to create and delete accounts.
reseller_prefix = <i>AUTH</i>	The naming scope for the auth service. Swift
service_roles =	When present, this option requires that the X-Service-Token header supplies a token from a user who has a role listed in service_roles. This parameter may be prefixed with an appropriate prefix.
use = <i>egg:swift#keystoneauth</i>	Entry point of paste.deploy in the server

Table 10.59. Description of configuration options for **[filter-list-endpoints]** in **proxy-server.conf**

Configuration option = Default value	Description
list_endpoints_path = <i>/endpoints/</i>	No help text available for this option.
use = <i>egg:swift#list_endpoints</i>	Entry point of paste.deploy in the server

Table 10.60. Description of configuration options for **[filter-proxy-logging]** in **proxy-server.conf**

Configuration option = Default value	Description
access_log_address = <i>/dev/log</i>	No help text available for this option.
access_log_facility = <i>LOG_LOCAL0</i>	No help text available for this option.
access_log_headers = <i>false</i>	No help text available for this option.
access_log_headers_only =	If access_log_headers is True and access_log_headers_only is set only these headers are logged. Multiple headers can be defined as comma separated list like this: access_log_headers_only = Host, X-Object-Meta-Mtime
access_log_level = <i>INFO</i>	No help text available for this option.

Configuration option = Default value	Description
<code>access_log_name = swift</code>	No help text available for this option.
<code>access_log_statsd_default_sample_rate = 1.0</code>	No help text available for this option.
<code>access_log_statsd_host = localhost</code>	No help text available for this option.
<code>access_log_statsd_metric_prefix =</code>	No help text available for this option.
<code>access_log_statsd_port = 8125</code>	No help text available for this option.
<code>access_log_statsd_sample_rate_factor = 1.0</code>	No help text available for this option.
<code>access_log_udp_host =</code>	No help text available for this option.
<code>access_log_udp_port = 514</code>	No help text available for this option.
<code>log_statsd_valid_http_methods = GET,HEAD,POST,PUT,DELETE,COPY,OPTIONS</code>	No help text available for this option.
<code>logged with access_log_headers = True.</code>	No help text available for this option.
<code>reveal_sensitive_prefix = 16</code>	The X-Auth-Token is sensitive data. If revealed to an unauthorised person, they can now make requests against an account until the token expires. Set <code>reveal_sensitive_prefix</code> to the number of characters of the token that are logged. For example <code>reveal_sensitive_prefix = 12</code> so only first 12 characters of the token are logged. Or, set to 0 to completely remove the token.
<code>use = egg:swift#proxy_logging</code>	Entry point of paste.deploy in the server

Table 10.61. Description of configuration options for [filter-tempauth] in proxy-server.conf

Configuration option = Default value	Description
<code>allow_overrides = true</code>	This option allows middleware higher in the WSGI pipeline to override auth processing, useful for middleware such as tempurl and formpost. If you know you are not going to use such middleware and you want a bit of extra security, you can set this to False.

Configuration option = Default value	Description
auth_prefix = <i>/auth/</i>	The HTTP request path prefix for the auth service. Swift itself reserves anything beginning with the letter
require_group =	No help text available for this option.
reseller_prefix = <i>AUTH</i>	The naming scope for the auth service. Swift
set log_address = <i>/dev/log</i>	Location where syslog sends the logs to
set log_facility = <i>LOG_LOCAL0</i>	Syslog log facility
set log_headers = <i>false</i>	If True, log headers in each request
set log_level = <i>INFO</i>	Log level
set log_name = <i>tempauth</i>	Label to use when logging
storage_url_scheme = <i>default</i>	Scheme to return with storage urls: http, https, or default (chooses based on what the server is running as) This can be useful with an SSL load balancer in front of a non-SSL server.
token_life = <i>86400</i>	The number of seconds a token is valid.
use = <i>egg:swift#tempauth</i>	Entry point of paste.deploy in the server
user_admin_admin = <i>admin.admin</i> <i>.reseller_admin</i>	No help text available for this option.
user_test2_tester2 = <i>testing2.admin</i>	No help text available for this option.
user_test5_tester5 = <i>testing5 service</i>	No help text available for this option.
user_test_tester = <i>testing.admin</i>	No help text available for this option.
user_test_tester3 = <i>testing3</i>	No help text available for this option.

Table 10.62. Description of configuration options for `[filter-xprofile]` in `proxy-server.conf`

Configuration option = Default value	Description
dump_interval = <i>5.0</i>	No help text available for this option.

Configuration option = Default value	Description
dump_timestamp = <i>false</i>	No help text available for this option.
flush_at_shutdown = <i>false</i>	No help text available for this option.
log_filename_prefix = <i>/tmp/log/swift/profile/default.profile</i>	No help text available for this option.
path = <i>/__profile__</i>	No help text available for this option.
profile_module = <i>eventlet.green.profile</i>	No help text available for this option.
unwind = <i>false</i>	No help text available for this option.
use = <i>egg:swift#xprofile</i>	Entry point of paste.deploy in the server

10.9.1. Sample proxy server configuration file

```
[DEFAULT]
# bind_ip = 0.0.0.0
bind_port = 8080
# bind_timeout = 30
# backlog = 4096
# swift_dir = /etc/swift
# user = swift

# Enables exposing configuration settings via HTTP GET /info.
# expose_info = true

# Key to use for admin calls that are HMAC signed. Default is empty,
# which will disable admin calls to /info.
# admin_key = secret_admin_key
#
# Allows the ability to withhold sections from showing up in the public
calls
# to /info. You can withhold subsections by separating the dict level
with a
# ".". The following would cause the sections 'container_quotas' and
'tempurl'
# to not be listed, and the key max_failed_deletes would be removed from
# bulk_delete. Default value is 'swift.valid_api_versions' which allows
all
# registered features to be listed via HTTP GET /info except
# swift.valid_api_versions information
# disallowed_sections = swift.valid_api_versions, container_quotas,
tempurl

# Use an integer to override the number of pre-forked processes that will
# accept connections. Should default to the number of effective cpu
# cores in the system. It's worth noting that individual workers will
# use many eventlet co-routines to service multiple concurrent requests.
```

```
# workers = auto
#
# Maximum concurrent requests per worker
# max_clients = 1024
#
# Set the following two lines to enable SSL. This is for testing only.
# cert_file = /etc/swift/proxy.crt
# key_file = /etc/swift/proxy.key
#
# expiring_objects_container_divisor = 86400
# expiring_objects_account_name = expiring_objects
#
# You can specify default log routing here if you want:
# log_name = swift
# log_facility = LOG_LOCAL0
# log_level = INFO
# log_headers = false
# log_address = /dev/log
# The following caps the length of log lines to the value given; no limit
if
# set to 0, the default.
# log_max_line_length = 0
#
# This optional suffix (default is empty) that would be appended to the
swift transaction
# id allows one to easily figure out from which cluster that X-Trans-Id
belongs to.
# This is very useful when one is managing more than one swift cluster.
# trans_id_suffix =
#
# comma separated list of functions to call to setup custom log handlers.
# functions get passed: conf, name, log_to_console, log_route, fmt,
logger,
# adapted_logger
# log_custom_handlers =
#
# If set, log_udp_host will override log_address
# log_udp_host =
# log_udp_port = 514
#
# You can enable StatsD logging here:
# log_statsd_host = localhost
# log_statsd_port = 8125
# log_statsd_default_sample_rate = 1.0
# log_statsd_sample_rate_factor = 1.0
# log_statsd_metric_prefix =
#
# Use a comma separated list of full url
(http://foo.bar:1234,https://foo.bar)
# cors_allow_origin =
# strict_cors_mode = True
#
# client_timeout = 60
# eventlet_debug = false

[pipeline:main]
```



```

pipeline = catch_errors gatekeeper healthcheck proxy-logging cache
container_sync bulk tempurl ratelimit tempauth container-quotas account-
quotas slo dlo proxy-logging proxy-server

[app:proxy-server]
use = egg:swift#proxy
# You can override the default log routing for this app here:
# set log_name = proxy-server
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_address = /dev/log
#
# log_handoffs = true
# recheck_account_existence = 60
# recheck_container_existence = 60
# object_chunk_size = 65536
# client_chunk_size = 65536
#
# How long the proxy server will wait on responses from the a/c/o servers.
# node_timeout = 10
#
# How long the proxy server will wait for an initial response and to read
a
# chunk of data from the object servers while serving GET / HEAD requests.
# Timeouts from these requests can be recovered from so setting this to
# something lower than node_timeout would provide quicker error recovery
# while allowing for a longer timeout for non-recoverable requests (PUTs).
# Defaults to node_timeout, should be overridden if node_timeout is set to
a
# high number to prevent client timeouts from firing before the proxy
server
# has a chance to retry.
# recoverable_node_timeout = node_timeout
#
# conn_timeout = 0.5
#
# How long to wait for requests to finish after a quorum has been
established.
# post_quorum_timeout = 0.5
#
# How long without an error before a node's error count is reset. This
will
# also be how long before a node is reenabled after suppression is
triggered.
# error_suppression_interval = 60
#
# How many errors can accumulate before a node is temporarily ignored.
# error_suppression_limit = 10
#
# If set to 'true' any authorized user may create and delete accounts; if
# 'false' no one, even authorized, can.
# allow_account_management = false
#
# Set object_post_as_copy = false to turn on fast posts where only the
metadata
# changes are stored anew and the original data file is kept in place.

```

```
This
# makes for quicker posts; but since the container metadata isn't updated
in
# this mode, features like container sync won't be able to sync posts.
# object_post_as_copy = true
#
# If set to 'true' authorized accounts that do not yet exist within the
Swift
# cluster will be automatically created.
# account_autocreate = false
#
# If set to a positive value, trying to create a container when the
account
# already has at least this maximum containers will result in a 403
Forbidden.
# Note: This is a soft limit, meaning a user might exceed the cap for
# recheck_account_existence before the 403s kick in.
# max_containers_per_account = 0
#
# This is a comma separated list of account hashes that ignore the
# max_containers_per_account cap.
# max_containers_whitelist =
#
# Comma separated list of Host headers to which the proxy will deny
requests.
# deny_host_headers =
#
# Prefix used when automatically creating accounts.
# auto_create_account_prefix = .
#
# Depth of the proxy put queue.
# put_queue_depth = 10
#
# Storage nodes can be chosen at random (shuffle), by using timing
# measurements (timing), or by using an explicit match (affinity).
# Using timing measurements may allow for lower overall latency, while
# using affinity allows for finer control. In both the timing and
# affinity cases, equally-sorting nodes are still randomly chosen to
# spread load.
# The valid values for sorting_method are "affinity", "shuffle", and
"timing".
# sorting_method = shuffle
#
# If the "timing" sorting_method is used, the timings will only be valid
for
# the number of seconds configured by timing_expiry.
# timing_expiry = 300
#
# The maximum time (seconds) that a large object connection is allowed to
last.
# max_large_object_get_time = 86400
#
# Set to the number of nodes to contact for a normal request. You can use
# '* replicas' at the end to have it use the number given times the number
of
# replicas for the ring being used for the request.
```

```

# request_node_count = 2 * replicas
#
# Which backend servers to prefer on reads. Format is r<N> for region
# N or r<N>z<M> for region N, zone M. The value after the equals is
# the priority; lower numbers are higher priority.
#
# Example: first read from region 1 zone 1, then region 1 zone 2, then
# anything in region 2, then everything else:
# read_affinity = r1z1=100, r1z2=200, r2=300
# Default is empty, meaning no preference.
# read_affinity =
#
# Which backend servers to prefer on writes. Format is r<N> for region
# N or r<N>z<M> for region N, zone M. If this is set, then when
# handling an object PUT request, some number (see setting
# write_affinity_node_count) of local backend servers will be tried
# before any nonlocal ones.
#
# Example: try to write to regions 1 and 2 before writing to any other
# nodes:
# write_affinity = r1, r2
# Default is empty, meaning no preference.
# write_affinity =
#
# The number of local (as governed by the write_affinity setting)
# nodes to attempt to contact first, before any non-local ones. You
# can use '* replicas' at the end to have it use the number given
# times the number of replicas for the ring being used for the
# request.
# write_affinity_node_count = 2 * replicas
#
# These are the headers whose values will only be shown to swift_owners.
# The
# exact definition of a swift_owner is up to the auth system in use, but
# usually indicates administrative responsibilities.
# swift_owner_headers = x-container-read, x-container-write, x-container-
# sync-key, x-container-sync-to, x-account-meta-temp-url-key, x-account-
# meta-temp-url-key-2, x-container-meta-temp-url-key, x-container-meta-temp-
# url-key-2, x-account-access-control

[filter:tempauth]
use = egg:swift#tempauth
# You can override the default log routing for this filter here:
# set log_name = tempauth
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_headers = false
# set log_address = /dev/log
#
# The reseller prefix will verify a token begins with this prefix before
# even
# attempting to validate it. Also, with authorization, only Swift storage
# accounts with this prefix will be authorized by this middleware. Useful
# if
# multiple auth systems are in use for one Swift cluster.
# The reseller_prefix may contain a comma separated list of items. The

```

```

first
# item is used for the token as mentioned above. If second and subsequent
# items exist, the middleware will handle authorization for an account
with
# that prefix. For example, for prefixes "AUTH, SERVICE", a path of
# /v1/SERVICE_account is handled the same as /v1/AUTH_account. If an empty
# (blank) reseller prefix is required, it must be first in the list. Two
# single quote characters indicates an empty (blank) reseller prefix.
# reseller_prefix = AUTH

#
# The require_group parameter names a group that must be presented by
# either X-Auth-Token or X-Service-Token. Usually this parameter is
# used only with multiple reseller prefixes (e.g.,
SERVICE_require_group=blah).
# By default, no group is needed. Do not use .admin.
# require_group =

# The auth prefix will cause requests beginning with this prefix to be
routed
# to the auth subsystem, for granting tokens, etc.
# auth_prefix = /auth/
# token_life = 86400
#
# This allows middleware higher in the WSGI pipeline to override auth
# processing, useful for middleware such as tempurl and formpost. If you
know
# you're not going to use such middleware and you want a bit of extra
security,
# you can set this to false.
# allow_overrides = true
#
# This specifies what scheme to return with storage urls:
# http, https, or default (chooses based on what the server is running as)
# This can be useful with an SSL load balancer in front of a non-SSL
server.
# storage_url_scheme = default
#
# Lastly, you need to list all the accounts/users you want here. The
format is:
#   user_<account>_<user> = <key> [group] [group] [...] [storage_url]
# or if you want underscores in <account> or <user>, you can base64 encode
them
# (with no equal signs) and use this format:
#   user64_<account_b64>_<user_b64> = <key> [group] [group] [...]
[storage_url]
# There are special groups of:
#   .reseller_admin = can do anything to any account for this auth
#   .admin = can do anything within the account
# If neither of these groups are specified, the user can only access
containers
# that have been explicitly allowed for them by a .admin or
.reseller_admin.
# The trailing optional storage_url allows you to specify an alternate url
to
# hand back to the user upon authentication. If not specified, this

```

```

defaults to
# $HOST/v1/<reseller_prefix>_<account> where $HOST will do its best to
# resolve
# to what the requester would need to use to reach this host.
# Here are example entries, required for running the tests:
user_admin_admin = admin .admin .reseller_admin
user_test_tester = testing .admin
user_test2_tester2 = testing2 .admin
user_test_tester3 = testing3
user_test5_tester5 = testing5 service

# To enable Keystone authentication you need to have the auth token
# middleware first to be configured. Here is an example below, please
# refer to the keystone's documentation for details about the
# different settings.
#
# You'll need to have as well the keystoneauth middleware enabled
# and have it in your main pipeline so instead of having tempauth in
# there you can change it to: authtoken keystoneauth
#
# [filter:authtoken]
# paste.filter_factory = keystonemiddleware.auth_token:filter_factory
# identity_uri = http://keystonehost:35357/
# auth_uri = http://keystonehost:5000/
# admin_tenant_name = service
# admin_user = swift
# admin_password = password
#
# delay_auth_decision defaults to False, but leaving it as false will
# prevent other auth systems, staticweb, tempurl, formpost, and ACLs from
# working. This value must be explicitly set to True.
# delay_auth_decision = False
#
# cache = swift.cache
# include_service_catalog = False
#
# [filter:keystoneauth]
# use = egg:swift#keystoneauth
# The reseller_prefix option lists account namespaces that this middleware
# is
# responsible for. The prefix is placed before the Keystone project id.
# For example, for project 12345678, and prefix AUTH, the account is
# named AUTH_12345678 (i.e., path is /v1/AUTH_12345678/...).
# Several prefixes are allowed by specifying a comma-separated list
# as in: "reseller_prefix = AUTH, SERVICE". The empty string indicates a
# single blank/empty prefix. If an empty prefix is required in a list of
# prefixes, a value of '' (two single quote characters) indicates a
# blank/empty prefix. Except for the blank/empty prefix, an underscore
# ('_')
# character is appended to the value unless already present.
# reseller_prefix = AUTH
#
# The user must have at least one role named by operator_roles on a
# project in order to create, delete and modify containers and objects
# and to set and read privileged headers such as ACLs.
# If there are several reseller prefix items, you can prefix the

```

```

# parameter so it applies only to those accounts (for example
# the parameter SERVICE_operator_roles applies to the
/v1/SERVICE_<project>
# path). If you omit the prefix, the option applies to all reseller
# prefix items. For the blank/empty prefix, prefix with '' (do not put
# underscore after the two single quote characters).
# operator_roles = admin, swiftoperator
#
# The reseller admin role has the ability to create and delete accounts
# reseller_admin_role = ResellerAdmin
#
# This allows middleware higher in the WSGI pipeline to override auth
# processing, useful for middleware such as tempurl and formpost. If you
know
# you're not going to use such middleware and you want a bit of extra
security,
# you can set this to false.
# allow_overrides = true
#
# If is_admin is true, a user whose username is the same as the project
name
# and who has any role on the project will have access rights elevated to
be
# the same as if the user had an operator role. Note that the condition
# compares names rather than UUIDs. This option is deprecated.
# is_admin = false
#
# If the service_roles parameter is present, an X-Service-Token must be
# present in the request that when validated, grants at least one role
listed
# in the parameter. The X-Service-Token may be scoped to any project.
# If there are several reseller prefix items, you can prefix the
# parameter so it applies only to those accounts (for example
# the parameter SERVICE_service_roles applies to the /v1/SERVICE_<project>
# path). If you omit the prefix, the option applies to all reseller
# prefix items. For the blank/empty prefix, prefix with '' (do not put
# underscore after the two single quote characters).
# By default, no service_roles are required.
# service_roles =
#
# For backwards compatibility, keystoneauth will match names in cross-
tenant
# access control lists (ACLs) when both the requesting user and the tenant
# are in the default domain i.e the domain to which existing tenants are
# migrated. The default_domain_id value configured here should be the same
as
# the value used during migration of tenants to keystone domains.
# default_domain_id = default
#
# For a new installation, or an installation in which keystone projects
may
# move between domains, you should disable backwards compatible name
matching
# in ACLs by setting allow_names_in_acls to false:
# allow_names_in_acls = true

```

```

[filter:healthcheck]
use = egg:swift#healthcheck
# An optional filesystem path, which if present, will cause the
healthcheck
# URL to return "503 Service Unavailable" with a body of "DISABLED BY
FILE".
# This facility may be used to temporarily remove a Swift node from a load
# balancer pool during maintenance or upgrade (remove the file to allow
the
# node back into the load balancer pool).
# disable_path =

[filter:cache]
use = egg:swift#memcache
# You can override the default log routing for this filter here:
# set log_name = cache
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_headers = false
# set log_address = /dev/log
#
# If not set here, the value for memcache_servers will be read from
# memcache.conf (see memcache.conf-sample) or lacking that file, it will
# default to the value below. You can specify multiple servers separated
with
# commas, as in: 10.1.2.3:11211,10.1.2.4:11211
# memcache_servers = 127.0.0.1:11211
#
# Sets how memcache values are serialized and deserialized:
# 0 = older, insecure pickle serialization
# 1 = json serialization but pickles can still be read (still insecure)
# 2 = json serialization only (secure and the default)
# If not set here, the value for memcache_serialization_support will be
read
# from /etc/swift/memcache.conf (see memcache.conf-sample).
# To avoid an instant full cache flush, existing installations should
# upgrade with 0, then set to 1 and reload, then after some time (24
hours)
# set to 2 and reload.
# In the future, the ability to use pickle serialization will be removed.
# memcache_serialization_support = 2
#
# Sets the maximum number of connections to each memcached server per
worker
# memcache_max_connections = 2
#
# More options documented in memcache.conf-sample

[filter:ratelimit]
use = egg:swift#ratelimit
# You can override the default log routing for this filter here:
# set log_name = ratelimit
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_headers = false
# set log_address = /dev/log

```

```

#
# clock_accuracy should represent how accurate the proxy servers' system
# clocks
# are with each other. 1000 means that all the proxies' clock are accurate
# to
# each other within 1 millisecond. No ratelimit should be higher than the
# clock accuracy.
# clock_accuracy = 1000
#
# max_sleep_time_seconds = 60
#
# log_sleep_time_seconds of 0 means disabled
# log_sleep_time_seconds = 0
#
# allows for slow rates (e.g. running up to 5 sec's behind) to catch up.
# rate_buffer_seconds = 5
#
# account_ratelimit of 0 means disabled
# account_ratelimit = 0

# DEPRECATED- these will continue to work but will be replaced
# by the X-Account-Sysmeta-Global-Write-Ratelimit flag.
# Please see ratelimiting docs for details.
# these are comma separated lists of account names
# account_whitelist = a,b
# account_blacklist = c,d

# with container_limit_x = r
# for containers of size x limit write requests per second to r. The
# container
# rate will be linearly interpolated from the values given. With the
# values
# below, a container of size 5 will get a rate of 75.
# container_ratelimit_0 = 100
# container_ratelimit_10 = 50
# container_ratelimit_50 = 20

# Similarly to the above container-level write limits, the following will
# limit
# container GET (listing) requests.
# container_listing_ratelimit_0 = 100
# container_listing_ratelimit_10 = 50
# container_listing_ratelimit_50 = 20

[filter:domain_remap]
use = egg:swift#domain_remap
# You can override the default log routing for this filter here:
# set log_name = domain_remap
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_headers = false
# set log_address = /dev/log
#
# storage_domain = example.com
# path_root = v1
# reseller_prefixes = AUTH

```



```

[filter:catch_errors]
use = egg:swift#catch_errors
# You can override the default log routing for this filter here:
# set log_name = catch_errors
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_headers = false
# set log_address = /dev/log

[filter:cname_lookup]
# Note: this middleware requires python-dnspython
use = egg:swift#cname_lookup
# You can override the default log routing for this filter here:
# set log_name = cname_lookup
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_headers = false
# set log_address = /dev/log
#
# Specify the storage_domain that match your cloud, multiple domains
# can be specified separated by a comma
# storage_domain = example.com
#
# lookup_depth = 1

# Note: Put staticweb just after your auth filter(s) in the pipeline
[filter:staticweb]
use = egg:swift#staticweb

# Note: Put tempurl before dlo, slo and your auth filter(s) in the
# pipeline
[filter:tempurl]
use = egg:swift#tempurl
# The methods allowed with Temp URLs.
# methods = GET HEAD PUT POST DELETE
#
# The headers to remove from incoming requests. Simply a whitespace
# delimited
# list of header names and names can optionally end with '*' to indicate a
# prefix match. incoming_allow_headers is a list of exceptions to these
# removals.
# incoming_remove_headers = x-timestamp
#
# The headers allowed as exceptions to incoming_remove_headers. Simply a
# whitespace delimited list of header names and names can optionally end
# with
# '*' to indicate a prefix match.
# incoming_allow_headers =
#
# The headers to remove from outgoing responses. Simply a whitespace
# delimited
# list of header names and names can optionally end with '*' to indicate a
# prefix match. outgoing_allow_headers is a list of exceptions to these
# removals.
# outgoing_remove_headers = x-object-meta-*

```

```

#
# The headers allowed as exceptions to outgoing_remove_headers. Simply a
# whitespace delimited list of header names and names can optionally end
# with
# '*' to indicate a prefix match.
# outgoing_allow_headers = x-object-meta-public-*

# Note: Put formpost just before your auth filter(s) in the pipeline
[filter:formpost]
use = egg:swift#formpost

# Note: Just needs to be placed before the proxy-server in the pipeline.
[filter:name_check]
use = egg:swift#name_check
# forbidden_chars = '"`<>
# maximum_length = 255
# forbidden_regexp = /\./|/\.\./|/\.$|/\.\.$

[filter:list-endpoints]
use = egg:swift#list_endpoints
# list_endpoints_path = /endpoints/

[filter:proxy-logging]
use = egg:swift#proxy_logging
# If not set, logging directives from [DEFAULT] without "access_" will be
# used
# access_log_name = swift
# access_log_facility = LOG_LOCAL0
# access_log_level = INFO
# access_log_address = /dev/log
#
# If set, access_log_udp_host will override access_log_address
# access_log_udp_host =
# access_log_udp_port = 514
#
# You can use log_statsd_* from [DEFAULT] or override them here:
# access_log_statsd_host = localhost
# access_log_statsd_port = 8125
# access_log_statsd_default_sample_rate = 1.0
# access_log_statsd_sample_rate_factor = 1.0
# access_log_statsd_metric_prefix =
# access_log_headers = false
#
# If access_log_headers is True and access_log_headers_only is set only
# these headers are logged. Multiple headers can be defined as comma
# separated
# list like this: access_log_headers_only = Host, X-Object-Meta-Mtime
# access_log_headers_only =
#
# By default, the X-Auth-Token is logged. To obscure the value,
# set reveal_sensitive_prefix to the number of characters to log.
# For example, if set to 12, only the first 12 characters of the
# token appear in the log. An unauthorized access of the log file
# won't allow unauthorized usage of the token. However, the first
# 12 or so characters is unique enough that you can trace/debug
# token usage. Set to 0 to suppress the token completely (replaced

```

```

# by '...' in the log).
# Note: reveal_sensitive_prefix will not affect the value
# logged with access_log_headers=True.
# reveal_sensitive_prefix = 16
#
# What HTTP methods are allowed for StatsD logging (comma-sep); request
methods
# not in this list will have "BAD_METHOD" for the <verb> portion of the
metric.
# log_statsd_valid_http_methods = GET,HEAD,POST,PUT,DELETE,COPY,OPTIONS
#
# Note: The double proxy-logging in the pipeline is not a mistake. The
# left-most proxy-logging is there to log requests that were handled in
# middleware and never made it through to the right-most middleware (and
# proxy server). Double logging is prevented for normal requests. See
# proxy-logging docs.

# Note: Put before both ratelimit and auth in the pipeline.
[filter:bulk]
use = egg:swift#bulk
# max_containers_per_extraction = 10000
# max_failed_extractions = 1000
# max_deletes_per_request = 10000
# max_failed_deletes = 1000

# In order to keep a connection active during a potentially long bulk
request,
# Swift may return whitespace prepended to the actual response body. This
# whitespace will be yielded no more than every yield_frequency seconds.
# yield_frequency = 10

# Note: The following parameter is used during a bulk delete of objects
and
# their container. This would frequently fail because it is very likely
# that all replicated objects have not been deleted by the time the
middleware got a
# successful response. It can be configured the number of retries. And the
# number of seconds to wait between each retry will be 1.5**retry

# delete_container_retry_count = 0

# Note: Put after auth and staticweb in the pipeline.
[filter:slo]
use = egg:swift#slo
# max_manifest_segments = 1000
# max_manifest_size = 2097152
# min_segment_size = 1048576
# Start rate-limiting SLO segment serving after the Nth segment of a
# segmented object.
# rate_limit_after_segment = 10
#
# Once segment rate-limiting kicks in for an object, limit segments served
# to N per second. 0 means no rate-limiting.
# rate_limit_segments_per_sec = 0
#
# Time limit on GET requests (seconds)

```

```

# max_get_time = 86400

# Note: Put after auth and staticweb in the pipeline.
# If you don't put it in the pipeline, it will be inserted for you.
[filter:dlo]
use = egg:swift#dlo
# Start rate-limiting DLO segment serving after the Nth segment of a
# segmented object.
# rate_limit_after_segment = 10
#
# Once segment rate-limiting kicks in for an object, limit segments served
# to N per second. 0 means no rate-limiting.
# rate_limit_segments_per_sec = 1
#
# Time limit on GET requests (seconds)
# max_get_time = 86400

# Note: Put after auth in the pipeline.
[filter:container-quotas]
use = egg:swift#container_quotas

# Note: Put after auth in the pipeline.
[filter:account-quotas]
use = egg:swift#account_quotas

[filter:gatekeeper]
use = egg:swift#gatekeeper
# You can override the default log routing for this filter here:
# set log_name = gatekeeper
# set log_facility = LOG_LOCAL0
# set log_level = INFO
# set log_headers = false
# set log_address = /dev/log

[filter:container_sync]
use = egg:swift#container_sync
# Set this to false if you want to disallow any full url values to be set
# for
# any new X-Container-Sync-To headers. This will keep any new full urls
# from
# coming in, but won't change any existing values already in the cluster.
# Updating those will have to be done manually, as knowing what the true
# realm
# endpoint should be cannot always be guessed.
# allow_full_urls = true
# Set this to specify this clusters //realm/cluster as "current" in /info
# current = //REALM/CLUSTER

# Note: Put it at the beginning of the pipeline to profile all middleware.
# But
# it is safer to put this after catch_errors, gatekeeper and healthcheck.
[filter:xprofile]
use = egg:swift#xprofile
# This option enable you to switch profilers which should inherit from
# python
# standard profiler. Currently the supported value can be 'cProfile',

```

```

# 'eventlet.green.profile' etc.
# profile_module = eventlet.green.profile
#
# This prefix will be used to combine process ID and timestamp to name the
# profile data file. Make sure the executing user has permission to write
# into this path (missing path segments will be created, if necessary).
# If you enable profiling in more than one type of daemon, you must
# override
# it with an unique value like: /var/log/swift/profile/proxy.profile
# log_filename_prefix = /tmp/log/swift/profile/default.profile
#
# the profile data will be dumped to local disk based on above naming rule
# in this interval.
# dump_interval = 5.0
#
# Be careful, this option will enable profiler to dump data into the file
# with
# time stamp which means there will be lots of files piled up in the
# directory.
# dump_timestamp = false
#
# This is the path of the URL to access the mini web UI.
# path = /__profile__
#
# Clear the data when the wsgi server shutdown.
# flush_at_shutdown = false
#
# unwind the iterator of applications
# unwind = false

```

10.10. PROXY SERVER MEMCACHE CONFIGURATION

Find an example memcache configuration for the proxy server at `etc/memcache.conf-sample` in the source code repository.

The available configuration options are:

Table 10.63. Description of configuration options for [memcache] in `memcache.conf`

Configuration option = Default value	Description
<code>connect_timeout = 0.3</code>	Timeout in seconds (float) for connection
<code>io_timeout = 2.0</code>	Timeout in seconds (float) for read and write
<code>memcache_max_connections = 2</code>	Max number of connections to each memcached server per worker services
<code>memcache_serialization_support = 2</code>	Sets how memcache values are serialized and deserialized
<code>memcache_servers = 127.0.0.1:11211</code>	Comma-separated list of memcached servers ip:port services

Configuration option = Default value	Description
pool_timeout = 1.0	Timeout in seconds (float) for pooled connection
tries = 3	Number of servers to retry on failures getting a pooled connection

10.11. RSYNC D CONFIGURATION

Find an example rsyncd configuration at `etc/rsyncd.conf-sample` in the source code repository.

The available configuration options are:

Table 10.64. Description of configuration options for [account] in rsyncd.conf

Configuration option = Default value	Description
lock file = <code>/var/lock/account.lock</code>	No help text available for this option.
max connections = 2	No help text available for this option.
path = <code>/srv/node</code>	No help text available for this option.
read only = <code>false</code>	No help text available for this option.

Table 10.65. Description of configuration options for [container] in rsyncd.conf

Configuration option = Default value	Description
lock file = <code>/var/lock/container.lock</code>	No help text available for this option.
max connections = 4	No help text available for this option.
path = <code>/srv/node</code>	No help text available for this option.
read only = <code>false</code>	No help text available for this option.

Table 10.66. Description of configuration options for [object] in rsyncd.conf

Configuration option = Default value	Description
lock file = <code>/var/lock/object.lock</code>	No help text available for this option.
max connections = 8	No help text available for this option.
path = <code>/srv/node</code>	No help text available for this option.

Configuration option = Default value	Description
<code>read_only = false</code>	No help text available for this option.
<code>rsync_module =</code> <code>``{replication_ip}::object_{device}``</code>	Format of the rsync module where the replicator will send data. The configuration value can include some variables that will be extracted from the ring. Variables must follow the format {NAME} where NAME is one of: ip, port, replication_ip, replication_port, region, zone, device, meta. See etc/rsyncd.conf-sample for some examples. uses what's set here, or what's set in the DEFAULT section, or 10 (though other sections use 3 as the final default).

10.12. CONFIGURE OBJECT STORAGE FEATURES

10.12.1. Object Storage zones

In OpenStack Object Storage, data is placed across different tiers of failure domains. First, data is spread across regions, then zones, then servers, and finally across drives. Data is placed to get the highest failure domain isolation. If you deploy multiple regions, the Object Storage service places the data across the regions. Within a region, each replica of the data should be stored in unique zones, if possible. If there is only one zone, data should be placed on different servers. And if there is only one server, data should be placed on different drives.

Regions are widely separated installations with a high-latency or otherwise constrained network link between them. Zones are arbitrarily assigned, and it is up to the administrator of the Object Storage cluster to choose an isolation level and attempt to maintain the isolation level through appropriate zone assignment. For example, a zone may be defined as a rack with a single power source. Or a zone may be a DC room with a common utility provider. Servers are identified by a unique IP/port. Drives are locally attached storage volumes identified by mount point.

In small clusters (five nodes or fewer), everything is normally in a single zone. Larger Object Storage deployments may assign zone designations differently; for example, an entire cabinet or rack of servers may be designated as a single zone to maintain replica availability if the cabinet becomes unavailable (for example, due to failure of the top of rack switches or a dedicated circuit). In very large deployments, such as service provider level deployments, each zone might have an entirely autonomous switching and power infrastructure, so that even the loss of an electrical circuit or switching aggregator would result in the loss of a single replica at most.

10.12.2. RAID controller configuration

OpenStack Object Storage does not require RAID. In fact, most RAID configurations cause significant performance degradation. The main reason for using a RAID controller is the battery-backed cache. It is very important for data integrity reasons that when the operating system confirms a write has been committed that the write has actually been committed to a persistent location. Most disks lie about hardware commits by default, instead writing to a faster write cache for performance reasons. In most cases, that write cache exists only in non-persistent memory. In the case of a loss of power, this data may never actually get committed to disk, resulting in discrepancies that the underlying file system must handle.

OpenStack Object Storage works best on the XFS file system, and this document assumes that the hardware being used is configured appropriately to be mounted with the `nobarrriers` option. For more information, refer to the XFS FAQ: http://xfs.org/index.php/XFS_FAQ

To get the most out of your hardware, it is essential that every disk used in OpenStack Object Storage is configured as a standalone, individual RAID 0 disk; in the case of 6 disks, you would have six RAID 0s or one JBOD. Some RAID controllers do not support JBOD or do not support battery backed cache with JBOD. To ensure the integrity of your data, you must ensure that the individual drive caches are disabled and the battery backed cache in your RAID card is configured and used. Failure to configure the controller properly in this case puts data at risk in the case of sudden loss of power.

You can also use hybrid drives or similar options for battery backed up cache configurations without a RAID controller.

10.12.3. Throttle resources through rate limits

Rate limiting in OpenStack Object Storage is implemented as a pluggable middleware that you configure on the proxy server. Rate limiting is performed on requests that result in database writes to the account and container SQLite databases. It uses memcached and is dependent on the proxy servers having highly synchronized time. The rate limits are limited by the accuracy of the proxy server clocks.

10.12.3.1. Configure rate limiting

All configuration is optional. If no account or container limits are provided, no rate limiting occurs. Available configuration options include:

Table 10.67. Description of configuration options for `[filter-ratelimit]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>account_blacklist = c,d</code>	Comma separated lists of account names that will not be allowed. Returns a 497 response. <code>r</code> : for containers of size <code>x</code> , limit requests per second to <code>r</code> . Will limit PUT, DELETE, and POST requests to <code>/a/c/o</code> . <code>container_listing_ratelimit_x = r</code> : for containers of size <code>x</code> , limit listing requests per second to <code>r</code> . Will limit GET requests to <code>/a/c</code> .
<code>account_ratelimit = 0</code>	If set, will limit PUT and DELETE requests to <code>/account_name/container_name</code> . Number is in requests per second.
<code>account_whitelist = a,b</code>	Comma separated lists of account names that will not be rate limited.
<code>clock_accuracy = 1000</code>	Represents how accurate the proxy servers' system clocks are with each other. 1000 means that all the proxies' clock are accurate to each other within 1 millisecond. No <code>ratelimit</code> should be higher than the clock accuracy.

Configuration option = Default value	Description
<code>container_listing_ratelimit_0 = 100</code>	No help text available for this option.
<code>container_listing_ratelimit_10 = 50</code>	No help text available for this option.
<code>container_listing_ratelimit_50 = 20</code>	No help text available for this option.
<code>container_ratelimit_0 = 100</code>	No help text available for this option.
<code>container_ratelimit_10 = 50</code>	No help text available for this option.
<code>container_ratelimit_50 = 20</code>	No help text available for this option.
<code>log_sleep_time_seconds = 0</code>	To allow visibility into rate limiting set this value > 0 and all sleeps greater than the number will be logged.
<code>max_sleep_time_seconds = 60</code>	App will immediately return a 498 response if the necessary sleep time ever exceeds the given <code>max_sleep_time_seconds</code> .
<code>rate_buffer_seconds = 5</code>	Number of seconds the rate counter can drop and be allowed to catch up (at a faster than listed rate). A larger number will result in larger spikes in rate but better average accuracy.
<code>set log_address = /dev/log</code>	Location where syslog sends the logs to
<code>set log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>set log_headers = false</code>	If True, log headers in each request
<code>set log_level = INFO</code>	Log level
<code>set log_name = ratelimit</code>	Label to use when logging
<code>use = egg:swift#ratelimit</code>	Entry point of paste.deploy in the server
<code>with container_limit_x = r</code>	No help text available for this option.

The container rate limits are linearly interpolated from the values given. A sample container rate limiting could be:

`container_ratelimit_100 = 100`

`container_ratelimit_200 = 50`

`container_ratelimit_500 = 20`

This would result in:

Table 10.68. Values for Rate Limiting with Sample Configuration Settings

Container Size	Rate Limit
0-99	No limiting
100	100
150	75
500	20
1000	20

10.12.4. Health check

Provides an easy way to monitor whether the Object Storage proxy server is alive. If you access the proxy with the path `/healthcheck`, it responds with **OK** in the response body, which monitoring tools can use.

Table 10.69. Description of configuration options for `[filter-healthcheck]` in `account-server.conf`

Configuration option = Default value	Description
<code>disable_path =</code>	No help text available for this option.
<code>use = egg:swift#healthcheck</code>	Entry point of paste.deploy in the server

10.12.5. Domain remap

Middleware that translates container and account parts of a domain to path parameters that the proxy server understands.

Table 10.70. Description of configuration options for `[filter-domain_remap]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>default_reseller_prefix =</code>	No help text available for this option.
<code>path_root = v1</code>	Root path
<code>reseller_prefixes = AUTH</code>	Reseller prefix
<code>set log_address = /dev/log</code>	Location where syslog sends the logs to

Configuration option = Default value	Description
<code>set log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>set log_headers = false</code>	If True, log headers in each request
<code>set log_level = INFO</code>	Log level
<code>set log_name = domain_remap</code>	Label to use when logging
<code>storage_domain = example.com</code>	Domain that matches your cloud. Multiple domains can be specified using a comma-separated list.
<code>use = egg:swift#domain_remap</code>	Entry point of paste.deploy in the server

10.12.6. CNAME lookup

Middleware that translates an unknown domain in the host header to something that ends with the configured `storage_domain` by looking up the given domain's CNAME record in DNS.

Table 10.71. Description of configuration options for `[filter-cname_lookup]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>lookup_depth = 1</code>	Because CNAMEs can be recursive, specifies the number of levels through which to search.
<code>set log_address = /dev/log</code>	Location where syslog sends the logs to
<code>set log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>set log_headers = false</code>	If True, log headers in each request
<code>set log_level = INFO</code>	Log level
<code>set log_name = cname_lookup</code>	Label to use when logging
<code>storage_domain = example.com</code>	Domain that matches your cloud. Multiple domains can be specified using a comma-separated list.
<code>use = egg:swift#cname_lookup</code>	Entry point of paste.deploy in the server

10.12.7. Temporary URL

Allows the creation of URLs to provide temporary access to objects. For example, a website may wish to provide a link to download a large object in OpenStack Object Storage, but the Object Storage account has no public access. The website can generate a URL that provides GET access for a limited

time to the resource. When the web browser user clicks on the link, the browser downloads the object directly from Object Storage, eliminating the need for the website to act as a proxy for the request. If the user shares the link with all his friends, or accidentally posts it on a forum, the direct access is limited to the expiration time set when the website created the link.

A temporary URL is the typical URL associated with an object, with two additional query parameters:

temp_url_sig

A cryptographic signature

temp_url_expires

An expiration date, in Unix time

An example of a temporary URL:

```
https://swift-cluster.example.com/v1/AUTH_a422b2-91f3-2f46-74b7-
d7c9e8958f5d30/container/object?
temp_url_sig=da39a3ee5e6b4b0d3255bfef95601890afd80709&
temp_url_expires=1323479485
```

To create temporary URLs, first set the **X-Account-Meta-Temp-URL-Key** header on your Object Storage account to an arbitrary string. This string serves as a secret key. For example, to set a key of **b3968d0207b54ece87cccc06515a89d4** using the `swift` command-line tool:

```
$ swift post -m "Temp-URL-Key:b3968d0207b54ece87cccc06515a89d4"
```

Next, generate an HMAC-SHA1 (RFC 2104) signature to specify:

- Which HTTP method to allow (typically **GET** or **PUT**)
- The expiry date as a Unix timestamp
- The full path to the object
- The secret key set as the **X-Account-Meta-Temp-URL-Key**

Here is code generating the signature for a GET for 24 hours on **/v1/AUTH_account/container/object**:

```
import hmac
from hashlib import sha1
from time import time
method = 'GET'
duration_in_seconds = 60*60*24
expires = int(time() + duration_in_seconds)
path = '/v1/AUTH_a422b2-91f3-2f46-74b7-d7c9e8958f5d30/container/object'
key = 'mykey'
hmac_body = '%s\n%s\n%s' % (method, expires, path)
sig = hmac.new(key, hmac_body, sha1).hexdigest()
s = 'https://{host}/{path}?temp_url_sig={sig}&temp_url_expires={expires}'
url = s.format(host='swift-cluster.example.com', path=path, sig=sig,
expires=expires)
```

Any alteration of the resource path or query arguments results in a 401 Unauthorized error. Similarly, a PUT where GET was the allowed method returns a 401. HEAD is allowed if GET or PUT is allowed. Using this in combination with browser form post translation middleware could also allow direct-from-browser uploads to specific locations in Object Storage.



NOTE

Changing the **X-Account-Meta-Temp-URL-Key** invalidates any previously generated temporary URLs within 60 seconds (the memcache time for the key). Object Storage supports up to two keys, specified by **X-Account-Meta-Temp-URL-Key** and **X-Account-Meta-Temp-URL-Key-2**. Signatures are checked against both keys, if present. This is to allow for key rotation without invalidating all existing temporary URLs.

Object Storage includes a script called **swift-temp-url** that generates the query parameters automatically:

```
$ bin/swift-temp-url GET 3600 /v1/AUTH_account/container/object mykey
/v1/AUTH_account/container/object?
temp_url_sig=5c4cc8886f36a9d0919d708ade98bf0cc71c9e91&
temp_url_expires=1374497657
```

Because this command only returns the path, you must prefix the Object Storage host name (for example, <https://swift-cluster.example.com>).

With GET Temporary URLs, a **Content-Disposition** header is set on the response so that browsers interpret this as a file attachment to be saved. The file name chosen is based on the object name, but you can override this with a **filename** query parameter. The following example specifies a filename of **My Test File.pdf**:

```
https://swift-cluster.example.com/v1/AUTH_a422b2-91f3-2f46-74b7-
d7c9e8958f5d30/container/object?
temp_url_sig=da39a3ee5e6b4b0d3255bfef95601890afd80709&
temp_url_expires=1323479485&
filename=My+Test+File.pdf
```

If you do not want the object to be downloaded, you can cause **Content-Disposition: inline** to be set on the response by adding the **inline** parameter to the query string, as follows:

```
https://swift-cluster.example.com/v1/AUTH_account/container/object?
temp_url_sig=da39a3ee5e6b4b0d3255bfef95601890afd80709&
temp_url_expires=1323479485&inline
```

To enable Temporary URL functionality, edit **/etc/swift/proxy-server.conf** to add **tempurl** to the **pipeline** variable defined in the **[pipeline:main]** section. The **tempurl** entry should appear immediately before the authentication filters in the pipeline, such as **authtoken**, **tempauth** or **keystoneauth**. For example:

```
[pipeline:main]
pipeline = pipeline = healthcheck cache tempurl authtoken keystoneauth
proxy-server
```

Table 10.72. Description of configuration options for **[filter-tempurl]** in **proxy-server.conf**

Configuration option = Default value	Description
incoming_allow_headers =	Headers allowed as exceptions to incoming_remove_headers. Simply a whitespace delimited list of header names and names can optionally end with '*' to indicate a prefix match.
incoming_remove_headers = x-timestamp	Headers to remove from incoming requests. Simply a whitespace delimited list of header names and names can optionally end with '*' to indicate a prefix match.
methods = GET HEAD PUT POST DELETE	HTTP methods allowed with Temporary URLs
outgoing_allow_headers = ``x-object-meta-public-*``	Headers allowed as exceptions to outgoing_remove_headers. Simply a whitespace delimited list of header names and names can optionally end with '*' to indicate a prefix match.
outgoing_remove_headers = ``x-object-meta-*``	Headers to remove from outgoing responses. Simply a whitespace delimited list of header names and names can optionally end with '*' to indicate a prefix match.
use = egg:swift#tempurl	Entry point of paste.deploy in the server

10.12.8. Name Check filter

Name Check is a filter that disallows any paths that contain defined forbidden characters or that exceed a defined length.

Table 10.73. Description of configuration options for [filter-name_check] in proxy-server.conf

Configuration option = Default value	Description
forbidden_chars = <code>``"'"`<>``</code>	Characters that are not allowed in a name
forbidden_regexp = <code>``\./ /\.\./ /\.\$ /\.\.\$``</code>	Substrings to forbid, using regular expression syntax
maximum_length = 255	Maximum length of a name
use = <code>egg:swift#name_check</code>	Entry point of paste.deploy in the server

10.12.9. Constraints

To change the OpenStack Object Storage internal limits, update the values in the **swift-constraints** section in the **swift.conf** file. Use caution when you update these values because they affect the performance in the entire cluster.

Table 10.74. Description of configuration options for [swift-constraints] in swift.conf

Configuration option = Default value	Description
<code>account_listing_limit = 10000</code>	The default (and maximum) number of items returned for an account listing request.
<code>container_listing_limit = 10000</code>	The default (and maximum) number of items returned for a container listing request.
<code>extra_header_count = 0</code>	By default the maximum number of allowed headers depends on the number of max allowed metadata settings plus a default value of 32 for regular http headers. If for some reason this is not enough (custom middleware for example) it can be increased with the <code>extra_header_count</code> constraint.
<code>max_account_name_length = 256</code>	The maximum number of bytes in the utf8 encoding of an account name.
<code>max_container_name_length = 256</code>	The maximum number of bytes in the utf8 encoding of a container name.
<code>max_file_size = 5368709122</code>	The largest normal object that can be saved in the cluster. This is also the limit on the size of each segment of a large object when using the large object manifest support. This value is set in bytes. Setting it to lower than 1MiB will cause some tests to fail. It is STRONGLY recommended to leave this value at the default ($5 * 2^{30} + 2$).
<code>max_header_size = 8192</code>	The max number of bytes in the utf8 encoding of each header. Using 8192 as default because eventlet use 8192 as maximum size of header line. You may need to increase this value when using identity v3 API tokens including more than 7 catalog entries. See also <code>include_service_catalog</code> in <code>proxy-server.conf-sample</code> (documented in <code>overview_auth.rst</code>).
<code>max_meta_count = 90</code>	The max number of metadata keys that can be stored on a single account, container, or object.
<code>max_meta_name_length = 128</code>	The max number of bytes in the utf8 encoding of the name portion of a metadata header.
<code>max_meta_overall_size = 4096</code>	The max number of bytes in the utf8 encoding of the metadata (keys + values).
<code>max_meta_value_length = 256</code>	The max number of bytes in the utf8 encoding of a metadata value.

Configuration option = Default value	Description
<code>max_object_name_length = 1024</code>	The max number of bytes in the utf8 encoding of an object name.
<code>valid_api_versions = v0,v1,v2</code>	No help text available for this option.

10.12.10. Cluster health

Use the `swift-dispersion-report` tool to measure overall cluster health. This tool checks if a set of deliberately distributed containers and objects are currently in their proper places within the cluster. For instance, a common deployment has three replicas of each object. The health of that object can be measured by checking if each replica is in its proper place. If only two of the three is in place, the object's health can be said to be at 66.66%, where 100% would be perfect. A single object's health, especially an older object, usually reflects the health of that entire partition the object is in. If you make enough objects on a distinct percentage of the partitions in the cluster, you get a good estimate of the overall cluster health. In practice, about 1% partition coverage seems to balance well between accuracy and the amount of time it takes to gather results. First, to provide this health value, create a new account solely for this usage. Next, place the containers and objects throughout the system so that they are on distinct partitions. The `swift-dispersion-populate` tool does this by making up random container and object names until they fall on distinct partitions. Last, and repeatedly for the life of the cluster, you must run the `swift-dispersion-report` tool to check the health of each of these containers and objects. These tools need direct access to the entire cluster and to the ring files (installing them on a proxy server suffices). The `swift-dispersion-populate` and `swift-dispersion-report` commands both use the same configuration file, `/etc/swift/dispersion.conf`. Example `dispersion.conf` file:

```
[dispersion]
auth_url = http://localhost:8080/auth/v1.0
auth_user = test:tester
auth_key = testing
```

There are also configuration options for specifying the dispersion coverage, which defaults to 1%, retries, concurrency, and so on. However, the defaults are usually fine. Once the configuration is in place, run `swift-dispersion-populate` to populate the containers and objects throughout the cluster. Now that those containers and objects are in place, you can run `swift-dispersion-report` to get a dispersion report, or the overall health of the cluster. Here is an example of a cluster in perfect health:

```
$ swift-dispersion-report
Queried 2621 containers for dispersion reporting, 19s, 0 retries
100.00% of container copies found (7863 of 7863)
Sample represents 1.00% of the container partition space

Queried 2619 objects for dispersion reporting, 7s, 0 retries
100.00% of object copies found (7857 of 7857)
Sample represents 1.00% of the object partition space
```

Now, deliberately double the weight of a device in the object ring (with replication turned off) and re-run the dispersion report to show what impact that has:

```
$ swift-ring-builder object.builder set_weight d0 200
```



```
$ swift-ring-builder object.builder rebalance
...
$ swift-dispersion-report
Queried 2621 containers for dispersion reporting, 8s, 0 retries
100.00% of container copies found (7863 of 7863)
Sample represents 1.00% of the container partition space

Queried 2619 objects for dispersion reporting, 7s, 0 retries
There were 1763 partitions missing one copy.
77.56% of object copies found (6094 of 7857)
Sample represents 1.00% of the object partition space
```

You can see the health of the objects in the cluster has gone down significantly. Of course, this test environment has just four devices, in a production environment with many devices the impact of one device change is much less. Next, run the replicators to get everything put back into place and then rerun the dispersion report:

```
... start object replicators and monitor logs until they're caught up ...
$ swift-dispersion-report
Queried 2621 containers for dispersion reporting, 17s, 0 retries
100.00% of container copies found (7863 of 7863)
Sample represents 1.00% of the container partition space

Queried 2619 objects for dispersion reporting, 7s, 0 retries
100.00% of object copies found (7857 of 7857)
Sample represents 1.00% of the object partition space
```

Alternatively, the dispersion report can also be output in JSON format. This allows it to be more easily consumed by third-party utilities:

```
$ swift-dispersion-report -j
{"object": {"retries": 0, "missing_two": 0, "copies_found": 7863,
"missing_one": 0,
"copies_expected": 7863, "pct_found": 100.0, "overlapping": 0,
"missing_all": 0}, "container":
{"retries": 0, "missing_two": 0, "copies_found": 12534, "missing_one": 0,
"copies_expected":
12534, "pct_found": 100.0, "overlapping": 15, "missing_all": 0}}
```

Table 10.75. Description of configuration options for [dispersion] in dispersion.conf

Configuration option = Default value	Description
auth_key = <i>testing</i>	No help text available for this option.
auth_url = <i>http://localhost:8080/auth/v1.0</i>	Endpoint for auth server, such as keystone
auth_user = <i>test:tester</i>	Default user for dispersion in this context
auth_version = <i>1.0</i>	Indicates which version of auth
concurrency = <i>25</i>	Number of replication workers to spawn

Configuration option = Default value	Description
container_populate = <i>yes</i>	No help text available for this option.
container_report = <i>yes</i>	No help text available for this option.
dispersion_coverage = <i>1.0</i>	No help text available for this option.
dump_json = <i>no</i>	No help text available for this option.
endpoint_type = <i>publicURL</i>	Indicates whether endpoint for auth is public or internal
keystone_api_insecure = <i>no</i>	Allow accessing insecure keystone server. The keystone's certificate will not be verified.
object_populate = <i>yes</i>	No help text available for this option.
object_report = <i>yes</i>	No help text available for this option.
project_domain_name = <i>project_domain</i>	No help text available for this option.
project_name = <i>project</i>	No help text available for this option.
retries = <i>5</i>	No help text available for this option.
swift_dir = <i>/etc/swift</i>	Swift configuration directory
user_domain_name = <i>user_domain</i>	No help text available for this option.

10.12.11. Static Large Object (SLO) support

This feature is similar to Dynamic Large Object (DLO) support in that it enables the user to upload many objects concurrently and afterwards download them as a single object. It is different in that it does not rely on eventually consistent container listings to do so. Instead, a user-defined manifest of the object segments is used.

Table 10.76. Description of configuration options for `[filter-slo]` in `proxy-server.conf`

Configuration option = Default value	Description
max_get_time = <i>86400</i>	No help text available for this option.
max_manifest_segments = <i>1000</i>	No help text available for this option.
max_manifest_size = <i>2097152</i>	No help text available for this option.
min_segment_size = <i>1048576</i>	No help text available for this option.

Configuration option = Default value	Description
<code>rate_limit_after_segment = 10</code>	Rate limit the download of large object segments after this segment is downloaded.
<code>rate_limit_segments_per_sec = 0</code>	Rate limit large object downloads at this rate. contact for a normal request. You can use '*' replicas' at the end to have it use the number given times the number of replicas for the ring being used for the request. paste.deploy to use for auth. To use tempauth set to:
<code>use = egg:swift#slo</code>	Entry point of paste.deploy in the server

10.12.12. Container quotas

The `container_quotas` middleware implements simple quotas that can be imposed on Object Storage containers by a user with the ability to set container metadata, most likely the account administrator. This can be useful for limiting the scope of containers that are delegated to non-admin users, exposed to formpost uploads, or just as a self-imposed sanity check.

Any object PUT operations that exceed these quotas return a 403 response (forbidden).

Quotas are subject to several limitations: eventual consistency, the timeliness of the cached `container_info` (60 second TTL by default), and it is unable to reject chunked transfer uploads that exceed the quota (though once the quota is exceeded, new chunked transfers are refused).

Set quotas by adding meta values to the container. These values are validated when you set them:

- `X-Container-Meta-Quota-Bytes`: Maximum size of the container, in bytes.
- `X-Container-Meta-Quota-Count`: Maximum object count of the container.

Table 10.77. Description of configuration options for `[filter-container-quotas]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>use = egg:swift#container_quotas</code>	Entry point of paste.deploy in the server

10.12.13. Account quotas

The `x-account-meta-quota-bytes` metadata entry must be requests (PUT, POST) if a given account quota (in bytes) is exceeded while DELETE requests are still allowed.

The `x-account-meta-quota-bytes` metadata entry must be set to store and enable the quota. Write requests to this metadata entry are only permitted for resellers. There is no account quota limitation on a reseller account even if `x-account-meta-quota-bytes` is set.

Any object PUT operations that exceed the quota return a 413 response (request entity too large) with

a descriptive body.

The following command uses an admin account that owns the Reseller role to set a quota on the test account:

```
$ swift -A http://127.0.0.1:8080/auth/v1.0 -U admin:admin -K admin \ --os-storage-url http://127.0.0.1:8080/v1/AUTH_test post -m quota-bytes:10000
```

Here is the stat listing of an account where quota has been set:

```
$ swift -A http://127.0.0.1:8080/auth/v1.0 -U test:tester -K testing stat
Account: AUTH_test
Containers: 0
Objects: 0
Bytes: 0
Meta Quota-Bytes: 10000
X-Timestamp: 1374075958.37454
X-Trans-Id: tx602634cf478546a39b1be-0051e6bc7a
```

This command removes the account quota:

```
$ swift -A http://127.0.0.1:8080/auth/v1.0 -U admin:admin -K admin --os-storage-url http://127.0.0.1:8080/v1/AUTH_test post -m quota-bytes:
```

10.12.14. Bulk delete

Use **bulk-delete** to delete multiple files from an account with a single request. Responds to DELETE requests with a header 'X-Bulk-Delete: true_value'. The body of the DELETE request is a new line-separated list of files to delete. The files listed must be URL encoded and in the form:

```
/container_name/obj_name
```

If all files are successfully deleted (or did not exist), the operation returns **HTTPOk**. If any files failed to delete, the operation returns **HTTPBadGateway**. In both cases, the response body is a JSON dictionary that shows the number of files that were successfully deleted or not found. The files that failed are listed.

Table 10.78. Description of configuration options for `[filter-bulk]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>delete_container_retry_count = 0</code>	No help text available for this option.
<code>max_containers_per_extraction = 10000</code>	No help text available for this option.
<code>max_deletes_per_request = 10000</code>	No help text available for this option.
<code>max_failed_deletes = 1000</code>	No help text available for this option.
<code>max_failed_extractions = 1000</code>	No help text available for this option.

Configuration option = Default value	Description
<code>use = egg:swift#bulk</code>	Entry point of paste.deploy in the server
<code>yield_frequency = 10</code>	No help text available for this option.

10.12.15. Drive audit

The `swift-drive-audit` configuration items reference a script that can be run by using `cron` to watch for bad drives. If errors are detected, it unmounts the bad drive, so that OpenStack Object Storage can work around it. It takes the following options:

Table 10.79. Description of configuration options for `[drive-audit]` in `drive-audit.conf`

Configuration option = Default value	Description
<code>device_dir = /srv/node</code>	Directory devices are mounted under
<code>error_limit = 1</code>	Number of errors to find before a device is unmounted
<code>log_address = /dev/log</code>	Location where syslog sends the logs to
<code>log_facility = LOG_LOCAL0</code>	Syslog log facility
<code>log_file_pattern = ``/var/log/kern.*[!][!g][!z]``</code>	Location of the log file with globbing pattern to check against device errors locate device blocks with errors in the log file
<code>log_level = INFO</code>	Logging level
<code>log_max_line_length = 0</code>	Caps the length of log lines to the value given; no limit if set to 0, the default.
<code>log_name = drive-audit</code>	Label used when logging
<code>log_to_console = False</code>	No help text available for this option.
<code>minutes = 60</code>	Number of minutes to look back in
<code>recon_cache_path = /var/cache/swift</code>	Directory where stats for a few items will be stored
<code>regex_pattern_1 = ``\berror\b.*\b(dm-[0-9]{1,2}\d?)\b``</code>	No help text available for this option.
<code>unmount_failed_device = True</code>	No help text available for this option.

10.12.16. Form post

10.12.16. Form post

Middleware that enables you to upload objects to a cluster by using an HTML form **POST**.

The format of the form is:

```
<![CDATA[
<form action="<swift-url>" method="POST"
      enctype="multipart/form-data">
  <input type="hidden" name="redirect" value="<redirect-url>" />
  <input type="hidden" name="max_file_size" value="<bytes>" />
  <input type="hidden" name="max_file_count" value="<count>" />
  <input type="hidden" name="expires" value="<unix-timestamp>" />
  <input type="hidden" name="signature" value="<hmac>" />
  <input type="hidden" name="x_delete_at" value="<unix-timestamp>"/>
  <input type="hidden" name="x_delete_after" value="<seconds>"/>
  <input type="file" name="file1" /><br />
  <input type="submit" />
</form>]]>
```

In the form:

- **action="<swift-url>"**

The URL to the Object Storage destination, such as `https://swift-cluster.example.com/v1/AUTH_account/container/object_prefix`.

The name of each uploaded file is appended to the specified **swift-url**. So, you can upload directly to the root of container with a URL like `https://swift-cluster.example.com/v1/AUTH_account/container/`.

Optionally, you can include an object prefix to separate different users' uploads, such as `https://swift-cluster.example.com/v1/AUTH_account/container/object_prefix`.

- **method="POST"**

The form **method** must be **POST**.

- **enctype="multipart/form-data"**

The **enctype** must be set to **multipart/form-data**.

- **name="redirect"**

The URL to which to redirect the browser after the upload completes. The URL has status and message query parameters added to it that indicate the HTTP status code for the upload and, optionally, additional error information. The 2nn status code indicates success. If an error occurs, the URL might include error information, such as `"max_file_size exceeded"`.

- **name="max_file_size"**

Required. The maximum number of bytes that can be uploaded in a single file upload.

- **name="max_file_count"**

Required. The maximum number of files that can be uploaded with the form.

- **name="expires"**

The expiration date and time for the form in [UNIX Epoch time stamp format](#). After this date and time, the form is no longer valid.

For example, **1440619048** is equivalent to **Mon, Wed, 26 Aug 2015 19:57:28 GMT**.

- **name="signature"**

The HMAC-SHA1 signature of the form. This sample Python code shows how to compute the signature:

```
import hmac
from hashlib import sha1
from time import time
path = '/v1/account/container/object_prefix'
redirect = 'https://myserver.com/some-page'
max_file_size = 104857600
max_file_count = 10
expires = int(time() + 600)
key = 'mykey'
hmac_body = '%s\n%s\n%s\n%s\n%s' % (path, redirect,
    max_file_size, max_file_count, expires)
signature = hmac.new(key, hmac_body, sha1).hexdigest()
```

The key is the value of the **X-Account-Meta-Temp-URL-Key** header on the account.

Use the full path from the **/v1/** value and onward.

During testing, you can use the **swift-form-signature** command-line tool to compute the **expires** and **signature** values.

- **name="x_delete_at"**

The date and time in [UNIX Epoch time stamp format](#) when the object will be removed.

For example, **1440619048** is equivalent to **Mon, Wed, 26 Aug 2015 19:57:28 GMT**.

This attribute enables you to specify the **X-Delete-At** header value in the form **POST**.

- **name="x_delete_after"**

The number of seconds after which the object is removed. Internally, the Object Storage system stores this value in the **X-Delete-At** metadata item. This attribute enables you to specify the **X-Delete-After** header value in the form **POST**.

- **type="file" name="filexx"**

Optional. One or more files to upload. Must appear after the other attributes to be processed correctly. If attributes come after the **file** attribute, they are not sent with the sub-request because on the server side, all attributes in the file cannot be parsed unless the whole file is read into memory and the server does not have enough memory to service these requests. So, attributes that follow the **file** attribute are ignored.

Table 10.80. Description of configuration options for `[filter-formpost]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>use = egg:swift#formpost</code>	Entry point of paste.deploy in the server

10.12.17. Static web sites

When configured, this middleware serves container data as a static web site with index file and error file resolution and optional file listings. This mode is normally only active for anonymous requests.

Table 10.81. Description of configuration options for `[filter-staticweb]` in `proxy-server.conf`

Configuration option = Default value	Description
<code>use = egg:swift#staticweb</code>	Entry point of paste.deploy in the server

10.12.18. Cross-origin resource sharing

Cross-Origin Resource Sharing (CORS) is a mechanism that allows code (JavaScript, for example) running in a browser to make requests to a domain, other than the one it was originated from. OpenStack Object Storage supports CORS requests to containers and objects within the containers using metadata held on the container.

In addition to the metadata on containers, you can use the `cors_allow_origin` option in the `proxy-server.conf` file to set a list of hosts that are included with any CORS request by default.

10.12.19. Endpoint listing middleware

The endpoint listing middleware enables third-party services that use data locality information to integrate with OpenStack Object Storage. This middleware reduces network overhead and is designed for third-party services that run inside the firewall. Deploy this middleware on a proxy server because usage of this middleware is not authenticated.

Format requests for endpoints, as follows:

```
/endpoints/{account}/{container}/{object} /endpoints/{account}/{container}
/endpoints/{account}
```

Use the `list_endpoints_path` configuration option in the `proxy_server.conf` file to customize the `/endpoints/` path.

Responses are JSON-encoded lists of endpoints, as follows:

```
http://{server}:{port}/{dev}/{part}/{acc}/{cont}/{obj}
http://{server}:{port}/{dev}/{part}/{acc}/{cont}
http://{server}:{port}/{dev}/{part}/{acc}
```

An example response is:


```
http://10.1.1.1:6000/sda1/2/a/c2/o1  
http://10.1.1.1:6000/sda1/2/a/c2  
http://10.1.1.1:6000/sda1/2/a
```

10.13. NEW, UPDATED AND DEPRECATED OPTIONS IN LIBERTY FOR OPENSTACK OBJECT STORAGE

There are no new, updated, and deprecated options in Liberty for OpenStack Object Storage.

CHAPTER 11. ORCHESTRATION

The Orchestration service is designed to manage the lifecycle of infrastructure and applications within OpenStack clouds. Its various agents and services are configured in the `/etc/heat/heat.conf` file.

The following tables provide a comprehensive list of the Orchestration configuration options.

Table 11.1. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(String) Service user password.
admin_tenant_name = <i>admin</i>	(String) Service tenant name.
admin_token = <i>None</i>	(String) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(String) Service username.
auth_admin_prefix =	(String) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
auth_host = <i>127.0.0.1</i>	(String) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_port = <i>35357</i>	(Integer) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_protocol = <i>https</i>	(String) Protocol of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_section = <i>None</i>	(Unknown) Config Section from which to load plugin specific options
auth_type = <i>None</i>	(Unknown) Authentication type to load
auth_uri = <i>None</i>	(String) Complete public Identity API endpoint.
auth_version = <i>None</i>	(String) API version of the admin Identity API endpoint.
cache = <i>None</i>	(String) Env key for the swift cache.

Configuration option = Default value	Description
cafile = <i>None</i>	(String) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(String) Required if identity server requires client certificate
check_revocations_for_cached = <i>False</i>	(Boolean) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.
delay_auth_decision = <i>False</i>	(Boolean) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(String) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(List) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(Integer) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(Integer) How many times are we trying to reconnect when communicating with Identity API Server.
identity_uri = <i>None</i>	(String) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/

Configuration option = Default value	Description
include_service_catalog = <i>True</i>	(Boolean) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(Boolean) Verify HTTPS connections.
keyfile = <i>None</i>	(String) Required if identity server requires client certificate
memcache_pool_conn_get_timeout = <i>10</i>	(Integer) (Optional) Number of seconds that an operation will wait to get a memcached client connection from the pool.
memcache_pool_dead_retry = <i>300</i>	(Integer) (Optional) Number of seconds memcached server is considered dead before it is tried again.
memcache_pool_maxsize = <i>10</i>	(Integer) (Optional) Maximum total number of open connections to every memcached server.
memcache_pool_socket_timeout = <i>3</i>	(Integer) (Optional) Socket timeout in seconds for communicating with a memcached server.
memcache_pool_unused_timeout = <i>60</i>	(Integer) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
memcache_secret_key = <i>None</i>	(String) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
memcache_security_strategy = <i>None</i>	(String) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.
memcache_use_advanced_pool = <i>False</i>	(Boolean) (Optional) Use the advanced (eventlet safe) memcached client pool. The advanced pool will only work under python 2.x.
memcached_servers = <i>None</i>	(List) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.

Configuration option = Default value	Description
region_name = <i>None</i>	(String) The region in which the identity server can be found.
revocation_cache_time = 10	(Integer) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
signing_dir = <i>None</i>	(String) Directory used to cache files related to PKI tokens.
token_cache_time = 300	(Integer) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 11.2. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
client_retry_limit = 2	(Integer) Number of times to retry when a client encounters an expected intermittent error. Set to 0 to disable retries.
convergence_engine = <i>False</i>	(Boolean) Enables engine with convergence architecture. All stacks with this option will be created using convergence engine.
default_deployment_signal_transport = <i>CFN_SIGNAL</i>	(String) Template default for how the server should signal to heat with the deployment output values. CFN_SIGNAL will allow an HTTP POST to a CFN keypair signed URL (requires enabled heat-api-cfn). TEMP_URL_SIGNAL will create a Swift TempURL to be signaled via HTTP PUT (requires object-store endpoint which supports TempURL). HEAT_SIGNAL will allow calls to the Heat API resource-signal using the provided keystone credentials. ZAQAR_SIGNAL will create a dedicated zaqar queue to be signaled using the provided keystone credentials.

Configuration option = Default value	Description
default_software_config_transport = <i>POLL_SERVER_CFN</i>	(String) Template default for how the server should receive the metadata required for software configuration. <i>POLL_SERVER_CFN</i> will allow calls to the cfn API action <i>DescribeStackResource</i> authenticated with the provided keypair (requires enabled <i>heat-api-cfn</i>). <i>POLL_SERVER_HEAT</i> will allow calls to the Heat API resource-show using the provided keystone credentials (requires keystone v3 API, and configured <i>stack_user_*</i> config options). <i>POLL_TEMP_URL</i> will create and populate a Swift TempURL with metadata for polling (requires object-store endpoint which supports TempURL). <i>ZAQAR_MESSAGE</i> will create a dedicated zaqar queue and post the metadata for polling.
deferred_auth_method = <i>trusts</i>	(String) Select deferred auth method, stored password or trusts.
environment_dir = <i>/etc/heat/environment.d</i>	(String) The directory to search for environment files.
error_wait_time = <i>240</i>	(Integer) Error wait time in seconds for stack action (ie. create or update).
event_purge_batch_size = <i>10</i>	(Integer) Controls how many events will be pruned whenever a stack's events exceed <i>max_events_per_stack</i> . Set this lower to keep more events at the expense of more frequent purges.
executor_thread_pool_size = <i>64</i>	(Integer) Size of executor thread pool.
host = <i>localhost</i>	(String) Name of the engine node. This can be an opaque identifier. It is not necessarily a hostname, FQDN, or IP address.
keystone_backend = <i>heat.common.heat_keystoneclient.KeystoneClientV3</i>	(String) Fully qualified class name to use as a keystone backend.
max_interface_check_attempts = <i>10</i>	(Integer) Number of times to check whether an interface has been attached or detached.
memcached_servers = <i>None</i>	(List) Memcached servers or None for in process cache.
periodic_interval = <i>60</i>	(Integer) Seconds between running periodic tasks.
plugin_dirs = <i>/usr/lib64/heat, /usr/lib/heat, /usr/local/lib/heat, /usr/local/lib64/heat</i>	(List) List of directories to search for plug-ins.

Configuration option = Default value	Description
reauthentication_auth_method =	(String) Allow reauthentication on token expiry, such that long-running tasks may complete. Note this defeats the expiry of any provided user tokens.
watch_log_file = <i>False</i>	(Boolean) Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if <code>log_file</code> option is specified and Linux platform is used. This option is ignored if <code>log_config_append</code> is set.
[cache]	
backend = <i>dogpile.cache.null</i>	(String) Dogpile.cache backend module. It is recommended that Memcache with pooling (<code>oslo_cache.memcache_pool</code>) or Redis (<code>dogpile.cache.redis</code>) be used in production deployments. Small workloads (single process) like devstack can use the <code>dogpile.cache.memory</code> backend.
backend_argument = []	(Multi-valued) Arguments supplied to the backend module. Specify this option once per argument to be passed to the <code>dogpile.cache</code> backend. Example format: "<argname>:<value>".
config_prefix = <i>cache.oslo</i>	(String) Prefix for building the configuration dictionary for the cache region. This should not need to be changed unless there is another <code>dogpile.cache</code> region with the same configuration name.
debug_cache_backend = <i>False</i>	(Boolean) Extra debugging from the cache backend (cache keys, get/set/delete/etc calls). This is only really useful if you need to see the specific cache-backend get/set/delete calls with the keys/values. Typically this should be left set to false.
enabled = <i>False</i>	(Boolean) Global toggle for caching.
expiration_time = 600	(Integer) Default TTL, in seconds, for any cached item in the <code>dogpile.cache</code> region. This applies to any cached method that doesn't have an explicit cache expiration time defined for it.
memcache_dead_retry = 300	(Integer) Number of seconds memcached server is considered dead before it is tried again. (<code>dogpile.cache.memcache</code> and <code>oslo_cache.memcache_pool</code> backends only).

Configuration option = Default value	Description
memcache_pool_connection_get_timeout = 10	(Integer) Number of seconds that an operation will wait to get a memcache client connection.
memcache_pool_maxsize = 10	(Integer) Max total number of open connections to every memcached server. (oslo_cache.memcache_pool backend only).
memcache_pool_unused_timeout = 60	(Integer) Number of seconds a connection to memcached is held unused in the pool before it is closed. (oslo_cache.memcache_pool backend only).
memcache_servers = localhost:11211	(List) Memcache servers in the format of "host:port". (dogpile.cache.memcache and oslo_cache.memcache_pool backends only).
memcache_socket_timeout = 3	(Integer) Timeout in seconds for every call to a server. (dogpile.cache.memcache and oslo_cache.memcache_pool backends only).
proxies =	(List) Proxy classes to import that will affect the way the dogpile.cache backend functions. See the dogpile.cache documentation on changing-backend-behavior.
[constraint_validation_cache]	
caching = True	(Boolean) Toggle to enable/disable caching when Orchestration Engine validates property constraints of stack. During property validation with constraints Orchestration Engine caches requests to other OpenStack services. Please note that the global toggle for oslo.cache(enabled=True in [cache] group) must be enabled to use this feature.
expiration_time = 60	(Integer) TTL, in seconds, for any cached item in the dogpile.cache region used for caching of validation constraints.
[resource_finder_cache]	
caching = True	(Boolean) Toggle to enable/disable caching when Orchestration Engine looks for other OpenStack service resources using name or id. Please note that the global toggle for oslo.cache(enabled=True in [cache] group) must be enabled to use this feature.
expiration_time = 3600	(Integer) TTL, in seconds, for any cached item in the dogpile.cache region used for caching of OpenStack service finder functions.

Configuration option = Default value	Description
[revision]	
heat_revision = <i>unknown</i>	(String) Heat build revision. If you would prefer to manage your build revision separately, you can move this section to a different file and add it as another config option.
[service_extension_cache]	
caching = <i>True</i>	(Boolean) Toggle to enable/disable caching when Orchestration Engine retrieves extensions from other OpenStack services. Please note that the global toggle for oslo.cache(enabled=True in [cache] group) must be enabled to use this feature.
expiration_time = <i>3600</i>	(Integer) TTL, in seconds, for any cached item in the dogpile.cache region used for caching of service extensions.

Table 11.3. Description of CORS configuration options

Configuration option = Default value	Description
[cors]	
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(Integer) Maximum cache age of CORS preflight requests.
[cors.subdomain]	

Configuration option = Default value	Description
allow_credentials = <i>True</i>	(Boolean) Indicate that the actual request can include user credentials
allow_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which header field names may be used during the actual request.
allow_methods = <i>GET, POST, PUT, DELETE, OPTIONS</i>	(List) Indicate which methods can be used during the actual request.
allowed_origin = <i>None</i>	(List) Indicate whether this resource may be shared with the domain received in the requests "origin" header.
expose_headers = <i>Content-Type, Cache-Control, Content-Language, Expires, Last-Modified, Pragma</i>	(List) Indicate which headers are safe to expose to the API. Defaults to HTTP Simple Headers.
max_age = <i>3600</i>	(Integer) Maximum cache age of CORS preflight requests.

Table 11.4. Description of crypt configuration options

Configuration option = Default value	Description
[DEFAULT]	
auth_encryption_key = <i>notgood but just long enough i t</i>	(String) Key used to encrypt authentication info in the database. Length of this key must be 32 characters.

Table 11.5. Description of database configuration options

Configuration option = Default value	Description
[database]	
backend = <i>sqlalchemy</i>	(String) The back end to use for the database.
connection = <i>None</i>	(String) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(Integer) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(Boolean) Add Python stack traces to SQL as comment strings.

Configuration option = Default value	Description
db_inc_retry_interval = <i>True</i>	(Boolean) If True, increases the interval between retries of a database operation up to db_max_retry_interval .
db_max_retries = 20	(Integer) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = 10	(Integer) If db_inc_retry_interval is set, the maximum seconds between retries of a database operation.
db_retry_interval = 1	(Integer) Seconds between retries of a database transaction.
idle_timeout = 3600	(Integer) Timeout before idle SQL connections are reaped.
max_overflow = 50	(Integer) If set, use this value for max_overflow with SQLAlchemy.
max_pool_size = <i>None</i>	(Integer) Maximum number of SQL connections to keep open in a pool.
max_retries = 10	(Integer) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
min_pool_size = 1	(Integer) Minimum number of SQL connections to keep open in a pool.
mysql_sql_mode = <i>TRADITIONAL</i>	(String) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: mysql_sql_mode =
pool_timeout = <i>None</i>	(Integer) If set, use this value for pool_timeout with SQLAlchemy.
retry_interval = 10	(Integer) Interval between retries of opening a SQL connection.
slave_connection = <i>None</i>	(String) The SQLAlchemy connection string to use to connect to the slave database.
sqlite_db = <i>oslo.sqlite</i>	(String) The file name to use with SQLite.

Configuration option = Default value	Description
sqlite_synchronous = <i>True</i>	(Boolean) If True, SQLite uses synchronous mode.
use_db_reconnect = <i>False</i>	(Boolean) Enable the experimental use of database reconnect on connection lost.

Table 11.6. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
backdoor_port = <i>None</i>	(StrOpt) Enable eventlet backdoor. Acceptable values are 0, <port>, and <start>:<end>, where 0 results in listening on a random tcp port number; <port> results in listening on the specified port number (and not enabling backdoor if that port is in use); and <start>:<end> results in listening on the smallest unused port number within the specified range of port numbers. The chosen port is displayed in the service's log file.

Table 11.7. Description of load balancer configuration options

Configuration option = Default value	Description
[DEFAULT]	
loadbalancer_template = <i>None</i>	(String) Custom template for the built-in loadbalancer nested stack.

Table 11.8. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(Boolean) If set to true, the logging level will be set to DEBUG instead of the default INFO level.

Configuration option = Default value	Description
default_log_levels = <i>amqp=WARN, amqplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, requests.packages.urllib3.util.retry=WARN, urllib3.util.retry=WARN, keystonemiddleware=WARN, routes.middleware=WARN, stevedore=WARN, taskflow=WARN, keystoneauth=WARN, oslo.cache=INFO, dogpile.core.dogpile=INFO</i>	(List) List of package logging levels in logger=LEVEL pairs. This option is ignored if log_config_append is set.
fatal_deprecations = <i>False</i>	(Boolean) Enables or disables fatal status of deprecations.
instance_format = <i>"[instance: %(uuid)s] "</i>	(String) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(String) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(String) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, logging_context_format_string).
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(String) Defines the format string for <code>%(asctime)s</code> in log records. Default: <code>%(default)s</code> . This option is ignored if log_config_append is set.
log_dir = <i>None</i>	(String) (Optional) The base directory used for relative log_file paths. This option is ignored if log_config_append is set.
log_file = <i>None</i>	(String) (Optional) Name of log file to send logging output to. If no default is set, logging will go to stderr as defined by use_stderr. This option is ignored if log_config_append is set.
logging_context_format_string = <i>%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%(message)s</i>	(String) Format string to use for log messages with context.

Configuration option = Default value	Description
logging_debug_format_suffix = % (funcName)s %(pathname)s:%(lineno)d	(String) Additional data to append to log message when logging level for the message is DEBUG.
logging_default_format_string = % (asctime)s.%(msecs)03d %(process)d %(levelname)s % (name)s [-] %(instance)s%(message)s	(String) Format string to use for log messages when context is undefined.
logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d ERROR %(name)s % (instance)s	(String) Prefix each line of exception output with this format.
logging_user_identity_format = %(user)s %(tenant)s %(domain)s %(user_domain)s % (project_domain)s	(String) Defines the format string for % (user_identity)s that is used in logging_context_format_string.
publish_errors = <i>False</i>	(Boolean) Enables or disables publication of error events.
syslog_log_facility = <i>LOG_USER</i>	(String) Syslog facility to receive log lines. This option is ignored if log_config_append is set.
use_stderr = <i>True</i>	(Boolean) Log output to standard error. This option is ignored if log_config_append is set.
use_syslog = <i>False</i>	(Boolean) Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.
verbose = <i>True</i>	(Boolean) DEPRECATED: If set to false, the logging level will be set to WARNING instead of the default INFO level.

Table 11.9. Description of oslo_middleware configuration options

Configuration option = Default value	Description
[oslo_middleware]	
max_request_body_size = <i>114688</i>	(IntOpt) The maximum body size for each request, in bytes.

Table 11.10. Description of quota configuration options

Configuration option = Default value	Description
[DEFAULT]	
max_events_per_stack = 1000	(Integer) Maximum events that will be available per stack. Older events will be deleted when this is reached. Set to 0 for unlimited events per stack.
max_nested_stack_depth = 5	(Integer) Maximum depth allowed when using nested stacks.
max_resources_per_stack = 1000	(Integer) Maximum resources allowed per top-level stack. -1 stands for unlimited.
max_stacks_per_tenant = 100	(Integer) Maximum number of stacks any one tenant may have active at one time.
max_template_size = 524288	(Integer) Maximum raw byte size of any template.

Table 11.11. Description of Redis configuration options

Configuration option = Default value	Description
[matchmaker_redis]	
check_timeout = 20000	(Integer) Time in ms to wait before the transaction is killed.
host = 127.0.0.1	(String) Host to locate redis.
password =	(String) Password for Redis server (optional).
port = 6379	(Port number) Use this port to connect to redis host.
sentinel_group_name = oslo-messaging-zeromq	(String) Redis replica set name.
sentinel_hosts =	(List) List of Redis Sentinel hosts (fault tolerance mode) e.g. [host:port, host1:port ...]
socket_timeout = 1000	(Integer) Timeout in ms on blocking socket operations
wait_timeout = 500	(Integer) Time in ms to wait between connection attempts.

Table 11.12. Description of testing configuration options

Configuration option = Default value	Description
[profiler]	
enabled = <i>False</i>	(Boolean) Enables the profiling for all services on this node. Default value is False (fully disable the profiling feature). Possible values: * True: Enables the feature * False: Disables the feature. The profiling cannot be started via this project operations. If the profiling is triggered by another project, this project part will be empty.
hmac_keys = <i>SECRET_KEY</i>	(String) Secret key(s) to use for encrypting context data for performance profiling. This string value should have the following format: <key1>[,<key2>,...<keyn>], where each key is some random string. A user who triggers the profiling via the REST API has to set one of these keys in the headers of the REST API call to include profiling results of this node for this particular project. Both "enabled" flag and "hmac_keys" config options should be set to enable profiling. Also, to generate correct profiling information across all services at least one key needs to be consistent between OpenStack projects. This ensures it can be used from client side to generate the trace, containing information from all possible resources.
trace_sqlalchemy = <i>False</i>	(Boolean) Enables SQL requests profiling in services. Default value is False (SQL requests won't be traced). Possible values: * True: Enables SQL requests profiling. Each SQL query will be part of the trace and can be analyzed by how much time was spent for that. * False: Disables SQL requests profiling. The spent time is only shown on a higher level of operations. Single SQL queries cannot be analyzed this way.

Table 11.13. Description of trustee configuration options

Configuration option = Default value	Description
[trustee]	
auth_plugin = <i>None</i>	(String) Name of the plugin to load
auth_section = <i>None</i>	(String) Config Section from which to load plugin specific options

11.1. CONFIGURE APIS

The following options allow configuration of the APIs that Orchestration supports. Currently this includes compatibility APIs for CloudFormation and CloudWatch and a native API.

Table 11.14. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
action_retry_limit = 5	(Integer) Number of times to retry to bring a resource to a non-error state. Set to 0 to disable retries.
enable_stack_abandon = False	(Boolean) Enable the preview Stack Abandon feature.
enable_stack_adapt = False	(Boolean) Enable the preview Stack Adopt feature.
encrypt_parameters_and_properties = False	(Boolean) Encrypt template parameters that were marked as hidden and also all the resource properties before storing them in database.
heat_metadata_server_url = None	(String) URL of the Heat metadata server. NOTE: Setting this is only needed if you require instances to use a different endpoint than in the keystone catalog
heat_stack_user_role = heat_stack_user	(String) Keystone role for heat template-defined users.
heat_waitcondition_server_url = None	(String) URL of the Heat waitcondition server.
heat_watch_server_url =	(String) URL of the Heat CloudWatch server.
hidden_stack_tags = data-processing-cluster	(List) Stacks containing these tag names will be hidden. Multiple tags should be given in a comma-delimited list (eg. hidden_stack_tags=hide_me,me_too).
max_json_body_size = 1048576	(Integer) Maximum raw byte size of JSON request body. Should be larger than max_template_size.
num_engine_workers = None	(Integer) Number of heat-engine processes to fork and run.
observe_on_update = False	(Boolean) On update, enables heat to collect existing resource properties from reality and converge to updated template.

Configuration option = Default value	Description
stack_action_timeout = 3600	(Integer) Timeout in seconds for stack action (ie. create or update).
stack_domain_admin = None	(String) Keystone username, a user with roles sufficient to manage users and projects in the stack_user_domain.
stack_domain_admin_password = None	(String) Keystone password for stack_domain_admin user.
stack_scheduler_hints = False	(Boolean) When this feature is enabled, scheduler hints identifying the heat stack context of a server or volume resource are passed to the configured schedulers in nova and cinder, for creates done using heat resource types OS::Cinder::Volume, OS::Nova::Server, and AWS::EC2::Instance. heat_root_stack_id will be set to the id of the root stack of the resource, heat_stack_id will be set to the id of the resource's parent stack, heat_stack_name will be set to the name of the resource's parent stack, heat_path_in_stack will be set to a list of tuples, (stackresource name, stackname) with list[0] being (None, rootstackname), heat_resource_name will be set to the resource's name, and heat_resource_uuid will be set to the resource's orchestration id.
stack_user_domain_id = None	(String) Keystone domain ID which contains heat template-defined users. If this option is set, stack_user_domain_name option will be ignored.
stack_user_domain_name = None	(String) Keystone domain name which contains heat template-defined users. If `stack_user_domain_id` option is set, this option is ignored.
stale_token_duration = 30	(Integer) Gap, in seconds, to determine whether the given token is about to expire.
trusts_delegated_roles =	(List) Subset of trustor roles to be delegated to heat. If left unset, all roles of a user will be delegated to heat when creating a stack.
[auth_password]	
allowed_auth_uris =	(List) Allowed keystone endpoints for auth_uri when multi_cloud is enabled. At least one endpoint needs to be specified.
multi_cloud = False	(Boolean) Allow orchestration of multiple clouds.

Configuration option = Default value	Description
[ec2authtoken]	
allowed_auth_uris =	(List) Allowed keystone endpoints for auth_uri when multi_cloud is enabled. At least one endpoint needs to be specified.
auth_uri = None	(String) Authentication Endpoint URI.
ca_file = None	(String) Optional CA cert file to use in SSL connections.
cert_file = None	(String) Optional PEM-formatted certificate chain file.
insecure = False	(Boolean) If set, then the server's certificate will not be verified.
key_file = None	(String) Optional PEM-formatted file that contains the private key.
multi_cloud = False	(Boolean) Allow orchestration of multiple clouds.
[eventlet_opts]	
client_socket_timeout = 900	(Integer) Timeout for client connections' socket operations. If an incoming connection is idle for this number of seconds it will be closed. A value of '0' means wait forever.
wsgi_keep_alive = True	(Boolean) If False, closes the client socket connection explicitly.
[heat_api]	
backlog = 4096	(Integer) Number of backlog requests to configure the socket with.
bind_host = 0.0.0.0	(Unknown) Address to bind the server. Useful when selecting a particular network interface.
bind_port = 8004	(Port number) The port on which the server will listen.
cert_file = None	(String) Location of the SSL certificate file to use for SSL mode.

Configuration option = Default value	Description
key_file = <i>None</i>	(String) Location of the SSL key file to use for enabling SSL mode.
max_header_line = <i>16384</i>	(Integer) Maximum line size of message headers to be accepted. max_header_line may need to be increased when using large tokens (typically those generated by the Keystone v3 API with big service catalogs).
tcp_keepidle = <i>600</i>	(Integer) The value for the socket option TCP_KEEPIDLE. This is the time in seconds that the connection must be idle before TCP starts sending keepalive probes.
workers = <i>0</i>	(Integer) Number of workers for Heat service. Default value 0 means, that service will start number of workers equal number of cores on server.
[oslo_middleware]	
max_request_body_size = <i>114688</i>	(Integer) The maximum body size for each request, in bytes.
secure_proxy_ssl_header = <i>X-Forwarded-Proto</i>	(String) DEPRECATED: The HTTP Header that will be used to determine what the original request protocol scheme was, even if it was hidden by an SSL termination proxy.
[oslo_policy]	
policy_default_rule = <i>default</i>	(String) Default rule. Enforced when a requested rule is not found.
policy_dirs = <i>['policy.d']</i>	(Multi-valued) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the config_dir option, or absolute paths. The file defined by policy_file must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(String) The JSON file that defines policies.
[oslo_versionedobjects]	
fatal_exception_format_errors = <i>False</i>	(Boolean) Make exception message format errors fatal
[paste_deploy]	

Configuration option = Default value	Description
api_paste_config = <i>api-paste.ini</i>	(String) The API paste config file to use.
flavor = <i>None</i>	(String) The flavor to use.

Table 11.15. Description of Cloudformation-compatible API configuration options

Configuration option = Default value	Description
[DEFAULT]	
instance_connection_https_validate_certificates = <i>1</i>	(String) Instance connection to CFN/CW API validate certs if SSL is used.
instance_connection_is_secure = <i>0</i>	(String) Instance connection to CFN/CW API via https.
[heat_api_cfn]	
backlog = <i>4096</i>	(Integer) Number of backlog requests to configure the socket with.
bind_host = <i>0.0.0.0</i>	(Unknown) Address to bind the server. Useful when selecting a particular network interface.
bind_port = <i>8000</i>	(Port number) The port on which the server will listen.
cert_file = <i>None</i>	(String) Location of the SSL certificate file to use for SSL mode.
key_file = <i>None</i>	(String) Location of the SSL key file to use for enabling SSL mode.
max_header_line = <i>16384</i>	(Integer) Maximum line size of message headers to be accepted. max_header_line may need to be increased when using large tokens (typically those generated by the Keystone v3 API with big service catalogs).
tcp_keepidle = <i>600</i>	(Integer) The value for the socket option TCP_KEEPIIDLE. This is the time in seconds that the connection must be idle before TCP starts sending keepalive probes.
workers = <i>1</i>	(Integer) Number of workers for Heat service.

Table 11.16. Description of CloudWatch API configuration options

Configuration option = Default value	Description
[DEFAULT]	
enable_cloud_watch_lite = <i>False</i>	(Boolean) Enable the legacy OS::Heat::CWLiteAlarm resource.
heat_watch_server_url =	(String) URL of the Heat CloudWatch server.
[heat_api_cloudwatch]	
backlog = 4096	(Integer) Number of backlog requests to configure the socket with.
bind_host = 0.0.0.0	(Unknown) Address to bind the server. Useful when selecting a particular network interface.
bind_port = 8003	(Port number) The port on which the server will listen.
cert_file = <i>None</i>	(String) Location of the SSL certificate file to use for SSL mode.
key_file = <i>None</i>	(String) Location of the SSL key file to use for enabling SSL mode.
max_header_line = 16384	(Integer) Maximum line size of message headers to be accepted. max_header_line may need to be increased when using large tokens (typically those generated by the Keystone v3 API with big service catalogs.)
tcp_keepidle = 600	(Integer) The value for the socket option TCP_KEEPIDLE. This is the time in seconds that the connection must be idle before TCP starts sending keepalive probes.
workers = 1	(Integer) Number of workers for Heat service.

Table 11.17. Description of metadata API configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
<code>heat_metadata_server_url = None</code>	(String) URL of the Heat metadata server. NOTE: Setting this is only needed if you require instances to use a different endpoint than in the keystone catalog

Table 11.18. Description of waitcondition API configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>heat_waitcondition_server_url = None</code>	(String) URL of the Heat waitcondition server.

11.2. CONFIGURE CLIENTS

The following options allow configuration of the clients that Orchestration uses to talk to other services.

Table 11.19. Description of clients configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>region_name_for_services = None</code>	(String) Default region name used to get services endpoints.
[clients]	
<code>ca_file = None</code>	(String) Optional CA cert file to use in SSL connections.
<code>cert_file = None</code>	(String) Optional PEM-formatted certificate chain file.
<code>endpoint_type = publicURL</code>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
<code>insecure = False</code>	(Boolean) If set, then the server's certificate will not be verified.
<code>key_file = None</code>	(String) Optional PEM-formatted file that contains the private key.

Table 11.20. Description of client backends configuration options

Configuration option = Default value	Description
[DEFAULT]	
cloud_backend = <i>heat.engine.clients.OpenStackClients</i>	(String) Fully qualified class name to use as a client backend.

Table 11.21. Description of ceilometer clients configuration options

Configuration option = Default value	Description
[clients_ceilometer]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.22. Description of cinder clients configuration options

Configuration option = Default value	Description
[clients_cinder]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
http_log_debug = <i>False</i>	(Boolean) Allow client's debug log output.

Configuration option = Default value	Description
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.23. Description of glance clients configuration options

Configuration option = Default value	Description
[clients_glance]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.24. Description of heat clients configuration options

Configuration option = Default value	Description
[clients_heat]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.

Configuration option = Default value	Description
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.
url =	(String) Optional heat url in format like <code>http://0.0.0.0:8004/v1/(tenant_id)s</code> .

Table 11.25. Description of keystone clients configuration options

Configuration option = Default value	Description
[clients_keystone]	
auth_uri =	(String) Unversioned keystone url in format like <code>http://0.0.0.0:5000</code> .
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.26. Description of neutron clients configuration options

Configuration option = Default value	Description
[clients_neutron]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.

Configuration option = Default value	Description
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.27. Description of nova clients configuration options

Configuration option = Default value	Description
[clients_nova]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
http_log_debug = <i>False</i>	(Boolean) Allow client's debug log output.
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.28. Description of sahara clients configuration options

Configuration option = Default value	Description
[clients_sahara]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.

Configuration option = Default value	Description
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.29. Description of swift clients configuration options

Configuration option = Default value	Description
[clients_swift]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.
key_file = <i>None</i>	(String) Optional PEM-formatted file that contains the private key.

Table 11.30. Description of trove clients configuration options

Configuration option = Default value	Description
[clients_trove]	
ca_file = <i>None</i>	(String) Optional CA cert file to use in SSL connections.
cert_file = <i>None</i>	(String) Optional PEM-formatted certificate chain file.
endpoint_type = <i>None</i>	(String) Type of endpoint in Identity service catalog to use for communication with the OpenStack service.
insecure = <i>None</i>	(Boolean) If set, then the server's certificate will not be verified.

Configuration option = Default value	Description
<code>key_file = None</code>	(String) Optional PEM-formatted file that contains the private key.

11.3. CONFIGURE THE RPC MESSAGING SYSTEM

OpenStack projects use an open standard for messaging middleware known as AMQP. This messaging middleware enables the OpenStack services that run on multiple servers to talk to each other. OpenStack Oslo RPC supports two implementations of AMQP: **RabbitMQ** and **Qpid**.

11.3.1. Configure RabbitMQ

OpenStack Oslo RPC uses **RabbitMQ** by default. Use these options to configure the **RabbitMQ** message system. The `rpc_backend` option is optional as long as **RabbitMQ** is the default messaging system. However, if it is included in the configuration, you must set it to `heat.openstack.common.rpc.impl_kombu`.

```
rpc_backend = heat.openstack.common.rpc.impl_kombu
```

Use these options to configure the **RabbitMQ** messaging system. You can configure messaging communication for different installation scenarios, tune retries for RabbitMQ, and define the size of the RPC thread pool. To monitor notifications through RabbitMQ, you must set the `notification_driver` option to `heat.openstack.common.notifier.rpc_notifier` in the `heat.conf` file:

Table 11.31. Description of RabbitMQ configuration options

Configuration option = Default value	Description
<code>[oslo_messaging_rabbit]</code>	
<code>amqp_auto_delete = False</code>	(Boolean) Auto-delete queues in AMQP.
<code>amqp_durable_queues = False</code>	(Boolean) Use durable queues in AMQP.
<code>channel_max = None</code>	(Integer) Maximum number of channels to allow
<code>default_notification_exchange = \${control_exchange}_notification</code>	(String) Exchange name for for sending notifications
<code>default_notification_retry_attempts = -1</code>	(Integer) Reconnecting retry count in case of connectivity problem during sending notification, -1 means infinite retry.
<code>default_rpc_exchange = \${control_exchange}_rpc</code>	(String) Exchange name for sending RPC messages

Configuration option = Default value	Description
default_rpc_retry_attempts = -1	(Integer) Reconnecting retry count in case of connectivity problem during sending RPC message, -1 means infinite retry. If actual retry attempts in not 0 the rpc request could be processed more then one time
fake_rabbit = <i>False</i>	(Boolean) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
frame_max = <i>None</i>	(Integer) The maximum byte size for an AMQP frame
heartbeat_interval = 1	(Integer) How often to send heartbeats for consumer's connections
heartbeat_rate = 2	(Integer) How often times during the <code>heartbeat_timeout_threshold</code> we check the heartbeat.
heartbeat_timeout_threshold = 60	(Integer) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disable the heartbeat). EXPERIMENTAL
host_connection_reconnect_delay = 0.25	(Floating point) Set delay for reconnection to some host which has connection error
kombu_compression = <i>None</i>	(String) EXPERIMENTAL: Possible values are: <code>gzip</code> , <code>bz2</code> . If not set compression will not be used. This option may not be available in future versions.
kombu_failover_strategy = <i>round-robin</i>	(String) Determines how the next RabbitMQ node is chosen in case the one we are currently connected to becomes unavailable. Takes effect only if more than one RabbitMQ node is provided in config.
kombu_missing_consumer_retry_timeout = 60	(Integer) How long to wait a missing client before abandoning to send it its replies. This value should not be longer than <code>rpc_response_timeout</code> .
kombu_reconnect_delay = 1.0	(Floating point) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_ssl_ca_certs =	(String) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(String) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(String) SSL key file (valid only if SSL enabled).

Configuration option = Default value	Description
kombu_ssl_version =	(String) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be available on some distributions.
notification_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to notification listener.
notification_persistence = False	(Boolean) Persist notification messages.
notification_retry_delay = 0.25	(Floating point) Reconnecting retry delay in case of connectivity problem during sending notification message
pool_max_overflow = 0	(Integer) Maximum number of connections to create above `pool_max_size`.
pool_max_size = 10	(Integer) Maximum number of connections to keep queued.
pool_recycle = 600	(Integer) Lifetime of a connection (since creation) in seconds or None for no recycling. Expired connections are closed on acquire.
pool_stale = 60	(Integer) Threshold at which inactive (since release) connections are considered stale in seconds or None for no staleness. Stale connections are closed on acquire.
pool_timeout = 30	(Integer) Default number of seconds to wait for a connections to available
rabbit_ha_queues = False	(Boolean) Try to use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database. In RabbitMQ 3.0, queue mirroring is no longer controlled by the x-ha-policy argument when declaring a queue. If you just want to make sure that all queues (except those with auto-generated names) are mirrored across all nodes, run: <code>"rabbitmqctl set_policy HA '^(?!amq\\.).*' '{"ha-mode": "all"}' "</code>
rabbit_host = localhost	(String) The RabbitMQ broker address where a single node is used.
rabbit_hosts = \$rabbit_host:\$rabbit_port	(List) RabbitMQ HA cluster host:port pairs.

Configuration option = Default value	Description
rabbit_interval_max = 30	(Integer) Maximum interval of RabbitMQ connection retries. Default is 30 seconds.
rabbit_login_method = <i>AMQPLAIN</i>	(String) The RabbitMQ login method.
rabbit_max_retries = 0	(Integer) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = <i>guest</i>	(String) The RabbitMQ password.
rabbit_port = 5672	(Port number) The RabbitMQ broker port where a single node is used.
rabbit_qos_prefetch_count = 0	(Integer) Specifies the number of messages to prefetch. Setting to zero allows unlimited messages.
rabbit_retry_backoff = 2	(Integer) How long to backoff for between retries when connecting to RabbitMQ.
rabbit_retry_interval = 1	(Integer) How frequently to retry connecting with RabbitMQ.
rabbit_transient_queues_ttl = 1800	(Integer) Positive integer representing duration in seconds for queue TTL (x-expires). Queues which are unused for the duration of the TTL are automatically deleted. The parameter affects only reply and fanout queues.
rabbit_use_ssl = <i>False</i>	(Boolean) Connect over SSL for RabbitMQ.
rabbit_userid = <i>guest</i>	(String) The RabbitMQ userid.
rabbit_virtual_host = /	(String) The RabbitMQ virtual host.
rpc_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc listener.
rpc_queue_expiration = 60	(Integer) Time to live for rpc queues without consumers in seconds.
rpc_reply_exchange = <i>\${control_exchange}_rpc_reply</i>	(String) Exchange name for receiving RPC replies
rpc_reply_listener_prefetch_count = 100	(Integer) Max number of not acknowledged message which RabbitMQ can send to rpc reply listener.

Configuration option = Default value	Description
<code>rpc_reply_retry_attempts = -1</code>	(Integer) Reconnecting retry count in case of connectivity problem during sending reply. -1 means infinite retry during <code>rpc_timeout</code>
<code>rpc_reply_retry_delay = 0.25</code>	(Floating point) Reconnecting retry delay in case of connectivity problem during sending reply.
<code>rpc_retry_delay = 0.25</code>	(Floating point) Reconnecting retry delay in case of connectivity problem during sending RPC message
<code>socket_timeout = 0.25</code>	(Floating point) Set socket timeout in seconds for connection's socket
<code>ssl = None</code>	(Boolean) Enable SSL
<code>ssl_options = None</code>	(Dict) Arguments passed to <code>ssl.wrap_socket</code>
<code>tcp_user_timeout = 0.25</code>	(Floating point) Set <code>TCP_USER_TIMEOUT</code> in seconds for connection's socket

11.3.2. Configure Qpid

Use these options to configure the **Qpid** messaging system for OpenStack Oslo RPC. **Qpid** is not the default messaging system, so you must enable it by setting the `rpc_backend` option in the `heat.conf` file:

```
rpc_backend=heat.openstack.common.rpc.impl_qpid
```

This critical option points the compute nodes to the **Qpid** broker (server). Set the `qpid_hostname` option to the host name where the broker runs in the `heat.conf` file.



NOTE

The `qpid_hostname` option accepts a host name or IP address value.

```
qpid_hostname = hostname.example.com
```

If the **Qpid** broker listens on a port other than the AMQP default of **5672**, you must set the `qpid_port` option to that value:

```
qpid_port = 12345
```

If you configure the **Qpid** broker to require authentication, you must add a user name and password to the configuration:

```
qpid_username = username
qpid_password = password
```

By default, TCP is used as the transport. To enable SSL, set the `qpid_protocol` option:

```
qpid_protocol = ssl
```

Use these additional options to configure the Qpid messaging driver for OpenStack Oslo RPC. These options are used infrequently.

Table 11.32. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
<code>amqp_auto_delete = False</code>	(BoolOpt) Auto-delete queues in AMQP.
<code>amqp_durable_queues = False</code>	(BoolOpt) Use durable queues in AMQP.
<code>qpid_heartbeat = 60</code>	(IntOpt) Seconds between connection keepalive heartbeats.
<code>qpid_hostname = localhost</code>	(StrOpt) Qpid broker hostname.
<code>qpid_hosts = \$qpid_hostname:\$qpid_port</code>	(ListOpt) Qpid HA cluster host:port pairs.
<code>qpid_password =</code>	(StrOpt) Password for Qpid connection.
<code>qpid_port = 5672</code>	(IntOpt) Qpid broker port.
<code>qpid_protocol = tcp</code>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
<code>qpid_receiver_capacity = 1</code>	(IntOpt) The number of prefetched messages held by receiver.
<code>qpid_sasl_mechanisms =</code>	(StrOpt) Space separated list of SASL mechanisms to use for auth.
<code>qpid_tcp_nodelay = True</code>	(BoolOpt) Whether to disable the Nagle algorithm.
<code>qpid_topology_version = 1</code>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
<code>qpid_username =</code>	(StrOpt) Username for Qpid connection.
<code>rpc_conn_pool_size = 30</code>	(IntOpt) Size of RPC connection pool.

11.3.3. Configure messaging

Use these common options to configure the **RabbitMQ** and **Qpid** messaging drivers:

Table 11.33. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>openstack</i>	(String) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
default_notification_level = <i>INFO</i>	(String) Default notification level for outgoing notifications.
default_publisher_id = <i>None</i>	(String) Default publisher_id for outgoing notifications.
transport_url = <i>None</i>	(String) A URL representing the messaging driver to use and its full configuration. If not set, we fall back to the <code>rpc_backend</code> option and driver specific configuration.

Table 11.34. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
engine_life_check_timeout = 2	(Integer) RPC timeout for the engine liveness check that is used for stack locking.
rpc_backend = <i>rabbit</i>	(String) The messaging driver to use, defaults to rabbit. Other drivers include <code>amqp</code> and <code>zmq</code> .
rpc_cast_timeout = -1	(Integer) Seconds to wait before a cast expires (TTL). The default value of -1 specifies an infinite linger period. The value of 0 specifies no linger period. Pending messages shall be discarded immediately when the socket is closed. Only supported by <code>impl_zmq</code> .
rpc_conn_pool_size = 30	(Integer) Size of RPC connection pool.
rpc_poll_timeout = 1	(Integer) The default number of seconds that poll should wait. Poll raises timeout exception when timeout expired.

Configuration option = Default value	Description
rpc_response_timeout = <i>60</i>	(Integer) Seconds to wait for a response from a call.
[oslo_concurrency]	
disable_process_locking = <i>False</i>	(Boolean) Enables or disables inter-process locks.
lock_path = <i>None</i>	(String) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(Boolean) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>	(String) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(String) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(String) address prefix when sending to any server in group
idle_timeout = <i>0</i>	(Integer) Timeout for inactive connections (in seconds)
password =	(String) Password for message broker authentication
sasl_config_dir =	(String) Path to directory that contains the SASL configuration
sasl_config_name =	(String) Name of configuration file (without .conf suffix)
sasl_mechanisms =	(String) Space separated list of acceptable SASL mechanisms
server_request_prefix = <i>exclusive</i>	(String) address prefix used when sending to a specific server
ssl_ca_file =	(String) CA certificate PEM file to verify server certificate

Configuration option = Default value	Description
ssl_cert_file =	(String) Identifying certificate PEM file to present to clients
ssl_key_file =	(String) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(String) Password for decrypting ssl_key_file (if encrypted)
trace = <i>False</i>	(Boolean) Debug: dump AMQP frames to stdout
username =	(String) User name for message broker authentication
[oslo_messaging_notifications]	
driver = []	(Multi-valued) The Drivers(s) to handle sending notifications. Possible values are messaging, messagingv2, routing, log, test, noop
topics = <i>notifications</i>	(List) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(String) A URL representing the messaging driver to use for notifications. If not set, we fall back to the same configuration used for RPC.

Table 11.35. Description of notification configuration options

Configuration option = Default value	Description
[DEFAULT]	
onready = <i>None</i>	(String) Deprecated.

11.4. ORCHESTRATION LOG FILES

The corresponding log file of each Orchestration service is stored in the `/var/log/heat/` directory of the host on which each service runs.

Table 11.36. Log files used by Orchestration services

Configuration option = Default value	Description
heat-api.log	Orchestration service API Service
heat-engine.log	Orchestration service Engine Service

Configuration option = Default value	Description
<code>heat-manage.log</code>	Orchestration service events

11.5. NEW, UPDATED, AND DEPRECATED OPTIONS IN MITAKA FOR ORCHESTRATION SERVICE

There are no new, updated, and deprecated options in Mitaka for OpenStack Object Storage.

CHAPTER 12. TELEMETRY

The Telemetry service collects measurements within OpenStack. Its various agents and services are configured in the `/etc/ceilometer/ceilometer.conf` file.

To install Telemetry, run the following command:

```
# yum install -y mongodb-server openstack-ceilometer-* python-ceilometer
python-ceilometerclient
```

The following tables provide a comprehensive list of the Telemetry configuration options.

Table 12.1. Description of alarm configuration options

Configuration option = Default value	Description
[alarm]	
evaluation_interval = 60	(IntOpt) Period of evaluation cycle, should be \geq than configured pipeline interval for collection of underlying metrics.
evaluation_service = default	(StrOpt) Driver to use for alarm evaluation service. DEPRECATED: "singleton" and "partitioned" alarm evaluator services will be removed in Kilo in favour of the default alarm evaluation service using tooz for partitioning.
notifier_rpc_topic = alarm_notifier	(StrOpt) The topic that ceilometer uses for alarm notifier messages.
partition_rpc_topic = alarm_partition_coordination	(StrOpt) The topic that ceilometer uses for alarm partition coordination messages. DEPRECATED: RPC-based partitioned alarm evaluation service will be removed in Kilo in favour of the default alarm evaluation service using tooz for partitioning.
project_alarm_quota = None	(IntOpt) Maximum number of alarms defined for a project.
record_history = True	(BoolOpt) Record alarm change events.
rest_notifier_certificate_file =	(StrOpt) SSL Client certificate for REST notifier.
rest_notifier_certificate_key =	(StrOpt) SSL Client private key for REST notifier.
rest_notifier_max_retries = 0	(IntOpt) Number of retries for REST notifier
rest_notifier_ssl_verify = True	(BoolOpt) Whether to verify the SSL Server certificate when calling alarm action.

Configuration option = Default value	Description
user_alarm_quota = <i>None</i>	(IntOpt) Maximum number of alarms defined for a user.

Table 12.2. Description of alarms configuration options

Configuration option = Default value	Description
[alarms]	
gnocchi_url = <i>http://localhost:8041</i>	(StrOpt) URL to Gnocchi.

Table 12.3. Description of AMQP configuration options

Configuration option = Default value	Description
[DEFAULT]	
control_exchange = <i>openstack</i>	(StrOpt) The default exchange under which topics are scoped. May be overridden by an exchange name specified in the <code>transport_url</code> option.
notification_driver = <i>[]</i>	(MultiStrOpt) Driver or drivers to handle sending notifications.
notification_topics = <i>notifications</i>	(ListOpt) AMQP topic used for OpenStack notifications.
transport_url = <i>None</i>	(StrOpt) A URL representing the messaging driver to use and its full configuration. If not set, fall back to the <code>rpc_backend</code> option and driver specific configuration.

Table 12.4. Description of API configuration options

Configuration option = Default value	Description
[DEFAULT]	
api_paste_config = <i>api_paste.ini</i>	(StrOpt) Configuration file for WSGI definition of API.
api_workers = <i>1</i>	(IntOpt) Number of workers for Ceilometer API server.
event_pipeline_cfg_file = <i>event_pipeline.yaml</i>	(StrOpt) Configuration file for event pipeline definition.

Configuration option = Default value	Description
pipeline_cfg_file = <i>pipeline.yaml</i>	(StrOpt) Configuration file for pipeline definition.
reserved_metadata_keys =	(ListOpt) List of metadata keys reserved for metering use. And these keys are additional to the ones included in the namespace.
reserved_metadata_length = 256	(IntOpt) Limit on length of reserved metadata values.
reserved_metadata_namespace = <i>metering</i> .	(ListOpt) List of metadata prefixes reserved for metering use.
[api]	
host = 0.0.0.0	(StrOpt) The listen IP for the ceilometer API server.
pecan_debug = <i>False</i>	(BoolOpt) Toggle Pecan Debug Middleware.
port = 8777	(IntOpt) The port for the ceilometer API server.

Table 12.5. Description of authorization configuration options

Configuration option = Default value	Description
[service_credentials]	
insecure = <i>False</i>	(BoolOpt) Disables X.509 certificate validation when an SSL connection to Identity Service is established.
os_auth_url = <i>http://localhost:5000/v2.0</i>	(StrOpt) Auth URL to use for OpenStack service access.
os_cacert = <i>None</i>	(StrOpt) Certificate chain for SSL validation.
os_endpoint_type = <i>publicURL</i>	(StrOpt) Type of endpoint in Identity service catalog to use for communication with OpenStack services.
os_password = <i>admin</i>	(StrOpt) Password to use for OpenStack service access.
os_region_name = <i>None</i>	(StrOpt) Region name to use for OpenStack service endpoints.
os_tenant_id =	(StrOpt) Tenant ID to use for OpenStack service access.

Configuration option = Default value	Description
os_tenant_name = <i>admin</i>	(StrOpt) Tenant name to use for OpenStack service access.
os_username = <i>ceilometer</i>	(StrOpt) User name to use for OpenStack service access.

Table 12.6. Description of authorization token configuration options

Configuration option = Default value	Description
[keystone_authtoken]	
admin_password = <i>None</i>	(StrOpt) Service user password.
admin_tenant_name = <i>admin</i>	(StrOpt) Service tenant name.
admin_token = <i>None</i>	(StrOpt) This option is deprecated and may be removed in a future release. Single shared secret with the Keystone configuration used for bootstrapping a Keystone installation, or otherwise bypassing the normal authentication process. This option should not be used, use <code>`admin_user`</code> and <code>`admin_password`</code> instead.
admin_user = <i>None</i>	(StrOpt) Service username.
auth_admin_prefix =	(StrOpt) Prefix to prepend at the beginning of the path. Deprecated, use <code>identity_uri</code> .
auth_host = <i>127.0.0.1</i>	(StrOpt) Host providing the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_plugin = <i>None</i>	(StrOpt) Name of the plugin to load
auth_port = <i>35357</i>	(IntOpt) Port of the admin Identity API endpoint. Deprecated, use <code>identity_uri</code> .
auth_protocol = <i>https</i>	(StrOpt) Protocol of the admin Identity API endpoint (http or https). Deprecated, use <code>identity_uri</code> .
auth_section = <i>None</i>	(StrOpt) Config Section from which to load plugin specific options
auth_uri = <i>None</i>	(StrOpt) Complete public Identity API endpoint.

Configuration option = Default value	Description
auth_version = <i>None</i>	(StrOpt) API version of the admin Identity API endpoint.
cache = <i>None</i>	(StrOpt) Env key for the swift cache.
cafile = <i>None</i>	(StrOpt) A PEM encoded Certificate Authority to use when verifying HTTPs connections. Defaults to system CAs.
certfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate
check_revocations_for_cached = <i>False</i>	(BoolOpt) If true, the revocation list will be checked for cached tokens. This requires that PKI tokens are configured on the identity server.
delay_auth_decision = <i>False</i>	(BoolOpt) Do not handle authorization requests within the middleware, but delegate the authorization decision to downstream WSGI components.
enforce_token_bind = <i>permissive</i>	(StrOpt) Used to control the use and type of token binding. Can be set to: "disabled" to not check token binding. "permissive" (default) to validate binding information if the bind type is of a form known to the server and ignore it if not. "strict" like "permissive" but if the bind type is unknown the token will be rejected. "required" any form of token binding is needed to be allowed. Finally the name of a binding method that must be present in tokens.
hash_algorithms = <i>md5</i>	(ListOpt) Hash algorithms to use for hashing PKI tokens. This may be a single algorithm or multiple. The algorithms are those supported by Python standard hashlib.new(). The hashes will be tried in the order given, so put the preferred one first for performance. The result of the first hash will be stored in the cache. This will typically be set to multiple values only while migrating from a less secure algorithm to a more secure one. Once all the old tokens are expired this option should be set to a single value for better performance.
http_connect_timeout = <i>None</i>	(IntOpt) Request timeout value for communicating with Identity API server.
http_request_max_retries = <i>3</i>	(IntOpt) How many times to try to reconnect when communicating with Identity API Server.

Configuration option = Default value	Description
identity_uri = <i>None</i>	(StrOpt) Complete admin Identity API endpoint. This should specify the unversioned root endpoint e.g. https://localhost:35357/
include_service_catalog = <i>True</i>	(BoolOpt) (Optional) Indicate whether to set the X-Service-Catalog header. If False, middleware will not ask for service catalog on token validation and will not set the X-Service-Catalog header.
insecure = <i>False</i>	(BoolOpt) Verify HTTPS connections.
keyfile = <i>None</i>	(StrOpt) Required if identity server requires client certificate
memcache_pool_conn_get_timeout = <i>10</i>	(IntOpt) (Optional) Number of seconds that an operation will wait to get a memcache client connection from the pool.
memcache_pool_dead_retry = <i>300</i>	(IntOpt) (Optional) Number of seconds memcached server is considered dead before it is tried again.
memcache_pool_maxsize = <i>10</i>	(IntOpt) (Optional) Maximum total number of open connections to every memcached server.
memcache_pool_socket_timeout = <i>3</i>	(IntOpt) (Optional) Socket timeout in seconds for communicating with a memcache server.
memcache_pool_unused_timeout = <i>60</i>	(IntOpt) (Optional) Number of seconds a connection to memcached is held unused in the pool before it is closed.
memcache_secret_key = <i>None</i>	(StrOpt) (Optional, mandatory if <code>memcache_security_strategy</code> is defined) This string is used for key derivation.
memcache_security_strategy = <i>None</i>	(StrOpt) (Optional) If defined, indicate whether token data should be authenticated or authenticated and encrypted. Acceptable values are MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in the cache. If ENCRYPT, token data is encrypted and authenticated in the cache. If the value is not one of these options or empty, <code>auth_token</code> will raise an exception on initialization.
memcache_use_advanced_pool = <i>False</i>	(BoolOpt) (Optional) Use the advanced (eventlet safe) memcache client pool. The advanced pool will only work under python 2.x.

Configuration option = Default value	Description
revocation_cache_time = 10	(IntOpt) Determines the frequency at which the list of revoked tokens is retrieved from the Identity service (in seconds). A high number of revocation events combined with a low cache duration may significantly reduce performance.
signing_dir = <i>None</i>	(StrOpt) Directory used to cache files related to PKI tokens.
token_cache_time = 300	(IntOpt) In order to prevent excessive effort spent validating tokens, the middleware caches previously-seen tokens for a configurable duration (in seconds). Set to -1 to disable caching completely.

Table 12.7. Description of collector configuration options

Configuration option = Default value	Description
[DEFAULT]	
collector_workers = 1	(IntOpt) Number of workers for collector service. A single collector is enabled by default.
[collector]	
requeue_event_on_dispatcher_error = <i>False</i>	(BoolOpt) Requeue the event on the collector event queue when the collector fails to dispatch it.
requeue_sample_on_dispatcher_error = <i>False</i>	(BoolOpt) Requeue the sample on the collector sample queue when the collector fails to dispatch it. This is only valid if the sample come from the notifier publisher.
udp_address = 0.0.0.0	(StrOpt) Address to which the UDP socket is bound. Set to an empty string to disable.
udp_port = 4952	(IntOpt) Port to which the UDP socket is bound.
[dispatcher_file]	
backup_count = 0	(IntOpt) The max number of the files to keep.
file_path = <i>None</i>	(StrOpt) Name and the location of the file to record meters.
max_bytes = 0	(IntOpt) The max size of the file.

Table 12.8. Description of common configuration options

Configuration option = Default value	Description
[DEFAULT]	
host = <i>localhost</i>	(StrOpt) Name of this node, which must be valid in an AMQP key. Can be an opaque identifier.
http_timeout = <i>600</i>	(IntOpt) Timeout seconds for HTTP requests. Set it to None to disable timeout.
memcached_servers = <i>None</i>	(ListOpt) Memcached servers or None for in process cache.
notification_workers = <i>1</i>	(IntOpt) Number of workers for notification service. A single notification agent is enabled by default.
polling_namespaces = <i>['compute', 'central']</i>	(MultiChoicesOpt) Polling namespace(s) to be used while resource polling
pollster_list = <i>[]</i>	(MultiChoicesOpt) List of pollsters (or wildcard templates) to be used while polling
rootwrap_config = <i>/etc/ceilometer/rootwrap.conf</i>	(StrOpt) Path to the rootwrap configuration file to use for running commands as root
shuffle_time_before_polling_task = <i>0</i>	(IntOpt) To reduce large requests at same time to Nova or other components from different compute agents, shuffle start time of polling task.
sql_expire_samples_only = <i>False</i>	(BoolOpt) Indicates if expirer expires only samples. If set true, expired samples will be deleted, but residual resource and meter definition data will remain.
[compute]	
workload_partitioning = <i>False</i>	(BoolOpt) Enable work-load partitioning, allowing multiple compute agents to be run simultaneously.
[coordination]	
backend_url = <i>None</i>	(StrOpt) The backend URL to use for distributed coordination. If left empty, per-deployment central agent and per-host compute agent will not do workload partitioning and will only function correctly if a single instance of that service is running.

Configuration option = Default value	Description
check_watchers = 10.0	(FloatOpt) Number of seconds between checks to see if group membership has changed
heartbeat = 1.0	(FloatOpt) Number of seconds between heartbeats for distributed coordination.
[keystone_authtoken]	
memcached_servers = None	(ListOpt) Optionally specify a list of memcached server(s) to use for caching. If left undefined, tokens will instead be cached in-process.
[polling]	
partitioning_group_prefix = None	(StrOpt) Work-load partitioning group prefix. Use only if you want to run multiple polling agents with different config files. For each sub-group of the agent pool with the same partitioning_group_prefix a disjoint subset of pollsters should be loaded.

Table 12.9. Description of concurrency configuration options

Configuration option = Default value	Description
[oslo_concurrency]	
disable_process_locking = False	(BoolOpt) Enables or disables inter-process locks.
lock_path = None	(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.

Table 12.10. Description of database configuration options

Configuration option = Default value	Description
[DEFAULT]	
database_connection = None	(StrOpt) DEPRECATED - Database connection string.
[database]	

Configuration option = Default value	Description
alarm_connection = <i>None</i>	(StrOpt) The connection string used to connect to the alarm database. (if unset, connection is used)
backend = <i>sqlalchemy</i>	(StrOpt) The back end to use for the database.
connection = <i>None</i>	(StrOpt) The SQLAlchemy connection string to use to connect to the database.
connection_debug = <i>0</i>	(IntOpt) Verbosity of SQL debugging information: 0=None, 100=Everything.
connection_trace = <i>False</i>	(BoolOpt) Add Python stack traces to SQL as comment strings.
db2nosql_resource_id_maxlen = <i>512</i>	(IntOpt) The max length of resources id in DB2 nosql, the value should be larger than $\text{len}(\text{hostname}) * 2$ as compute node's resource id is <code><hostname>_<nodename></code> .
db_inc_retry_interval = <i>True</i>	(BoolOpt) If True, increases the interval between retries of a database operation up to <code>db_max_retry_interval</code> .
db_max_retries = <i>20</i>	(IntOpt) Maximum retries in case of connection error or deadlock error before error is raised. Set to -1 to specify an infinite retry count.
db_max_retry_interval = <i>10</i>	(IntOpt) If <code>db_inc_retry_interval</code> is set, the maximum seconds between retries of a database operation.
db_retry_interval = <i>1</i>	(IntOpt) Seconds between retries of a database transaction.
event_connection = <i>None</i>	(StrOpt) The connection string used to connect to the event database. (if unset, connection is used)
event_time_to_live = <i>-1</i>	(IntOpt) Number of seconds that events are kept in the database for (<code><= 0</code> means forever).
idle_timeout = <i>3600</i>	(IntOpt) Timeout before idle SQL connections are reaped.
max_overflow = <i>None</i>	(IntOpt) If set, use this value for <code>max_overflow</code> with SQLAlchemy.
max_pool_size = <i>None</i>	(IntOpt) Maximum number of SQL connections to keep open in a pool.

Configuration option = Default value	Description
<code>max_retries = 10</code>	(IntOpt) Maximum number of database connection retries during startup. Set to -1 to specify an infinite retry count.
<code>metering_connection = None</code>	(StrOpt) The connection string used to connect to the metering database. (if unset, connection is used)
<code>metering_time_to_live = -1</code>	(IntOpt) Number of seconds that samples are kept in the database for (<= 0 means forever).
<code>min_pool_size = 1</code>	(IntOpt) Minimum number of SQL connections to keep open in a pool.
<code>mongodb_replica_set =</code>	(StrOpt) The name of the replica set which is used to connect to MongoDB database. If it is set, MongoReplicaSetClient will be used instead of MongoClient.
<code>mysql_sql_mode = TRADITIONAL</code>	(StrOpt) The SQL mode to be used for MySQL sessions. This option, including the default, overrides any server-set SQL mode. To use whatever SQL mode is set by the server configuration, set this to no value. Example: <code>mysql_sql_mode=</code>
<code>pool_timeout = None</code>	(IntOpt) If set, use this value for pool_timeout with SQLAlchemy.
<code>retry_interval = 10</code>	(IntOpt) Interval between retries of opening a SQL connection.
<code>slave_connection = None</code>	(StrOpt) The SQLAlchemy connection string to use to connect to the slave database.
<code>sqlite_db = oslo.sqlite</code>	(StrOpt) The file name to use with SQLite.
<code>sqlite_synchronous = True</code>	(BoolOpt) If True, SQLite uses synchronous mode.
<code>use_db_reconnect = False</code>	(BoolOpt) Enable the experimental use of database reconnect on connection lost.

Table 12.11. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
backdoor_port = <i>None</i>	(StrOpt) Enable eventlet backdoor. Acceptable values are 0, <port>, and <start>:<end>, where 0 results in listening on a random tcp port number; <port> results in listening on the specified port number (and not enabling backdoor if that port is in use); and <start>:<end> results in listening on the smallest unused port number within the specified range of port numbers. The chosen port is displayed in the service's log file.
nova_http_log_debug = <i>False</i>	(BoolOpt) Allow novaclient's debug log output.

Table 12.12. Description of HTTP dispatcher configuration options

Configuration option = Default value	Description
[dispatcher_http]	
cadf_only = <i>False</i>	(BoolOpt) The flag that indicates if only cadf message should be posted. If false, all meters will be posted.
event_target = <i>None</i>	(StrOpt) The target for event data where the http request will be sent to. If this is not set, it will default to same as Sample target.
target =	(StrOpt) The target where the http request will be sent. If this is not set, no data will be posted. For example: target = http://hostname:1234/path
timeout = 5	(IntOpt) The max time in seconds to wait for a request to timeout.

Table 12.13. Description of events configuration options

Configuration option = Default value	Description
[event]	
definitions_cfg_file = <i>event_definitions.yaml</i>	(StrOpt) Configuration file for event definitions.
drop_unmatched_notifications = <i>False</i>	(BoolOpt) Drop notifications if no event definition matches. (Otherwise, they cannot be converted with only the default traits.)

Configuration option = Default value	Description
store_raw = []	(MultiStrOpt) Store the raw notification for select priority levels (info and/or error). By default, raw details are not captured.
[notification]	
ack_on_event_error = <i>True</i>	(BoolOpt) Acknowledge message when event persistence fails.
store_events = <i>False</i>	(BoolOpt) Save event details.
workload_partitioning = <i>False</i>	(BoolOpt) Enable workload partitioning, allowing multiple notification agents to be run simultaneously.

Table 12.14. Description of exchange configuration options

Configuration option = Default value	Description
[DEFAULT]	
cinder_control_exchange = <i>cinder</i>	(StrOpt) Exchange name for Cinder notifications.
glance_control_exchange = <i>glance</i>	(StrOpt) Exchange name for Glance notifications.
heat_control_exchange = <i>heat</i>	(StrOpt) Exchange name for Heat notifications
http_control_exchanges = [<i>'nova'</i> , <i>'glance'</i> , <i>'neutron'</i> , <i>'cinder'</i>]	(MultiStrOpt) Exchanges name to listen for notifications.
ironic_exchange = <i>ironic</i>	(StrOpt) Exchange name for Ironic notifications.
keystone_control_exchange = <i>keystone</i>	(StrOpt) Exchange name for Keystone notifications.
neutron_control_exchange = <i>neutron</i>	(StrOpt) Exchange name for Neutron notifications.
nova_control_exchange = <i>nova</i>	(StrOpt) Exchange name for Nova notifications.
sahara_control_exchange = <i>sahara</i>	(StrOpt) Exchange name for Data Processing notifications.
sample_source = <i>openstack</i>	(StrOpt) Source for samples emitted on this instance.
swift_control_exchange = <i>swift</i>	(StrOpt) Exchange name for Swift notifications.

Configuration option = Default value	Description
<code>trove_control_exchange = trove</code>	(StrOpt) Exchange name for DBaaS notifications.

Table 12.15. Description of glance configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>glance_page_size = 0</code>	(IntOpt) Number of items to request in each paginated Glance API request (parameter used by glanceclient). If this is less than or equal to 0, page size is not specified (default value in glanceclient is used).

Table 12.16. Description of inspector configuration options

Configuration option = Default value	Description
[DEFAULT]	
<code>hypervisor_inspector = libvirt</code>	(StrOpt) Inspector to use for inspecting the hypervisor layer.
<code>libvirt_type = kvm</code>	(StrOpt) Libvirt domain type.
<code>libvirt_uri =</code>	(StrOpt) Override the default libvirt URI (which is dependent on libvirt_type).

Table 12.17. Description of IPMI configuration options

Configuration option = Default value	Description
[ipmi]	
<code>node_manager_init_retry = 3</code>	(IntOpt) Number of retries upon Intel Node Manager initialization failure
<code>polling_retry = 3</code>	(IntOpt) Tolerance of IPMI/NM polling failures before disable this pollster. Negative indicates retrying forever.

Table 12.18. Description of oslo_middleware configuration options

Configuration option = Default value	Description
[oslo_middleware]	
max_request_body_size = 114688	(IntOpt) The maximum body size for each request, in bytes.

Table 12.19. Description of logging configuration options

Configuration option = Default value	Description
[DEFAULT]	
debug = <i>False</i>	(BoolOpt) Print debugging output (set logging level to DEBUG instead of default WARNING level).
default_log_levels = <i>amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN, urllib3.connectionpool=WARN, websocket=WARN, keystone.middleware=WARN, routes.middleware=WARN, stevedore=WARN</i>	(ListOpt) List of logger=LEVEL pairs.
fatal_deprecations = <i>False</i>	(BoolOpt) Enables or disables fatal status of deprecations.
instance_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance that is passed with the log message.
instance_uuid_format = <i>"[instance: %(uuid)s] "</i>	(StrOpt) The format for an instance UUID that is passed with the log message.
log_config_append = <i>None</i>	(StrOpt) The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation.
log_date_format = <i>%Y-%m-%d %H:%M:%S</i>	(StrOpt) Format string for %(asctime)s in log records. Default: %(default)s .
log_dir = <i>None</i>	(StrOpt) (Optional) The base directory used for relative --log-file paths.
log_file = <i>None</i>	(StrOpt) (Optional) Name of log file to output to. If no default is set, logging will go to stdout.

Configuration option = Default value	Description
log_format = <i>None</i>	(StrOpt) DEPRECATED. A logging.Formatter log message format string which may use any of the available logging.LogRecord attributes. This option is deprecated. Use logging_context_format_string and logging_default_format_string instead.
logging_context_format_string = % (asctime)s.(msecs)03d %(process)d %(levelname)s % (name)s [% (request_id)s %(user_identity)s] % (instance)s%(message)s	(StrOpt) Format string to use for log messages with context.
logging_debug_format_suffix = % (funcName)s %(pathname)s: %(lineno)d	(StrOpt) Data to append to log format when level is DEBUG.
logging_default_format_string = % (asctime)s.(msecs)03d %(process)d %(levelname)s % (name)s [-] %(instance)s%(message)s	(StrOpt) Format string to use for log messages without context.
logging_exception_prefix = %(asctime)s.(msecs)03d %(process)d TRACE %(name)s % (instance)s	(StrOpt) Prefix each line of exception output with this format.
publish_errors = <i>False</i>	(BoolOpt) Enables or disables publication of error events.
syslog_log_facility = <i>LOG_USER</i>	(StrOpt) Syslog facility to receive log lines.
use_stderr = <i>True</i>	(BoolOpt) Log output to standard error.
use_syslog = <i>False</i>	(BoolOpt) Use syslog for logging. Existing syslog format is DEPRECATED during I, and will change in J to honor RFC5424.
use_syslog_rfc_format = <i>False</i>	(BoolOpt) (Optional) Enables or disables syslog rfc5424 format for logging. If enabled, prefixes the MSG part of the syslog message with APP-NAME (RFC5424). The format without the APP-NAME is deprecated in I, and will be removed in J.
verbose = <i>False</i>	(BoolOpt) Print more verbose output (set logging level to INFO instead of default WARNING level).

Table 12.20. Description of MagetoDB configuration options

Configuration option = Default value	Description
[DEFAULT]	

Configuration option = Default value	Description
magnetodb_control_exchange = <i>magnetodb</i>	(StrOpt) Exchange name for Magnetodb notifications.

Table 12.21. Description of notification configuration options

Configuration option = Default value	Description
[notification]	
disable_non_metric_meters = <i>False</i>	(BoolOpt) WARNING: Ceilometer historically offered the ability to store events as meters. This usage is NOT advised as it can flood the metering database and cause performance degradation. This option disables the collection of non-metric meters and will be the default behavior in Liberty.

Table 12.22. Description of policy configuration options

Configuration option = Default value	Description
[oslo_policy]	
policy_default_rule = <i>default</i>	(StrOpt) Default rule. Enforced when a requested rule is not found.
policy_dirs = <i>['policy.d']</i>	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the <code>config_dir</code> option, or absolute paths. The file defined by <code>policy_file</code> must exist for these directories to be searched. Missing or empty directories are ignored.
policy_file = <i>policy.json</i>	(StrOpt) The JSON file that defines policies.

Table 12.23. Description of Qpid configuration options

Configuration option = Default value	Description
[oslo_messaging_qpid]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
qpid_heartbeat = <i>60</i>	(IntOpt) Seconds between connection keepalive heartbeats.

Configuration option = Default value	Description
qpid_hostname = <i>localhost</i>	(StrOpt) Qpid broker hostname.
qpid_hosts = <i>\$qpid_hostname:\$qpid_port</i>	(ListOpt) Qpid HA cluster host:port pairs.
qpid_password =	(StrOpt) Password for Qpid connection.
qpid_port = <i>5672</i>	(IntOpt) Qpid broker port.
qpid_protocol = <i>tcp</i>	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
qpid_receiver_capacity = <i>1</i>	(IntOpt) The number of prefetched messages held by receiver.
qpid_sasl_mechanisms =	(StrOpt) Space separated list of SASL mechanisms to use for auth.
qpid_tcp_nodelay = <i>True</i>	(BoolOpt) Whether to disable the Nagle algorithm.
qpid_topology_version = <i>1</i>	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
qpid_username =	(StrOpt) Username for Qpid connection.
rpc_conn_pool_size = <i>30</i>	(IntOpt) Size of RPC connection pool.

Table 12.24. Description of RabbitMQ configuration options

Configuration option = Default value	Description
[oslo_messaging_rabbit]	
amqp_auto_delete = <i>False</i>	(BoolOpt) Auto-delete queues in AMQP.
amqp_durable_queues = <i>False</i>	(BoolOpt) Use durable queues in AMQP.
fake_rabbit = <i>False</i>	(BoolOpt) Deprecated, use <code>rpc_backend=kombu+memory</code> or <code>rpc_backend=fake</code>
heartbeat_rate = <i>2</i>	(IntOpt) How often times during the <code>heartbeat_timeout_threshold</code> to check the heartbeat.

Configuration option = Default value	Description
heartbeat_timeout_threshold = 0	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disables the heartbeat, >0 enables it. Enabling heartbeats requires kombu>=3.0.7 and amqp>=1.4.0). EXPERIMENTAL
kombu_reconnect_delay = 1.0	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).
kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 are also available.
rabbit_ha_queues = False	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
rabbit_host = localhost	(StrOpt) The RabbitMQ broker address where a single node is used.
rabbit_hosts = \$rabbit_host:\$rabbit_port	(ListOpt) RabbitMQ HA cluster host:port pairs.
rabbit_login_method = AMQPLAIN	(StrOpt) The RabbitMQ login method.
rabbit_max_retries = 0	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
rabbit_password = guest	(StrOpt) The RabbitMQ password.
rabbit_port = 5672	(IntOpt) The RabbitMQ broker port where a single node is used.
rabbit_retry_backoff = 2	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
rabbit_retry_interval = 1	(IntOpt) How frequently to retry connecting with RabbitMQ.
rabbit_use_ssl = False	(BoolOpt) Connect over SSL for RabbitMQ.

Configuration option = Default value	Description
rabbit_userid = <i>guest</i>	(StrOpt) The RabbitMQ userid.
rabbit_virtual_host = <i>/</i>	(StrOpt) The RabbitMQ virtual host.
rpc_conn_pool_size = <i>30</i>	(IntOpt) Size of RPC connection pool.

Table 12.25. Description of Redis configuration options

Configuration option = Default value	Description
[matchmaker_redis]	
host = <i>127.0.0.1</i>	(StrOpt) Host to locate redis.
password = <i>None</i>	(StrOpt) Password for Redis server (optional).
port = <i>6379</i>	(IntOpt) Use this port to connect to redis host.
[matchmaker_ring]	
ringfile = <i>/etc/oslo/matchmaker_ring.json</i>	(StrOpt) Matchmaker ring file (JSON).

Table 12.26. Description of Rados gateway configuration options

Configuration option = Default value	Description
[rgw_admin_credentials]	
access_key = <i>None</i>	(StrOpt) Access key for Radosgw Admin.
secret_key = <i>None</i>	(StrOpt) Secret key for Radosgw Admin.

Table 12.27. Description of RPC configuration options

Configuration option = Default value	Description
[DEFAULT]	
dispatcher = <i>['database']</i>	(MultiStrOpt) Dispatcher to process data.
matchmaker_heartbeat_freq = <i>300</i>	(IntOpt) Heartbeat frequency.
matchmaker_heartbeat_ttl = <i>600</i>	(IntOpt) Heartbeat time-to-live.

Configuration option = Default value	Description
rpc_backend = <i>rabbit</i>	(StrOpt) The messaging driver to use, defaults to rabbit. Other drivers include qpid and zmq.
rpc_cast_timeout = 30	(IntOpt) Seconds to wait before a cast expires (TTL). Only supported by impl_zmq.
rpc_response_timeout = 60	(IntOpt) Seconds to wait for a response from a call.
rpc_thread_pool_size = 64	(IntOpt) Size of RPC thread pool.
[notification]	
messaging_urls = []	(MultiStrOpt) Messaging URLs to listen for notifications. Example: transport://user:pass@host1:port[,hostN:portN]/virtual_host (DEFAULT/transport_url is used if empty)
[oslo_messaging_amqp]	
allow_insecure_clients = <i>False</i>	(BoolOpt) Accept clients using either SSL or plain TCP
broadcast_prefix = <i>broadcast</i>	(StrOpt) address prefix used when broadcasting to all servers
container_name = <i>None</i>	(StrOpt) Name for the AMQP container
group_request_prefix = <i>unicast</i>	(StrOpt) address prefix when sending to any server in group
idle_timeout = 0	(IntOpt) Timeout for inactive connections (in seconds)
server_request_prefix = <i>exclusive</i>	(StrOpt) address prefix used when sending to a specific server
ssl_ca_file =	(StrOpt) CA certificate PEM file for verifying server certificate
ssl_cert_file =	(StrOpt) Identifying certificate PEM file to present to clients
ssl_key_file =	(StrOpt) Private key PEM file used to sign cert_file certificate
ssl_key_password = <i>None</i>	(StrOpt) Password for decrypting ssl_key_file (if encrypted)

Configuration option = Default value	Description
trace = <i>False</i>	(BoolOpt) Debug: dump AMQP frames to stdout
[publisher]	
telemetry_secret = <i>change this for valid signing</i>	(StrOpt) Secret value for signing messages. Set value empty if signing is not required to avoid computational overhead.
[publisher_notifier]	
event_topic = <i>event</i>	(StrOpt) The topic that ceilometer uses for event notifications.
metering_topic = <i>metering</i>	(StrOpt) The topic that ceilometer uses for metering notifications.
telemetry_driver = <i>messagingv2</i>	(StrOpt) The driver that ceilometer uses for metering notifications.
[publisher_rpc]	
metering_topic = <i>metering</i>	(StrOpt) The topic that ceilometer uses for metering messages.

Table 12.28. Description of service types configuration options

Configuration option = Default value	Description
[service_types]	
glance = <i>image</i>	(StrOpt) Glance service type.
kwapi = <i>energy</i>	(StrOpt) Kwapi service type.
neutron = <i>network</i>	(StrOpt) Neutron service type.
nova = <i>compute</i>	(StrOpt) Nova service type.
radosgw = <i>object-store</i>	(StrOpt) Radosgw service type.
swift = <i>object-store</i>	(StrOpt) Swift service type.

Table 12.29. Description of swift configuration options

Configuration option = Default value	Description
[DEFAULT]	
reseller_prefix = <i>AUTH_</i>	(StrOpt) Swift reseller prefix. Must be on par with reseller_prefix in <i>proxy-server.conf</i> .

Table 12.30. Description of TripleO configuration options

Configuration option = Default value	Description
[hardware]	
readonly_user_name = <i>ro_snmp_user</i>	(StrOpt) SNMPd user name of all nodes running in the cloud.
readonly_user_password = <i>password</i>	(StrOpt) SNMPd password of all the nodes running in the cloud.
url_scheme = <i>snmp://</i>	(StrOpt) URL scheme to use for hardware nodes.

Table 12.31. Description of VMware configuration options

Configuration option = Default value	Description
[vmware]	
api_retry_count = <i>10</i>	(IntOpt) Number of times a VMware Vsphere API may be retried.
host_ip =	(StrOpt) IP address of the VMware Vsphere host.
host_password =	(StrOpt) Password of VMware Vsphere.
host_port = <i>443</i>	(IntOpt) Port of the VMware Vsphere host.
host_username =	(StrOpt) Username of VMware Vsphere.
task_poll_interval = <i>0.5</i>	(FloatOpt) Sleep time in seconds for polling an ongoing async task.
wSDL_location = <i>None</i>	(StrOpt) Optional vim service WSDL location e.g <i>http://<server>/vimService.wsdl</i> . Optional override to default location for bug work-arounds.

Table 12.32. Description of XenAPI configuration options

Configuration option = Default value	Description
[xenapi]	
connection_password = <i>None</i>	(StrOpt) Password for connection to XenServer/Xen Cloud Platform.
connection_url = <i>None</i>	(StrOpt) URL for connection to XenServer/Xen Cloud Platform.
connection_username = <i>root</i>	(StrOpt) Username for connection to XenServer/Xen Cloud Platform.
login_timeout = <i>10</i>	(IntOpt) Timeout in seconds for XenAPI login.

Table 12.33. Description of Zaqar configuration options

Configuration option = Default value	Description
[DEFAULT]	
zaqar_control_exchange = <i>zaqar</i>	(StrOpt) Exchange name for Messaging service notifications.

12.1. TELEMETRY SAMPLE CONFIGURATION FILES

All the files in this section can be found in the `/etc/ceilometer/` directory.

12.1.1. ceilometer.conf

The configuration for the Telemetry services and agents is found in the `ceilometer.conf` file.

This file must be modified after installation.

```
[DEFAULT]

#
# From ceilometer
#

# To reduce large requests at same time to Nova or other components
# from different compute agents, shuffle start time of polling task.
# (integer value)
#shuffle_time_before_polling_task = 0

# Configuration file for WSGI definition of API. (string value)
#api_paste_config = api_paste.ini

# Number of workers for Ceilometer API server. (integer value)
```

```

#api_workers = 1

# Polling namespace(s) to be used while resource polling (unknown
# type)
#polling_namespaces = ['compute', 'central']

# List of pollsters (or wildcard templates) to be used while polling
# (unknown type)
#pollster_list = []

# Exchange name for Nova notifications. (string value)
#nova_control_exchange = nova

# List of metadata prefixes reserved for metering use. (list value)
#reserved_metadata_namespace = metering.

# Limit on length of reserved metadata values. (integer value)
#reserved_metadata_length = 256

# List of metadata keys reserved for metering use. And these keys are
# additional to the ones included in the namespace. (list value)
#reserved_metadata_keys =

# Inspector to use for inspecting the hypervisor layer. (string value)
#hypervisor_inspector = libvirt

# Libvirt domain type. (string value)
# Allowed values: kvm, lxc, qemu, uml, xen
#libvirt_type = kvm

# Override the default libvirt URI (which is dependent on
# libvirt_type). (string value)
#libvirt_uri =

# Exchange name for Data Processing notifications. (string value)
#sahara_control_exchange = sahara

# Dispatcher to process data. (multi valued)
# Deprecated group/name - [collector]/dispatcher
#dispatcher = database

# Exchange name for Keystone notifications. (string value)
#keystone_control_exchange = keystone

# Number of items to request in each paginated Glance API request
# (parameter used by glanceclient). If this is less than or equal to
# 0, page size is not specified (default value in glanceclient is
# used). (integer value)
#glance_page_size = 0

# Exchange name for Glance notifications. (string value)
#glance_control_exchange = glance

# Exchange name for Ironic notifications. (string value)
#ironic_exchange = ironic

```

```

# Exchanges name to listen for notifications. (multi valued)
#http_control_exchanges = nova
#http_control_exchanges = glance
#http_control_exchanges = neutron
#http_control_exchanges = cinder

# Exchange name for Neutron notifications. (string value)
# Deprecated group/name - [DEFAULT]/quantum_control_exchange
#neutron_control_exchange = neutron

# Allow novaclient's debug log output. (boolean value)
#nova_http_log_debug = false

# Swift reseller prefix. Must be on par with reseller_prefix in proxy-
# server.conf. (string value)
#reseller_prefix = AUTH_

# Enable eventlet backdoor. Acceptable values are 0, <port>, and
# <start>:<end>, where 0 results in listening on a random tcp port
# number; <port> results in listening on the specified port number
# (and not enabling backdoor if that port is in use); and
# <start>:<end> results in listening on the smallest unused port
# number within the specified range of port numbers. The chosen port
# is displayed in the service's log file. (string value)
#backdoor_port = <None>

# Print debugging output (set logging level to DEBUG instead of
# default WARNING level). (boolean value)
#debug = false

# Print more verbose output (set logging level to INFO instead of
# default WARNING level). (boolean value)
#verbose = false

# Log output to standard error. (boolean value)
#use_stderr = true

# The name of a logging configuration file. This file is appended to
# any existing logging configuration files. For details about logging
# configuration files, see the Python logging module documentation.
# (string value)
# Deprecated group/name - [DEFAULT]/log_config
#log_config_append = <None>

# DEPRECATED. A logging.Formatter log message format string which may
# use any of the available logging.LogRecord attributes. This option
# is deprecated. Please use logging_context_format_string and
# logging_default_format_string instead. (string value)
#log_format = <None>

# Format string for %(asctime)s in log records. Default: %(default)s
# . (string value)
#log_date_format = %Y-%m-%d %H:%M:%S

# (Optional) Name of log file to output to. If no default is set,
# logging will go to stdout. (string value)

```



```

# Deprecated group/name - [DEFAULT]/logfile
#log_file = <None>

# (Optional) The base directory used for relative --log-file paths.
# (string value)
# Deprecated group/name - [DEFAULT]/logdir
#log_dir = <None>

# Use syslog for logging. Existing syslog format is DEPRECATED during
# I, and will change in J to honor RFC5424. (boolean value)
#use_syslog = false

# (Optional) Enables or disables syslog rfc5424 format for logging. If
# enabled, prefixes the MSG part of the syslog message with APP-NAME
# (RFC5424). The format without the APP-NAME is deprecated in I, and
# will be removed in J. (boolean value)
#use_syslog_rfc_format = false

# Syslog facility to receive log lines. (string value)
#syslog_log_facility = LOG_USER

# Format string to use for log messages with context. (string value)
#logging_context_format_string = %(asctime)s.%(msecs)03d %(process)d %
#(levelname)s %(name)s [%(request_id)s %(user_identity)s] %(instance)s%
#(message)s

# Format string to use for log messages without context. (string
# value)
#logging_default_format_string = %(asctime)s.%(msecs)03d %(process)d %
#(levelname)s %(name)s [-] %(instance)s%(message)s

# Data to append to log format when level is DEBUG. (string value)
#logging_debug_format_suffix = %(funcName)s %(pathname)s:%(lineno)d

# Prefix each line of exception output with this format. (string
# value)
#logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d TRACE %
#(name)s %(instance)s

# List of logger=LEVEL pairs. (list value)
#default_log_levels =
amqp=WARN, amqpplib=WARN, boto=WARN, qpid=WARN, sqlalchemy=WARN, suds=INFO, oslo.
messaging=INFO, iso8601=WARN, requests.packages.urllib3.connectionpool=WARN,
urllib3.connectionpool=WARN, websocket=WARN, keystonemiddleware=WARN, routes.
middleware=WARN, stevedore=WARN

# Enables or disables publication of error events. (boolean value)
#publish_errors = false

# Enables or disables fatal status of deprecations. (boolean value)
#fatal_deprecations = false

# The format for an instance that is passed with the log message.
# (string value)
#instance_format = "[instance: %(uuid)s] "

```

```
# The format for an instance UUID that is passed with the log message.
# (string value)
#instance_uuid_format = "[instance: %(uuid)s] "

# Exchange name for Heat notifications (string value)
#heat_control_exchange = heat

# Configuration file for pipeline definition. (string value)
#pipeline_cfg_file = pipeline.yaml

# Configuration file for event pipeline definition. (string value)
#event_pipeline_cfg_file = event_pipeline.yaml

# Exchange name for DBaaS notifications. (string value)
#trove_control_exchange = trove

# Exchange name for Messaging service notifications. (string value)
#zaqar_control_exchange = zaqar

# Source for samples emitted on this instance. (string value)
# Deprecated group/name - [DEFAULT]/counter_source
#sample_source = openstack

# Name of this node, which must be valid in an AMQP key. Can be an
# opaque identifier. For ZeroMQ only, must be a valid host name, FQDN,
# or IP address. (string value)
#host = shock

# Number of workers for collector service. A single collector is
# enabled by default. (integer value)
#collector_workers = 1

# Number of workers for notification service. A single notification
# agent is enabled by default. (integer value)
#notification_workers = 1

# Timeout seconds for HTTP requests. Set it to None to disable
# timeout. (integer value)
#http_timeout = 600

# DEPRECATED - Database connection string. (string value)
#database_connection = <None>

# Path to the rootwrap configuration file touse for running commands
# as root (string value)
#rootwrap_config = /etc/ceilometer/rootwrap.conf

# Exchange name for Cinder notifications. (string value)
#cinder_control_exchange = cinder

#
# From oslo.messaging
#

# ZeroMQ bind address. Should be a wildcard (*), an ethernet
# interface, or IP. The "host" option should point or resolve to this
```

```

# address. (string value)
#rpc_zmq_bind_address = *

# MatchMaker driver. (string value)
#rpc_zmq_matchmaker = local

# ZeroMQ receiver listening port. (integer value)
#rpc_zmq_port = 9501

# Number of ZeroMQ contexts, defaults to 1. (integer value)
#rpc_zmq_contexts = 1

# Maximum number of ingress messages to locally buffer per topic.
# Default is unlimited. (integer value)
#rpc_zmq_topic_backlog = <None>

# Directory for holding IPC sockets. (string value)
#rpc_zmq_ipc_dir = /var/run/openstack

# Name of this node. Must be a valid hostname, FQDN, or IP address.
# Must match "host" option, if running Nova. (string value)
#rpc_zmq_host = localhost

# Seconds to wait before a cast expires (TTL). Only supported by
# impl_zmq. (integer value)
#rpc_cast_timeout = 30

# Heartbeat frequency. (integer value)
#matchmaker_heartbeat_freq = 300

# Heartbeat time-to-live. (integer value)
#matchmaker_heartbeat_ttl = 600

# Size of RPC thread pool. (integer value)
#rpc_thread_pool_size = 64

# Driver or drivers to handle sending notifications. (multi valued)
#notification_driver =

# AMQP topic used for OpenStack notifications. (list value)
# Deprecated group/name - [rpc_notifier2]/topics
#notification_topics = notifications

# Seconds to wait for a response from a call. (integer value)
#rpc_response_timeout = 60

# A URL representing the messaging driver to use and its full
# configuration. If not set, we fall back to the rpc_backend option
# and driver specific configuration. (string value)
#transport_url = <None>

# The messaging driver to use, defaults to rabbit. Other drivers
# include qpuid and zmq. (string value)
#rpc_backend = rabbit

# The default exchange under which topics are scoped. May be

```

```

# overridden by an exchange name specified in the transport_url
# option. (string value)
#control_exchange = openstack

[alarm]

#
# From ceilometer
#

# SSL Client certificate for REST notifier. (string value)
#rest_notifier_certificate_file =

# SSL Client private key for REST notifier. (string value)
#rest_notifier_certificate_key =

# Whether to verify the SSL Server certificate when calling alarm
# action. (boolean value)
#rest_notifier_ssl_verify = true

# Number of retries for REST notifier (integer value)
#rest_notifier_max_retries = 0

# Period of evaluation cycle, should be >= than configured pipeline
# interval for collection of underlying metrics. (integer value)
# Deprecated group/name - [alarm]/threshold_evaluation_interval
#evaluation_interval = 60

# The topic that ceilometer uses for alarm notifier messages. (string
# value)
#notifier_rpc_topic = alarm_notifier

# The topic that ceilometer uses for alarm partition coordination
# messages. DEPRECATED: RPC-based partitionedalarm evaluation service
# will be removed in Kilo in favour of the default alarm evaluation
# service using tooz for partitioning. (string value)
#partition_rpc_topic = alarm_partition_coordination

# URL to Gnocchi. (string value)
#gnocchi_url = http://localhost:8041

# Record alarm change events. (boolean value)
#record_history = true

# Maximum number of alarms defined for a user. (integer value)
#user_alarm_quota = <None>

# Maximum number of alarms defined for a project. (integer value)
#project_alarm_quota = <None>

# Driver to use for alarm evaluation service. DEPRECATED: "singleton"
# and "partitioned" alarm evaluator services will be removed in Kilo
# in favour of the default alarm evaluation service using tooz for
# partitioning. (string value)
#evaluation_service = default

```

```

[api]

#
# From ceilometer
#

# The port for the ceilometer API server. (integer value)
# Deprecated group/name - [DEFAULT]/metering_api_port
#port = 8777

# The listen IP for the ceilometer API server. (string value)
#host = 0.0.0.0

# Toggle Pecan Debug Middleware. (boolean value)
#pecan_debug = false


[central]

#
# From ceilometer
#

# Work-load partitioning group prefix. Use only if you want to run
# multiple polling agents with different config files. For each sub-
# group of the agent pool with the same partitioning_group_prefix a
# disjoint subset of pollsters should be loaded. (string value)
# Deprecated group/name - [central]/partitioning_group_prefix
#partitioning_group_prefix = <None>


[collector]

#
# From ceilometer
#

# Address to which the UDP socket is bound. Set to an empty string to
# disable. (string value)
#udp_address = 0.0.0.0

# Port to which the UDP socket is bound. (integer value)
#udp_port = 4952

# Requeue the sample on the collector sample queue when the collector
# fails to dispatch it. This is only valid if the sample come from the
# notifier publisher. (boolean value)
#requeue_sample_on_dispatcher_error = false

# Requeue the event on the collector event queue when the collector
# fails to dispatch it. (boolean value)
#requeue_event_on_dispatcher_error = false

```

[compute]

```
#
# From ceilometer
#

# Enable work-load partitioning, allowing multiple compute agents to
# be run simultaneously. (boolean value)
#workload_partitioning = false
```

[coordination]

```
#
# From ceilometer
#

# The backend URL to use for distributed coordination. If left empty,
# per-deployment central agent and per-host compute agent won't do
# workload partitioning and will only function correctly if a single
# instance of that service is running. (string value)
#backend_url = <None>

# Number of seconds between heartbeats for distributed coordination.
# (floating point value)
#heartbeat = 1.0

# Number of seconds between checks to see if group membership has
# changed (floating point value)
#check_watchers = 10.0
```

[database]

```
#
# From ceilometer
#

# Number of seconds that samples are kept in the database for (<= 0
# means forever). (integer value)
# Deprecated group/name - [database]/time_to_live
#metering_time_to_live = -1

# Number of seconds that events are kept in the database for (<= 0
# means forever). (integer value)
#event_time_to_live = -1

# The connection string used to connect to the metering database. (if
# unset, connection is used) (string value)
#metering_connection = <None>

# The connection string used to connect to the alarm database. (if
# unset, connection is used) (string value)
#alarm_connection = <None>

# The connection string used to connect to the event database. (if
```

```

# unset, connection is used) (string value)
#event_connection = <None>

# The name of the replica set which is used to connect to MongoDB
# database. If it is set, MongoReplicaSetClient will be used instead
# of MongoClient. (string value)
#mongodb_replica_set =

# The max length of resources id in DB2 nosql, the value should be
# larger than len(hostname) * 2 as compute node's resource id is
# <hostname>_<nodename>. (integer value)
#db2nosql_resource_id_maxlen = 512

#
# From oslo.db
#

# The file name to use with SQLite. (string value)
# Deprecated group/name - [DEFAULT]/sqlite_db
#sqlite_db = oslo.sqlite

# If True, SQLite uses synchronous mode. (boolean value)
# Deprecated group/name - [DEFAULT]/sqlite_synchronous
#sqlite_synchronous = true

# The back end to use for the database. (string value)
# Deprecated group/name - [DEFAULT]/db_backend
#backend = sqlalchemy

# The SQLAlchemy connection string to use to connect to the database.
# (string value)
# Deprecated group/name - [DEFAULT]/sql_connection
# Deprecated group/name - [DATABASE]/sql_connection
# Deprecated group/name - [sql]/connection
#connection = <None>

# The SQLAlchemy connection string to use to connect to the slave
# database. (string value)
#slave_connection = <None>

# The SQL mode to be used for MySQL sessions. This option, including
# the default, overrides any server-set SQL mode. To use whatever SQL
# mode is set by the server configuration, set this to no value.
# Example: mysql_sql_mode= (string value)
#mysql_sql_mode = TRADITIONAL

# Timeout before idle SQL connections are reaped. (integer value)
# Deprecated group/name - [DEFAULT]/sql_idle_timeout
# Deprecated group/name - [DATABASE]/sql_idle_timeout
# Deprecated group/name - [sql]/idle_timeout
#idle_timeout = 3600

# Minimum number of SQL connections to keep open in a pool. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_min_pool_size
# Deprecated group/name - [DATABASE]/sql_min_pool_size

```

```

#min_pool_size = 1

# Maximum number of SQL connections to keep open in a pool. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_max_pool_size
# Deprecated group/name - [DATABASE]/sql_max_pool_size
#max_pool_size = <None>

# Maximum number of database connection retries during startup. Set to
# -1 to specify an infinite retry count. (integer value)
# Deprecated group/name - [DEFAULT]/sql_max_retries
# Deprecated group/name - [DATABASE]/sql_max_retries
#max_retries = 10

# Interval between retries of opening a SQL connection. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_retry_interval
# Deprecated group/name - [DATABASE]/reconnect_interval
#retry_interval = 10

# If set, use this value for max_overflow with SQLAlchemy. (integer
# value)
# Deprecated group/name - [DEFAULT]/sql_max_overflow
# Deprecated group/name - [DATABASE]/sqlalchemy_max_overflow
#max_overflow = <None>

# Verbosity of SQL debugging information: 0=None, 100=Everything.
# (integer value)
# Deprecated group/name - [DEFAULT]/sql_connection_debug
#connection_debug = 0

# Add Python stack traces to SQL as comment strings. (boolean value)
# Deprecated group/name - [DEFAULT]/sql_connection_trace
#connection_trace = false

# If set, use this value for pool_timeout with SQLAlchemy. (integer
# value)
# Deprecated group/name - [DATABASE]/sqlalchemy_pool_timeout
#pool_timeout = <None>

# Enable the experimental use of database reconnect on connection
# lost. (boolean value)
#use_db_reconnect = false

# Seconds between retries of a database transaction. (integer value)
#db_retry_interval = 1

# If True, increases the interval between retries of a database
# operation up to db_max_retry_interval. (boolean value)
#db_inc_retry_interval = true

# If db_inc_retry_interval is set, the maximum seconds between retries
# of a database operation. (integer value)
#db_max_retry_interval = 10

# Maximum retries in case of connection error or deadlock error before

```



```

# error is raised. Set to -1 to specify an infinite retry count.
# (integer value)
#db_max_retries = 20

[dispatcher_file]

#
# From ceilometer
#

# Name and the location of the file to record meters. (string value)
#file_path = <None>

# The max size of the file. (integer value)
#max_bytes = 0

# The max number of the files to keep. (integer value)
#backup_count = 0

[event]

#
# From ceilometer
#

# Configuration file for event definitions. (string value)
#definitions_cfg_file = event_definitions.yaml

# Drop notifications if no event definition matches. (Otherwise, we
# convert them with just the default traits) (boolean value)
#drop_unmatched_notifications = false

# Store the raw notification for select priority levels (info and/or
# error). By default, raw details are not captured. (multi valued)
#store_raw =

[hardware]

#
# From ceilometer
#

# URL scheme to use for hardware nodes. (string value)
#url_scheme = snmp://

# SNMPd user name of all nodes running in the cloud. (string value)
#readonly_user_name = ro_snmp_user

# SNMPd password of all the nodes running in the cloud. (string value)
#readonly_user_password = password

[ipmi]

```

```

#
# From ceilometer
#

# Number of retries upon Intel Node Manager initialization failure
# (integer value)
#node_manager_init_retry = 3

# Tolerance of IPMI/NM polling failures before disable this pollster.
# Negative indicates retrying forever. (integer value)
#polling_retry = 3

[keystone_authtoken]

#
# From keystonemiddleware.auth_token
#

# Complete public Identity API endpoint. (string value)
#auth_uri = <None>

# API version of the admin Identity API endpoint. (string value)
#auth_version = <None>

# Do not handle authorization requests within the middleware, but
# delegate the authorization decision to downstream WSGI components.
# (boolean value)
#delay_auth_decision = false

# Request timeout value for communicating with Identity API server.
# (integer value)
#http_connect_timeout = <None>

# How many times are we trying to reconnect when communicating with
# Identity API Server. (integer value)
#http_request_max_retries = 3

# Env key for the swift cache. (string value)
#cache = <None>

# Required if identity server requires client certificate (string
# value)
#certfile = <None>

# Required if identity server requires client certificate (string
# value)
#keyfile = <None>

# A PEM encoded Certificate Authority to use when verifying HTTPS
# connections. Defaults to system CAs. (string value)
#cafile = <None>

# Verify HTTPS connections. (boolean value)
#insecure = false

```

```

# Directory used to cache files related to PKI tokens. (string value)
#signing_dir = <None>

# Optionally specify a list of memcached server(s) to use for caching.
# If left undefined, tokens will instead be cached in-process. (list
# value)
# Deprecated group/name - [DEFAULT]/memcache_servers
#memcached_servers = <None>

# In order to prevent excessive effort spent validating tokens, the
# middleware caches previously-seen tokens for a configurable duration
# (in seconds). Set to -1 to disable caching completely. (integer
# value)
#token_cache_time = 300

# Determines the frequency at which the list of revoked tokens is
# retrieved from the Identity service (in seconds). A high number of
# revocation events combined with a low cache duration may
# significantly reduce performance. (integer value)
#revocation_cache_time = 10

# (Optional) If defined, indicate whether token data should be
# authenticated or authenticated and encrypted. Acceptable values are
# MAC or ENCRYPT. If MAC, token data is authenticated (with HMAC) in
# the cache. If ENCRYPT, token data is encrypted and authenticated in
# the cache. If the value is not one of these options or empty,
# auth_token will raise an exception on initialization. (string value)
#memcache_security_strategy = <None>

# (Optional, mandatory if memcache_security_strategy is defined) This
# string is used for key derivation. (string value)
#memcache_secret_key = <None>

# (Optional) Number of seconds memcached server is considered dead
# before it is tried again. (integer value)
#memcache_pool_dead_retry = 300

# (Optional) Maximum total number of open connections to every
# memcached server. (integer value)
#memcache_pool_maxsize = 10

# (Optional) Socket timeout in seconds for communicating with a
# memcache server. (integer value)
#memcache_pool_socket_timeout = 3

# (Optional) Number of seconds a connection to memcached is held
# unused in the pool before it is closed. (integer value)
#memcache_pool_unused_timeout = 60

# (Optional) Number of seconds that an operation will wait to get a
# memcache client connection from the pool. (integer value)
#memcache_pool_conn_get_timeout = 10

# (Optional) Use the advanced (eventlet safe) memcache client pool.
# The advanced pool will only work under python 2.x. (boolean value)

```

```

#memcache_use_advanced_pool = false

# (Optional) Indicate whether to set the X-Service-Catalog header. If
# False, middleware will not ask for service catalog on token
# validation and will not set the X-Service-Catalog header. (boolean
# value)
#include_service_catalog = true

# Used to control the use and type of token binding. Can be set to:
# "disabled" to not check token binding. "permissive" (default) to
# validate binding information if the bind type is of a form known to
# the server and ignore it if not. "strict" like "permissive" but if
# the bind type is unknown the token will be rejected. "required" any
# form of token binding is needed to be allowed. Finally the name of a
# binding method that must be present in tokens. (string value)
#enforce_token_bind = permissive

# If true, the revocation list will be checked for cached tokens. This
# requires that PKI tokens are configured on the identity server.
# (boolean value)
#check_revocations_for_cached = false

# Hash algorithms to use for hashing PKI tokens. This may be a single
# algorithm or multiple. The algorithms are those supported by Python
# standard hashlib.new(). The hashes will be tried in the order given,
# so put the preferred one first for performance. The result of the
# first hash will be stored in the cache. This will typically be set
# to multiple values only while migrating from a less secure algorithm
# to a more secure one. Once all the old tokens are expired this
# option should be set to a single value for better performance. (list
# value)
#hash_algorithms = md5

# Prefix to prepend at the beginning of the path. Deprecated, use
# identity_uri. (string value)
#auth_admin_prefix =

# Host providing the admin Identity API endpoint. Deprecated, use
# identity_uri. (string value)
#auth_host = 127.0.0.1

# Port of the admin Identity API endpoint. Deprecated, use
# identity_uri. (integer value)
#auth_port = 35357

# Protocol of the admin Identity API endpoint (http or https).
# Deprecated, use identity_uri. (string value)
#auth_protocol = https

# Complete admin Identity API endpoint. This should specify the
# unversioned root endpoint e.g. https://localhost:35357/ (string
# value)
#identity_uri = <None>

# This option is deprecated and may be removed in a future release.
# Single shared secret with the Keystone configuration used for

```

```

# bootstrapping a Keystone installation, or otherwise bypassing the
# normal authentication process. This option should not be used, use
# `admin_user` and `admin_password` instead. (string value)
#admin_token = <None>

# Service username. (string value)
#admin_user = <None>

# Service user password. (string value)
#admin_password = <None>

# Service tenant name. (string value)
#admin_tenant_name = admin

[matchmaker_redis]

#
# From oslo.messaging
#

# Host to locate redis. (string value)
#host = 127.0.0.1

# Use this port to connect to redis host. (integer value)
#port = 6379

# Password for Redis server (optional). (string value)
#password = <None>

[matchmaker_ring]

#
# From oslo.messaging
#

# Matchmaker ring file (JSON). (string value)
# Deprecated group/name - [DEFAULT]/matchmaker_ringfile
#ringfile = /etc/oslo/matchmaker_ring.json

[notification]

#
# From ceilometer
#

# Acknowledge message when event persistence fails. (boolean value)
# Deprecated group/name - [collector]/ack_on_event_error
#ack_on_event_error = true

# Save event details. (boolean value)
# Deprecated group/name - [collector]/store_events
#store_events = false

```

```

# WARNING: Ceilometer historically offered the ability to store events
# as meters. This usage is NOT advised as it can flood the metering
# database and cause performance degradation. This option disables the
# collection of non-metric meters and will be the default behavior in
# Liberty. (boolean value)
#disable_non_metric_meters = false

# Enable workload partitioning, allowing multiple notification agents
# to be run simultaneously. (boolean value)
#workload_partitioning = false

# Messaging URLs to listen for notifications. Example:
# transport://user:pass@host1:port[,hostN:portN]/virtual_host
# (DEFAULT/transport_url is used if empty) (multi valued)
#messaging_urls =

[oslo_concurrency]

#
# From oslo.concurrency
#

# Enables or disables inter-process locks. (boolean value)
# Deprecated group/name - [DEFAULT]/disable_process_locking
#disable_process_locking = false

# Directory to use for lock files. For security, the specified
# directory should only be writable by the user running the processes
# that need locking. Defaults to environment variable OSLO_LOCK_PATH.
# If external locks are used, a lock path must be set. (string value)
# Deprecated group/name - [DEFAULT]/lock_path
#lock_path = <None>

[oslo_messaging_amqp]

#
# From oslo.messaging
#

# address prefix used when sending to a specific server (string value)
# Deprecated group/name - [amqp1]/server_request_prefix
#server_request_prefix = exclusive

# address prefix used when broadcasting to all servers (string value)
# Deprecated group/name - [amqp1]/broadcast_prefix
#broadcast_prefix = broadcast

# address prefix when sending to any server in group (string value)
# Deprecated group/name - [amqp1]/group_request_prefix
#group_request_prefix = unicast

# Name for the AMQP container (string value)
# Deprecated group/name - [amqp1]/container_name
#container_name = <None>

```

```

# Timeout for inactive connections (in seconds) (integer value)
# Deprecated group/name - [amqp1]/idle_timeout
#idle_timeout = 0

# Debug: dump AMQP frames to stdout (boolean value)
# Deprecated group/name - [amqp1]/trace
#trace = false

# CA certificate PEM file for verifying server certificate (string
# value)
# Deprecated group/name - [amqp1]/ssl_ca_file
#ssl_ca_file =

# Identifying certificate PEM file to present to clients (string
# value)
# Deprecated group/name - [amqp1]/ssl_cert_file
#ssl_cert_file =

# Private key PEM file used to sign cert_file certificate (string
# value)
# Deprecated group/name - [amqp1]/ssl_key_file
#ssl_key_file =

# Password for decrypting ssl_key_file (if encrypted) (string value)
# Deprecated group/name - [amqp1]/ssl_key_password
#ssl_key_password = <None>

# Accept clients using either SSL or plain TCP (boolean value)
# Deprecated group/name - [amqp1]/allow_insecure_clients
#allow_insecure_clients = false

[oslo_messaging_qpid]

#
# From oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
#amqp_durable_queues = false

# Auto-delete queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/amqp_auto_delete
#amqp_auto_delete = false

# Size of RPC connection pool. (integer value)
# Deprecated group/name - [DEFAULT]/rpc_conn_pool_size
#rpc_conn_pool_size = 30

# Qpid broker hostname. (string value)
# Deprecated group/name - [DEFAULT]/qpid_hostname
#qpid_hostname = localhost

# Qpid broker port. (integer value)

```

```

# Deprecated group/name - [DEFAULT]/qpid_port
#qpid_port = 5672

# Qpid HA cluster host:port pairs. (list value)
# Deprecated group/name - [DEFAULT]/qpid_hosts
#qpid_hosts = $qpid_hostname:$qpid_port

# Username for Qpid connection. (string value)
# Deprecated group/name - [DEFAULT]/qpid_username
#qpid_username =

# Password for Qpid connection. (string value)
# Deprecated group/name - [DEFAULT]/qpid_password
#qpid_password =

# Space separated list of SASL mechanisms to use for auth. (string
# value)
# Deprecated group/name - [DEFAULT]/qpid_sasl_mechanisms
#qpid_sasl_mechanisms =

# Seconds between connection keepalive heartbeats. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_heartbeat
#qpid_heartbeat = 60

# Transport to use, either 'tcp' or 'ssl'. (string value)
# Deprecated group/name - [DEFAULT]/qpid_protocol
#qpid_protocol = tcp

# Whether to disable the Nagle algorithm. (boolean value)
# Deprecated group/name - [DEFAULT]/qpid_tcp_nodelay
#qpid_tcp_nodelay = true

# The number of prefetched messages held by receiver. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_receiver_capacity
#qpid_receiver_capacity = 1

# The qpid topology version to use. Version 1 is what was originally
# used by impl_qpid. Version 2 includes some backwards-incompatible
# changes that allow broker federation to work. Users should update
# to version 2 when they are able to take everything down, as it
# requires a clean break. (integer value)
# Deprecated group/name - [DEFAULT]/qpid_topology_version
#qpid_topology_version = 1

[oslo_messaging_rabbit]

#
# From oslo.messaging
#

# Use durable queues in AMQP. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
#amqp_durable_queues = false

# Auto-delete queues in AMQP. (boolean value)

```



```

# Deprecated group/name - [DEFAULT]/amqp_auto_delete
#amqp_auto_delete = false

# Size of RPC connection pool. (integer value)
# Deprecated group/name - [DEFAULT]/rpc_conn_pool_size
#rpc_conn_pool_size = 30

# SSL version to use (valid only if SSL enabled). Valid values are
# TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 may be
# available on some distributions. (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_version
#kombu_ssl_version =

# SSL key file (valid only if SSL enabled). (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_keyfile
#kombu_ssl_keyfile =

# SSL cert file (valid only if SSL enabled). (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_certfile
#kombu_ssl_certfile =

# SSL certification authority file (valid only if SSL enabled).
# (string value)
# Deprecated group/name - [DEFAULT]/kombu_ssl_ca_certs
#kombu_ssl_ca_certs =

# How long to wait before reconnecting in response to an AMQP consumer
# cancel notification. (floating point value)
# Deprecated group/name - [DEFAULT]/kombu_reconnect_delay
#kombu_reconnect_delay = 1.0

# The RabbitMQ broker address where a single node is used. (string
# value)
# Deprecated group/name - [DEFAULT]/rabbit_host
#rabbit_host = localhost

# The RabbitMQ broker port where a single node is used. (integer
# value)
# Deprecated group/name - [DEFAULT]/rabbit_port
#rabbit_port = 5672

# RabbitMQ HA cluster host:port pairs. (list value)
# Deprecated group/name - [DEFAULT]/rabbit_hosts
#rabbit_hosts = $rabbit_host:$rabbit_port

# Connect over SSL for RabbitMQ. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_use_ssl
#rabbit_use_ssl = false

# The RabbitMQ userid. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_userid
#rabbit_userid = guest

# The RabbitMQ password. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_password
#rabbit_password = guest

```

```

# The RabbitMQ login method. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_login_method
#rabbit_login_method = AMQPPLAIN

# The RabbitMQ virtual host. (string value)
# Deprecated group/name - [DEFAULT]/rabbit_virtual_host
#rabbit_virtual_host = /

# How frequently to retry connecting with RabbitMQ. (integer value)
#rabbit_retry_interval = 1

# How long to backoff for between retries when connecting to RabbitMQ.
# (integer value)
# Deprecated group/name - [DEFAULT]/rabbit_retry_backoff
#rabbit_retry_backoff = 2

# Maximum number of RabbitMQ connection retries. Default is 0
# (infinite retry count). (integer value)
# Deprecated group/name - [DEFAULT]/rabbit_max_retries
#rabbit_max_retries = 0

# Use HA queues in RabbitMQ (x-ha-policy: all). If you change this
# option, you must wipe the RabbitMQ database. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_ha_queues
#rabbit_ha_queues = false

# Number of seconds after which the Rabbit broker is considered down
# if heartbeat's keep-alive fails (0 disables the heartbeat, >0
# enables it. Enabling heartbeats requires kombu>=3.0.7 and
# amqp>=1.4.0). EXPERIMENTAL (integer value)
#heartbeat_timeout_threshold = 0

# How often times during the heartbeat_timeout_threshold we check the
# heartbeat. (integer value)
#heartbeat_rate = 2

# Deprecated, use rpc_backend=kombu+memory or rpc_backend=fake
# (boolean value)
# Deprecated group/name - [DEFAULT]/fake_rabbit
#fake_rabbit = false

[oslo_policy]

#
# From oslo.policy
#

# The JSON file that defines policies. (string value)
# Deprecated group/name - [DEFAULT]/policy_file
#policy_file = policy.json

# Default rule. Enforced when a requested rule is not found. (string
# value)
# Deprecated group/name - [DEFAULT]/policy_default_rule

```

```

#policy_default_rule = default

# Directories where policy configuration files are stored. They can be
# relative to any directory in the search path defined by the
# config_dir option, or absolute paths. The file defined by
# policy_file must exist for these directories to be searched.
# Missing or empty directories are ignored. (multi valued)
# Deprecated group/name - [DEFAULT]/policy_dirs
#policy_dirs = policy.d

[polling]

#
# From ceilometer
#

# Work-load partitioning group prefix. Use only if you want to run
# multiple polling agents with different config files. For each sub-
# group of the agent pool with the same partitioning_group_prefix a
# disjoint subset of pollsters should be loaded. (string value)
# Deprecated group/name - [central]/partitioning_group_prefix
#partitioning_group_prefix = <None>

[publisher]

#
# From ceilometer
#

# Secret value for signing messages. Set value empty if signing is not
# required to avoid computational overhead. (string value)
# Deprecated group/name - [DEFAULT]/metering_secret
# Deprecated group/name - [publisher_rpc]/metering_secret
# Deprecated group/name - [publisher]/metering_secret
#telemetry_secret = change this for valid signing

[publisher_notifier]

#
# From ceilometer
#

# The topic that ceilometer uses for metering notifications. (string
# value)
#metering_topic = metering

# The topic that ceilometer uses for event notifications. (string
# value)
#event_topic = event

# The driver that ceilometer uses for metering notifications. (string
# value)
# Deprecated group/name - [DEFAULT]/metering_driver

```

```
#telemetry_driver = messagingv2

[publisher_rpc]

#
# From ceilometer
#

# The topic that ceilometer uses for metering messages. (string value)
# Deprecated group/name - [DEFAULT]/metering_topic
#metering_topic = metering

[rgw_admin_credentials]

#
# From ceilometer
#

# Access key for Radosgw Admin. (string value)
#access_key = <None>

# Secret key for Radosgw Admin. (string value)
#secret_key = <None>

[service_credentials]

#
# From ceilometer
#

# User name to use for OpenStack service access. (string value)
# Deprecated group/name - [DEFAULT]/os_username
#os_username = ceilometer

# Password to use for OpenStack service access. (string value)
# Deprecated group/name - [DEFAULT]/os_password
#os_password = admin

# Tenant ID to use for OpenStack service access. (string value)
# Deprecated group/name - [DEFAULT]/os_tenant_id
#os_tenant_id =

# Tenant name to use for OpenStack service access. (string value)
# Deprecated group/name - [DEFAULT]/os_tenant_name
#os_tenant_name = admin

# Certificate chain for SSL validation. (string value)
#os_cacert = <None>

# Auth URL to use for OpenStack service access. (string value)
# Deprecated group/name - [DEFAULT]/os_auth_url
#os_auth_url = http://localhost:5000/v2.0
```

```

# Region name to use for OpenStack service endpoints. (string value)
# Deprecated group/name - [DEFAULT]/os_region_name
#os_region_name = <None>

# Type of endpoint in Identity service catalog to use for
# communication with OpenStack services. (string value)
#os_endpoint_type = publicURL

# Disables X.509 certificate validation when an SSL connection to
# Identity Service is established. (boolean value)
#insecure = false

[service_types]

#
# From ceilometer
#

# Kwapi service type. (string value)
#kwapi = energy

# Glance service type. (string value)
#glance = image

# Neutron service type. (string value)
#neutron = network

# Nova service type. (string value)
#nova = compute

# Radosgw service type. (string value)
#radosgw = object-store

# Swift service type. (string value)
#swift = object-store

[vmware]

#
# From ceilometer
#

# IP address of the VMware Vsphere host. (string value)
#host_ip =

# Port of the VMware Vsphere host. (integer value)
#host_port = 443

# Username of VMware Vsphere. (string value)
#host_username =

# Password of VMware Vsphere. (string value)
#host_password =

```

```

# Number of times a VMware Vsphere API may be retried. (integer value)
#api_retry_count = 10

# Sleep time in seconds for polling an ongoing async task. (floating
# point value)
#task_poll_interval = 0.5

# Optional vim service WSDL location e.g
# http://<server>/vimService.wsdl. Optional over-ride to default
# location for bug work-arounds. (string value)
#wsdl_location = <None>

[xenapi]

#
# From ceilometer
#

# URL for connection to XenServer/Xen Cloud Platform. (string value)
#connection_url = <None>

# Username for connection to XenServer/Xen Cloud Platform. (string
# value)
#connection_username = root

# Password for connection to XenServer/Xen Cloud Platform. (string
# value)
#connection_password = <None>

# Timeout in seconds for XenAPI login. (integer value)
#login_timeout = 10

```

12.1.2. event_definitions.yaml

The `event_definitions.yaml` file defines how events received from other OpenStack components should be translated to Telemetry events.

This file provides a standard set of events and corresponding traits that may be of interest. This file can be modified to add and drop traits that operators may find useful.

```

---
- event_type: compute.instance.*
  traits: &instance_traits
    tenant_id:
      fields: payload.tenant_id
    user_id:
      fields: payload.user_id
    instance_id:
      fields: payload.instance_id
    host:
      fields: publisher_id

```

```

    plugin:
      name: split
      parameters:
        segment: 1
        max_split: 1
  service:
    fields: publisher_id
    plugin: split
  memory_mb:
    type: int
    fields: payload.memory_mb
  disk_gb:
    type: int
    fields: payload.disk_gb
  root_gb:
    type: int
    fields: payload.root_gb
  ephemeral_gb:
    type: int
    fields: payload.ephemeral_gb
  vcpus:
    type: int
    fields: payload.vcpus
  instance_type_id:
    type: int
    fields: payload.instance_type_id
  instance_type:
    fields: payload.instance_type
  state:
    fields: payload.state
  os_architecture:
    fields: payload.image_meta.'org.openstack__1__architecture'
  os_version:
    fields: payload.image_meta.'org.openstack__1__os_version'
  os_distro:
    fields: payload.image_meta.'org.openstack__1__os_distro'
  launched_at:
    type: datetime
    fields: payload.launched_at
  deleted_at:
    type: datetime
    fields: payload.deleted_at
- event_type: compute.instance.exists
  traits:
    <<: *instance_traits
    audit_period_beginning:
      type: datetime
      fields: payload.audit_period_beginning
    audit_period_ending:
      type: datetime
      fields: payload.audit_period_ending
- event_type: ['volume.exists', 'volume.create.*', 'volume.delete.*',
'volume.resize.*', 'volume.attach.*', 'volume.detach.*',
'volume.update.*', 'snapshot.exists', 'snapshot.create.*',
'snapshot.delete.*', 'snapshot.update.*']
  traits: &cinder_traits

```

```

    user_id:
        fields: payload.user_id
    project_id:
        fields: payload.tenant_id
    availability_zone:
        fields: payload.availability_zone
    display_name:
        fields: payload.display_name
    replication_status:
        fields: payload.replication_status
    status:
        fields: payload.status
    created_at:
        fields: payload.created_at
- event_type: ['volume.exists', 'volume.create.*', 'volume.delete.*',
'volume.resize.*', 'volume.attach.*', 'volume.detach.*',
'volume.update.*']
    traits:
        <<: *cinder_traits
        resource_id:
            fields: payload.volume_id
        host:
            fields: payload.host
        size:
            fields: payload.size
        type:
            fields: payload.volume_type
        replication_status:
            fields: payload.replication_status
- event_type: ['snapshot.exists', 'snapshot.create.*',
'snapshot.delete.*', 'snapshot.update.*']
    traits:
        <<: *cinder_traits
        resource_id:
            fields: payload.snapshot_id
        volume_id:
            fields: payload.volume_id
- event_type: ['image.update', 'image.upload', 'image.delete']
    traits: &glance_crud
        project_id:
            fields: payload.owner
        resource_id:
            fields: payload.id
        name:
            fields: payload.name
        status:
            fields: payload.status
        created_at:
            fields: payload.created_at
        user_id:
            fields: payload.owner
        deleted_at:
            fields: payload.deleted_at
        size:
            fields: payload.size
- event_type: image.send

```



```

traits: &glance_send
  receiver_project:
    fields: payload.receiver_tenant_id
  receiver_user:
    fields: payload.receiver_user_id
  user_id:
    fields: payload.owner_id
  image_id:
    fields: payload.image_id
  destination_ip:
    fields: payload.destination_ip
  bytes_sent:
    fields: payload.bytes_sent
- event_type: orchestration.stack.*
traits: &orchestration_crud
  project_id:
    fields: payload.tenant_id
  user_id:
    fields: ['_context_trustor_user_id', '_context_user_id']
  resource_id:
    fields: payload.stack_identity
- event_type: sahara.cluster.*
traits: &sahara_crud
  project_id:
    fields: payload.project_id
  user_id:
    fields: _context_user_id
  resource_id:
    fields: payload.cluster_id
- event_type: ['identity.user.*', 'identity.project.*',
'identity.group.*', 'identity.role.*', 'identity.OS-TRUST:trust.*',
                'identity.region.*', 'identity.service.*',
'identity.endpoint.*', 'identity.policy.*']
traits: &identity_crud
  resource_id:
    fields: payload.resource_info
  initiator_id:
    fields: payload.initiator.id
  project_id:
    fields: payload.initiator.project_id
  domain_id:
    fields: payload.initiator.domain_id
- event_type: identity.role_assignment.*
traits: &identity_role_assignment
  role:
    fields: payload.role
  group:
    fields: payload.group
  domain:
    fields: payload.domain
  user:
    fields: payload.user
  project:
    fields: payload.project
- event_type: identity.authenticate
traits: &identity_authenticate

```

```

    typeURI:
      fields: payload.typeURI
    id:
      fields: payload.id
    action:
      fields: payload.action
    eventType:
      fields: payload.eventType
    eventTime:
      fields: payload.eventTime
    outcome:
      fields: payload.outcome
    initiator_typeURI:
      fields: payload.initiator.typeURI
    initiator_id:
      fields: payload.initiator.id
    initiator_name:
      fields: payload.initiator.name
    initiator_host_agent:
      fields: payload.initiator.host.agent
    initiator_host_addr:
      fields: payload.initiator.host.address
    target_typeURI:
      fields: payload.target.typeURI
    target_id:
      fields: payload.target.id
    observer_typeURI:
      fields: payload.observer.typeURI
    observer_id:
      fields: payload.observer.id
- event_type: objectstore.http.request
traits: &objectstore_request
    typeURI:
      fields: payload.typeURI
    id:
      fields: payload.id
    action:
      fields: payload.action
    eventType:
      fields: payload.eventType
    eventTime:
      fields: payload.eventTime
    outcome:
      fields: payload.outcome
    initiator_typeURI:
      fields: payload.initiator.typeURI
    initiator_id:
      fields: payload.initiator.id
    initiator_project_id:
      fields: payload.initiator.project_id
    target_typeURI:
      fields: payload.target.typeURI
    target_id:
      fields: payload.target.id
    target_action:
      fields: payload.target.action

```

```

target_metadata_path:
  fields: payload.target.metadata.path
target_metadata_version:
  fields: payload.target.metadata.version
target_metadata_container:
  fields: payload.target.metadata.container
target_metadata_object:
  fields: payload.target.metadata.object
observer_id:
  fields: payload.observer.id
- event_type: magnetodb.table.*
  traits: &kv_store
  resource_id:
    fields: payload.table_uuid
  user_id:
    fields: _context_user_id
  project_id:
    fields: _context_tenant
- event_type: ['network.*', 'subnet.*', 'port.*', 'router.*',
'floatingip.*', 'pool.*', 'vip.*', 'member.*', 'health_monitor.*',
'firewall.*', 'firewall_policy.*', 'firewall_rule.*', 'vpnservice.*',
'ipsecpolicy.*', 'ikepolicy.*', 'ipsec_site_connection.*']
  traits: &network_traits
  user_id:
    fields: _context_user_id
  project_id:
    fields: _context_tenant_id
- event_type: network.*
  traits:
    <<: *network_traits
  resource_id:
    fields: ['payload.network.id', 'payload.id']
- event_type: subnet.*
  traits:
    <<: *network_traits
  resource_id:
    fields: ['payload.subnet.id', 'payload.id']
- event_type: port.*
  traits:
    <<: *network_traits
  resource_id:
    fields: ['payload.port.id', 'payload.id']
- event_type: router.*
  traits:
    <<: *network_traits
  resource_id:
    fields: ['payload.router.id', 'payload.id']
- event_type: floatingip.*
  traits:
    <<: *network_traits
  resource_id:
    fields: ['payload.floatingip.id', 'payload.id']
- event_type: pool.*
  traits:
    <<: *network_traits
  resource_id:

```

```

        fields: ['payload.pool.id', 'payload.id']
- event_type: vip.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.vip.id', 'payload.id']
- event_type: member.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.member.id', 'payload.id']
- event_type: health_monitor.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.health_monitor.id', 'payload.id']
- event_type: firewall.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.firewall.id', 'payload.id']
- event_type: firewall_policy.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.firewall_policy.id', 'payload.id']
- event_type: firewall_rule.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.firewall_rule.id', 'payload.id']
- event_type: vpnservice.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.vpnservice.id', 'payload.id']
- event_type: ipsecpolicy.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.ipsecpolicy.id', 'payload.id']
- event_type: ikcpolicy.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.ikcpolicy.id', 'payload.id']
- event_type: ipsec_site_connection.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.ipsec_site_connection.id', 'payload.id']
- event_type: '*http.*'
  traits: &http_audit
  typeURI:
    fields: payload.typeURI
  eventType:

```

```

      fields: payload.eventType
action:
  fields: payload.action
outcome:
  fields: payload.outcome
id:
  fields: payload.id
eventTime:
  fields: payload.eventTime
requestPath:
  fields: payload.requestPath
observer_id:
  fields: payload.observer.id
target_id:
  fields: payload.target.id
target_typeURI:
  fields: payload.target.typeURI
target_name:
  fields: payload.target.name
initiator_typeURI:
  fields: payload.initiator.typeURI
initiator_id:
  fields: payload.initiator.id
initiator_name:
  fields: payload.initiator.name
initiator_host_address:
  fields: payload.initiator.host.address
- event_type: '*http.response'
traits:
  <<: *http_audit
  reason_code:
    fields: payload.reason.reasonCode

```

12.1.3. pipeline.yaml

Pipelines describe a coupling between sources of samples and the corresponding sinks for transformation and publication of the data. They are defined in the `pipeline.yaml` file.

This file can be modified to adjust polling intervals and the samples generated by the Telemetry module

```

---
sources:
  - name: meter_source
    interval: 600
    meters:
      - "*"
    sinks:
      - meter_sink
  - name: cpu_source
    interval: 600
    meters:
      - "cpu"
    sinks:
      - cpu_sink

```

```

- name: disk_source
  interval: 600
  meters:
    - "disk.read.bytes"
    - "disk.read.requests"
    - "disk.write.bytes"
    - "disk.write.requests"
    - "disk.device.read.bytes"
    - "disk.device.read.requests"
    - "disk.device.write.bytes"
    - "disk.device.write.requests"
  sinks:
    - disk_sink
- name: network_source
  interval: 600
  meters:
    - "network.incoming.bytes"
    - "network.incoming.packets"
    - "network.outgoing.bytes"
    - "network.outgoing.packets"
  sinks:
    - network_sink
sinks:
- name: meter_sink
  transformers:
  publishers:
    - notifier://
- name: cpu_sink
  transformers:
    - name: "rate_of_change"
      parameters:
        target:
          name: "cpu_util"
          unit: "%"
          type: "gauge"
          scale: "100.0 / (10**9 *
(resource_metadata.cpu_number or 1))"
      publishers:
        - notifier://
- name: disk_sink
  transformers:
    - name: "rate_of_change"
      parameters:
        source:
          map_from:
            name: "(disk\\.device|disk)\\. (read|write)\\.
(bytes|requests)"
            unit: "(B|request)"
          target:
            map_to:
              name: "\\1\\.\\2\\.\\3.rate"
              unit: "\\1/s"
              type: "gauge"
      publishers:
        - notifier://
- name: network_sink

```

```

    transformers:
      - name: "rate_of_change"
        parameters:
          source:
            map_from:
              name: "network\\. (incoming|outgoing)\\.
(bytes|packets)"
              unit: "(B|packet)"
            target:
              map_to:
                name: "network\\.\\1\\.\\2.rate"
                unit: "\\1/s"
              type: "gauge"
        publishers:
          - notifier://

```

12.1.4. event_pipeline.yaml

Event pipelines describe a coupling between notification event_types and the corresponding sinks for publication of the event data. They are defined in the `event_pipeline.yaml` file.

This file can be modified to adjust which notifications to capture and the and where to publish the events.

```

---
sources:
  - name: event_source
    events:
      - "*"
    sinks:
      - event_sink
sinks:
  - name: event_sink
    transformers:
    triggers:
    publishers:
      - direct://

```

12.1.5. policy.json

The `policy.json` file defines additional access controls that apply to the Telemetry service.

```

{
  "context_is_admin": "role:admin",
  "context_is_project": "project_id:$(target.project_id)s",
  "context_is_owner": "user_id:$(target.user_id)s",
  "segregation": "rule:context_is_admin",
  "default": ""
}

```

12.2. NEW, UPDATED AND DEPRECATED OPTIONS IN KILO FOR TELEMETRY

Table 12.34. New options

Option = default value	(Type) Help string
[DEFAULT] api_workers = 1	(IntOpt) Number of workers for Ceilometer API server.
[DEFAULT] event_pipeline_cfg_file = event_pipeline.yaml	(StrOpt) Configuration file for event pipeline definition.
[DEFAULT] magnetodb_control_exchange = magnetodb	(StrOpt) Exchange name for Magnetodb notifications.
[DEFAULT] polling_namespaces = ['compute', 'central']	(MultiChoicesOpt) Polling namespace(s) to be used while resource polling
[DEFAULT] pollster_list = []	(MultiChoicesOpt) List of pollsters (or wildcard templates) to be used while polling
[DEFAULT] reserved_metadata_keys =	(ListOpt) List of metadata keys reserved for metering use. And these keys are additional to the ones included in the namespace.
[DEFAULT] shuffle_time_before_polling_task = 0	(IntOpt) To reduce large requests at same time to Nova or other components from different compute agents, shuffle start time of polling task.
[DEFAULT] sql_expire_samples_only = False	(BoolOpt) Indicates if expirer expires only samples. If set true, expired samples will be deleted, but residual resource and meter definition data will remain.
[DEFAULT] swift_control_exchange = swift	(StrOpt) Exchange name for Swift notifications.
[DEFAULT] zaqar_control_exchange = zaqar	(StrOpt) Exchange name for Messaging service notifications.
[alarms] gnocchi_url = http://localhost:8041	(StrOpt) URL to Gnocchi.
[collector] requeue_event_on_dispatcher_error = False	(BoolOpt) Requeue the event on the collector event queue when the collector fails to dispatch it.
[coordination] check_watchers = 10.0	(FloatOpt) Number of seconds between checks to see if group membership has changed
[database] db2nosql_resource_id_maxlen = 512	(IntOpt) The max length of resources id in DB2 nosql, the value should be larger than len(hostname) * 2 as compute node's resource id is <hostname>_<nodename>.

Option = default value	(Type) Help string
[database] event_connection = None	(StrOpt) The connection string used to connect to the event database. (if unset, connection is used)
[database] event_time_to_live = -1	(IntOpt) Number of seconds that events are kept in the database for (<= 0 means forever).
[database] metering_time_to_live = -1	(IntOpt) Number of seconds that samples are kept in the database for (<= 0 means forever).
[database] mongodb_replica_set =	(StrOpt) The name of the replica set which is used to connect to MongoDB database. If it is set, MongoReplicaSetClient will be used instead of MongoClient.
[dispatcher_http] cadf_only = False	(BoolOpt) The flag that indicates if only cadf message should be posted. If false, all meters will be posted.
[dispatcher_http] event_target = None	(StrOpt) The target for event data where the http request will be sent to. If this is not set, it will default to same as Sample target.
[dispatcher_http] target =	(StrOpt) The target where the http request will be sent. If this is not set, no data will be posted. For example: target = http://hostname:1234/path
[dispatcher_http] timeout = 5	(IntOpt) The max time in seconds to wait for a request to timeout.
[event] store_raw = []	(MultiStrOpt) Store the raw notification for select priority levels (info and/or error). By default, raw details are not captured.
[ipmi] polling_retry = 3	(IntOpt) Tolerance of IPMI/NM polling failures before disable this pollster. Negative indicates retrying forever.
[notification] disable_non_metric_meters = False	(BoolOpt) WARNING: Ceilometer historically offered the ability to store events as meters. This usage is NOT advised as it can flood the metering database and cause performance degradation. This option disables the collection of non-metric meters and will be the default behavior in Liberty.
[notification] workload_partitioning = False	(BoolOpt) Enable workload partitioning, allowing multiple notification agents to be run simultaneously.

Option = default value	(Type) Help string
[oslo_concurrency] disable_process_locking = False	(BoolOpt) Enables or disables inter-process locks.
[oslo_concurrency] lock_path = None	(StrOpt) Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable OSLO_LOCK_PATH. If external locks are used, a lock path must be set.
[oslo_messaging_amqp] allow_insecure_clients = False	(BoolOpt) Accept clients using either SSL or plain TCP
[oslo_messaging_amqp] broadcast_prefix = broadcast	(StrOpt) address prefix used when broadcasting to all servers
[oslo_messaging_amqp] container_name = None	(StrOpt) Name for the AMQP container
[oslo_messaging_amqp] group_request_prefix = unicast	(StrOpt) address prefix when sending to any server in group
[oslo_messaging_amqp] idle_timeout = 0	(IntOpt) Timeout for inactive connections (in seconds)
[oslo_messaging_amqp] server_request_prefix = exclusive	(StrOpt) address prefix used when sending to a specific server
[oslo_messaging_amqp] ssl_ca_file =	(StrOpt) CA certificate PEM file for verifying server certificate
[oslo_messaging_amqp] ssl_cert_file =	(StrOpt) Identifying certificate PEM file to present to clients
[oslo_messaging_amqp] ssl_key_file =	(StrOpt) Private key PEM file used to sign cert_file certificate
[oslo_messaging_amqp] ssl_key_password = None	(StrOpt) Password for decrypting ssl_key_file (if encrypted)
[oslo_messaging_amqp] trace = False	(BoolOpt) Debug: dump AMQP frames to stdout
[oslo_messaging_qpid] amqp_auto_delete = False	(BoolOpt) Auto-delete queues in AMQP.
[oslo_messaging_qpid] amqp_durable_queues = False	(BoolOpt) Use durable queues in AMQP.
[oslo_messaging_qpid] qpid_heartbeat = 60	(IntOpt) Seconds between connection keepalive heartbeats.

Option = default value	(Type) Help string
[oslo_messaging_qpid] qpid_hostname = localhost	(StrOpt) Qpid broker hostname.
[oslo_messaging_qpid] qpid_hosts = \$qpid_hostname:\$qpid_port	(ListOpt) Qpid HA cluster host:port pairs.
[oslo_messaging_qpid] qpid_password =	(StrOpt) Password for Qpid connection.
[oslo_messaging_qpid] qpid_port = 5672	(IntOpt) Qpid broker port.
[oslo_messaging_qpid] qpid_protocol = tcp	(StrOpt) Transport to use, either 'tcp' or 'ssl'.
[oslo_messaging_qpid] qpid_receiver_capacity = 1	(IntOpt) The number of prefetched messages held by receiver.
[oslo_messaging_qpid] qpid_sasl_mechanisms =	(StrOpt) Space separated list of SASL mechanisms to use for auth.
[oslo_messaging_qpid] qpid_tcp_nodelay = True	(BoolOpt) Whether to disable the Nagle algorithm.
[oslo_messaging_qpid] qpid_topology_version = 1	(IntOpt) The qpid topology version to use. Version 1 is what was originally used by impl_qpid. Version 2 includes some backwards-incompatible changes that allow broker federation to work. Users should update to version 2 when they are able to take everything down, as it requires a clean break.
[oslo_messaging_qpid] qpid_username =	(StrOpt) Username for Qpid connection.
[oslo_messaging_qpid] rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
[oslo_messaging_rabbit] amqp_auto_delete = False	(BoolOpt) Auto-delete queues in AMQP.
[oslo_messaging_rabbit] amqp_durable_queues = False	(BoolOpt) Use durable queues in AMQP.
[oslo_messaging_rabbit] fake_rabbit = False	(BoolOpt) Deprecated, use rpc_backend=kombu+memory or rpc_backend=fake
[oslo_messaging_rabbit] heartbeat_rate = 2	(IntOpt) How often times during the heartbeat_timeout_threshold to check the heartbeat.
[oslo_messaging_rabbit] heartbeat_timeout_threshold = 0	(IntOpt) Number of seconds after which the Rabbit broker is considered down if heartbeat's keep-alive fails (0 disables the heartbeat, >0 enables it. Enabling heartbeats requires kombu>=3.0.7 and amqp>=1.4.0). EXPERIMENTAL

Option = default value	(Type) Help string
[oslo_messaging_rabbit] kombu_reconnect_delay = 1.0	(FloatOpt) How long to wait before reconnecting in response to an AMQP consumer cancel notification.
[oslo_messaging_rabbit] kombu_ssl_ca_certs =	(StrOpt) SSL certification authority file (valid only if SSL enabled).
[oslo_messaging_rabbit] kombu_ssl_certfile =	(StrOpt) SSL cert file (valid only if SSL enabled).
[oslo_messaging_rabbit] kombu_ssl_keyfile =	(StrOpt) SSL key file (valid only if SSL enabled).
[oslo_messaging_rabbit] kombu_ssl_version =	(StrOpt) SSL version to use (valid only if SSL enabled). Valid values are TLSv1 and SSLv23. SSLv2, SSLv3, TLSv1_1, and TLSv1_2 are also available.
[oslo_messaging_rabbit] rabbit_ha_queues = False	(BoolOpt) Use HA queues in RabbitMQ (x-ha-policy: all). If you change this option, you must wipe the RabbitMQ database.
[oslo_messaging_rabbit] rabbit_host = localhost	(StrOpt) The RabbitMQ broker address where a single node is used.
[oslo_messaging_rabbit] rabbit_hosts = \$rabbit_host:\$rabbit_port	(ListOpt) RabbitMQ HA cluster host:port pairs.
[oslo_messaging_rabbit] rabbit_login_method = AMQPLAIN	(StrOpt) The RabbitMQ login method.
[oslo_messaging_rabbit] rabbit_max_retries = 0	(IntOpt) Maximum number of RabbitMQ connection retries. Default is 0 (infinite retry count).
[oslo_messaging_rabbit] rabbit_password = guest	(StrOpt) The RabbitMQ password.
[oslo_messaging_rabbit] rabbit_port = 5672	(IntOpt) The RabbitMQ broker port where a single node is used.
[oslo_messaging_rabbit] rabbit_retry_backoff = 2	(IntOpt) How long to backoff for between retries when connecting to RabbitMQ.
[oslo_messaging_rabbit] rabbit_retry_interval = 1	(IntOpt) How frequently to retry connecting with RabbitMQ.
[oslo_messaging_rabbit] rabbit_use_ssl = False	(BoolOpt) Connect over SSL for RabbitMQ.
[oslo_messaging_rabbit] rabbit_userid = guest	(StrOpt) The RabbitMQ userid.
[oslo_messaging_rabbit] rabbit_virtual_host = /	(StrOpt) The RabbitMQ virtual host.

Option = default value	(Type) Help string
[oslo_messaging_rabbit] rpc_conn_pool_size = 30	(IntOpt) Size of RPC connection pool.
[oslo_middleware] max_request_body_size = 114688	(IntOpt) The maximum body size for each request, in bytes.
[oslo_policy] policy_default_rule = default	(StrOpt) Default rule. Enforced when a requested rule is not found.
[oslo_policy] policy_dirs = ['policy.d']	(MultiStrOpt) Directories where policy configuration files are stored. They can be relative to any directory in the search path defined by the config_dir option, or absolute paths. The file defined by policy_file must exist for these directories to be searched. Missing or empty directories are ignored.
[oslo_policy] policy_file = policy.json	(StrOpt) The JSON file that defines policies.
[polling] partitioning_group_prefix = None	(StrOpt) Work-load partitioning group prefix. Use only if you want to run multiple polling agents with different config files. For each sub-group of the agent pool with the same partitioning_group_prefix a disjoint subset of pollsters should be loaded.
[publisher] telemetry_secret = change this for valid signing	(StrOpt) Secret value for signing messages. Set value empty if signing is not required to avoid computational overhead.
[publisher_notifier] event_topic = event	(StrOpt) The topic that ceilometer uses for event notifications.
[publisher_notifier] telemetry_driver = messagingv2	(StrOpt) The driver that ceilometer uses for metering notifications.
[rgw_admin_credentials] access_key = None	(StrOpt) Access key for Radosgw Admin.
[rgw_admin_credentials] secret_key = None	(StrOpt) Secret key for Radosgw Admin.
[service_types] radosgw = object-store	(StrOpt) Radosgw service type.
[vmware] host_port = 443	(IntOpt) Port of the VMware Vsphere host.

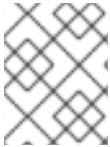
Table 12.35. New default values

Option	Previous default value	New default value
[DEFAULT] rpc_zmq_matchmaker	oslo.messaging._drivers.matchmaker.MatchMakerLocalhost	local

Table 12.36. Deprecated options

Deprecated option	New Option
[alarm] evaluation_service	None
[DEFAULT] log_format	None
[database] time_to_live	[database] metering_time_to_live
[DEFAULT] database_connection	None
[publisher] metering_secret	[publisher] telemetry_secret
[alarm] partition_rpc_topic	None
[DEFAULT] use_syslog	None

APPENDIX A. THE POLICY.JSON FILE



NOTE

The following functionality is limited to Nova API v2.0, and currently does *not* apply to Nova API v2.1.

Each OpenStack service, Identity, Compute, Networking and so on, has its own role-based access policies. They determine which user can access which objects in which way, and are defined in the service's `policy.json` file.

Whenever an API call to an OpenStack service is made, the service's policy engine uses the appropriate policy definitions to determine if the call can be accepted. Any changes to `policy.json` are effective immediately, which allows new policies to be implemented while the service is running.

A `policy.json` file is a text file in JSON (Javascript Object Notation) format. Each policy is defined by a one-line statement in the form "`<target>`" : "`<rule>`".

The policy target, also named "action", represents an API call like "start an instance" or "attach a volume".

Action names are usually qualified. Example: OpenStack Compute features API calls to list instances, volumes and networks. In `/etc/nova/policy.json`, these APIs are represented by `compute:get_all`, `volume:get_all` and `network:get_all`, respectively.

The mapping between API calls and actions is not generally documented.

The policy rule determines under which circumstances the API call is permitted. Usually this involves the user who makes the call (hereafter named the "API user") and often the object on which the API call operates. A typical rule checks if the API user is the object's owner.

MODIFYING THE POLICY

While recipes for editing `policy.json` files are found on blogs, modifying the policy can have unexpected side effects and is not encouraged.

A.1. EXAMPLES

A simple rule might look like this:

```
"compute:get_all" : ""
```

The target is "`compute:get_all`", the "list all instances" API of the Compute service. The rule is an empty string meaning "always". This policy allows anybody to list instances.

You can also decline permission to use an API:

```
"compute:shelve": "!"
```

The exclamation mark stands for "never" or "nobody", which effectively disables the Compute API "shelve an instance".

Many APIs can only be called by admin users. This can be expressed by the rule `"role:admin"`. The following policy ensures that only administrators can create new users in the Identity database:

```
"identity:create_user" : "role:admin"
```

You can limit APIs to any role. For example, the Orchestration service defines a role named `heat_stack_user`. Whoever has this role isn't allowed to create stacks:

```
"stacks:create": "not role:heat_stack_user"
```

This rule makes use of the boolean operator `not`. More complex rules can be built using operators `and`, `or` and parentheses.

You can define aliases for rules:

```
"deny_stack_user": "not role:heat_stack_user"
```

The policy engine understands that `"deny_stack_user"` is not an API and consequently interprets it as an alias. The stack creation policy above can then be written as:

```
"stacks:create": "rule:deny_stack_user"
```

This is taken verbatim from `/etc/heat/policy.json`.

Rules can compare API attributes to object attributes. For example:

```
"compute:start" : "user_id:%(user_id)s"
```

states that only the owner of an instance can start it up. The `user_id` string before the colon is an API attribute, namely the user ID of the API user. It is compared with the user ID of the object (in this case, an instance); more precisely, it is compared with the `user_id` field of that object in the database. If the two values are equal, permission is granted.

An admin user always has permission to call APIs. This is how `/etc/keystone/policy.json` makes this policy explicit:

```
"admin_required": "role:admin or is_admin:1",
"owner" : "user_id:%(user_id)s",
"admin_or_owner": "rule:admin_required or rule:owner",
"identity:change_password": "rule:admin_or_owner"
```

The first line defines an alias for "user is an admin user". The `is_admin` flag is only used when setting up the Identity service for the first time. It indicates that the user has admin privileges granted by the service token (`--os-token` parameter of the `keystone` command line client).

The second line creates an alias for "user owns the object" by comparing the API's user ID with the object's user ID.

Line 3 defines a third alias `admin_or_owner`, combining the two first aliases with the Boolean operator `or`.

Line 4 sets up the policy that a password can only be modified by its owner or an admin user.

As a final example, let's examine a more complex rule:


```
"identity:ec2_delete_credential": "rule:admin_required or
                                   (rule:owner and user_id:%(target.credential.user_id)s)"
```

This rule determines who can use the Identity API "delete EC2 credential". Here, boolean operators and parentheses combine three simpler rules. `admin_required` and `owner` are the same aliases as in the previous example. `user_id:%(target.credential.user_id)s` compares the API user with the user ID of the credential object associated with the target.

A.2. SYNTAX

A `policy.json` file consists of policies and aliases of the form `target:rule` or `alias:definition`, separated by commas and enclosed in curly braces:

```
{
    "alias 1" : "definition 1",
    "alias 2" : "definition 2",
    ...
    "target 1" : "rule 1",
    "target 2" : "rule 2",
    ....
}
```

Targets are APIs and are written `"service:API"` or simply `"API"`. For example, `"compute:create"` or `"add_image"`.

Rules determine whether the API call is allowed.

Rules can be:

- always true. The action is always permitted. This can be written as `"` (empty string), `[]`, or `"@"`.
- always false. The action is never permitted. Written as `"!"`.
- a special check
- a comparison of two values
- boolean expressions based on simpler rules

Special checks are

- `<role>:<role name>`, a test whether the API credentials contain this role.
- `<rule>:<rule name>`, the definition of an alias.
- `http:<target URL>`, which delegates the check to a remote server. The API is authorized when the server returns True.

Developers can define additional special checks.

Two values are compared in the following way:

```
"value1 : value2"
```

Possible values are

- constants: Strings, numbers, `true`, `false`
- API attributes
- target object attributes
- the flag `is_admin`

API attributes can be `project_id`, `user_id` or `domain_id`.

Target object attributes are fields from the object description in the database. For example in the case of the `"compute:start"` API, the object is the instance to be started. The policy for starting instances could use the `%(project_id)s` attribute, that is the project that owns the instance. The trailing `s` indicates this is a string.

`is_admin` indicates that administrative privileges are granted via the admin token mechanism (the `--os-token` option of the `keystone` command). The admin token allows initialisation of the identity database before the admin role exists.

The alias construct exists for convenience. An alias is short name for a complex or hard to understand rule. It is defined in the same way as a policy:

```
alias name : alias definition
```

Once an alias is defined, use the `rule` keyword to use it in a policy rule.

A.3. OLDER SYNTAX

You may encounter older `policy.json` files that feature a different syntax, where JavaScript arrays are used instead of boolean operators. For example, the EC2 credentials rule above would have been written as follows:

```
"identity:ec2_delete_credential": [ [ "rule:admin_required ],  
                                   [ "rule:owner", "user_id:%(target.credential.user_id)s" ] ]
```

The rule is an array of arrays. The innermost arrays are or'ed together, whereas elements inside the innermost arrays are and'ed.

While the old syntax is still supported, we recommend using the newer, more intuitive syntax.

APPENDIX B. FIREWALLS AND DEFAULT PORTS

On some deployments, such as ones where restrictive firewalls are in place, you might need to manually configure a firewall to permit OpenStack service traffic.

To manually configure a firewall, you must permit traffic through the ports that each OpenStack service uses. This table lists the default ports that each OpenStack service uses:

Table B.1. Default ports that OpenStack components use

OpenStack service	Default ports	Port type
Block Storage (cinder)	8776	publicurl and adminurl
Compute (nova) endpoints	8774	publicurl and adminurl
Compute API (nova-api)	8773, 8775	
Compute ports for access to virtual machine consoles	5900-5999	
Compute VNC proxy for browsers (openstack-nova-novncproxy)	6080	
Compute VNC proxy for traditional VNC clients (openstack-nova-xvncproxy)	6081	
Proxy port for HTML5 console used by Compute service	6082	
Data processing service (sahara) endpoint	8386	publicurl and adminurl
Identity service (keystone) administrative endpoint	35357	adminurl
Identity service public endpoint	5000	publicurl
Image service (glance) API	9292	publicurl and adminurl
Image service registry	9191	
Networking (neutron)	9696	publicurl and adminurl
Object Storage (swift)	6000, 6001, 6002	
Orchestration (heat) endpoint	8004	publicurl and adminurl
Orchestration AWS CloudFormation-compatible API (openstack-heat-api-cfn)	8000	

OpenStack service	Default ports	Port type
Orchestration AWS CloudWatch-compatible API (openstack-heat-api-cloudwatch)	8003	
Telemetry (ceilometer)	8777	publicurl and adminurl

To function properly, some OpenStack components depend on other, non-OpenStack services. For example, the OpenStack dashboard uses HTTP for non-secure communication. In this case, you must configure the firewall to allow traffic to and from HTTP.

This table lists the ports that other OpenStack components use:

Table B.2. Default ports that secondary services related to OpenStack components use

Service	Default port	Used by
HTTP	80	OpenStack dashboard (Horizon) when it is not configured to use secure access.
HTTP alternate	8080	OpenStack Object Storage (swift) service.
HTTPS	443	Any OpenStack service that is enabled for SSL, especially secure-access dashboard.
rsync	873	OpenStack Object Storage. Required.
iSCSI target	3260	OpenStack Block Storage. Required.
MySQL database service	3306	Most OpenStack components.
Message Broker (AMQP traffic)	5672	OpenStack Block Storage, Networking, Orchestration, and Compute.

On some deployments, the default port used by a service may fall within the defined local port range of a host. To check a host's local port range:

```
$ sysctl -a | grep ip_local_port_range
```

If a service's default port falls within this range, run the following program to check if the port has already been assigned to another application:

```
$ lsof -i :PORT
```

Configure the service to use a different port if the default port is already being used by another application.

