



Red Hat Enterprise Linux OpenStack Platform 5 Command-Line Interface Reference Guide

Command-line clients for the Red Hat Enterprise Linux OpenStack
Platform

07 Jul 2014

Red Hat Documentation Team

Red Hat Enterprise Linux OpenStack Platform 5 Command-Line Interface Reference Guide

Command-line clients for the Red Hat Enterprise Linux OpenStack Platform

07 Jul 2014

Red Hat Documentation Team

Legal Notice

Copyright © 2014 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, MetaMatrix, 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 guide documents the OpenStack command-line clients.

Table of Contents

Chapter 1. OpenStack command-line clients	14
1. Overview	14
2. Install the OpenStack command-line clients	15
3. Discover the version number for a client	17
4. Set environment variables using the OpenStack RC file	17
Chapter 2. Block Storage command-line client	20
1. cinder usage	20
2. cinder optional arguments	24
3. cinder absolute-limits command	25
4. cinder availability-zone-list command	25
5. cinder backup-create command	25
6. cinder backup-delete command	26
7. cinder backup-list command	26
8. cinder backup-restore command	26
9. cinder backup-show command	27
10. cinder create command	27
11. cinder credentials command	28
12. cinder delete command	28
13. cinder encryption-type-create command	28
14. cinder encryption-type-delete command	29
15. cinder encryption-type-list command	29
16. cinder encryption-type-show command	29
17. cinder endpoints command	29
18. cinder extend command	29
19. cinder extra-specs-list command	30
20. cinder force-delete command	30
21. cinder list command	30
22. cinder list-extensions command	31
23. cinder metadata command	31
24. cinder metadata-show command	31
25. cinder metadata-update-all command	31
26. cinder migrate command	32
27. cinder qos-associate command	32
28. cinder qos-create command	32
29. cinder qos-delete command	33
30. cinder qos-disassociate command	33
31. cinder qos-disassociate-all command	33
32. cinder qos-get-association command	34
33. cinder qos-key command	34
34. cinder qos-list command	34
35. cinder qos-show command	34
36. cinder quota-class-show command	35
37. cinder quota-class-update command	35
38. cinder quota-defaults command	35
39. cinder quota-show command	36
40. cinder quota-update command	36
41. cinder quota-usage command	36
42. cinder rate-limits command	37
43. cinder readonly-mode-update command	37
44. cinder rename command	37
45. cinder reset-state command	38

45. cinder reset-state command	38
46. cinder service-disable command	38
47. cinder service-enable command	38
48. cinder service-list command	38
49. cinder show command	39
50. cinder snapshot-create command	39
51. cinder snapshot-delete command	40
52. cinder snapshot-list command	40
53. cinder snapshot-metadata command	40
54. cinder snapshot-metadata-show command	41
55. cinder snapshot-metadata-update-all command	41
56. cinder snapshot-rename command	41
57. cinder snapshot-reset-state command	42
58. cinder snapshot-show command	42
59. cinder transfer-accept command	42
60. cinder transfer-create command	43
61. cinder transfer-delete command	43
62. cinder transfer-list command	43
63. cinder transfer-show command	43
64. cinder type-create command	43
65. cinder type-delete command	44
66. cinder type-key command	44
67. cinder type-list command	44
68. cinder upload-to-image command	44
Chapter 3. Compute command-line client	46
1. nova usage	46
2. nova optional arguments	58
3. nova absolute-limits command	59
4. nova add-fixed-ip command	59
5. nova add-secgroup command	60
6. nova agent-create command	60
7. nova agent-delete command	61
8. nova agent-list command	61
9. nova agent-modify command	61
10. nova aggregate-add-host command	61
11. nova aggregate-create command	62
12. nova aggregate-delete command	62
13. nova aggregate-details command	62
14. nova aggregate-list command	63
15. nova aggregate-remove-host command	63
16. nova aggregate-set-metadata command	63
17. nova aggregate-update command	63
18. nova availability-zone-list command	64
19. nova backup command	64
20. nova baremetal-interface-add command	64
21. nova baremetal-interface-list command	65
22. nova baremetal-interface-remove command	65
23. nova baremetal-node-create command	65
24. nova baremetal-node-delete command	66
25. nova baremetal-node-list command	66
26. nova baremetal-node-show command	66
27. nova boot command	67
28. nova cell-capacities command	69

29. nova cell-show command	69
30. nova clear-password command	69
31. nova cloudpipe-configure command	69
32. nova cloudpipe-create command	70
33. nova cloudpipe-list command	70
34. nova console-log command	70
35. nova credentials command	70
36. nova delete command	71
37. nova diagnostics command	71
38. nova dns-create command	71
39. nova dns-create-private-domain command	71
40. nova dns-create-public-domain command	72
41. nova dns-delete command	72
42. nova dns-delete-domain command	72
43. nova dns-domains command	73
44. nova dns-list command	73
45. nova endpoints command	73
46. nova evacuate command	73
47. nova fixed-ip-get command	74
48. nova fixed-ip-reserve command	74
49. nova fixed-ip-unreserve command	74
50. nova flavor-access-add command	75
51. nova flavor-access-list command	75
52. nova flavor-access-remove command	75
53. nova flavor-create command	76
54. nova flavor-delete command	76
55. nova flavor-key command	77
56. nova flavor-list command	77
57. nova flavor-show command	77
58. nova floating-ip-associate command	78
59. nova floating-ip-bulk-create command	78
60. nova floating-ip-bulk-delete command	78
61. nova floating-ip-bulk-list command	79
62. nova floating-ip-create command	79
63. nova floating-ip-delete command	79
64. nova floating-ip-disassociate command	79
65. nova floating-ip-list command	80
66. nova floating-ip-pool-list command	80
67. nova force-delete command	80
68. nova get-password command	80
69. nova get-rdp-console command	80
70. nova get-spice-console command	81
71. nova get-vnc-console command	81
72. nova host-action command	81
73. nova host-describe command	82
74. nova host-evacuate command	82
75. nova host-list command	82
76. nova host-meta command	83
77. nova host-servers-migrate command	83
78. nova host-update command	83
79. nova hypervisor-list command	84
80. nova hypervisor-servers command	84
81. nova hypervisor-show command	84

82. nova hypervisor-stats command	84
83. nova hypervisor-uptime command	84
84. nova image-create command	85
85. nova image-delete command	85
86. nova image-list command	85
87. nova image-meta command	86
88. nova image-show command	86
89. nova instance-action command	86
90. nova instance-action-list command	87
91. nova interface-attach command	87
92. nova interface-detach command	87
93. nova interface-list command	88
94. nova keypair-add command	88
95. nova keypair-delete command	88
96. nova keypair-list command	88
97. nova keypair-show command	89
98. nova list command	89
99. nova list-extensions command	90
100. nova list-secgroup command	90
101. nova live-migration command	90
102. nova lock command	91
103. nova meta command	91
104. nova migrate command	91
105. nova migration-list command	92
106. nova net command	92
107. nova net-create command	92
108. nova net-delete command	93
109. nova net-list command	93
110. nova network-associate-host command	93
111. nova network-associate-project command	93
112. nova network-create command	94
113. nova network-disassociate command	95
114. nova network-list command	95
115. nova network-show command	95
116. nova pause command	96
117. nova quota-class-show command	96
118. nova quota-class-update command	96
119. nova quota-defaults command	97
120. nova quota-delete command	97
121. nova quota-show command	98
122. nova quota-update command	98
123. nova rate-limits command	99
124. nova reboot command	99
125. nova rebuild command	100
126. nova refresh-network command	100
127. nova remove-fixed-ip command	101
128. nova remove-secgroup command	101
129. nova rename command	101
130. nova rescue command	102
131. nova reset-network command	102
132. nova reset-state command	102
133. nova resize command	102
134. nova resize-confirm command	103

135. nova resize-revert command	103
136. nova restore command	103
137. nova resume command	104
138. nova root-password command	104
139. nova scrub command	104
140. nova secgroup-add-group-rule command	104
141. nova secgroup-add-rule command	105
142. nova secgroup-create command	105
143. nova secgroup-delete command	106
144. nova secgroup-delete-group-rule command	106
145. nova secgroup-delete-rule command	106
146. nova secgroup-list command	107
147. nova secgroup-list-rules command	107
148. nova secgroup-update command	107
149. nova server-group-create	108
150. nova server-group-delete	108
151. nova server-group-get	108
152. nova server-group-list	109
153. nova service-disable command	109
154. nova service-enable command	109
155. nova service-list command	109
156. nova shelve command	110
157. nova shelve-offload command	110
158. nova show command	110
159. nova ssh command	110
160. nova start command	111
161. nova stop command	111
162. nova suspend command	112
163. nova unlock command	112
164. nova unpause command	112
165. nova unrescue command	112
166. nova unshelve command	113
167. nova usage command	113
168. nova usage-list command	113
169. nova volume-attach command	114
170. nova volume-create command	114
171. nova volume-delete command	115
172. nova volume-detach command	115
173. nova volume-list command	115
174. nova volume-show command	115
175. nova volume-snapshot-create command	116
176. nova volume-snapshot-delete command	116
177. nova volume-snapshot-list command	116
178. nova volume-snapshot-show command	116
179. nova volume-type-create command	117
180. nova volume-type-delete command	117
181. nova volume-type-list command	117
182. nova volume-update command	117
183. nova x509-create-cert command	118
184. nova x509-get-root-cert command	118
Chapter 4. Identity service command-line client	119
1. keystone usage	119

2. keystone optional arguments	121
3. keystone bootstrap command	123
4. keystone catalog command	123
5. keystone discover command	123
6. keystone ec2-credentials-create command	124
7. keystone ec2-credentials-delete command	124
8. keystone ec2-credentials-get command	124
9. keystone ec2-credentials-list command	124
10. keystone endpoint-create command	125
11. keystone endpoint-delete command	125
12. keystone endpoint-get command	125
13. keystone endpoint-list command	126
14. keystone password-update command	126
15. keystone role-create command	126
16. keystone role-delete command	127
17. keystone role-get command	127
18. keystone role-list command	127
19. keystone service-create command	127
20. keystone service-delete command	128
21. keystone service-get command	128
22. keystone service-list command	128
23. keystone tenant-create command	128
24. keystone tenant-delete command	129
25. keystone tenant-get command	129
26. keystone tenant-list command	129
27. keystone tenant-update command	129
28. keystone token-get command	130
29. keystone user-create command	130
30. keystone user-delete command	130
31. keystone user-get command	131
32. keystone user-list command	131
33. keystone user-password-update command	131
34. keystone user-role-add command	131
35. keystone user-role-list command	132
36. keystone user-role-remove command	132
37. keystone user-update command	133
Chapter 5. Image Service command-line client	134
1. glance usage	134
2. glance optional arguments	135
3. glance image-create command	138
4. glance image-delete command	139
5. glance image-list command	139
6. glance image-show command	141
7. glance image-update command	141
8. glance member-create command	143
9. glance member-delete command	143
10. glance member-list command	143
Chapter 6. Image Service property keys	145
Chapter 7. Networking command-line client	153
1. neutron usage	153
2. neutron optional arguments	153

3. neutron API v2.0 commands	154
4. neutron agent-delete command	164
5. neutron agent-list command	164
6. neutron agent-show command	165
7. neutron agent-update command	165
8. neutron cisco-credential-create command	166
9. neutron cisco-credential-delete command	166
10. neutron cisco-credential-list command	167
11. neutron cisco-credential-show command	167
12. neutron cisco-network-profile-create command	168
13. neutron cisco-network-profile-delete command	169
14. neutron cisco-network-profile-list command	169
15. neutron cisco-network-profile-show command	170
16. neutron cisco-network-profile-update command	170
17. neutron cisco-policy-profile-list command	171
18. neutron cisco-policy-profile-show command	171
19. neutron cisco-policy-profile-update command	172
20. neutron dhcp-agent-list-hosting-net command	172
21. neutron dhcp-agent-network-add command	173
22. neutron dhcp-agent-network-remove command	173
23. neutron ext-list command	174
24. neutron ext-show command	174
25. neutron firewall-create command	175
26. neutron firewall-delete command	175
27. neutron firewall-list command	176
28. neutron firewall-policy-create command	176
29. neutron firewall-policy-delete command	177
30. neutron firewall-policy-insert-rule command	178
31. neutron firewall-policy-list command	178
32. neutron firewall-policy-remove-rule command	179
33. neutron firewall-policy-show command	179
34. neutron firewall-policy-update command	180
35. neutron firewall-rule-create command	180
36. neutron firewall-rule-delete command	182
37. neutron firewall-rule-list command	182
38. neutron firewall-rule-show command	183
39. neutron firewall-rule-update command	183
40. neutron firewall-show command	184
41. neutron firewall-update command	184
42. neutron floatingip-associate command	185
43. neutron floatingip-create command	185
44. neutron floatingip-delete command	186
45. neutron floatingip-disassociate command	186
46. neutron floatingip-list command	187
47. neutron floatingip-show command	187
48. neutron ipsec-site-connection-create command	188
49. neutron ipsec-site-connection-delete command	189
50. neutron ipsec-site-connection-list command	190
51. neutron ipsec-site-connection-show command	191
52. neutron ipsec-site-connection-update command	191
53. neutron l3-agent-list-hosting-router command	192
54. neutron l3-agent-router-add command	192
55. neutron l3-agent-router-remove command	193

56. neutron lb-agent-hosting-pool command	193
57. neutron lb-healthmonitor-associate command	194
58. neutron lb-healthmonitor-create command	194
59. neutron lb-healthmonitor-delete command	195
60. neutron lb-healthmonitor-disassociate command	196
61. neutron lb-healthmonitor-list command	196
62. neutron lb-healthmonitor-show command	197
63. neutron lb-healthmonitor-update command	197
64. neutron lb-member-create command	198
65. neutron lb-member-delete command	199
66. neutron lb-member-list command	199
67. neutron lb-member-show command	200
68. neutron lb-member-update command	200
69. neutron lb-pool-create command	201
70. neutron lb-pool-delete command	202
71. neutron lb-pool-list command	202
72. neutron lb-pool-list-on-agent command	203
73. neutron lb-pool-show command	203
74. neutron lb-pool-stats command	204
75. neutron lb-pool-update command	204
76. neutron lb-vip-create command	205
77. neutron lb-vip-delete command	206
78. neutron lb-vip-list command	206
79. neutron lb-vip-show command	207
80. neutron lb-vip-update command	208
81. neutron meter-label-create command	208
82. neutron meter-label-delete command	209
83. neutron meter-label-list command	209
84. neutron meter-label-rule-create command	210
85. neutron meter-label-rule-delete command	210
86. neutron meter-label-rule-list command	211
87. neutron meter-label-rule-show command	212
88. neutron meter-label-show command	212
89. neutron net-create command	213
90. neutron net-delete command	213
91. neutron net-external-list command	214
92. neutron net-gateway-connect command	214
93. neutron net-gateway-create command	215
94. neutron net-gateway-delete command	216
95. neutron net-gateway-disconnect command	216
96. neutron net-gateway-list command	217
97. neutron net-gateway-show command	217
98. neutron net-gateway-update command	218
99. neutron net-list command	218
100. neutron net-list-on-dhcp-agent command	219
101. neutron net-show command	220
102. neutron net-update command	220
103. neutron port-create command	221
104. neutron port-delete command	222
105. neutron port-list command	222
106. neutron port-show command	223
107. neutron port-update command	223
108. neutron queue-create command	224

109. neutron queue-delete command	225
110. neutron queue-list command	225
111. neutron queue-show command	226
112. neutron quota-delete command	226
113. neutron quota-list command	227
114. neutron quota-show command	227
115. neutron quota-update command	228
116. neutron router-create command	228
117. neutron router-delete command	229
118. neutron router-gateway-clear command	229
119. neutron router-gateway-set command	230
120. neutron router-interface-add command	230
121. neutron router-interface-delete command	231
122. neutron router-list command	232
123. neutron router-list-on-l3-agent command	232
124. neutron router-port-list command	233
125. neutron router-show command	234
126. neutron router-update command	234
127. neutron security-group-create command	235
128. neutron security-group-delete command	235
129. neutron security-group-list command	236
130. neutron security-group-rule-create command	236
131. neutron security-group-rule-delete command	237
132. neutron security-group-rule-list command	238
133. neutron security-group-rule-show command	238
134. neutron security-group-show command	239
135. neutron security-group-update command	240
136. neutron service-provider-list command	240
137. neutron subnet-create command	241
138. neutron subnet-delete command	242
139. neutron subnet-list command	242
140. neutron subnet-show command	243
141. neutron subnet-update command	244
142. neutron vpn-ikepolicy-create command	244
143. neutron vpn-ikepolicy-delete command	245
144. neutron vpn-ikepolicy-list command	246
145. neutron vpn-ikepolicy-show command	246
146. neutron vpn-ikepolicy-update command	247
147. neutron vpn-ipsecpolicy-create command	247
148. neutron vpn-ipsecpolicy-delete command	248
149. neutron vpn-ipsecpolicy-list command	249
150. neutron vpn-ipsecpolicy-show command	250
151. neutron vpn-ipsecpolicy-update command	250
152. neutron vpn-service-create command	251
153. neutron vpn-service-delete command	251
154. neutron vpn-service-list command	252
155. neutron vpn-service-show command	252
156. neutron vpn-service-update command	253
Chapter 8. neutron-debug command-line client	254
1. neutron-debug usage	254
2. neutron-debug optional arguments	254
3. neutron-debug probe-create command	256

4. neutron-debug probe-list command	256
5. neutron-debug probe-clear command	256
6. neutron-debug probe-delete command	256
7. neutron-debug probe-exec command	256
8. neutron-debug ping-all command	256
9. neutron-debug example	257
Chapter 9. Object Storage command-line client	258
1. swift usage	258
2. swift examples	258
3. swift optional arguments	259
4. swift delete command	260
5. swift download command	261
6. swift list command	262
7. swift post command	263
8. swift stat command	263
9. swift upload command	264
Chapter 10. Orchestration command-line client	266
1. heat usage	266
2. heat optional arguments	268
3. heat action-resume command	270
4. heat action-suspend command	270
5. heat build-info command	270
6. heat event-list command	270
7. heat event-show command	271
8. heat output-list command	271
9. heat output-show command	271
10. heat resource-list command	271
11. heat resource-metadata command	272
12. heat resource-show command	272
13. heat resource-signal command	272
14. heat resource-template command	273
15. heat resource-type-list command	273
16. heat resource-type-show command	273
17. heat stack-abandon command	274
18. heat stack-adopt command	274
19. heat stack-create command	275
20. heat stack-delete command	276
21. heat stack-list command	276
22. heat stack-preview command	276
23. heat stack-show command	277
24. heat stack-update command	277
25. heat template-show command	278
26. heat template-validate command	278
Chapter 11. Telemetry command-line client	280
1. ceilometer usage	280
2. ceilometer optional arguments	282
3. ceilometer alarm-combination-create command	283
4. ceilometer alarm-combination-update command	284
5. ceilometer alarm-delete command	286
6. ceilometer alarm-history command	286
7. ceilometer alarm-list command	286

8. ceilometer alarm-show command	287
9. ceilometer alarm-state-get command	287
10. ceilometer alarm-state-set command	287
11. ceilometer alarm-threshold-create command	287
12. ceilometer alarm-threshold-update command	289
13. ceilometer alarm-update command	291
14. ceilometer event-list command	292
15. ceilometer event-show command	293
16. ceilometer event-type-list command	293
17. ceilometer meter-list command	293
18. ceilometer query-alarm-history command	293
19. ceilometer query-alarms command	294
20. ceilometer query-samples command	294
21. ceilometer resource-list command	294
22. ceilometer resource-show command	295
23. ceilometer sample-create command	295
24. ceilometer sample-list command	296
25. ceilometer statistics command	296
26. ceilometer trait-description-list command	297
27. ceilometer trait-list command	297
Chapter 12. Database Service command-line client	298
1. trove usage	298
2. trove optional arguments	301
3. trove backup-create command	302
4. trove backup-delete command	303
5. trove backup-list command	303
6. trove backup-list-instance command	303
7. trove backup-show command	304
8. trove configuration-attach command	304
9. trove configuration-create command	304
10. trove configuration-default command	305
11. trove configuration-delete command	305
12. trove configuration-detach command	305
13. trove configuration-instances command	305
14. trove configuration-list command	306
15. trove configuration-parameter-list command	306
16. trove configuration-parameter-show command	306
17. trove configuration-patch command	307
18. trove configuration-show command	307
19. trove configuration-update command	307
20. trove create command	308
21. trove database-create command	309
22. trove database-delete command	309
23. trove database-list command	309
24. trove datastore-list command	310
25. trove datastore-show command	310
26. trove datastore-version-list command	310
27. trove datastore-version-show command	310
28. trove delete command	311
29. trove flavor-list command	311
30. trove flavor-show command	311
31. trove limit-list command	311

32. trove list command	311
33. trove resize-flavor command	312
34. trove resize-volume command	312
35. trove restart command	312
36. trove root-enable command	312
37. trove root-show command	313
38. trove secgroup-add-rule command	313
39. trove secgroup-delete-rule command	313
40. trove secgroup-list command	313
41. trove secgroup-show command	314
42. trove show command	314
43. trove user-create command	314
44. trove user-delete command	315
45. trove user-grant-access command	315
46. trove user-list command	315
47. trove user-revoke-access command	316
48. trove user-show command	316
49. trove user-show-access command	317
50. trove user-update-attributes command	317
Revision History	319

Chapter 1. OpenStack command-line clients

1. Overview

You can use the OpenStack command-line clients to run simple commands that make API calls. You can run these commands from the command line or in scripts to automate tasks. If you provide OpenStack credentials, you can run these commands on any computer.

Internally, each client command runs cURL commands that embed API requests. The OpenStack APIs are RESTful APIs that use the HTTP protocol, including methods, URIs, media types, and response codes.

These open-source Python clients run on Linux or Mac OS X systems and are easy to learn and use. Each OpenStack service has its own command-line client. On some client commands, you can specify a **debug** parameter to show the underlying API request for the command. This is a good way to become familiar with the OpenStack API calls.

The following table lists the command-line client for each OpenStack service with its package name and description.

Table 1.1. OpenStack services and clients

Service	Client	Package	Description
Block Storage	cinder	python-cinderclient	Create and manage volumes.
Compute	nova	python-novaclient	Create and manage images, instances, and flavors.
Database Service	trove	python-troveclient	Create and manage databases.
Identity	keystone	python-keystoneclient	Create and manage users, tenants, roles, endpoints, and credentials.
Image Service	glance	python-glanceclient	Create and manage images.
Networking	neutron	python-neutronclient	Configure networks for guest servers. This client was previously called quantum .
Object Storage	swift	python-swiftclient	Gather statistics, list items, update metadata, and upload, download, and delete files stored by the Object Storage service. Gain access to an Object Storage installation for ad hoc processing.
Orchestration	heat	python-heatclient	Launch stacks from templates, view details of running stacks including events and resources, and update and delete stacks.
Telemetry	ceilometer	python-ceilometerclient	Create and collect measurements across OpenStack.

An OpenStack **common** client is in development.

For client installation instructions, see [Section 2, “Install the OpenStack command-line clients”](#). For information about the OpenStack RC file, see the *Red Hat Enterprise Linux OpenStack Platform End User Guide*.

2. Install the OpenStack command-line clients

Install the prerequisite software and the Python package for each OpenStack client.

2.1. Install the prerequisite software

The following table lists the software that you need to have to run the command-line clients, and provides installation instructions as needed.

Table 1.2. Prerequisite software

Prerequisite	Description
Python 2.6 or later	Currently, the clients do not support Python 3.
setuptools package	Many Linux distributions provide packages to make setuptools easy to install. Search your package manager for setuptools to find an installation package. If you cannot find one, download the setuptools package directly from http://pypi.python.org/pypi/setuptools .
pip package	<p>To install the clients on Red Hat Enterprise Linux, use pip. It is easy to use, ensures that you get the latest version of the clients from the Python Package Index, and lets you update or remove the packages later on.</p> <p>Install pip through the package manager:</p> <p>Red Hat Enterprise Linux.</p> <p>A packaged version enables you to use yum to install the clients, or you can install pip and use it to manage client installation:</p> <pre># yum install python-pip</pre>

2.2. Install the clients

When following the instructions in this section, replace *PROJECT* with the lowercase name of the client to install, such as **nova**. Repeat for each client. The following values are valid:

❏ **ceilometer** - Telemetry API

- ✧ **cinder** - Block Storage API and extensions
- ✧ **glance** - Image Service API
- ✧ **heat** - Orchestration API
- ✧ **keystone** - Identity service API and extensions
- ✧ **neutron** - Networking API
- ✧ **nova** - Compute API and extensions
- ✧ **swift** - Object Storage API
- ✧ **trove** - Database Service API

The following example shows the command for installing the nova client with *pip*.

```
# pip install python-novaclient
```

2.2.1. Installing with *pip*

Use *pip* to install the OpenStack clients on a Red Hat Enterprise Linux system. It is easy to use and ensures that you get the latest version of the client from the [Python Package Index](#). Also, *pip* enables you to update or remove a package.

Install each client separately by using the following command:

- ✧ Red Hat Enterprise Linux:

```
# pip install python-PROJECTclient
```

2.2.2. Installing from packages

On Red Hat Enterprise Linux, use **yum** to install the clients:

```
# yum install python-PROJECTclient
```

2.3. Upgrade or remove clients

To upgrade a client, add the **--upgrade** option to the **pip install** command:

```
# pip install --upgrade python-PROJECTclient
```

To remove the a client, run the **pip uninstall** command:

```
# pip uninstall python-PROJECTclient
```

2.4. What's next

Before you can run client commands, you must create and source the **PROJECT-openrc.sh** file to set environment variables. See [Section 4, “Set environment variables using the OpenStack RC file”](#).

3. Discover the version number for a client

Run the following command to discover the version number for a client:

```
$ PROJECT --version
```

For example, to see the version number for the **nova** client, run the following command:

```
$ nova --version
```

The version number (2.15.0 in the example) is returned.

```
2.15.0
```

4. Set environment variables using the OpenStack RC file

To set the required environment variables for the OpenStack command-line clients, you must create an environment file called an OpenStack rc file, or **openrc.sh** file. If your OpenStack installation provides it, you can download the file from the OpenStack dashboard as an administrative user or any other user. This project-specific environment file contains the credentials that all OpenStack services use.

When you source the file, environment variables are set for your current shell. The variables enable the OpenStack client commands to communicate with the OpenStack services that run in the cloud.

Note

Defining environment variables using an environment file is not a common practice on Microsoft Windows. Environment variables are usually defined in the **Advanced** tab of the System Properties dialog box.

4.1. Download and source the OpenStack RC file

1. Log in to the OpenStack dashboard, choose the project for which you want to download the OpenStack RC file, and click **Access & Security**.
2. On the API Access tab, click **Download OpenStack RC File** and save the file. The filename will be of the form **PROJECT-openrc.sh** where **PROJECT** is the name of the project for which you downloaded the file.
3. Copy the **PROJECT-openrc.sh** file to the computer from which you want to run OpenStack commands.

For example, copy the file to the computer from which you want to upload an image with a **glance** client command.

4. On any shell from which you want to run OpenStack commands, source the **PROJECT-openrc.sh** file for the respective project.

In the following example, the **demo-openrc.sh** file is sourced for the demo project:

```
$ source demo-openrc.sh
```

- When you are prompted for an OpenStack password, enter the password for the user who downloaded the **PROJECT-openrc.sh** file.

4.2. Create and source the OpenStack RC file

Alternatively, you can create the **PROJECT-openrc.sh** file from scratch, if for some reason you cannot download the file from the dashboard.

- In a text editor, create a file named **PROJECT-openrc.sh** file and add the following authentication information:

```
export OS_USERNAME=username
export OS_PASSWORD=password
export OS_TENANT_NAME=projectName
export OS_AUTH_URL=https://identityHost:portNumber/v2.0
# The following lines can be omitted
export OS_TENANT_ID=tenantIDString
export OS_REGION_NAME=regionName
```

The following example shows the information for a project called **admin**, where the OS username is also **admin**, and the identity host is located at **controller**.

```
export OS_USERNAME=admin
export OS_PASSWORD=ADMIN_PASS
export OS_TENANT_NAME=admin
export OS_AUTH_URL=http://controller:35357/v2.0
```

- On any shell from which you want to run OpenStack commands, source the **PROJECT-openrc.sh** file for the respective project. In this example, you source the **admin-openrc.sh** file for the *admin* project:

```
$ source admin-openrc.sh
```

Note

You are not prompted for the password with this method. The password lives in clear text format in the **PROJECT-openrc.sh** file. Restrict the permissions on this file to avoid security problems. You can also remove the **OS_PASSWORD** variable from the file, and use the **--password** parameter with OpenStack client commands instead.

4.3. Override environment variable values

When you run OpenStack client commands, you can override some environment variable settings by using the options that are listed at the end of the **help** output of the various client commands. For example, you can override the **OS_PASSWORD** setting in the **PROJECT-openrc.sh** file by specifying a password on a **keystone** command, as follows:

```
$ keystone --os-password PASSWORD service-list
```

Where *PASSWORD* is your password.

Chapter 2. Block Storage command-line client

The **cinder** client is the command-line interface (CLI) for the OpenStack Block Storage API and its extensions. This chapter documents **cinder** version 1.0.8.

For help on a specific **cinder** command, enter:

```
$ cinder help COMMAND
```

1. cinder usage

```
usage: cinder [--version] [--debug] [--os-username <auth-user-name>]
[--os-password <auth-password>] [--os-tenant-name <auth-tenant-
name>] [--os-tenant-id <auth-tenant-id>] [--os-auth-url <auth-url>]
[--os-region-name <region-name>] [--service-type <service-type>] [--
service-name <service-name>] [--volume-service-name <volume-service-
name>] [--endpoint-type <endpoint-type>] [--os-volume-api-version
<volume-api-ver>] [--os-cacert <ca-certificate>] [--retries
<retries>] <subcommand> ...
```

Subcommands

absolute-limits

Print a list of absolute limits for a user

availability-zone-list

List all the availability zones.

backup-create

Creates a backup.

backup-delete

Remove a backup.

backup-list

List all the backups.

backup-restore

Restore a backup.

backup-show

Show details about a backup.

create

Add a new volume.

credentials

Show user credentials returned from auth.

delete

Remove volume(s).

encryption-type-create

Create a new encryption type for a volume type (Admin Only).

encryption-type-delete

Delete the encryption type for a volume type (Admin Only).

encryption-type-list

List encryption type information for all volume types (Admin Only).

encryption-type-show

Show the encryption type information for a volume type (Admin Only).

endpoints

Discover endpoints that get returned from the authenticate services.

extend

Attempt to extend the size of an existing volume.

extra-specs-list

Print a list of current 'volume types and extra specs' (Admin Only).

force-delete

Attempt forced removal of volume(s), regardless of the state(s).

list

List all the volumes.

metadata

Set or Delete metadata on a volume.

metadata-show

Show metadata of given volume.

metadata-update-all

Update all metadata of a volume.

migrate

Migrate the volume to the new host.

qos-associate

Associate qos specs with specific volume type.

qos-create

Create a new qos specs.

qos-delete

Delete a specific qos specs.

qos-disassociate

Disassociate qos specs from specific volume type.

qos-disassociate-all

Disassociate qos specs from all of its associations.

qos-get-association

Get all associations of specific qos specs.

qos-key

Set or unset specifications for a qos spec.

qos-list

Get full list of qos specs.

qos-show

Get a specific qos specs.

quota-class-show

List the quotas for a quota class.

quota-class-update

Update the quotas for a quota class.

quota-defaults

List the default quotas for a tenant.

quota-show

List the quotas for a tenant.

quota-update

Update the quotas for a tenant.

quota-usage

List the quota usage for a tenant.

rate-limits

Print a list of rate limits for a user

readonly-mode-update

Update volume read-only access mode read_only.

rename

Rename a volume.

reset-state

Explicitly update the state of a volume.

service-disable

Disable the service.

service-enable

Enable the service.

service-list

List all the services. Filter by host & service binary.

show

Show details about a volume.

snapshot-create

Add a new snapshot.

snapshot-delete

Remove a snapshot.

snapshot-list

List all the snapshots.

snapshot-metadata

Set or Delete metadata of a snapshot.

snapshot-metadata-show

Show metadata of given snapshot.

snapshot-metadata-update-all

Update all metadata of a snapshot.

snapshot-rename

Rename a snapshot.

snapshot-reset-state

Explicitly update the state of a snapshot.

snapshot-show

Show details about a snapshot.

transfer-accept

Accepts a volume transfer.

transfer-create

Creates a volume transfer.

transfer-delete

Undo a transfer.

transfer-list

List all the transfers.

transfer-show

Show details about a transfer.

type-create

Create a new volume type.

type-delete

Delete a specific volume type.

type-key

Set or unset extra_spec for a volume type.

type-list

Print a list of available 'volume types'.

upload-to-image

Upload volume to image service as image.

bash-completion

Print arguments for bash_completion.

help

Display help about this program or one of its subcommands.

list-extensions

List all the os-api extensions that are available.

2. cinder optional arguments

--version

show program's version number and exit

--debug

Print debugging output

--os-username <auth-user-name>

Defaults to `env[OS_USERNAME]`.

--os-password <auth-password>

Defaults to `env[OS_PASSWORD]`.

--os-tenant-name <auth-tenant-name>
 Defaults to `env[OS_TENANT_NAME]`.

--os-tenant-id <auth-tenant-id>
 Defaults to `env[OS_TENANT_ID]`.

--os-auth-url <auth-url>
 Defaults to `env[OS_AUTH_URL]`.

--os-region-name <region-name>
 Defaults to `env[OS_REGION_NAME]`.

--service-type <service-type>
 Defaults to volume for most actions

--service-name <service-name>
 Defaults to `env[CINDER_SERVICE_NAME]`

--volume-service-name <volume-service-name>
 Defaults to `env[CINDER_VOLUME_SERVICE_NAME]`

--endpoint-type <endpoint-type>
 Defaults to `env[CINDER_ENDPOINT_TYPE]` or publicURL.

--os-volume-api-version <volume-api-ver>
 Accepts 1 or 2, defaults to `env[OS_VOLUME_API_VERSION]`.

--os-cacert <ca-certificate>
 Specify a CA bundle file to use in verifying a TLS (https) server certificate. Defaults to `env[OS_CACERT]`

--retries <retries>
 Number of retries.

3. cinder absolute-limits command

```
usage: cinder absolute-limits
```

Print a list of absolute limits for a user

4. cinder availability-zone-list command

```
usage: cinder availability-zone-list
```

List all the availability zones.

5. cinder backup-create command

```
usage: cinder backup-create [--container <container>] [--display-name <display-name>] [--display-description <display-description>] <volume>
```

Creates a backup.

Positional arguments

<volume>

Name or ID of the volume to backup.

Optional arguments

--container <container>

Optional Backup container name. (Default=None)

--display-name <display-name>

Optional backup name. (Default=None)

--display-description <display-description>

Optional backup description. (Default=None)

6. cinder backup-delete command

```
usage: cinder backup-delete <backup>
```

Remove a backup.

Positional arguments

<backup>

Name or ID of the backup to delete.

7. cinder backup-list command

```
usage: cinder backup-list
```

List all the backups.

8. cinder backup-restore command

```
usage: cinder backup-restore [--volume-id <volume>] <backup>
```

Restore a backup.

Positional arguments

<backup>

ID of the backup to restore.

Optional arguments**--volume-id <volume>**

Optional ID(or name) of the volume to restore to.

9. cinder backup-show command

```
usage: cinder backup-show <backup>
```

Show details about a backup.

Positional arguments**<backup>**

Name or ID of the backup.

10. cinder create command

```
usage: cinder create [--snapshot-id <snapshot-id>] [--source-volid
<source-volid>] [--image-id <image-id>] [--display-name <display-
name>] [--display-description <display-description>] [--volume-type
<volume-type>] [--availability-zone <availability-zone>] [--
metadata [<key=value> [<key=value> ...]]] <size>
```

Add a new volume.

Positional arguments**<size>**

Size of volume in GB

Optional arguments**--snapshot-id <snapshot-id>**

Create volume from snapshot id (Optional, Default=None)

--source-volid <source-volid>

Create volume from volume id (Optional, Default=None)

--image-id <image-id>

Create volume from image id (Optional, Default=None)

--display-name <display-name>

Volume name (Optional, Default=None)

--display-description <display-description>

Volume description (Optional, Default=None)

--volume-type <volume-type>

Volume type (Optional, Default=None)

--availability-zone <availability-zone>

Availability zone for volume (Optional, Default=None)

--metadata [<key=value> [<key=value> ...]]

Metadata key=value pairs (Optional, Default=None)

11. cinder credentials command

usage: cinder credentials

Show user credentials returned from auth.

12. cinder delete command

usage: cinder delete <volume> [<volume> ...]

Remove volume(s).

Positional arguments

<volume>

Name or ID of the volume(s) to delete.

13. cinder encryption-type-create command

usage: cinder encryption-type-create [--cipher <cipher>] [--key_size <key_size>] [--control_location <control_location>] <volume_type> <provider>

Create a new encryption type for a volume type (Admin Only).

Positional arguments

<volume_type>

Name or ID of the volume type

<provider>

Class providing encryption support (e.g. LuksEncryptor)

Optional arguments

--cipher <cipher>

Encryption algorithm/mode to use (e.g., aes-xts-plain64) (Optional, Default=None)

--key_size <key_size>

Size of the encryption key, in bits (e.g., 128, 256) (Optional, Default=None)

--control_location <control_location>

Notional service where encryption is performed (e.g., front-end=Nova). Values: 'front-end', 'back-end' (Optional, Default=None)

14. cinder encryption-type-delete command

usage: cinder encryption-type-delete <volume_type>

Delete the encryption type for a volume type (Admin Only).

Positional arguments**<volume_type>**

Name or ID of the volume type

15. cinder encryption-type-list command

usage: cinder encryption-type-list

List encryption type information for all volume types (Admin Only).

16. cinder encryption-type-show command

usage: cinder encryption-type-show <volume_type>

Show the encryption type information for a volume type (Admin Only).

Positional arguments**<volume_type>**

Name or ID of the volume type

17. cinder endpoints command

usage: cinder endpoints

Discover endpoints that get returned from the authenticate services.

18. cinder extend command

■

```
usage: cinder extend <volume> <new-size>
```

Attempt to extend the size of an existing volume.

Positional arguments

<volume>

Name or ID of the volume to extend.

<new-size>

New size of volume in GB

19. cinder extra-specs-list command

```
usage: cinder extra-specs-list
```

Print a list of current 'volume types and extra specs' (Admin Only).

20. cinder force-delete command

```
usage: cinder force-delete <volume> [<volume> ...]
```

Attempt forced removal of volume(s), regardless of the state(s).

Positional arguments

<volume>

Name or ID of the volume(s) to delete.

21. cinder list command

```
usage: cinder list [--all-tenants [<0|1>]] [--display-name <display-name>] [--status <status>] [--metadata [<key=value> [<key=value> ...]]]
```

List all the volumes.

Optional arguments

--all-tenants [<0|1>]

Display information from all tenants (Admin only).

--display-name <display-name>

Filter results by display-name

--status <status>

Filter results by status

```
--metadata [<key=value> [<key=value> ...]]
```

Filter results by metadata

22. cinder list-extensions command

```
usage: cinder list-extensions
```

List all the os-api extensions that are available.

23. cinder metadata command

```
usage: cinder metadata <volume> <action> <key=value> [<key=value> ...]
```

Set or Delete metadata on a volume.

Positional arguments

<volume>

Name or ID of the volume to update metadata on.

<action>

Actions: 'set' or 'unset'

<key=value>

Metadata to set/unset (only key is necessary on unset)

24. cinder metadata-show command

```
usage: cinder metadata-show <volume>
```

Show metadata of given volume.

Positional arguments

<volume>

ID of volume

25. cinder metadata-update-all command

```
usage: cinder metadata-update-all <volume> <key=value> [<key=value> ...]
```

Update all metadata of a volume.

Positional arguments

<volume>

ID of the volume to update metadata on.

<key=value>

Metadata entry/entries to update.

26. cinder migrate command

```
usage: cinder migrate [--force-host-copy <True|False>] <volume>
<host>
```

Migrate the volume to the new host.

Positional arguments

<volume>

ID of the volume to migrate

<host>

Destination host

Optional arguments

--force-host-copy <True|False>

Optional flag to force the use of the generic host- based migration mechanism, bypassing driver optimizations (Default=False).

27. cinder qos-associate command

```
usage: cinder qos-associate <qos_specs> <volume_type_id>
```

Associate qos specs with specific volume type.

Positional arguments

<qos_specs>

ID of qos_specs.

<volume_type_id>

ID of volume type to be associated with.

28. cinder qos-create command

```
usage: cinder qos-create <name> <key=value> [<key=value> ...]
```

Create a new qos specs.

Positional arguments**<name>**

Name of the new QoS specs

<key=value>

Specifications for QoS

29. cinder qos-delete command

```
usage: cinder qos-delete [--force <True|False>] <qos_specs>
```

Delete a specific qos specs.

Positional arguments**<qos_specs>**

ID of the qos_specs to delete.

Optional arguments**--force <True|False>**

Optional flag that indicates whether to delete specified qos specs even if it is in-use.

30. cinder qos-disassociate command

```
usage: cinder qos-disassociate <qos_specs> <volume_type_id>
```

Disassociate qos specs from specific volume type.

Positional arguments**<qos_specs>**

ID of qos_specs.

<volume_type_id>

ID of volume type to be associated with.

31. cinder qos-disassociate-all command

```
usage: cinder qos-disassociate-all <qos_specs>
```

Disassociate qos specs from all of its associations.

Positional arguments

<qos_specs>

ID of qos_specs to be operate on.

32. cinder qos-get-association command

usage: cinder qos-get-association <qos_specs>

Get all associations of specific qos specs.

Positional arguments

<qos_specs>

ID of the qos_specs.

33. cinder qos-key command

usage: cinder qos-key <qos_specs> <action> key=value [key=value ...]

Set or unset specifications for a qos spec.

Positional arguments

<qos_specs>

ID of qos specs

<action>

Actions: 'set' or 'unset'

key=value

QoS specs to set/unset (only key is necessary on unset)

34. cinder qos-list command

usage: cinder qos-list

Get full list of qos specs.

35. cinder qos-show command

usage: cinder qos-show <qos_specs>

Get a specific qos specs.

Positional arguments

<qos_specs>

ID of the qos_specs to show.

36. cinder quota-class-show command

```
usage: cinder quota-class-show <class>
```

List the quotas for a quota class.

Positional arguments

<class>

Name of quota class to list the quotas for.

37. cinder quota-class-update command

```
usage: cinder quota-class-update [--volumes <volumes>] [--snapshots
<snapshots>] [--gigabytes <gigabytes>] [--volume-type
<volume_type_name>] <class>
```

Update the quotas for a quota class.

Positional arguments

<class>

Name of quota class to set the quotas for.

Optional arguments

--volumes <volumes>

New value for the "volumes" quota.

--snapshots <snapshots>

New value for the "snapshots" quota.

--gigabytes <gigabytes>

New value for the "gigabytes" quota.

--volume-type <volume_type_name>

Volume type (Optional, Default=None)

38. cinder quota-defaults command

```
usage: cinder quota-defaults <tenant_id>
```

List the default quotas for a tenant.

Positional arguments

<tenant_id>

UUID of tenant to list the default quotas for.

39. cinder quota-show command

```
usage: cinder quota-show <tenant_id>
```

List the quotas for a tenant.

Positional arguments

<tenant_id>

UUID of tenant to list the quotas for.

40. cinder quota-update command

```
usage: cinder quota-update [--volumes <volumes>] [--snapshots  
<snapshots>] [--gigabytes <gigabytes>] [--volume-type  
<volume_type_name>] <tenant_id>
```

Update the quotas for a tenant.

Positional arguments

<tenant_id>

UUID of tenant to set the quotas for.

Optional arguments

--volumes <volumes>

New value for the "volumes" quota.

--snapshots <snapshots>

New value for the "snapshots" quota.

--gigabytes <gigabytes>

New value for the "gigabytes" quota.

--volume-type <volume_type_name>

Volume type (Optional, Default=None)

41. cinder quota-usage command

```
usage: cinder quota-usage <tenant_id>
```

List the quota usage for a tenant.

Positional arguments

<tenant_id>

UUID of tenant to list the quota usage for.

42. cinder rate-limits command

```
usage: cinder rate-limits
```

Print a list of rate limits for a user

43. cinder readonly-mode-update command

```
usage: cinder readonly-mode-update <volume> <True|true|False|false>
```

Update volume read-only access mode read_only.

Positional arguments

<volume>

ID of the volume to update.

<True|true|False|false>

Flag to indicate whether to update volume to read-only access mode.

44. cinder rename command

```
usage: cinder rename [--display-description <display-description>]
<volume> [<display-name>]
```

Rename a volume.

Positional arguments

<volume>

Name or ID of the volume to rename.

<display-name>

New display-name for the volume.

Optional arguments

--display-description <display-description>

Optional volume description. (Default=None)

45. cinder reset-state command

```
usage: cinder reset-state [--state <state>] <volume> [<volume> ...]
```

Explicitly update the state of a volume.

Positional arguments

<volume>

Name or ID of the volume to modify.

Optional arguments

--state <state>

Indicate which state to assign the volume. Options include available, error, creating, deleting, error_deleting. If no state is provided, available will be used.

46. cinder service-disable command

```
usage: cinder service-disable <hostname> <binary>
```

Disable the service.

Positional arguments

<hostname>

Name of host.

<binary>

Service binary.

47. cinder service-enable command

```
usage: cinder service-enable <hostname> <binary>
```

Enable the service.

Positional arguments

<hostname>

Name of host.

<binary>

Service binary.

48. cinder service-list command

```
usage: cinder service-list [--host <hostname>] [--binary <binary>]
```

List all the services. Filter by host & service binary.

Optional arguments

--host <hostname>

Name of host.

--binary <binary>

Service binary.

49. cinder show command

```
usage: cinder show <volume>
```

Show details about a volume.

Positional arguments

<volume>

Name or ID of the volume.

50. cinder snapshot-create command

```
usage: cinder snapshot-create [--force <True|False>] [--display-name  
<display-name>] [--display-description <display-description>]  
<volume>
```

Add a new snapshot.

Positional arguments

<volume>

Name or ID of the volume to snapshot

Optional arguments

--force <True|False>

Optional flag to indicate whether to snapshot a volume even if it's attached to an instance. (Default=False)

--display-name <display-name>

Optional snapshot name. (Default=None)

--display-description <display-description>

Optional snapshot description. (Default=None)

51. cinder snapshot-delete command

```
usage: cinder snapshot-delete <snapshot>
```

Remove a snapshot.

Positional arguments

<snapshot>

Name or ID of the snapshot to delete.

52. cinder snapshot-list command

```
usage: cinder snapshot-list [--all-tenants [<0|1>]] [--display-name  
<display-name>] [--status <status>] [--volume-id <volume-id>]
```

List all the snapshots.

Optional arguments

--all-tenants [<0|1>]

Display information from all tenants (Admin only).

--display-name <display-name>

Filter results by display-name

--status <status>

Filter results by status

--volume-id <volume-id>

Filter results by volume-id

53. cinder snapshot-metadata command

```
usage: cinder snapshot-metadata <snapshot> <action> <key=value>  
[<key=value> ...]
```

Set or Delete metadata of a snapshot.

Positional arguments

<snapshot>

ID of the snapshot to update metadata on.

<action>

Actions: 'set' or 'unset'

<key=value>

Metadata to set/unset (only key is necessary on unset)

54. cinder snapshot-metadata-show command

```
usage: cinder snapshot-metadata-show <snapshot>
```

Show metadata of given snapshot.

Positional arguments

<snapshot>

ID of snapshot

55. cinder snapshot-metadata-update-all command

```
usage: cinder snapshot-metadata-update-all <snapshot> <key=value>
[<key=value> ...]
```

Update all metadata of a snapshot.

Positional arguments

<snapshot>

ID of the snapshot to update metadata on.

<key=value>

Metadata entry/entries to update.

56. cinder snapshot-rename command

```
usage: cinder snapshot-rename [--display-description <display-
description>] <snapshot> [<display-name>]
```

Rename a snapshot.

Positional arguments

<snapshot>

Name or ID of the snapshot.

<display-name>

New display-name for the snapshot.

Optional arguments

--display-description <display-description>

Optional snapshot description. (Default=None)

57. cinder snapshot-reset-state command

```
usage: cinder snapshot-reset-state [--state <state>] <snapshot>
[<snapshot> ...]
```

Explicitly update the state of a snapshot.

Positional arguments

<snapshot>

Name or ID of the snapshot to modify.

Optional arguments

--state <state>

Indicate which state to assign the snapshot. Options include available, error, creating, deleting, error_deleting. If no state is provided, available will be used.

58. cinder snapshot-show command

```
usage: cinder snapshot-show <snapshot>
```

Show details about a snapshot.

Positional arguments

<snapshot>

Name or ID of the snapshot.

59. cinder transfer-accept command

```
usage: cinder transfer-accept <transfer> <auth_key>
```

Accepts a volume transfer.

Positional arguments

<transfer>

ID of the transfer to accept.

<auth_key>

Auth key of the transfer to accept.

60. cinder transfer-create command

```
usage: cinder transfer-create [--display-name <display-name>]  
    <volume>
```

Creates a volume transfer.

Positional arguments

<volume>

Name or ID of the volume to transfer.

Optional arguments

--display-name <display-name>

Optional transfer name. (Default=None)

61. cinder transfer-delete command

```
usage: cinder transfer-delete <transfer>
```

Undo a transfer.

Positional arguments

<transfer>

Name or ID of the transfer to delete.

62. cinder transfer-list command

```
usage: cinder transfer-list
```

List all the transfers.

63. cinder transfer-show command

```
usage: cinder transfer-show <transfer>
```

Show details about a transfer.

Positional arguments

<transfer>

Name or ID of the transfer to accept.

64. cinder type-create command

```
usage: cinder type-create <name>
```

Create a new volume type.

Positional arguments

<name>

Name of the new volume type

65. cinder type-delete command

```
usage: cinder type-delete <id>
```

Delete a specific volume type.

Positional arguments

<id>

Unique ID of the volume type to delete

66. cinder type-key command

```
usage: cinder type-key <vtype> <action> [<key=value> [<key=value>
...]]
```

Set or unset extra_spec for a volume type.

Positional arguments

<vtype>

Name or ID of the volume type

<action>

Actions: 'set' or 'unset'

<key=value>

Extra_specs to set/unset (only key is necessary on unset)

67. cinder type-list command

```
usage: cinder type-list
```

Print a list of available 'volume types'.

68. cinder upload-to-image command


```
usage: cinder upload-to-image [--force <True|False>] [--container-  
format <container-format>] [--disk-format <disk-format>] <volume>  
<image-name>
```

Upload volume to image service as image.

Positional arguments

<volume>

Name or ID of the volume to upload to an image

<image-name>

Name for created image

Optional arguments

--force <True|False>

Optional flag to indicate whether to upload a volume even if it's attached to an instance. (Default=False)

--container-format <container-format>

Optional type for container format (Default=bare)

--disk-format <disk-format>

Optional type for disk format (Default=raw)

Chapter 3. Compute command-line client

The **nova** client is the command-line interface (CLI) for the OpenStack Compute API and its extensions. This chapter documents **nova** version 2.17.0.

For help on a specific **nova** command, enter:

```
$ nova help COMMAND
```

1. nova usage

```
usage: nova [--version] [--debug] [--os-cache] [--timings] [--timeout
<seconds>] [--os-auth-token OS_AUTH_TOKEN] [--os-username <auth-
user-name>] [--os-password <auth-password>] [--os-tenant-name <auth-
tenant-name>] [--os-tenant-id <auth-tenant-id>] [--os-auth-url
<auth-url>] [--os-region-name <region-name>] [--os-auth-system
<auth-system>] [--service-type <service-type>] [--service-name
<service-name>] [--volume-service-name <volume-service-name>] [--
endpoint-type <endpoint-type>] [--os-compute-api-version <compute-
api-ver>] [--os-cacert <ca-certificate>] [--insecure] [--bypass-url
<bypass-url>] <subcommand> ...
```

Subcommands

absolute-limits

Print a list of absolute limits for a user

add-fixed-ip

Add new IP address on a network to server.

add-floating-ip

DEPRECATED, use floating-ip-associate instead.

add-secgroup

Add a Security Group to a server.

agent-create

Create new agent build.

agent-delete

Delete existing agent build.

agent-list

List all builds.

agent-modify

Modify existing agent build.

aggregate-add-host

Add the host to the specified aggregate.

aggregate-create

Create a new aggregate with the specified details.

aggregate-delete

Delete the aggregate.

aggregate-details

Show details of the specified aggregate.

aggregate-list

Print a list of all aggregates.

aggregate-remove-host

Remove the specified host from the specified aggregate.

aggregate-set-metadata

Update the metadata associated with the aggregate.

aggregate-update

Update the aggregate's name and optionally availability zone.

availability-zone-list

List all the availability zones.

backup

Backup a server by creating a 'backup' type snapshot.

boot

Boot a new server.

clear-password

Clear password for a server.

cloudpipe-configure

Update the VPN IP/port of a cloudpipe instance.

cloudpipe-create

Create a cloudpipe instance for the given project.

cloudpipe-list

Print a list of all cloudpipe instances.

console-log

Get console log output of a server.

credentials

Show user credentials returned from auth.

delete

Immediately shut down and delete specified server(s).

diagnostics

Retrieve server diagnostics.

dns-create

Create a DNS entry for domain, name and ip.

dns-create-private-domain

Create the specified DNS domain.

dns-create-public-domain

Create the specified DNS domain.

dns-delete

Delete the specified DNS entry.

dns-delete-domain

Delete the specified DNS domain.

dns-domains

Print a list of available dns domains.

dns-list

List current DNS entries for domain and ip or domain and name.

endpoints

Discover endpoints that get returned from the authenticate services.

evacuate

Evacuate server from failed host to specified one.

fixed-ip-get

Retrieve info on a fixed ip.

fixed-ip-reserve

Reserve a fixed IP.

fixed-ip-unreserve

Unreserve a fixed IP.

flavor-access-add

Add flavor access for the given tenant.

flavor-access-list

Print access information about the given flavor.

flavor-access-remove

Remove flavor access for the given tenant.

flavor-create

Create a new flavor

flavor-delete

Delete a specific flavor

flavor-key

Set or unset extra_spec for a flavor.

flavor-list

Print a list of available 'flavors' (sizes of servers).

flavor-show

Show details about the given flavor.

floating-ip-associate

Associate a floating IP address to a server.

floating-ip-bulk-create

Bulk create floating ips by range.

floating-ip-bulk-delete

Bulk delete floating ips by range.

floating-ip-bulk-list

List all floating ips.

floating-ip-create

Allocate a floating IP for the current tenant.

floating-ip-delete

De-allocate a floating IP.

floating-ip-disassociate

Disassociate a floating IP address from a server.

floating-ip-list

List floating ips for this tenant.

floating-ip-pool-list

List all floating ip pools.

get-password

Get password for a server.

get-rdp-console

Get a rdp console to a server.

get-spice-console

Get a spice console to a server.

get-vnc-console

Get a vnc console to a server.

host-action

Perform a power action on a host.

host-describe

Describe a specific host.

host-list

List all hosts by service.

host-update

Update host settings.

hypervisor-list

List hypervisors.

hypervisor-servers

List servers belonging to specific hypervisors.

hypervisor-show

Display the details of the specified hypervisor.

hypervisor-stats

Get hypervisor statistics over all compute nodes.

hypervisor-uptime

Display the uptime of the specified hypervisor.

image-create

Create a new image by taking a snapshot of a running server.

image-delete

Delete specified image(s).

image-list

Print a list of available images to boot from.

image-meta

Set or Delete metadata on an image.

image-show

Show details about the given image.

interface-attach

Attach a network interface to a server.

interface-detach

Detach a network interface from a server.

interface-list

List interfaces attached to a server.

keypair-add

Create a new key pair for use with servers.

keypair-delete

Delete keypair given by its name.

keypair-list

Print a list of keypairs for a user

keypair-show

Show details about the given keypair.

list

List active servers.

list-secgroup

List Security Group(s) of a server.

live-migration

Migrate running server to a new machine.

lock

Lock a server.

meta

Set or Delete metadata on a server.

migrate

Migrate a server. The new host will be selected by the scheduler.

network-associate-host

Associate host with network.

network-associate-project

Associate project with network.

network-create

Create a network.

network-disassociate

Disassociate host and/or project from the given network.

network-list

Print a list of available networks.

network-show

Show details about the given network.

pause

Pause a server.

quota-class-show

List the quotas for a quota class.

quota-class-update

Update the quotas for a quota class.

quota-defaults

List the default quotas for a tenant.

quota-delete

Delete quota for a tenant/user so their quota will Revert back to default.

quota-show

List the quotas for a tenant/user.

quota-update

Update the quotas for a tenant/user.

rate-limits

Print a list of rate limits for a user

reboot

Reboot a server.

rebuild

Shutdown, re-image, and re-boot a server.

refresh-network

Refresh server network information.

remove-fixed-ip

Remove an IP address from a server.

remove-floating-ip

DEPRECATED, use floating-ip-disassociate instead.

remove-secgroup

Remove a Security Group from a server.

rename

Rename a server.

rescue

Rescue a server.

reset-network

Reset network of a server.

reset-state

Reset the state of a server.

resize

Resize a server.

resize-confirm

Confirm a previous resize.

resize-revert

Revert a previous resize (and return to the previous VM).

resume

Resume a server.

root-password

Change the root password for a server.

scrub

Delete data associated with the project.

secgroup-add-group-rule

Add a source group rule to a security group.

secgroup-add-rule

Add a rule to a security group.

secgroup-create

Create a security group.

secgroup-delete

Delete a security group.

secgroup-delete-group-rule

Delete a source group rule from a security group.

secgroup-delete-rule

Delete a rule from a security group.

secgroup-list

List security groups for the current tenant.

secgroup-list-rules

List rules for a security group.

secgroup-update

Update a security group.

server-group-create

Create a new server group with the specified details.

server-group-delete

Delete specific server group(s).

server-group-get

Get a specific server group.

server-group-list

Print a list of all server groups.

service-disable

Disable the service.

service-enable

Enable the service.

service-list

Show a list of all running services. Filter by host & binary.

shelve

Shelve a server.

shelve-offload

Remove a shelved server from the compute node.

show

Show details about the given server.

ssh

SSH into a server.

start

Start a server.

stop

Stop a server.

suspend

Suspend a server.

unlock

Unlock a server.

unpause

Unpause a server.

unrescue

Unrescue a server.

unshelve

Unshelve a server.

usage

Show usage data for a single tenant.

usage-list

List usage data for all tenants.

volume-attach

Attach a volume to a server.

volume-create

Add a new volume.

volume-delete

Remove volume(s).

volume-detach

Detach a volume from a server.

volume-list

List all the volumes.

volume-show

Show details about a volume.

volume-snapshot-create

Add a new snapshot.

volume-snapshot-delete

Remove a snapshot.

volume-snapshot-list

List all the snapshots.

volume-snapshot-show

Show details about a snapshot.

volume-type-create

Create a new volume type.

volume-type-delete

Delete a specific flavor

volume-type-list

Print a list of available 'volume types'.

volume-update

Update volume attachment.

x509-create-cert

Create x509 cert for a user in tenant.

x509-get-root-cert

Fetch the x509 root cert.

bash-completion

Prints all of the commands and options to stdout so that the nova.bash_completion script doesn't have to hard code them.

help

Display help about this program or one of its subcommands.

force-delete

Force delete a server.

restore

Restore a soft-deleted server.

net

Show a network

net-create

Create a network

net-delete

Delete a network

net-list

List networks

baremetal-interface-add

Add a network interface to a baremetal node.

baremetal-interface-list

List network interfaces associated with a baremetal node.

baremetal-interface-remove

Remove a network interface from a baremetal node.

baremetal-node-create

Create a baremetal node.

baremetal-node-delete

Remove a baremetal node and any associated interfaces.

baremetal-node-list

Print list of available baremetal nodes.

baremetal-node-show

Show information about a baremetal node.

host-evacuate

Evacuate all instances from failed host to specified one.

instance-action

Show an action.

instance-action-list

List actions on a server.

migration-list

Print a list of migrations.

host-servers-migrate

Migrate all instances of the specified host to other available hosts.

cell-capacities

Get cell capacities for all cells or a given cell.

cell-show

Show details of a given cell.

host-meta

Set or Delete metadata on all instances of a host.

list-extensions

List all the os-api extensions that are available.

2. nova optional arguments

--version

show program's version number and exit

--debug

Print debugging output

--os-cache

Use the auth token cache. Defaults to False if **env[OS_CACHE]** is not set.

--timings

Print call timing info

--timeout <seconds>

Set HTTP call timeout (in seconds)

--os-auth-token OS_AUTH_TOKEN

Defaults to **env[OS_AUTH_TOKEN]**

--os-username <auth-user-name>

Defaults to **env[OS_USERNAME]**.

--os-password <auth-password>

Defaults to **env[OS_PASSWORD]**.

--os-tenant-name <auth-tenant-name>

Defaults to **env[OS_TENANT_NAME]**.

--os-tenant-id <auth-tenant-id>

Defaults to **env[OS_TENANT_ID]**.

--os-auth-url <auth-url>

Defaults to **env[OS_AUTH_URL]**.

--os-region-name <region-name>

Defaults to **env[OS_REGION_NAME]**.

--os-auth-system <auth-system>

Defaults to **env[OS_AUTH_SYSTEM]**.

--service-type <service-type>

Defaults to compute for most actions

--service-name <service-name>

Defaults to `env[NOVA_SERVICE_NAME]`

--volume-service-name <volume-service-name>

Defaults to `env[NOVA_VOLUME_SERVICE_NAME]`

--endpoint-type <endpoint-type>

Defaults to `env[NOVA_ENDPOINT_TYPE]` or `publicURL`.

--os-compute-api-version <compute-api-ver>

Accepts 1.1 or 3, defaults to `env[OS_COMPUTE_API_VERSION]`.

--os-cacert <ca-certificate>

Specify a CA bundle file to use in verifying a TLS (https) server certificate. Defaults to `env[OS_CACERT]`

--insecure

Explicitly allow novaclient to perform "insecure" SSL (https) requests. The server's certificate will not be verified against any certificate authorities. This option should be used with caution.

--bypass-url <bypass-url>

Use this API endpoint instead of the Service Catalog

3. nova absolute-limits command

```
usage: nova absolute-limits [--tenant [<tenant>]] [--reserved]
```

Print a list of absolute limits for a user

Optional arguments

--tenant [<tenant>]

Display information from single tenant (Admin only).

--reserved

Include reservations count.

4. nova add-fixed-ip command

```
usage: nova add-fixed-ip <server> <network-id>
```

Add new IP address on a network to server.

Positional arguments

<server>

Name or ID of server.

<network-id>

Network ID.

5. nova add-secgroup command

```
usage: nova add-secgroup <server> <secgroup>
```

Add a Security Group to a server.

Positional arguments

<server>

Name or ID of server.

<secgroup>

Name of Security Group.

6. nova agent-create command

```
usage: nova agent-create <os> <architecture> <version> <url>  
       <md5hash> <hypervisor>
```

Create new agent build.

Positional arguments

<os>

type of os.

<architecture>

type of architecture

<version>

version

<url>

url

<md5hash>

md5 hash

<hypervisor>

type of hypervisor.

7. nova agent-delete command

```
usage: nova agent-delete <id>
```

Delete existing agent build.

Positional arguments

<id>

id of the agent-build

8. nova agent-list command

```
usage: nova agent-list [--hypervisor <hypervisor>]
```

List all builds.

Optional arguments

--hypervisor <hypervisor>

type of hypervisor.

9. nova agent-modify command

```
usage: nova agent-modify <id> <version> <url> <md5hash>
```

Modify existing agent build.

Positional arguments

<id>

id of the agent-build

<version>

version

<url>

url

<md5hash>

md5hash

10. nova aggregate-add-host command

```
usage: nova aggregate-add-host <aggregate> <host>
```

Add the host to the specified aggregate.

Positional arguments

<aggregate>

Name or ID of aggregate.

<host>

The host to add to the aggregate.

11. nova aggregate-create command

```
usage: nova aggregate-create <name> [<availability-zone>]
```

Create a new aggregate with the specified details.

Positional arguments

<name>

Name of aggregate.

<availability-zone>

The availability zone of the aggregate (optional).

12. nova aggregate-delete command

```
usage: nova aggregate-delete <aggregate>
```

Delete the aggregate.

Positional arguments

<aggregate>

Name or ID of aggregate to delete.

13. nova aggregate-details command

```
usage: nova aggregate-details <aggregate>
```

Show details of the specified aggregate.

Positional arguments

<aggregate>

Name or ID of aggregate.

14. nova aggregate-list command

```
usage: nova aggregate-list
```

Print a list of all aggregates.

15. nova aggregate-remove-host command

```
usage: nova aggregate-remove-host <aggregate> <host>
```

Remove the specified host from the specified aggregate.

Positional arguments

<aggregate>

Name or ID of aggregate.

<host>

The host to remove from the aggregate.

16. nova aggregate-set-metadata command

```
usage: nova aggregate-set-metadata <aggregate> <key=value>
[<key=value> ...]
```

Update the metadata associated with the aggregate.

Positional arguments

<aggregate>

Name or ID of aggregate to update.

<key=value>

Metadata to add/update to aggregate

17. nova aggregate-update command

```
usage: nova aggregate-update <aggregate> <name> [<availability-
zone>]
```

Update the aggregate's name and optionally availability zone.

Positional arguments

<aggregate>

Name or ID of aggregate to update.

<name>

Name of aggregate.

<availability-zone>

The availability zone of the aggregate.

18. nova availability-zone-list command

```
usage: nova availability-zone-list
```

List all the availability zones.

19. nova backup command

```
usage: nova backup <server> <name> <backup-type> <rotation>
```

Backup a server by creating a 'backup' type snapshot.

Positional arguments

<server>

Name or ID of server.

<name>

Name of the backup image.

<backup-type>

The backup type, like "daily" or "weekly".

<rotation>

Int parameter representing how many backups to keep around.

20. nova baremetal-interface-add command

```
usage: nova baremetal-interface-add [--datapath_id <datapath_id>] [-  
-port_no <port_no>] <node> <address>
```

Add a network interface to a baremetal node.

Positional arguments

<node>

ID of node

<address>

MAC address of interface

Optional arguments

--datapath_id <datapath_id>

OpenFlow Datapath ID of interface

--port_no <port_no>

OpenFlow port number of interface

21. nova baremetal-interface-list command

```
usage: nova baremetal-interface-list <node>
```

List network interfaces associated with a baremetal node.

Positional arguments

<node>

ID of node

22. nova baremetal-interface-remove command

```
usage: nova baremetal-interface-remove <node> <address>
```

Remove a network interface from a baremetal node.

Positional arguments

<node>

ID of node

<address>

MAC address of interface

23. nova baremetal-node-create command

```
usage: nova baremetal-node-create [--pm_address <pm_address>] [--pm_user <pm_user>] [--pm_password <pm_password>] [--terminal_port <terminal_port>] <service_host> <cpus> <memory_mb> <local_gb> <prov_mac_address>
```

Create a baremetal node.

Positional arguments

<service_host>

Name of nova compute host which will control this baremetal node

<cpus>

Number of CPUs in the node

<memory_mb>

Megabytes of RAM in the node

<local_gb>

Gigabytes of local storage in the node

<prov_mac_address>

MAC address to provision the node

Optional arguments**--pm_address <pm_address>**

Power management IP for the node

--pm_user <pm_user>

Username for the node's power management

--pm_password <pm_password>

Password for the node's power management

--terminal_port <terminal_port>

ShellInABox port?

24. nova baremetal-node-delete command

```
usage: nova baremetal-node-delete <node>
```

Remove a baremetal node and any associated interfaces.

Positional arguments**<node>**

ID of the node to delete.

25. nova baremetal-node-list command

```
usage: nova baremetal-node-list
```

Print list of available baremetal nodes.

26. nova baremetal-node-show command

```
usage: nova baremetal-node-show <node>
```

Show information about a baremetal node.

Positional arguments

<node>

ID of node

27. nova boot command

```
usage: nova boot [--flavor <flavor>] [--image <image>] [--image-with
<key=value>] [--boot-volume <volume_id>] [--snapshot <snapshot_id>]
[--num-instances <number>] [--meta <key=value>] [--file <dst-
path=src-path>] [--key-name <key-name>] [--user-data <user-data>] [-
-availability-zone <availability-zone>] [--security-groups
<security-groups>] [--block-device-mapping <dev-name=mapping>] [--
block-device key1=value1[,key2=value2...]] [--swap <swap_size>] [--
ephemeral size=<size>[,format=<format>]] [--hint <key=value>] [--nic
<net-id=net-uuid,v4-fixed-ip=ip-addr,port-id=port-uuid>] [--config-
drive <value>] [--poll] <name>
```

Boot a new server.

Positional arguments

<name>

Name for the new server

Optional arguments

--flavor <flavor>

Name or ID of flavor (see 'nova flavor-list').

--image <image>

Name or ID of image (see 'nova image-list').

--image-with <key=value>

Image metadata property (see 'nova image-show').

--boot-volume <volume_id>

Volume ID to boot from.

--snapshot <snapshot_id>

Snapshot ID to boot from (will create a volume).

--num-instances <number>

boot multiple servers at a time (limited by quota).

--meta <key=value>

Record arbitrary key/value metadata to /meta.js on the new server. Can be specified

multiple times.

--file <dst-path=src-path>

Store arbitrary files from <src-path> locally to <dst-path> on the new server. You may store up to 5 files.

--key-name <key-name>

Key name of keypair that should be created earlier with the command `keypair-add`

--user-data <user-data>

user data file to pass to be exposed by the metadata server.

--availability-zone <availability-zone>

The availability zone for server placement.

--security-groups <security-groups>

Comma separated list of security group names.

--block-device-mapping <dev-name=mapping>

Block device mapping in the format <dev-name>=<id>:<type>:<size(GB)>:<delete-on-terminate>.

--block-device

key1=value1[,key2=value2...] Block device mapping with the keys: id=image_id, snapshot_id or volume_id, source=source type (image, snapshot, volume or blank), dest=destination type of the block device (volume or local), bus=device's bus, device=name of the device (e.g. vda, xda, ...), size=size of the block device in GB, format=device will be formatted (e.g. swap, ext3, ntfs, ...), bootindex=integer used for ordering the boot disks, type=device type (e.g. disk, cdrom, ...) and shutdown=shutdown behaviour (either preserve or remove).

--swap <swap_size>

Create and attach a local swap block device of <swap_size> MB.

--ephemeral

size=<size>[,format=<format>] Create and attach a local ephemeral block device of <size> GB and format it to <format>.

--hint <key=value>

Send arbitrary key/value pairs to the scheduler for custom use.

--nic <net-id=net-uuid,v4-fixed-ip=ip-addr,port-id=port-uuid>

Create a NIC on the server. Specify option multiple times to create multiple NICs. net-id: attach NIC to network with this UUID (required if no port-id), v4-fixed-ip: IPv4 fixed address for NIC (optional), port-id: attach NIC to port with this UUID (required if no net-id)

--config-drive <value>

Enable config drive

--poll

Blocks while server builds so progress can be reported.

28. nova cell-capacities command

```
usage: nova cell-capacities [--cell <cell-name>]
```

Get cell capacities for all cells or a given cell.

Optional arguments

--cell <cell-name>

Name of the cell to get the capacities.

29. nova cell-show command

```
usage: nova cell-show <cell-name>
```

Show details of a given cell.

Positional arguments

<cell-name>

Name of the cell.

30. nova clear-password command

```
usage: nova clear-password <server>
```

Clear password for a server.

Positional arguments

<server>

Name or ID of server.

31. nova cloudpipe-configure command

```
usage: nova cloudpipe-configure <ip address> <port>
```

Update the VPN IP/port of a cloudpipe instance.

Positional arguments

<ip address>

New IP Address.

<port>

New Port.

32. nova cloudpipe-create command

usage: nova cloudpipe-create <project_id>

Create a cloudpipe instance for the given project.

Positional arguments

<project_id>

UUID of the project to create the cloudpipe for.

33. nova cloudpipe-list command

usage: nova cloudpipe-list

Print a list of all cloudpipe instances.

34. nova console-log command

usage: nova console-log [--length <length>] <server>

Get console log output of a server.

Positional arguments

<server>

Name or ID of server.

Optional arguments

--length <length>

Length in lines to tail.

35. nova credentials command

usage: nova credentials [--wrap <integer>]

Show user credentials returned from auth.

Optional arguments

--wrap <integer>

wrap PKI tokens to a specified length, or 0 to disable

36. nova delete command

```
usage: nova delete <server> [<server> ...]
```

Immediately shut down and delete specified server(s).

Positional arguments

<server>

Name or ID of server(s).

37. nova diagnostics command

```
usage: nova diagnostics <server>
```

Retrieve server diagnostics.

Positional arguments

<server>

Name or ID of server.

38. nova dns-create command

```
usage: nova dns-create [--type <type>] <ip> <name> <domain>
```

Create a DNS entry for domain, name and ip.

Positional arguments

<ip>

ip address

<name>

DNS name

<domain>

DNS domain

Optional arguments

--type <type>

dns type (e.g. "A")

39. nova dns-create-private-domain command

```
usage: nova dns-create-private-domain [--availability-zone
<availability-zone>] <domain>
```

Create the specified DNS domain.

Positional arguments

<domain>

DNS domain

Optional arguments

--availability-zone <availability-zone>

Limit access to this domain to servers in the specified availability zone.

40. nova dns-create-public-domain command

```
usage: nova dns-create-public-domain [--project <project>] <domain>
```

Create the specified DNS domain.

Positional arguments

<domain>

DNS domain

Optional arguments

--project <project>

Limit access to this domain to users of the specified project.

41. nova dns-delete command

```
usage: nova dns-delete <domain> <name>
```

Delete the specified DNS entry.

Positional arguments

<domain>

DNS domain

<name>

DNS name

42. nova dns-delete-domain command

```
usage: nova dns-delete-domain <domain>
```

Delete the specified DNS domain.

Positional arguments

<domain>

DNS domain

43. nova dns-domains command

```
usage: nova dns-domains
```

Print a list of available dns domains.

44. nova dns-list command

```
usage: nova dns-list [--ip <ip>] [--name <name>] <domain>
```

List current DNS entries for domain and ip or domain and name.

Positional arguments

<domain>

DNS domain

Optional arguments

--ip <ip>

ip address

--name <name> DNS

name

45. nova endpoints command

```
usage: nova endpoints
```

Discover endpoints that get returned from the authenticate services.

46. nova evacuate command

```
usage: nova evacuate [--password <password>] [--on-shared-storage]
<server> <host>
```

Evacuate server from failed host to specified one.

Positional arguments

<server>

Name or ID of server.

<host>

Name or ID of target host.

Optional arguments

--password <password>

Set the provided password on the evacuated server. Not applicable with on-shared-storage flag

--on-shared-storage

Specifies whether server files are located on shared storage

47. nova fixed-ip-get command

```
usage: nova fixed-ip-get <fixed_ip>
```

Retrieve info on a fixed ip.

Positional arguments

<fixed_ip>

Fixed IP Address.

48. nova fixed-ip-reserve command

```
usage: nova fixed-ip-reserve <fixed_ip>
```

Reserve a fixed IP.

Positional arguments

<fixed_ip>

Fixed IP Address.

49. nova fixed-ip-unreserve command

```
usage: nova fixed-ip-unreserve <fixed_ip>
```

Unreserve a fixed IP.

Positional arguments

<fixed_ip>

Fixed IP Address.

50. nova flavor-access-add command

```
usage: nova flavor-access-add <flavor> <tenant_id>
```

Add flavor access for the given tenant.

Positional arguments

<flavor>

Flavor name or ID to add access for the given tenant.

<tenant_id>

Tenant ID to add flavor access for.

51. nova flavor-access-list command

```
usage: nova flavor-access-list [--flavor <flavor>] [--tenant <tenant_id>]
```

Print access information about the given flavor.

Optional arguments

--flavor <flavor>

Filter results by flavor name or ID.

--tenant <tenant_id>

Filter results by tenant ID.

52. nova flavor-access-remove command

```
usage: nova flavor-access-remove <flavor> <tenant_id>
```

Remove flavor access for the given tenant.

Positional arguments

<flavor>

Flavor name or ID to remove access for the given tenant.

<tenant_id>

Tenant ID to remove flavor access for.

53. nova flavor-create command

53. nova flavor-create command

```
usage: nova flavor-create [--ephemeral <ephemeral>] [--swap <swap>]
[--rxtx-factor <factor>] [--is-public <is-public>] <name> <id> <ram>
<disk> <vcpus>
```

Create a new flavor

Positional arguments

<name>

Name of the new flavor

<id>

Unique ID (integer or UUID) for the new flavor. If specifying 'auto', a UUID will be generated as id

<ram>

Memory size in MB

<disk>

Disk size in GB

<vcpus>

Number of vcpus

Optional arguments

--ephemeral <ephemeral>

Ephemeral space size in GB (default 0)

--swap <swap>

Swap space size in MB (default 0)

--rxtx-factor <factor>

RX/TX factor (default 1)

--is-public <is-public>

Make flavor accessible to the public (default true)

54. nova flavor-delete command

```
usage: nova flavor-delete <flavor>
```

Delete a specific flavor

Positional arguments

<flavor>

Name or ID of the flavor to delete

55. nova flavor-key command

```
usage: nova flavor-key <flavor> <action> <key=value> [<key=value>
...]
```

Set or unset extra_spec for a flavor.

Positional arguments

<flavor>

Name or ID of flavor

<action>

Actions: 'set' or 'unset'

<key=value>

Extra_specs to set/unset (only key is necessary on unset)

56. nova flavor-list command

```
usage: nova flavor-list [--extra-specs] [--all]
```

Print a list of available 'flavors' (sizes of servers).

Optional arguments

--extra-specs

Get extra-specs of each flavor.

--all

Display all flavors (Admin only).

57. nova flavor-show command

```
usage: nova flavor-show <flavor>
```

Show details about the given flavor.

Positional arguments

<flavor>

Name or ID of flavor

58. nova floating-ip associate command

58. nova floating-ip-associate command

```
usage: nova floating-ip-associate [--fixed-address <fixed_address>]
<server> <address>
```

Associate a floating IP address to a server.

Positional arguments

<server>

Name or ID of server.

<address>

IP Address.

Optional arguments

--fixed-address <fixed_address>

Fixed IP Address to associate with.

59. nova floating-ip-bulk-create command

```
usage: nova floating-ip-bulk-create [--pool <pool>] [--interface
<interface>] <range>
```

Bulk create floating ips by range.

Positional arguments

<range>

Address range to create

Optional arguments

--pool <pool>

Pool for new Floating IPs

--interface <interface>

Interface for new Floating IPs

60. nova floating-ip-bulk-delete command

```
usage: nova floating-ip-bulk-delete <range>
```

Bulk delete floating ips by range.

Positional arguments

<range>

Address range to delete

61. nova floating-ip-bulk-list command

```
usage: nova floating-ip-bulk-list [--host <host>]
```

List all floating ips.

Optional arguments

--host <host>

Filter by host

62. nova floating-ip-create command

```
usage: nova floating-ip-create [<floating-ip-pool>]
```

Allocate a floating IP for the current tenant.

Positional arguments

<floating-ip-pool>

Name of Floating IP Pool. (Optional)

63. nova floating-ip-delete command

```
usage: nova floating-ip-delete <address>
```

De-allocate a floating IP.

Positional arguments

<address>

IP of Floating Ip.

64. nova floating-ip-disassociate command

```
usage: nova floating-ip-disassociate <server> <address>
```

Disassociate a floating IP address from a server.

Positional arguments

<server>

Name or ID of server.

<address>

IP Address.

65. nova floating-ip-list command

```
usage: nova floating-ip-list
```

List floating ips for this tenant.

66. nova floating-ip-pool-list command

```
usage: nova floating-ip-pool-list
```

List all floating ip pools.

67. nova force-delete command

```
usage: nova force-delete <server>
```

Force delete a server.

Positional arguments

<server>

Name or ID of server.

68. nova get-password command

```
usage: nova get-password <server> [<private-key>]
```

Get password for a server.

Positional arguments

<server>

Name or ID of server.

<private-key>

Private key (used locally to decrypt password) (Optional). When specified, the command displays the clear (decrypted) VM password. When not specified, the ciphered VM password is displayed.

69. nova get-rdp-console command

```
usage: nova get-rdp-console <server> <console-type>
```

Get a rdp console to a server.

Positional arguments

<server>

Name or ID of server.

<console-type>

Type of rdp console ("rdp-html5").

70. nova get-spice-console command

```
usage: nova get-spice-console <server> <console-type>
```

Get a spice console to a server.

Positional arguments

<server>

Name or ID of server.

<console-type>

Type of spice console ("spice-html5").

71. nova get-vnc-console command

```
usage: nova get-vnc-console <server> <console-type>
```

Get a vnc console to a server.

Positional arguments

<server>

Name or ID of server.

<console-type>

Type of vnc console ("novnc" or "xvpvnc").

72. nova host-action command

```
usage: nova host-action [--action <action>] <hostname>
```

Perform a power action on a host.

Positional arguments

<hostname>

Name of host.

Optional arguments

--action <action> A

power action: startup, reboot, or shutdown.

73. nova host-describe command

usage: nova host-describe <hostname>

Describe a specific host.

Positional arguments

<hostname>

Name of host.

74. nova host-evacuate command

usage: nova host-evacuate [--target_host <target_host>] [--on-shared-storage] <host>

Evacuate all instances from failed host to specified one.

Positional arguments

<host>

Name of host.

Optional arguments

--target_host <target_host>

Name of target host.

--on-shared-storage

Specifies whether all instances files are on shared storage

75. nova host-list command

usage: nova host-list [--zone <zone>]

List all hosts by service.

Optional arguments

--zone <zone>

Filters the list, returning only those hosts in the availability zone <zone>.

76. nova host-meta command

usage: nova host-meta <host> <action> <key=value> [<key=value> ...]

Set or Delete metadata on all instances of a host.

Positional arguments

<host>

Name of host.

<action>

Actions: 'set' or 'delete'

<key=value>

Metadata to set or delete (only key is necessary on delete)

77. nova host-servers-migrate command

usage: nova host-servers-migrate <host>

Migrate all instances of the specified host to other available hosts.

Positional arguments

<host>

Name of host.

78. nova host-update command

usage: nova host-update [--status <enable|disable>] [--maintenance <enable|disable>] <hostname>

Update host settings.

Positional arguments

<hostname>

Name of host.

Optional arguments

--status <enable|disable>

Either enable or disable a host.

--maintenance <enable|disable>

Either put or resume host to/from maintenance.

79. nova hypervisor-list command

usage: nova hypervisor-list [--matching <hostname>]

List hypervisors.

Optional arguments

--matching <hostname>

List hypervisors matching the given <hostname>.

80. nova hypervisor-servers command

usage: nova hypervisor-servers <hostname>

List servers belonging to specific hypervisors.

Positional arguments

<hostname>

The hypervisor hostname (or pattern) to search for.

81. nova hypervisor-show command

usage: nova hypervisor-show <hypervisor>

Display the details of the specified hypervisor.

Positional arguments

<hypervisor>

Name or ID of the hypervisor to show the details of.

82. nova hypervisor-stats command

usage: nova hypervisor-stats

Get hypervisor statistics over all compute nodes.

83. nova hypervisor-uptime command


```
usage: nova hypervisor-uptime <hypervisor>
```

Display the uptime of the specified hypervisor.

Positional arguments

<hypervisor>

Name or ID of the hypervisor to show the uptime of.

84. nova image-create command

```
usage: nova image-create [--show] [--poll] <server> <name>
```

Create a new image by taking a snapshot of a running server.

Positional arguments

<server>

Name or ID of server.

<name>

Name of snapshot.

Optional arguments

--show

Print image info.

--poll

Blocks while server snapshots so progress can be reported.

85. nova image-delete command

```
usage: nova image-delete <image> [<image> ...]
```

Delete specified image(s).

Positional arguments

<image>

Name or ID of image(s).

86. nova image-list command

```
usage: nova image-list [--limit <limit>]
```

Print a list of available images to boot from.

Optional arguments

--limit <limit>

number of images to return per request

87. nova image-meta command

```
usage: nova image-meta <image> <action> <key=value> [<key=value>
...]
```

Set or Delete metadata on an image.

Positional arguments

<image>

Name or ID of image

<action>

Actions: 'set' or 'delete'

<key=value>

Metadata to add/update or delete (only key is necessary on delete)

88. nova image-show command

```
usage: nova image-show <image>
```

Show details about the given image.

Positional arguments

<image>

Name or ID of image

89. nova instance-action command

```
usage: nova instance-action <server> <request_id>
```

Show an action.

Positional arguments

<server>

Name or UUID of the server to show an action for.

<request_id>

Request ID of the action to get.

90. nova instance-action-list command

```
usage: nova instance-action-list <server>
```

List actions on a server.

Positional arguments

<server>

Name or UUID of the server to list actions for.

91. nova interface-attach command

```
usage: nova interface-attach [--port-id <port_id>] [--net-id  
<net_id>] [--fixed-ip <fixed_ip>] <server>
```

Attach a network interface to a server.

Positional arguments

<server>

Name or ID of server.

Optional arguments

--port-id <port_id>

Port ID.

--net-id <net_id>

Network ID

--fixed-ip <fixed_ip>

Requested fixed IP.

92. nova interface-detach command

```
usage: nova interface-detach <server> <port_id>
```

Detach a network interface from a server.

Positional arguments

<server>

Name or ID of server.

<port_id>

Port ID.

93. nova interface-list command

```
usage: nova interface-list <server>
```

List interfaces attached to a server.

Positional arguments

<server>

Name or ID of server.

94. nova keypair-add command

```
usage: nova keypair-add [--pub-key <pub-key>] <name>
```

Create a new key pair for use with servers.

Positional arguments

<name>

Name of key.

Optional arguments

--pub-key <pub-key>

Path to a public ssh key.

95. nova keypair-delete command

```
usage: nova keypair-delete <name>
```

Delete keypair given by its name.

Positional arguments

<name>

Keypair name to delete.

96. nova keypair-list command

```
usage: nova keypair-list
```

Print a list of keypairs for a user

97. nova keypair-show command

```
usage: nova keypair-show <keypair>
```

Show details about the given keypair.

Positional arguments

<keypair>

Name or ID of keypair

98. nova list command

```
usage: nova list [--reservation-id <reservation-id>] [--ip <ip-regex>]
               [--ip6 <ip6-regex>] [--name <name-regex>] [--instance-name
               <name-regex>] [--status <status>] [--flavor <flavor>] [--image
               <image>] [--host <hostname>] [--all-tenants [<0|1>]] [--tenant
               <tenant>]] [--deleted] [--fields <fields>] [--minimal]
```

List active servers.

Optional arguments

--reservation-id <reservation-id>

Only return servers that match reservation-id.

--ip <ip-regex>

Search with regular expression match by IP address (Admin only).

--ip6 <ip6-regex>

Search with regular expression match by IPv6 address (Admin only).

--name <name-regex>

Search with regular expression match by name

--instance-name <name-regex>

Search with regular expression match by server name (Admin only).

--status <status>

Search by server status

--flavor <flavor>

Search by flavor name or ID

--image <image>

Search by image name or ID

--host <hostname>

Search servers by hostname to which they are assigned (Admin only).

--all-tenants [<0|1>]

Display information from all tenants (Admin only).

--tenant [<tenant>]

Display information from single tenant (Admin only).

--deleted

Only display deleted servers (Admin only).

--fields <fields>

Comma-separated list of fields to display. Use the show command to see which fields are available.

--minimal

Get only uuid and name.

99. nova list-extensions command

usage: nova list-extensions

List all the os-api extensions that are available.

100. nova list-secgroup command

usage: nova list-secgroup <server>

List Security Group(s) of a server.

Positional arguments

<server>

Name or ID of server.

101. nova live-migration command

usage: nova live-migration [--block-migrate] [--disk-over-commit]
<server> [<host>]

Migrate running server to a new machine.

Positional arguments

<server>

Name or ID of server.

<host>

destination host name.

Optional arguments**--block-migrate**

True in case of block_migration. (Default=False:live_migration)

--disk-over-commit

Allow overcommit.(Default=False)

102. nova lock command**usage:** nova lock <server>

Lock a server.

Positional arguments**<server>**

Name or ID of server.

103. nova meta command**usage:** nova meta <server> <action> <key=value> [<key=value> ...]

Set or Delete metadata on a server.

Positional arguments**<server>**

Name or ID of server

<action>

Actions: 'set' or 'delete'

<key=value>

Metadata to set or delete (only key is necessary on delete)

104. nova migrate command**usage:** nova migrate [--poll] <server>

Migrate a server. The new host will be selected by the scheduler.

Positional arguments

<server>

Name or ID of server.

Optional arguments

--poll

Blocks while server migrates so progress can be reported.

105. nova migration-list command

```
usage: nova migration-list [--host <host>] [--status <status>] [--  
cell_name <cell_name>]
```

Print a list of migrations.

Optional arguments

--host <host>

Fetch migrations for the given host.

--status <status>

Fetch migrations for the given status.

--cell_name <cell_name>

Fetch migrations for the given cell_name.

106. nova net command

```
usage: nova net <network_id>
```

Show a network

Positional arguments

<network_id>

ID of network

107. nova net-create command

```
usage: nova net-create <network_label> <cidr>
```


Create a network

Positional arguments**<network_label>**

Network label (ex. my_new_network)

<cidr>


IP block to allocate from (ex. 172.16.0.0/24 or 2001:DB8::/64)

108. nova net-delete command usage: nova net-delete <network_id>


Delete a network

Positional arguments**<network_id>**

ID of network

109. nova net-list command usage: nova net-list

List networks

110. nova network-associate-host command usage: nova network-associate-host <network> <host>


Associate host with network.

Positional arguments**<network>**

uuid of network

<host>

Name of host

111. nova network-associate-project command usage: nova network-associate-project <network>

Associate project with network.

Positional arguments

<network>

uuid of network

112. nova network-create command

```
usage: nova network-create [--fixed-range-v4 <x.x.x.x/yy>] [--fixed-range-v6 CIDR_V6] [--vlan <vlan id>] [--vpn <vpn start>] [--gateway GATEWAY] [--gateway-v6 GATEWAY_V6] [--bridge <bridge>] [--bridge-interface <bridge interface>] [--multi-host <'T' | 'F'>] [--dns1 <DNS Address>] [--dns2 <DNS Address>] [--uuid <network uuid>] [--fixed-cidr <x.x.x.x/yy>] [--project-id <project id>] [--priority <number>] <network_label>
```

Create a network.

Positional arguments

<network_label>

Label for network

Optional arguments

--fixed-range-v4 <x.x.x.x/yy>

IPv4 subnet (ex: 10.0.0.0/8)

--fixed-range-v6

CIDR_V6 IPv6 subnet (ex: fe80::/64)

--vlan <vlan id>

vlan id

--vpn <vpn start>

vpn start

--gateway GATEWAY

gateway

--gateway-v6

GATEWAY_V6 ipv6 gateway

--bridge <bridge>

VIFs on this network are connected to this bridge

--bridge-interface <bridge interface>

the bridge is connected to this interface

--multi-host <'T' | 'F'>

Multi host

--dns1 <DNS Address>

First DNS

--dns2 <DNS Address>

Second DNS

--uuid <network uuid>

Network UUID

--fixed-cidr <x.x.x.x/yy>

IPv4 subnet for fixed IPS (ex: 10.20.0.0/16)

--project-id <project id>

Project id

--priority <number>

Network interface priority

113. nova network-disassociate command

```
usage: nova network-disassociate [--host-only [<0|1>]] [--project-
only [<0|1>]] <network>
```

Disassociate host and/or project from the given network.

Positional arguments

<network>

uuid of network

Optional arguments

--host-only [<0|1>]

--project-only [<0|1>]

114. nova network-list command

```
usage: nova network-list
```

Print a list of available networks.

115. nova network-show command

```
usage: nova network-show <network>
```

Show details about the given network.

Positional arguments

<network>

uuid or label of network

116. nova pause command

```
usage: nova pause <server>
```

Pause a server.

Positional arguments

<server>

Name or ID of server.

117. nova quota-class-show command

```
usage: nova quota-class-show <class>
```

List the quotas for a quota class.

Positional arguments

<class>

Name of quota class to list the quotas for.

118. nova quota-class-update command

```
usage: nova quota-class-update [--instances <instances>] [--cores  
<cores>] [--ram <ram>] [--floating-ips <floating-ips>] [--metadata-  
items <metadata-items>] [--injected-files <injected-files>] [--  
injected-file-content-bytes <injected-file-content-bytes>] [--  
injected-file-path-bytes <injected-file-path-bytes>] [--key-pairs  
<key-pairs>] [--security-groups <security-groups>] [--security-  
group-rules <security-group-rules>] <class>
```

Update the quotas for a quota class.

Positional arguments

<class>

Name of quota class to set the quotas for.

Optional arguments

--instances <instances>

New value for the "instances" quota.

--cores <cores>

New value for the "cores" quota.

--ram <ram>

New value for the "ram" quota.

--floating-ips <floating-ips>

New value for the "floating-ips" quota.

--metadata-items <metadata-items>

New value for the "metadata-items" quota.

--injected-files <injected-files>

New value for the "injected-files" quota.

--injected-file-content-bytes <injected-file-content-bytes>

New value for the "injected-file-content-bytes" quota.

--injected-file-path-bytes <injected-file-path-bytes>

New value for the "injected-file-path-bytes" quota.

--key-pairs <key-pairs>

New value for the "key-pairs" quota.

--security-groups <security-groups>

New value for the "security-groups" quota.

--security-group-rules <security-group-rules>

New value for the "security-group-rules" quota.

119. nova quota-defaults command

```
usage: nova quota-defaults [--tenant <tenant-id>]
```

List the default quotas for a tenant.

Optional arguments

--tenant <tenant-id> ID

of tenant to list the default quotas for.

120. nova quota-delete command

```
usage: nova quota-delete [--tenant <tenant-id>] [--user <user-id>]
```

Delete quota for a tenant/user so their quota will Revert back to default.

Optional arguments

--tenant <tenant-id> ID

of tenant to delete quota for.

--user <user-id> ID

of user to delete quota for.

121. nova quota-show command

```
usage: nova quota-show [--tenant <tenant-id>] [--user <user-id>]
```

List the quotas for a tenant/user.

Optional arguments

--tenant <tenant-id> ID

of tenant to list the quotas for.

--user <user-id> ID

of user to list the quotas for.

122. nova quota-update command

```
usage: nova quota-update [--user <user-id>] [--instances
<instances>] [--cores <cores>] [--ram <ram>] [--floating-ips
<floating-ips>] [--fixed-ips <fixed-ips>] [--metadata-items
<metadata-items>] [--injected-files <injected-files>] [--injected-
file-content-bytes <injected-file-content-bytes>] [--injected-file-
path-bytes <injected-file-path-bytes>] [--key-pairs <key-pairs>] [--
security-groups <security-groups>] [--security-group-rules
<security-group-rules>] [--force] <tenant-id>
```

Update the quotas for a tenant/user.

Positional arguments

<tenant-id>

ID of tenant to set the quotas for.

Optional arguments

--user <user-id> ID

of user to set the quotas for.

--instances <instances>

New value for the "instances" quota.

--cores <cores>

New value for the "cores" quota.

--ram <ram>

New value for the "ram" quota.

--floating-ips <floating-ips>

New value for the "floating-ips" quota.

--fixed-ips <fixed-ips>

New value for the "fixed-ips" quota.

--metadata-items <metadata-items>

New value for the "metadata-items" quota.

--injected-files <injected-files>

New value for the "injected-files" quota.

--injected-file-content-bytes <injected-file-content-bytes>

New value for the "injected-file-content-bytes" quota.

--injected-file-path-bytes <injected-file-path-bytes>

New value for the "injected-file-path-bytes" quota.

--key-pairs <key-pairs>

New value for the "key-pairs" quota.

--security-groups <security-groups>

New value for the "security-groups" quota.

--security-group-rules <security-group-rules>

New value for the "security-group-rules" quota.

--force

Whether force update the quota even if the already used and reserved exceeds the new quota

123. nova rate-limits command

usage: nova rate-limits

Print a list of rate limits for a user

124. nova reboot command

```
usage: nova reboot [--hard] [--poll] <server>
```

Reboot a server.

Positional arguments

<server>

Name or ID of server.

Optional arguments

--hard

Perform a hard reboot (instead of a soft one).

--poll

Blocks while server is rebooting.

125. nova rebuild command

```
usage: nova rebuild [--rebuild-password <rebuild-password>] [--poll]
  [--minimal] [--preserve-ephemeral] <server> <image>
```

Shutdown, re-image, and re-boot a server.

Positional arguments

<server>

Name or ID of server.

<image>

Name or ID of new image.

Optional arguments

--rebuild-password <rebuild-password>

Set the provided password on the rebuild server.

--poll

Blocks while server rebuilds so progress can be reported.

--minimal

Skips flavor/image lookups when showing servers

--preserve-ephemeral

Preserve the default ephemeral storage partition on rebuild.

126. nova refresh-network command


```
usage: nova refresh-network <server>
```

Refresh server network information.

Positional arguments

<server>

Name or ID of a server for which the network cache should be refreshed from neutron (Admin only).

127. nova remove-fixed-ip command

```
usage: nova remove-fixed-ip <server> <address>
```

Remove an IP address from a server.

Positional arguments

<server>

Name or ID of server.

<address>

IP Address.

128. nova remove-secgroup command

```
usage: nova remove-secgroup <server> <secgroup>
```

Remove a Security Group from a server.

Positional arguments

<server>

Name or ID of server.

<secgroup>

Name of Security Group.

129. nova rename command

```
usage: nova rename <server> <name>
```

Rename a server.

Positional arguments

<server>

Name (old name) or ID of server.

<name>

New name for the server.

130. nova rescue command

usage: nova rescue <server>

Rescue a server.

Positional arguments

<server>

Name or ID of server.

131. nova reset-network command

usage: nova reset-network <server>

Reset network of a server.

Positional arguments

<server>

Name or ID of server.

132. nova reset-state command

usage: nova reset-state [--active] <server>

Reset the state of a server.

Positional arguments

<server>

Name or ID of server.

Optional arguments

--active

Request the server be reset to "active" state instead of "error" state (the default).

133. nova resize command

-

```
usage: nova resize [--poll] <server> <flavor>
```

Resize a server.

Positional arguments

<server>

Name or ID of server.

<flavor>

Name or ID of new flavor.

Optional arguments

--poll

Blocks while servers resizes so progress can be reported.

134. nova resize-confirm command

```
usage: nova resize-confirm <server>
```

Confirm a previous resize.

Positional arguments

<server>

Name or ID of server.

135. nova resize-revert command

```
usage: nova resize-revert <server>
```

Revert a previous resize (and return to the previous VM).

Positional arguments

<server>

Name or ID of server.

136. nova restore command

```
usage: nova restore <server>
```

Restore a soft-deleted server.

Positional arguments

<server>

Name or ID of server.

137. nova resume command

usage: nova resume <server>

Resume a server.

Positional arguments

<server>

Name or ID of server.

138. nova root-password command

usage: nova root-password <server>

Change the root password for a server.

Positional arguments

<server>

Name or ID of server.

139. nova scrub command

usage: nova scrub <project_id>

Delete data associated with the project.

Positional arguments

<project_id>

The ID of the project.

140. nova secgroup-add-group-rule command

usage: nova secgroup-add-group-rule <secgroup> <source-group> <ip-proto> <from-port> <to-port>

Add a source group rule to a security group.

Positional arguments

<secgroup>

ID or name of security group.

<source-group>

ID or name of source group.

<ip-proto>

IP protocol (icmp, tcp, udp).

<from-port>

Port at start of range.

<to-port>

Port at end of range.

141. nova secgroup-add-rule command

```
usage: nova secgroup-add-rule <secgroup> <ip-proto> <from-port>  
<to-port> <cidr>
```

Add a rule to a security group.

Positional arguments

<secgroup>

ID or name of security group.

<ip-proto>

IP protocol (icmp, tcp, udp).

<from-port>

Port at start of range.

<to-port>

Port at end of range.

<cidr>

CIDR for address range.

142. nova secgroup-create command

```
usage: nova secgroup-create <name> <description>
```

Create a security group.

Positional arguments

<name>

Name of security group.

<description>

Description of security group.

143. nova secgroup-delete command

```
usage: nova secgroup-delete <secgroup>
```

Delete a security group.

Positional arguments

<secgroup>

ID or name of security group.

144. nova secgroup-delete-group-rule command

```
usage: nova secgroup-delete-group-rule <secgroup> <source-group>
<ip-PROTO> <from-port> <to-port>
```

Delete a source group rule from a security group.

Positional arguments

<secgroup>

ID or name of security group.

<source-group>

ID or name of source group.

<ip-PROTO>

IP protocol (icmp, tcp, udp).

<from-port>

Port at start of range.

<to-port>

Port at end of range.

145. nova secgroup-delete-rule command

```
usage: nova secgroup-delete-rule <secgroup> <ip-PROTO> <from-port>
<to-port> <cidr>
```

Delete a rule from a security group.

Positional arguments**<secgroup>**

ID or name of security group.

<ip-proto>

IP protocol (icmp, tcp, udp).

<from-port>

Port at start of range.

<to-port>

Port at end of range.

<cidr>

CIDR for address range.

146. nova secgroup-list command

```
usage: nova secgroup-list [--all-tenants [<0|1>]]
```

List security groups for the current tenant.

Optional arguments**--all-tenants [<0|1>]**

Display information from all tenants (Admin only).

147. nova secgroup-list-rules command

```
usage: nova secgroup-list-rules <secgroup>
```

List rules for a security group.

Positional arguments**<secgroup>**

ID or name of security group.

148. nova secgroup-update command

```
usage: nova secgroup-update <secgroup> <name> <description>
```

Update a security group.

Positional arguments

<secgroup>

ID or name of security group.

<name>

Name of security group.

<description>

Description of security group.

149. nova server-group-create

```
usage: nova server-group-create [--policy <policy>] <name>
```

Create a new server group with the specified details

Positional arguments

<name>

Server group name.

Optional arguments

--policy <policy>

Policies for the server groups.

150. nova server-group-delete

```
usage: nova server-group-delete <id> [<id>]
```

Delete the specified server group or groups.

Positional arguments

<id>

Unique ID(s) of the server group to delete.

151. nova server-group-get

```
usage: nova server-group-get <id>
```

Get details of the specified server group.

Positional arguments

<id>

Unique ID of the server group to get.

152. nova server-group-list

```
usage: nova server-group-list
```

Print a list of all server groups.

153. nova service-disable command

```
usage: nova service-disable [--reason <reason>] <hostname> <binary>
```

Disable the service.

Positional arguments

<hostname>

Name of host.

<binary>

Service binary.

Optional arguments

--reason <reason>

Reason for disabling service.

154. nova service-enable command

```
usage: nova service-enable <hostname> <binary>
```

Enable the service.

Positional arguments

<hostname>

Name of host.

<binary>

Service binary.

155. nova service-list command

```
usage: nova service-list [--host <hostname>] [--binary <binary>]
```

Show a list of all running services. Filter by host & binary.

Optional arguments

--host <hostname>

Name of host.

--binary <binary>

Service binary.

156. nova shelve command

usage: nova shelve <server>

Shelve a server.

Positional arguments

<server>

Name or ID of server.

157. nova shelve-offload command

usage: nova shelve-offload <server>

Remove a shelved server from the compute node.

Positional arguments

<server>

Name or ID of server.

158. nova show command

usage: nova show [--minimal] <server>

Show details about the given server.

Positional arguments

<server>

Name or ID of server.

Optional arguments

--minimal

Skips flavor/image lookups when showing servers

159. nova ssh command

-

```
usage: nova ssh [--port PORT] [--private] [--ipv6] [--login <login>]
[-i IDENTITY] [--extra-opts EXTRA] <server>
```

SSH into a server.

Positional arguments

<server>

Name or ID of server.

Optional arguments

--port **PORT**

Optional flag to indicate which port to use for ssh. (Default=22)

--private

Optional flag to indicate whether to only use private address attached to an instance. (Default=False). If no public address is found try private address

--ipv6

Optional flag to indicate whether to use an IPv6 address attached to a server. (Defaults to IPv4 address)

--login **<login>**

Login to use.

-i **IDENTITY**, --identity **IDENTITY**

Private key file, same as the -i option to the ssh command.

--extra-opts **EXTRA**

Extra options to pass to ssh. see: man ssh

160. nova start command

```
usage: nova start <server>
```

Start a server.

Positional arguments

<server>

Name or ID of server.

161. nova stop command

```
usage: nova stop <server>
```

Stop a server.

Positional arguments

<server>

Name or ID of server.

162. nova suspend command

usage: nova suspend <server>

Suspend a server.

Positional arguments

<server>

Name or ID of server.

163. nova unlock command

usage: nova unlock <server>

Unlock a server.

Positional arguments

<server>

Name or ID of server.

164. nova unpause command

usage: nova unpause <server>

Unpause a server.

Positional arguments

<server>

Name or ID of server.

165. nova unrescue command

usage: nova unrescue <server>

Unrescue a server.

Positional arguments

<server>

Name or ID of server.

166. nova unshelve command

```
usage: nova unshelve <server>
```

Unshelve a server.

Positional arguments

<server>

Name or ID of server.

167. nova usage command

```
usage: nova usage [--start <start>] [--end <end>] [--tenant <tenant-id>]
```

Show usage data for a single tenant.

Optional arguments

--start <start>

Usage range start date ex 2012-01-20 (default: 4 weeks ago)

--end <end>

Usage range end date, ex 2012-01-20 (default: tomorrow)

--tenant <tenant-id> UUID

or name of tenant to get usage for.

168. nova usage-list command

```
usage: nova usage-list [--start <start>] [--end <end>]
```

List usage data for all tenants.

Optional arguments

--start <start>

Usage range start date ex 2012-01-20 (default: 4 weeks ago)

--end <end>

Usage range end date, ex 2012-01-20 (default: tomorrow)

169. nova volume-attach command

```
usage: nova volume-attach <server> <volume> [<device>]
```

Attach a volume to a server.

Positional arguments

<server>

Name or ID of server.

<volume>

ID of the volume to attach.

<device>

Name of the device e.g. /dev/vdb. Use "auto" for autoassign (if supported)

170. nova volume-create command

```
usage: nova volume-create [--snapshot-id <snapshot-id>] [--image-id  
<image-id>] [--display-name <display-name>] [--display-description  
<display-description>] [--volume-type <volume-type>] [--  
availability-zone <availability-zone>] <size>
```

Add a new volume.

Positional arguments

<size>

Size of volume in GB

Optional arguments

--snapshot-id <snapshot-id>

Optional snapshot id to create the volume from. (Default=None)

--image-id <image-id>

Optional image id to create the volume from. (Default=None)

--display-name <display-name>

Optional volume name. (Default=None)

--display-description <display-description>

Optional volume description. (Default=None)

--volume-type <volume-type>

Optional volume type. (Default=None)

--availability-zone <availability-zone>

Optional Availability Zone for volume. (Default=None)

171. nova volume-delete command

usage: nova volume-delete <volume> [<volume> ...]

Remove volume(s).

Positional arguments

<volume>

Name or ID of the volume(s) to delete.

172. nova volume-detach command

usage: nova volume-detach <server> <volume>

Detach a volume from a server.

Positional arguments

<server>

Name or ID of server.

<volume>

Attachment ID of the volume.

173. nova volume-list command

usage: nova volume-list [--all-tenants [<0|1>]]

List all the volumes.

Optional arguments

--all-tenants [<0|1>]

Display information from all tenants (Admin only).

174. nova volume-show command

usage: nova volume-show <volume>

Show details about a volume.

Positional arguments

<volume>

Name or ID of the volume.

175. nova volume-snapshot-create command

```
usage: nova volume-snapshot-create [--force <True|False>] [--display-name <display-name>] [--display-description <display-description>] <volume-id>
```

Add a new snapshot.

Positional arguments

<volume-id>

ID of the volume to snapshot

Optional arguments

--force <True|False>

Optional flag to indicate whether to snapshot a volume even if its attached to a server. (Default=False)

--display-name <display-name>

Optional snapshot name. (Default=None)

--display-description <display-description>

Optional snapshot description. (Default=None)

176. nova volume-snapshot-delete command

```
usage: nova volume-snapshot-delete <snapshot>
```

Remove a snapshot.

Positional arguments

<snapshot>

Name or ID of the snapshot to delete.

177. nova volume-snapshot-list command

```
usage: nova volume-snapshot-list
```

List all the snapshots.

178. nova volume-snapshot-show command


```
usage: nova volume-snapshot-show <snapshot>
```

Show details about a snapshot.

Positional arguments

<snapshot>

Name or ID of the snapshot.

179. nova volume-type-create command

```
usage: nova volume-type-create <name>
```

Create a new volume type.

Positional arguments

<name>

Name of the new flavor

180. nova volume-type-delete command

```
usage: nova volume-type-delete <id>
```

Delete a specific flavor

Positional arguments

<id>

Unique ID of the volume type to delete

181. nova volume-type-list command

```
usage: nova volume-type-list
```

Print a list of available 'volume types'.

182. nova volume-update command

```
usage: nova volume-update <server> <volume> <volume>
```

Update volume attachment.

Positional arguments

<server>

Name or ID of server.

<volume>

Attachment ID of the volume.

<volume>

ID of the volume to attach.

183. nova x509-create-cert command

```
usage: nova x509-create-cert [<private-key-filename>] [<x509-cert-filename>]
```

Create x509 cert for a user in tenant.

Positional arguments

<private-key-filename>

Filename for the private key [Default: pk.pem]

<x509-cert-filename>

Filename for the X.509 certificate [Default: cert.pem]

184. nova x509-get-root-cert command

```
usage: nova x509-get-root-cert [<filename>]
```

Fetch the x509 root cert.

Positional arguments

<filename>

Filename to write the x509 root cert.

Chapter 4. Identity service command-line client

The **keystone** client is the command-line interface (CLI) for the OpenStack Identity API and its extensions. This chapter documents **keystone** version 0.8.0.

For help on a specific **keystone** command, enter:

```
$ keystone help COMMAND
```

1. keystone usage

```
usage: keystone [--version] [--timeout <seconds>] [--os-username
<auth-user-name>] [--os-password <auth-password>] [--os-tenant-name
<auth-tenant-name>] [--os-tenant-id <tenant-id>] [--os-auth-url
<auth-url>] [--os-region-name <region-name>] [--os-identity-api-
version <identity-api-version>] [--os-token <service-token>] [--os-
endpoint <service-endpoint>] [--os-cacert <ca-certificate>] [--
insecure] [--os-cert <certificate>] [--os-key <key>] [--os-cache] [--
force-new-token] [--stale-duration <seconds>] <subcommand> ...
```

Subcommands

catalog

List service catalog, possibly filtered by service.

ec2-credentials-create

Create EC2-compatible credentials for user per tenant.

ec2-credentials-delete

Delete EC2-compatible credentials.

ec2-credentials-get

Display EC2-compatible credentials.

ec2-credentials-list

List EC2-compatible credentials for a user.

endpoint-create

Create a new endpoint associated with a service.

endpoint-delete

Delete a service endpoint.

endpoint-get

Find endpoint filtered by a specific attribute or service type.

endpoint-list

List configured service endpoints.

password -update

Update own password.

role -create

Create new role.

role -delete

Delete role.

role -get

Display role details.

role -list

List all roles.

service -create

Add service to Service Catalog.

service -delete

Delete service from Service Catalog.

service -get

Display service from Service Catalog.

service -list

List all services in Service Catalog.

tenant -create

Create new tenant.

tenant -delete

Delete tenant.

tenant -get

Display tenant details.

tenant -list

List all tenants.

tenant -update

Update tenant name, description, enabled status.

token -get

Display the current user token.

user -create

Create new user

user-delete

Delete user.

user-get

Display user details.

user-list

List users.

user-password -update

Update user password.

user-role-add

Add role to user.

user-role-list

List roles granted to a user.

user-role-remove

Remove role from user.

user-update

Update user's name, email, and enabled status.

discover

Discover Keystone servers, supported API versions and extensions.

bootstrap

Grants a new role to a new user on a new tenant, after creating each.

bash-completion

Prints all of the commands and options to stdout.

help

Display help about this program or one of its subcommands.

2. keystone optional arguments

--version

Shows the client version and exits.

--timeout <seconds>

Set request timeout (in seconds).

--os-username <auth-user-name>

Name used for authentication with the OpenStack Identity service. Defaults to `env[OS_USERNAME]`.

--os-password <auth-password>

Password used for authentication with the OpenStack Identity service. Defaults to **env[OS_PASSWORD]**.

--os-tenant-name <auth-tenant-name>

Tenant to request authorization on. Defaults to **env[OS_TENANT_NAME]**.

--os-tenant-id <tenant-id>

Tenant to request authorization on. Defaults to **env[OS_TENANT_ID]**.

--os-auth-url <auth-url>

Specify the Identity endpoint to use for authentication. Defaults to **env[OS_AUTH_URL]**.

--os-region-name <region-name>

Specify the region to use. Defaults to **env[OS_REGION_NAME]**.

--os-identity-api-version <identity-api-version>

Specify Identity API version to use. Defaults to **env[OS_IDENTITY_API_VERSION]** or 2.0.

--os-token <service-token>

Specify an existing token to use instead of retrieving one via authentication (e.g. with username & password). Defaults to **env[OS_SERVICE_TOKEN]**.

--os-endpoint <service-endpoint>

Specify an endpoint to use instead of retrieving one from the service catalog (via authentication). Defaults to **env[OS_SERVICE_ENDPOINT]**.

--os-cacert <ca-certificate>

Specify a CA bundle file to use in verifying a TLS (https) server certificate. Defaults to **env[OS_CACERT]**.

--insecure

Explicitly allow keystoneclient to perform "insecure" TLS (https) requests. The server's certificate will not be verified against any certificate authorities. This option should be used with caution.

--os-cert <certificate>

Defaults to **env[OS_CERT]**.

--os-key <key>

Defaults to **env[OS_KEY]**.

--os-cache

Use the auth token cache. Defaults to **env[OS_CACHE]**.

--force-new-token

If the keyring is available and in use, token will always be stored and fetched from the keyring until the token has expired. Use this option to request a new token and replace the existing one in the keyring.

--stale-duration <seconds>

Stale duration (in seconds) used to determine whether a token has expired when retrieving it from keyring. This is useful in mitigating process or network delays. Default is 30 seconds.

3. keystone bootstrap command

```
usage: keystone bootstrap [--user-name <user-name>] --pass <password>
      [--role-name <role-name>] [--tenant-name <tenant-name>]
```

Grants a new role to a new user on a new tenant, after creating each.

Arguments

--user-name <user-name>

The name of the user to be created (default="admin").

--pass <password>

The password for the new user.

--role-name <role-name>

The name of the role to be created and granted to the user (default="admin").

--tenant-name <tenant-name>

The name of the tenant to be created (default="admin").

4. keystone catalog command

```
usage: keystone catalog [--service <service-type>]
```

List service catalog, possibly filtered by service.

Arguments

--service <service-type>

Service type to return.

5. keystone discover command

```
usage: keystone discover
```

Discover Keystone servers, supported API versions and extensions.

6. keystone os? credentials create command

6. keystone ec2-credentials-create command

```
usage: keystone ec2-credentials-create [--user-id <user-id>] [--tenant-id <tenant-id>]
```

Create EC2-compatible credentials for user per tenant.

Arguments

--user-id <user-id>

User ID for which to create credentials. If not specified, the authenticated user will be used.

--tenant-id <tenant-id>

Tenant ID for which to create credentials. If not specified, the authenticated tenant ID will be used.

7. keystone ec2-credentials-delete command

```
usage: keystone ec2-credentials-delete [--user-id <user-id>] --access <access-key>
```

Delete EC2-compatible credentials.

Arguments

--user-id <user-id>

User ID.

--access <access-key>

Access Key.

8. keystone ec2-credentials-get command

```
usage: keystone ec2-credentials-get [--user-id <user-id>] --access <access-key>
```

Display EC2-compatible credentials.

Arguments

--user-id <user-id>

User ID.

--access <access-key>

Access Key.

9. keystone ec2-credentials-list command

```
usage: keystone ec2-credentials-list [--user-id <user-id>]
```

List EC2-compatible credentials for a user.

Arguments

--user-id <user-id>

User ID.

10. keystone endpoint-create command

```
usage: keystone endpoint-create [--region <endpoint-region>] --
service <service> --publicurl <public-url> [--adminurl <admin-url>]
[--internalurl <internal-url>]
```

Create a new endpoint associated with a service.

Arguments

--region <endpoint-region>

Endpoint region.

--service <service>, --service-id <service>, --service_id <service>

Name or ID of service associated with endpoint.

--publicurl <public-url>

Public URL endpoint.

--adminurl <admin-url>

Admin URL endpoint.

--internalurl <internal-url>

Internal URL endpoint.

11. keystone endpoint-delete command

```
usage: keystone endpoint-delete <endpoint-id>
```

Delete a service endpoint.

Arguments

<endpoint-id>

ID of endpoint to delete.

12. keystone endpoint-get command

```
usage: keystone endpoint-get --service <service-type> [--endpoint-  
type <endpoint-type>] [--attr <service-attribute>] [--value <value>]
```

Find endpoint filtered by a specific attribute or service type.

Arguments

--service <service-type>

Service type to select.

--endpoint-type <endpoint-type>

Endpoint type to select.

--attr <service-attribute>

Service attribute to match for selection.

--value <value>

Value of attribute to match.

13. keystone endpoint-list command

```
usage: keystone endpoint-list
```

List configured service endpoints.

14. keystone password-update command

```
usage: keystone password-update [--current-password <current-  
password>] [--new-password <new-password>]
```

Update own password.

Arguments

--current-password <current-password>

Current password, Defaults to the password as set by --os-password or **env[OS_PASSWORD]**.

--new-password <new-password>

Desired new password.

15. keystone role-create command

```
usage: keystone role-create --name <role-name>
```

Create new role.

Arguments

--name <role-name>

Name of new role.

16. keystone role-delete command

```
usage: keystone role-delete <role>
```

Delete role.

Arguments

<role>

Name or ID of role to delete.

17. keystone role-get command

```
usage: keystone role-get <role>
```

Display role details.

Arguments

<role>

Name or ID of role to display.

18. keystone role-list command

```
usage: keystone role-list
```

List all roles.

19. keystone service-create command

```
usage: keystone service-create --name <name> --type <type> [--  
description <service-description>]
```

Add service to Service Catalog.

Arguments

--name <name>

Name of new service (must be unique).

--type <type>

Service type (one of: identity, compute, network, image, object-store, or other service identifier string).

--description <service-description>

Description of service.

20. keystone service-delete command

```
usage: keystone service-delete <service>
```

Delete service from Service Catalog.

Arguments

<service>

Name or ID of service to delete.

21. keystone service-get command

```
usage: keystone service-get <service>
```

Display service from Service Catalog.

Arguments

<service>

Name or ID of service to display.

22. keystone service-list command

```
usage: keystone service-list
```

List all services in Service Catalog.

23. keystone tenant-create command

```
usage: keystone tenant-create --name <tenant-name> [--description  
<tenant-description>] [--enabled <true|false>]
```

Create new tenant.

Arguments

--name <tenant-name>

New tenant name (must be unique).

--description <tenant-description>

Description of new tenant. Default is none.

--enabled <true|false>

Initial tenant enabled status. Default is true.

24. keystone tenant-delete command

usage: keystone tenant-delete <tenant>

Delete tenant.

Arguments

<tenant>

Name or ID of tenant to delete.

25. keystone tenant-get command

usage: keystone tenant-get <tenant>

Display tenant details.

Arguments

<tenant>

Name or ID of tenant to display.

26. keystone tenant-list command

usage: keystone tenant-list

List all tenants.

27. keystone tenant-update command

usage: keystone tenant-update [--name <tenant_name>] [--description <tenant-description>] [--enabled <true|false>] <tenant>

Update tenant name, description, enabled status.

Arguments

--name <tenant_name>

Desired new name of tenant.

--description <tenant-description>

Desired new description of tenant.

--enabled <true|false>

Enable or disable tenant.

<tenant>

Name or ID of tenant to update.

28. keystone token-get command

```
usage: keystone token-get [--wrap <integer>]
```

Display the current user token.

Arguments

--wrap <integer>

Wrap PKI tokens to a specified length, or 0 to disable.

29. keystone user-create command

```
usage: keystone user-create --name <user-name> [--tenant <tenant>]
      [--pass [<pass>]] [--email <email>] [--enabled <true|false>]
```

Create new user

Arguments

--name <user-name>

New user name (must be unique).

--tenant <tenant>, --tenant-id <tenant>

New user default tenant.

--pass [<pass>]

New user password; required for some auth backends.

--email <email>

New user email address.

--enabled <true|false>

Initial user enabled status. Default is true.

30. keystone user-delete command

```
usage: keystone user-delete <user>
```

Delete user.

Arguments

<user>

Name or ID of user to delete.

31. keystone user-get command

```
usage: keystone user-get <user>
```

Display user details.

Arguments

<user>

Name or ID of user to display.

32. keystone user-list command

```
usage: keystone user-list [--tenant <tenant>]
```

List users.

Arguments

--tenant <tenant>, --tenant-id <tenant>

Tenant; lists all users if not specified.

33. keystone user-password-update command

```
usage: keystone user-password-update [--pass <password>] <user>
```

Update user password.

Arguments

--pass <password>

Desired new password.

<user>

Name or ID of user to update password.

34. keystone user-role-add command

```
usage: keystone user-role-add --user <user> --role <role> [--tenant <tenant>]
```

Add role to user.

Arguments

--user <user>, --user-id <user>, --user_id <user>

Name or ID of user.

--role <role>, --role-id <role>, --role_id <role>

Name or ID of role.

--tenant <tenant>, --tenant-id <tenant>

Name or ID of tenant.

35. keystone user-role-list command

```
usage: keystone user-role-list [--user <user>] [--tenant <tenant>]
```

List roles granted to a user.

Arguments

--user <user>, --user-id <user>

List roles granted to specified user.

--tenant <tenant>, --tenant-id <tenant>

List only roles granted on specified tenant.

36. keystone user-role-remove command

```
usage: keystone user-role-remove --user <user> --role <role> [--tenant <tenant>]
```

Remove role from user.

Arguments

--user <user>, --user-id <user>, --user_id <user>

Name or ID of user.

--role <role>, --role-id <role>, --role_id <role>

Name or ID of role.

--tenant <tenant>, --tenant-id <tenant>

Name or ID of tenant.

37. keystone user-update command

```
usage: keystone user-update [--name <user-name>] [--email <email>]
      [--enabled <true|false>] <user>
```

Update user's name, email, and enabled status.

Arguments

--name <user-name>

Desired new user name.

--email <email>

Desired new email address.

--enabled <true|false>

Enable or disable user.

<user>

Name or ID of user to update.

Chapter 5. Image Service command-line client

The **glance** client is the command-line interface (CLI) for the OpenStack Image Service API and its extensions. This chapter documents **glance** version 0.12.0.

For help on a specific **glance** command, enter:

```
$ glance help COMMAND
```

1. glance usage

```
usage: glance [--version] [-d] [-v] [--get-schema] [-k] [--cert-file
CERT_FILE] [--key-file KEY_FILE] [--os-cacert <ca-certificate-file>]
[--ca-file OS_CACERT] [--timeout TIMEOUT] [--no-ssl-compression] [-f]
[--dry-run] [--ssl] [-H ADDRESS] [-p PORT] [--os-username
OS_USERNAME] [-I OS_USERNAME] [--os-password OS_PASSWORD] [-K
OS_PASSWORD] [--os-tenant-id OS_TENANT_ID] [--os-tenant-name
OS_TENANT_NAME] [-T OS_TENANT_NAME] [--os-auth-url OS_AUTH_URL] [-N
OS_AUTH_URL] [--os-region-name OS_REGION_NAME] [-R OS_REGION_NAME]
[--os-auth-token OS_AUTH_TOKEN] [-A OS_AUTH_TOKEN] [--os-image-url
OS_IMAGE_URL] [-U OS_IMAGE_URL] [--os-image-api-version
OS_IMAGE_API_VERSION] [--os-service-type OS_SERVICE_TYPE] [--os-
endpoint-type OS_ENDPOINT_TYPE] [-S OS_AUTH_STRATEGY] <subcommand>
...
```

Subcommands

add

DEPRECATED! Use image-create instead.

clear

DEPRECATED!

delete

DEPRECATED! Use image-delete instead.

details

DEPRECATED! Use image-list instead.

image-create

Create a new image.

image-delete

Delete specified image(s).

image-download

Download a specific image.

image-list

List images you can access.

image-members

DEPRECATED! Use member-list instead.

image-show

Describe a specific image.

image-update

Update a specific image.

index

DEPRECATED! Use image-list instead.

member-add

DEPRECATED! Use member-create instead.

member-create

Share a specific image with a tenant.

member-delete

Remove a shared image from a tenant.

member-images

DEPRECATED! Use member-list instead.

member-list

Describe sharing permissions by image or tenant.

members-replace

DEPRECATED!

show

DEPRECATED! Use image-show instead.

update

DEPRECATED! Use image-update instead.

help

Display help about this program or one of its subcommands.

2. glance optional arguments

--version

show program's version number and exit

-d, --debug

Defaults to **env[GLANCECLIENT_DEBUG]**

-v, --verbose

Print more verbose output

--get-schema

Force retrieving the schema used to generate portions of the help text rather than using a cached copy. Ignored with api version 1

-k, --insecure

Explicitly allow glanceclient to perform "insecure SSL" (https) requests. The server's certificate will not be verified against any certificate authorities. This option should be used with caution.

--cert-file CERT_FILE

Path of certificate file to use in SSL connection. This file can optionally be prepended with the private key.

--key-file KEY_FILE

Path of client key to use in SSL connection. This option is not necessary if your key is prepended to your cert file.

--os-cacert <ca-certificate-file>

Path of CA TLS certificate(s) used to verify the remote server's certificate. Without this option glance looks for the default system CA certificates.

--ca-file OS_CACERT

DEPRECATED! Use --os-cacert.

--timeout TIMEOUT

Number of seconds to wait for a response

--no-ssl-compression

Disable SSL compression when using https.

-f, --force

Prevent select actions from requesting user confirmation.

--dry-run

DEPRECATED! Only used for deprecated legacy commands.

--ssl

DEPRECATED! Send a fully-formed endpoint using --os- image-url instead.

-H ADDRESS, --host ADDRESS

DEPRECATED! Send a fully-formed endpoint using --os- image-url instead.

-p PORT, --port PORT

DEPRECATED! Send a fully-formed endpoint using --os- image-url instead.

--os-username OS_USERNAME
Defaults to `env[OS_USERNAME]`

-I OS_USERNAME
DEPRECATED! Use `--os-username`.

--os-password OS_PASSWORD
Defaults to `env[OS_PASSWORD]`

-K OS_PASSWORD
DEPRECATED! Use `--os-password`.

--os-tenant-id OS_TENANT_ID
Defaults to `env[OS_TENANT_ID]`

--os-tenant-name OS_TENANT_NAME
Defaults to `env[OS_TENANT_NAME]`

-T OS_TENANT_NAME
DEPRECATED! Use `--os-tenant-name`.

--os-auth-url OS_AUTH_URL
Defaults to `env[OS_AUTH_URL]`

-N OS_AUTH_URL
DEPRECATED! Use `--os-auth-url`.

--os-region-name OS_REGION_NAME
Defaults to `env[OS_REGION_NAME]`

-R OS_REGION_NAME
DEPRECATED! Use `--os-region-name`.

--os-auth-token OS_AUTH_TOKEN
Defaults to `env[OS_AUTH_TOKEN]`

-A OS_AUTH_TOKEN, --auth-token OS_AUTH_TOKEN
DEPRECATED! Use `--os-auth-token`.

--os-image-url OS_IMAGE_URL
Defaults to `env[OS_IMAGE_URL]`

-U OS_IMAGE_URL, --url OS_IMAGE_URL
DEPRECATED! Use `--os-image-url`.

--os-image-api-version OS_IMAGE_API_VERSION
Defaults to `env[OS_IMAGE_API_VERSION]` or 1

--os-service-type OS_SERVICE_TYPE

Defaults to **env[OS_SERVICE_TYPE]**

--os-endpoint-type OS_ENDPOINT_TYPE

Defaults to **env[OS_ENDPOINT_TYPE]**

-S OS_AUTH_STRATEGY, --os_auth_strategy OS_AUTH_STRATEGY

DEPRECATED! This option is completely ignored.

3. glance image-create command

```
usage: glance image-create [--id <IMAGE_ID>] [--name <NAME>] [--store <STORE>] [--disk-format <DISK_FORMAT>] [--container-format <CONTAINER_FORMAT>] [--owner <TENANT_ID>] [--size <SIZE>] [--min-disk <DISK_GB>] [--min-ram <DISK_RAM>] [--location <IMAGE_URL>] [--file <FILE>] [--checksum <CHECKSUM>] [--copy-from <IMAGE_URL>] [--is-public {True,False}] [--is-protected {True,False}] [--property <key=value>] [--human-readable] [--progress]
```

Create a new image.

Optional arguments

--id <IMAGE_ID> ID

of image to reserve.

--name <NAME>

Name of image.

--store <STORE>

Store to upload image to.

--disk-format <DISK_FORMAT>

Disk format of image. Acceptable formats: ami, ari, aki, vhd, vmdk, raw, qcow2, vdi, and iso.

--container-format <CONTAINER_FORMAT>

Container format of image. Acceptable formats: ami, ari, aki, bare, and ovf.

--owner <TENANT_ID>

Tenant who should own image.

--size <SIZE>

Size of image data (in bytes). Only used with '--location' and '--copy_from'.

--min-disk <DISK_GB>

Minimum size of disk needed to boot image (in gigabytes).

--min-ram <DISK_RAM>

Minimum amount of ram needed to boot image (in megabytes).

--location <IMAGE_URL>

URL where the data for this image already resides. For example, if the image data is stored in swift, you could specify 'swift://account:key@example.com/container/obj'.

--file <FILE>

Local file that contains disk image to be uploaded during creation. Alternatively, images can be passed to the client via stdin.

--checksum <CHECKSUM>

Hash of image data used Glance can use for verification. Provide a md5 checksum here.

--copy-from <IMAGE_URL>

Similar to '--location' in usage, but this indicates that the Glance server should immediately copy the data and store it in its configured image store.

--is-public {True,False}

Make image accessible to the public.

--is-protected {True,False}

Prevent image from being deleted.

--property <key=value>

Arbitrary property to associate with image. May be used multiple times.

--human-readable

Print image size in a human-friendly format.

--progress

Show upload progress bar.

4. glance image-delete command

```
usage: glance image-delete <IMAGE> [<IMAGE> ...]
```

Delete specified image(s).

Positional arguments

<IMAGE>

Name or ID of image(s) to delete.

5. glance image-list command

```
usage: glance image-list [--name <NAME>] [--status <STATUS>] [--
container-format <CONTAINER_FORMAT>] [--disk-format <DISK_FORMAT>]
```

```
[--size-min <SIZE>] [--size-max <SIZE>] [--property-filter  
<KEY=VALUE>] [--page-size <SIZE>] [--human-readable] [--sort-key  
{name,status,container_format,disk_format,size,id,created_at,updated  
_at}] [--sort-dir {asc,desc}] [--is-public {True,False}] [--owner  
<TENANT_ID>] [--all-tenants]
```

List images you can access.

Optional arguments

--name <NAME>

Filter images to those that have this name.

--status <STATUS>

Filter images to those that have this status.

--container-format <CONTAINER_FORMAT>

Filter images to those that have this container format. Acceptable formats: ami, ari, aki, bare, and ovf.

--disk-format <DISK_FORMAT>

Filter images to those that have this disk format. Acceptable formats: ami, ari, aki, vhd, vmdk, raw, qcow2, vdi, and iso.

--size-min <SIZE>

Filter images to those with a size greater than this.

--size-max <SIZE>

Filter images to those with a size less than this.

--property-filter <KEY=VALUE>

Filter images by a user-defined image property.

--page-size <SIZE>

Number of images to request in each paginated request.

--human-readable

Print image size in a human-friendly format.

--sort-key

{name,status,container_format,disk_format,size,id,created_at,updated_at}

Sort image list by specified field.

--sort-dir {asc,desc}

Sort image list in specified direction.

--is-public {True,False}

Allows the user to select a listing of public or non public images.

--owner <TENANT_ID>

Display only images owned by this tenant id. Filtering occurs on the client side so may be inefficient. This option is mainly intended for admin use. Use an empty string ('') to list images with no owner. Note: This option overrides the --is-public argument if present. Note: the v2 API supports more efficient server-side owner based filtering.

--all-tenants

Allows the admin user to list all images irrespective of the image's owner or is_public value.

6. glance image-show command

```
usage: glance image-show [--human-readable] <IMAGE>
```

Describe a specific image.

Positional arguments

<IMAGE>

Name or ID of image to describe.

Optional arguments

--human-readable

Print image size in a human-friendly format.

7. glance image-update command

```
usage: glance image-update [--name <NAME>] [--disk-format
<DISK_FORMAT>] [--container-format <CONTAINER_FORMAT>] [--owner
<TENANT_ID>] [--size <SIZE>] [--min-disk <DISK_GB>] [--min-ram
<DISK_RAM>] [--location <IMAGE_URL>] [--file <FILE>] [--checksum
<CHECKSUM>] [--copy-from <IMAGE_URL>] [--is-public {True,False}] [--
is-protected {True,False}] [--property <key=value>] [--purge-props]
[--human-readable] [--progress] <IMAGE>
```

Update a specific image.

Positional arguments

<IMAGE>

Name or ID of image to modify.

Optional arguments

--name <NAME>

Name of image.

--disk-format <DISK_FORMAT>

Disk format of image. Acceptable formats: ami, ari, aki, vhd, vmdk, raw, qcow2, vdi, and iso.

--container-format <CONTAINER_FORMAT>

Container format of image. Acceptable formats: ami, ari, aki, bare, and ovf.

--owner <TENANT_ID>

Tenant who should own image.

--size <SIZE>

Size of image data (in bytes).

--min-disk <DISK_GB>

Minimum size of disk needed to boot image (in gigabytes).

--min-ram <DISK_RAM>

Minimum amount of ram needed to boot image (in megabytes).

--location <IMAGE_URL>

URL where the data for this image already resides. For example, if the image data is stored in swift, you could specify 'swift://account:key@example.com/container/obj'.

--file <FILE>

Local file that contains disk image to be uploaded during update. Alternatively, images can be passed to the client via stdin.

--checksum <CHECKSUM>

Hash of image data used Glance can use for verification.

--copy-from <IMAGE_URL>

Similar to '--location' in usage, but this indicates that the Glance server should immediately copy the data and store it in its configured image store.

--is-public {True,False}

Make image accessible to the public.

--is-protected {True,False}

Prevent image from being deleted.

--property <key=value>

Arbitrary property to associate with image. May be used multiple times.

--purge-props

If this flag is present, delete all image properties not explicitly set in the update request. Otherwise, those properties not referenced are preserved.

--human-readable

Print image size in a human-friendly format.

--progress

Show upload progress bar.

8. glance member-create command

```
usage: glance member-create [--can-share] <IMAGE> <TENANT_ID>
```

Share a specific image with a tenant.

Positional arguments**<IMAGE>**

Image to add member to.

<TENANT_ID>

Tenant to add as member

Optional arguments**--can-share**

Allow the specified tenant to share this image.

9. glance member-delete command

```
usage: glance member-delete <IMAGE> <TENANT_ID>
```

Remove a shared image from a tenant.

Positional arguments**<IMAGE>**

Image from which to remove member

<TENANT_ID>

Tenant to remove as member

10. glance member-list command

```
usage: glance member-list [--image-id <IMAGE_ID>] [--tenant-id <TENANT_ID>]
```

Describe sharing permissions by image or tenant.

Optional arguments**--image-id <IMAGE_ID>**

Filter results by an image ID.

--tenant-id <TENANT_ID>

Filter results by a tenant ID.

Chapter 6. Image Service property keys

The following keys, together with the components to which they are specific, can be used with the **property** option for both the **glance image-update** and **glance image-create** commands. For example:

```
$ glance image-update IMG-UUID --property architecture=x86_64
```

Note

Behavior set using image properties overrides behavior set using flavors. For more information, refer to the *Red Hat Enterprise Linux OpenStack Platform Cloud Admin Guide*.

Table 6.1. Property keys

Specific to	Key	Description	Supported values
All	architecture	The CPU architecture that must be supported by the hypervisor. For example, x86_64 , arm , or ppc64 . Run uname -m to get the architecture of a machine. We strongly recommend using the architecture data vocabulary defined by the libosinfo project for this purpose.	<ul style="list-style-type: none"> ✦ alpha—DEC 64-bit RISC ✦ armv7l—ARM Cortex-A7 MPCore ✦ cris—Ethernet, Token Ring, AXis—Code Reduced Instruction Set ✦ i686—Intel sixth-generation x86 (P6 micro architecture) ✦ ia64—Itanium ✦ lm32—Lattice Micro32 ✦ m68k—Motorola 68000 ✦ microblaze—Xilinx 32-bit FPGA (Big Endian) ✦ microblazeel—Xilinx 32-bit FPGA (Little Endian) ✦ mips—MIPS 32-bit RISC (Big Endian) ✦ mipsel—MIPS 32-bit RISC (Little Endian) ✦ mips64—MIPS 64-bit RISC (Big Endian) ✦ mips64el—MIPS 64-bit RISC (Little Endian)

Specific to	Key	Description	Supported values
			<ul style="list-style-type: none"> ✦ openrisc—OpenCores RISC ✦ parisc—HP Precision Architecture RISC ✦ parisc64—HP Precision Architecture 64-bit RISC ✦ ppc—PowerPC 32-bit ✦ ppc64—PowerPC 64-bit ✦ ppcemb—PowerPC (Embedded 32-bit) ✦ s390—IBM Enterprise Systems Architecture/390 ✦ s390x—S/390 64-bit ✦ sh4—SuperH SH-4 (Little Endian) ✦ sh4eb—SuperH SH-4 (Big Endian) ✦ sparc—Scalable Processor Architecture, 32-bit ✦ sparc64—Scalable Processor Architecture, 64-bit ✦ unicore32—Microprocessor Research and Development Center RISC Unicores32 ✦ x86_64—64-bit extension of IA-32 ✦ xtensa—Tensilica Xtensa configurable microprocessor core ✦ xtensaeb—Tensilica Xtensa configurable microprocessor core (Big Endian)
All	hypervisor_type	The hypervisor type.	xen , qemu , kvm , lxc , uml , vmware , or hyperv
All	instance_uuid	For snapshot images, this is the UUID of the server used to create this image.	Valid server UUID
All	kernel_id	The ID of an image stored in the Image Service that should be used as the kernel when booting an AMI-style image.	Valid image ID

All Specific to	os, distr o Key	Description	Supported values
		<p>The common name of the operating system distribution in lowercase (uses the same data vocabulary as the libosinfo project). Specify only a recognized value for this field. Deprecated values are listed to assist you in searching for the recognized value.</p>	<ul style="list-style-type: none"> ✦ arch—Arch Linux. Do not use archlinux or org.archlinux ✦ centos—Community Enterprise Operating System. Do not use org.centos or CentOS ✦ debian—Debian. Do not use Debian or org.debian ✦ fedora—Fedora. Do not use Fedora, org.fedora, or org.fedoraproject ✦ freebsd—FreeBSD. Do not use org.freebsd, freeBSD, or FreeBSD ✦ gentoo—Gentoo Linux. Do not use Gentoo or org.gentoo ✦ mandrake—Mandrakelinux (MandrakeSoft) distribution. Do not use mandrakelinux or MandrakeLinux ✦ mandriva—Mandriva Linux. Do not use mandrivalinux ✦ mes—Mandriva Enterprise Server. Do not use mandrivaent or mandrivaES ✦ msdos—Microsoft Disc Operating System. Do not use ms-dos ✦ netbsd—NetBSD. Do not use NetBSD or org.netbsd ✦ netware—Novell NetWare. Do not use novell or NetWare ✦ openbsd—OpenBSD. Do not use OpenBSD or org.openbsd ✦ opensolaris—OpenSolaris. Do not use OpenSolaris or org.opensolaris

Specific to	Key	Description	Supported values
			<ul style="list-style-type: none"> ✦ opensuse—openSUSE. Do not use suse, SuSE, or org.opensuse ✦ rhel—Red Hat Enterprise Linux. Do not use redhat, RedHat, or com.redhat ✦ sled—SUSE Linux Enterprise Desktop. Do not use com.suse ✦ ubuntu—Ubuntu. Do not use Ubuntu, com.ubuntu, org.ubuntu, or canonical ✦ windows—Microsoft Windows. Do not use com.microsoft.server
All	os_version	The operating system version as specified by the distributor.	Version number (for example, "11.10")
All	ramdisk_id	The ID of image stored in the Image Service that should be used as the ramdisk when booting an AMI-style image.	Valid image ID
All	vm_mode	The virtual machine mode. This represents the host/guest ABI (application binary interface) used for the virtual machine.	<ul style="list-style-type: none"> ✦ hvm—Fully virtualized. This is the mode used by QEMU and KVM. ✦ xen—Xen 3.0 paravirtualized. ✦ uml—User Mode Linux paravirtualized. ✦ exe—Executables in containers. This is the mode used by LXC.
libvirt API driver	hw_disk_bus	Specifies the type of disk controller to attach disk devices to.	One of scsi , virtio , uml , xen , ide , or usb .

Specific to	Key	Description	Supported values
libvirt API driver	hw_rng_model	<p>Adds a random-number generator device to the image's instances. The cloud administrator can enable and control device behavior by configuring the instance's flavor. By default:</p> <ul style="list-style-type: none"> ➤ The generator device is disabled. ➤ <code>/dev/random</code> is used as the default entropy source. To specify a physical HW RNG device, use the following option in the <code>nova.conf</code> file: <pre>rng_dev_path=/dev/hwrng</pre>	virtio , or other supported device.
libvirt API driver	hw_machine_type	Enables booting an ARM system using the specified machine type. By default, if an ARM image is used and its type is not specified, Compute uses vexpress-a15 (for ARMv7) or virt (for AArch64) machine types.	Libvirt machine type. Valid types can be viewed by using the virsh capabilities command (machine types are displayed in the machine tag).
libvirt API driver	hw_scsi_model	Enables the use of VirtIO SCSI (virtio-scsi) to provide block device access for compute instances; by default, instances use VirtIO Block (virtio-blk). VirtIO SCSI is a para-virtualized SCSI controller device that provides improved scalability and performance, and supports advanced SCSI hardware.	virtio-scsi
libvirt API driver	hw_video_model	The video image driver used.	vga , cirrus , vmvga , xen , or qxl
libvirt API driver	hw_video_ram	Maximum RAM for the video image. Used only if a hw_video:ram_max_mb value has been set in the flavor's extra_specs and that value is higher than the value set in hw_video_ram .	Integer in MB (for example, '64')

Specific to	Key	Description	Supported values
libvirt API driver	hw_watchdog_action	Enables a virtual hardware watchdog device that carries out the specified action if the server hangs. The watchdog uses the i6300esb device (emulating a PCI Intel 6300ESB). If hw_watchdog_action is not specified, the watchdog is disabled.	<ul style="list-style-type: none"> ✦ disabled—(default) The device is not attached. Allows the user to disable the watchdog for the image, even if it has been enabled using the image's flavor. ✦ reset—Forcefully reset the guest. ✦ poweroff—Forcefully power off the guest. ✦ pause—Pause the guest. ✦ none—Only enable the watchdog; do nothing if the server hangs.
libvirt API driver	os_command_line	The kernel command line to be used by the libvirt driver, instead of the default. For linux containers (LXC), the value is used as arguments for initialization. This key is valid only for Amazon kernel, ramdisk, or machine images (aki, ari, or ami).	
libvirt API driver and VMware API driver	hw_vif_model	Specifies the model of virtual network interface device to use.	<p>The valid options depend on the configured hypervisor.</p> <ul style="list-style-type: none"> ✦ KVM and QEMU: e1000, ne2k_pci, pcnet, rtl8139, and virtio. ✦ VMware: e1000, e1000e, VirtualE1000, VirtualE1000e, VirtualPCNet32, VirtualSriovEthernetCard, and VirtualVmxnet. ✦ Xen: e1000, netfront, ne2k_pci, pcnet, and rtl8139.
VMware API driver	vmware_adapter_type	The virtual SCSI or IDE controller used by the hypervisor.	lsiLogic , busLogic , or ide

Specific to	Key	Description	Supported values
VMware API driver	vmware_ostype	A VMware GuestID which describes the operating system installed in the image. This value is passed to the hypervisor when creating a virtual machine. If not specified, the key defaults to otherGuest .	See thinkvirt.com .
VMware API driver	vmware_image_version	Currently unused.	1
XenAPI driver	auto_disk_config	If true, the root partition on the disk is automatically resized before the instance boots. This value is only taken into account by the Compute service when using a Xen-based hypervisor with the XenAPI driver. The Compute service will only attempt to resize if there is a single partition on the image, and only if the partition is in ext3 or ext4 format.	true false
XenAPI driver	os_type	The operating system installed on the image. The XenAPI driver contains logic that takes different actions depending on the value of the os_type parameter of the image. For example, for os_type=windows images, it creates a FAT32-based swap partition instead of a Linux swap partition, and it limits the injected host name to less than 16 characters.	linux or windows

Chapter 7. Networking command-line client

The **neutron** client is the command-line interface (CLI) for the OpenStack Networking API and its extensions. This chapter documents **neutron** version 2.3.4.

For help on a specific **neutron** command, enter:

```
$ neutron help COMMAND
```

1. neutron usage

```
usage: neutron [--version] [-v] [-q] [-h] [--os-auth-strategy <auth-
strategy>] [--os-auth-url <auth-url>] [--os-tenant-name <auth-
tenant-name>] [--os-tenant-id <auth-tenant-id>] [--os-username
<auth-username>] [--os-password <auth-password>] [--os-region-name
<auth-region-name>] [--os-token <token>] [--endpoint-type <endpoint-
type>] [--os-url <url>] [--os-cacert <ca-certificate>] [--insecure]
```

2. neutron optional arguments

--version

show program's version number and exit

-v, --verbose, --debug

Increase verbosity of output and show tracebacks on errors. Can be repeated.

-q, --quiet

Suppress output except warnings and errors

-h, --help

Show this help message and exit

--os-auth-strategy <auth-strategy>

Authentication strategy (Env: OS_AUTH_STRATEGY, default keystone). For now, any other value will disable the authentication

--os-auth-url <auth-url>

Authentication URL (Env: OS_AUTH_URL)

--os-tenant-name <auth-tenant-name>

Authentication tenant name (Env: OS_TENANT_NAME)

--os-tenant-id <auth-tenant-id>

Authentication tenant name (Env: OS_TENANT_ID)

--os-username <auth-username>

Authentication username (Env: OS_USERNAME)

--os-password <auth-password>

Authentication password (Env: OS_PASSWORD)

--os-region-name <auth-region-name>

Authentication region name (Env: OS_REGION_NAME)

--os-token <token>

Defaults to `env[OS_TOKEN]`

--endpoint-type <endpoint-type>

Defaults to `env[OS_ENDPOINT_TYPE]` or `publicURL`.

--os-url <url>

Defaults to `env[OS_URL]`

--os-cacert <ca-certificate>

Specify a CA bundle file to use in verifying a TLS (https) server certificate. Defaults to `env[OS_CACERT]`

--insecure

Explicitly allow neutronclient to perform "insecure" SSL (https) requests. The server's certificate will not be verified against any certificate authorities. This option should be used with caution.

3. neutron API v2.0 commands

agent-delete

Delete a given agent.

agent-list

List agents.

agent-show

Show information of a given agent.

agent-update

Update a given agent.

cisco-credential-create

Creates a credential.

cisco-credential-delete

Delete a given credential.

cisco-credential-list

List credentials that belong to a given tenant.

cisco-credential-show

Show information of a given credential.

cisco-network-profile-create

Creates a network profile.

cisco-network-profile-delete

Delete a given network profile.

cisco-network-profile-list

List network profiles that belong to a given tenant.

cisco-network-profile-show

Show information of a given network profile.

cisco-network-profile-update

Update network profile's information.

cisco-policy-profile-list

List policy profiles that belong to a given tenant.

cisco-policy-profile-show

Show information of a given policy profile.

cisco-policy-profile-update

Update policy profile's information.

complete

print bash completion command

dhcp-agent-list-hosting-net

List DHCP agents hosting a network.

dhcp-agent-network-add

Add a network to a DHCP agent.

dhcp-agent-network-remove

Remove a network from a DHCP agent.

ext-list

List all extensions.

ext-show

Show information of a given resource.

firewall-create

Create a firewall.

firewall-delete

Delete a given firewall.

firewall-list

List firewalls that belong to a given tenant.

firewall-policy-create

Create a firewall policy.

firewall-policy-delete

Delete a given firewall policy.

firewall-policy-insert-rule

Insert a rule into a given firewall policy.

firewall-policy-list

List firewall policies that belong to a given tenant.

firewall-policy-remove-rule

Remove a rule from a given firewall policy.

firewall-policy-show

Show information of a given firewall policy.

firewall-policy-update

Update a given firewall policy.

firewall-rule-create

Create a firewall rule.

firewall-rule-delete

Delete a given firewall rule.

firewall-rule-list

List firewall rules that belong to a given tenant.

firewall-rule-show

Show information of a given firewall rule.

firewall-rule-update

Update a given firewall rule.

firewall-show

Show information of a given firewall.

firewall-update

Update a given firewall.

floatingip-associate

Create a mapping between a floating ip and a fixed ip.

floatingip-create

Create a floating ip for a given tenant.

floatingip-delete

Delete a given floating ip.

floatingip-disassociate

Remove a mapping from a floating ip to a fixed ip.

floatingip-list

List floating ips that belong to a given tenant.

floatingip-show

Show information of a given floating ip.

help

print detailed help for another command

ipsec-site-connection-create

Create an IPsecSiteConnection.

ipsec-site-connection-delete

Delete a given IPsecSiteConnection.

ipsec-site-connection-list

List IPsecSiteConnections that belong to a given tenant.

ipsec-site-connection-show

Show information of a given IPsecSiteConnection.

ipsec-site-connection-update

Update a given IPsecSiteConnection.

l3-agent-list-hosting-router

List L3 agents hosting a router.

l3-agent-router-add

Add a router to a L3 agent.

l3-agent-router-remove

Remove a router from a L3 agent.

lb-agent-hosting-pool

Get loadbalancer agent hosting a pool.

lb-healthmonitor-associate

Create a mapping between a health monitor and a pool.

lb-healthmonitor-create

Create a healthmonitor.

lb-healthmonitor-delete

Delete a given healthmonitor.

lb-healthmonitor-disassociate

Remove a mapping from a health monitor to a pool.

lb-healthmonitor-list

List healthmonitors that belong to a given tenant.

lb-healthmonitor-show

Show information of a given healthmonitor.

lb-healthmonitor-update

Update a given healthmonitor.

lb-member-create

Create a member.

lb-member-delete

Delete a given member.

lb-member-list

List members that belong to a given tenant.

lb-member-show

Show information of a given member.

lb-member-update

Update a given member.

lb-pool-create

Create a pool.

lb-pool-delete

Delete a given pool.

lb-pool-list

List pools that belong to a given tenant.

lb-pool-list-on-agent

List the pools on a loadbalancer agent.

lb-pool-show

Show information of a given pool.

lb-pool-stats

Retrieve stats for a given pool.

lb-pool-update

Update a given pool.

lb-vip-create

Create a vip.

lb-vip-delete

Delete a given vip.

lb-vip-list

List vips that belong to a given tenant.

lb-vip-show

Show information of a given vip.

lb-vip-update

Update a given vip.

meter-label-create

Create a metering label for a given tenant.

meter-label-delete

Delete a given metering label.

meter-label-list

List metering labels that belong to a given tenant.

meter-label-rule-create

Create a metering label rule for a given label.

meter-label-rule-delete

Delete a given metering label.

meter-label-rule-list

List metering labels that belong to a given label.

meter-label-rule-show

Show information of a given metering label rule.

meter-label-show

Show information of a given metering label.

net-create

Create a network for a given tenant.

net-delete

Delete a given network.

net-external-list

List external networks that belong to a given tenant.

net-gateway-connect

Add an internal network interface to a router.

net-gateway-create

Create a network gateway.

net-gateway-delete

Delete a given network gateway.

net-gateway-disconnect

Remove a network from a network gateway.

net-gateway-list

List network gateways for a given tenant.

net-gateway-show

Show information of a given network gateway.

net-gateway-update

Update the name for a network gateway.

net-list

List networks that belong to a given tenant.

net-list-on-dhcp-agent

List the networks on a DHCP agent.

net-show

Show information of a given network.

net-update

Update network's information.

port-create

Create a port for a given tenant.

port-delete

Delete a given port.

port-list

List ports that belong to a given tenant.

port-show

Show information of a given port.

port-update

Update port's information.

queue-create

Create a queue.

queue-delete

Delete a given queue.

queue-list

List queues that belong to a given tenant.

queue-show

Show information of a given queue.

quota-delete

Delete defined quotas of a given tenant.

quota-list

List quotas of all tenants who have non-default quota values.

quota-show

Show quotas of a given tenant

quota-update

Define tenant's quotas not to use defaults.

router-create

Create a router for a given tenant.

router-delete

Delete a given router.

router-gateway-clear

Remove an external network gateway from a router.

router-gateway-set

Set the external network gateway for a router.

router-interface-add

Add an internal network interface to a router.

router-interface-delete

Remove an internal network interface from a router.

router-list

List routers that belong to a given tenant.

router-list-on-l3-agent

List the routers on a L3 agent.

router-port-list

List ports that belong to a given tenant, with specified router.

router-show

Show information of a given router.

router-update

Update router's information.

security-group-create

Create a security group.

security-group-delete

Delete a given security group.

security-group-list

List security groups that belong to a given tenant.

security-group-rule-create

Create a security group rule.

security-group-rule-delete

Delete a given security group rule.

security-group-rule-list

List security group rules that belong to a given tenant.

security-group-rule-show

Show information of a given security group rule.

security-group-show

Show information of a given security group.

security-group-update

Update a given security group.

service-provider-list

List service providers.

subnet-create

Create a subnet for a given tenant.

subnet-delete

Delete a given subnet.

subnet-list

List subnets that belong to a given tenant.

subnet-show

Show information of a given subnet.

subnet-update

Update subnet's information.

vpn-ikepolicy-create

Create an IKEPolicy.

vpn-ikepolicy-delete

Delete a given IKE Policy.

vpn-ikepolicy-list

List IKEPolicies that belong to a tenant.

vpn-ikepolicy-show

Show information of a given IKEPolicy.

vpn-ikepolicy-update

Update a given IKE Policy.

vpn-ipsecpolicy-create

Create an ipsecpolicy.

vpn-ipsecpolicy-delete

Delete a given ipsecpolicy.

vpn-ipsecpolicy-list

List ipsecpolicies that belongs to a given tenant connection.

vpn-ipsecpolicy-show

Show information of a given ipsecpolicy.

vpn-ipsecpolicy-update

Update a given ipsec policy.

vpn-service-create

Create a VPNService.

vpn-service-delete

Delete a given VPNService.

vpn-service-list

List VPNService configurations that belong to a given tenant.

vpn-service-show

Show information of a given VPNService.

vpn-service-update

Update a given VPNService.

4. neutron agent-delete command

```
usage: neutron agent-delete [-h] [--request-format {json,xml}] AGENT
```

Delete a given agent.

Positional arguments

AGENT

ID of agent to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

5. neutron agent-list command

```
usage: neutron agent-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}]
                        [--request-format {json,xml}] [-D] [-F FIELD]
```

List agents.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

6. neutron agent-show command

```
usage: neutron agent-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] AGENT
```

Show information of a given agent.

Positional arguments

AGENT

ID of agent to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

7. neutron agent-update command

```
usage: neutron agent-update [-h] [--request-format {json,xml}] AGENT
```

Update a given agent.

Positional arguments

AGENT

ID or name of agent to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

8. neutron cisco-credential-create command

```
usage: neutron cisco-credential-create [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--username USERNAME] [--
password PASSWORD] credential_name credential_type
```

Creates a credential.

Positional arguments

credential_name

Name/Ip address for Credential

credential_type

Type of the Credential

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--username USERNAME

Username for the credential

--password PASSWORD

Password for the credential

9. neutron cisco-credential-delete command

```
usage: neutron cisco-credential-delete [-h] [--request-format
{json,xml}] CREDENTIAL
```

Delete a given credential.

Positional arguments

CREDENTIAL

ID of credential to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

10. neutron cisco-credential-list command

```
usage: neutron cisco-credential-list [-h] [-f {csv,table}] [-c
COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD]
```

List credentials that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

11. neutron cisco-credential-show command

```
usage: neutron cisco-credential-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] CREDENTIAL
```

Show information of a given credential.

Positional arguments

CREDENTIAL

ID of credential to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

12. neutron cisco-network-profile-create command

```
usage: neutron cisco-network-profile-create [-h] [-f {shell,table}]
[-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--sub_type SUB_TYPE] [--
segment_range SEGMENT_RANGE] [--physical_network PHYSICAL_NETWORK]
[--multicast_ip_range MULTICAST_IP_RANGE] [--add-tenant ADD_TENANT]
name {vlan,overlay,multi-segment,trunk}
```

Creates a network profile.

Positional arguments

name

Name for Network Profile

{vlan,overlay,multi-segment,trunk}

Segment type

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--sub_type SUB_TYPE

Sub-type for the segment. Available sub-types for overlay segments: native, enhanced; For trunk segments: vlan, overlay.

--segment_range SEGMENT_RANGE

Range for the Segment

--physical_network PHYSICAL_NETWORK

Name for the Physical Network

--multicast_ip_range MULTICAST_IP_RANGE

Multicast IPv4 Range

--add-tenant ADD_TENANT

Add tenant to the network profile

13. neutron cisco-network-profile-delete command

```
usage: neutron cisco-network-profile-delete [-h] [--request-format {json,xml}] NETWORK_PROFILE
```

Delete a given network profile.

Positional arguments

NETWORK_PROFILE

ID or name of network_profile to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

14. neutron cisco-network-profile-list command

```
usage: neutron cisco-network-profile-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD]
```

List network profiles that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

15. neutron cisco-network-profile-show command

```
usage: neutron cisco-network-profile-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] NETWORK_PROFILE
```

Show information of a given network profile.

Positional arguments

NETWORK_PROFILE

ID or name of network_profile to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

16. neutron cisco-network-profile-update command

```
usage: neutron cisco-network-profile-update [-h] [--request-format
{json,xml}] NETWORK_PROFILE
```

Update network profile's information.

Positional arguments

NETWORK_PROFILE

ID or name of network_profile to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

17. neutron cisco-policy-profile-list command

```
usage: neutron cisco-policy-profile-list [-h] [-f {csv,table}] [-c
COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD]
```

List policy profiles that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

18. neutron cisco-policy-profile-show command

```
usage: neutron cisco-policy-profile-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] POLICY_PROFILE
```

Show information of a given policy profile.

Positional arguments

POLICY_PROFILE

ID or name of policy_profile to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

19. neutron cisco-policy-profile-update command

```
usage: neutron cisco-policy-profile-update [-h] [--request-format {json,xml}] POLICY_PROFILE
```

Update policy profile's information.

Positional arguments

POLICY_PROFILE

ID or name of policy_profile to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

20. neutron dhcp-agent-list-hosting-net command

```
usage: neutron dhcp-agent-list-hosting-net [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] network
```

List DHCP agents hosting a network.

Positional arguments

network

Network to query

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

21. neutron dhcp-agent-network-add command

```
usage: neutron dhcp-agent-network-add [-h] [--request-format
{json,xml}] dhcp_agent network
```

Add a network to a DHCP agent.

Positional arguments

dhcp_agent

ID of the DHCP agent

network

Network to add

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

22. neutron dhcp-agent-network-remove command

```
usage: neutron dhcp-agent-network-remove [-h] [--request-format
{json,xml}] dhcp_agent network
```

Remove a network from a DHCP agent.

Positional arguments

dhcp_agent

ID of the DHCP agent

network

Network to remove

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

23. neutron ext-list command

```
usage: neutron ext-list [-h] [-f {csv,table}] [-c COLUMN] [--quote
{all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-
F FIELD]
```

List all extensions.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

24. neutron ext-show command

```
usage: neutron ext-show [-h] [-f {shell,table}] [-c COLUMN] [--
variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[-D] [-F FIELD] EXT-ALIAS
```

Show information of a given resource.

Positional arguments

EXT -ALIAS

The extension alias

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

25. neutron firewall-create command

```
usage: neutron firewall-create [-h] [-f {shell,table}] [-c COLUMN] [-  
-variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]  
[--tenant-id TENANT_ID] [--name NAME] [--description DESCRIPTION] [-  
-shared] [--admin-state-down] POLICY
```

Create a firewall.

Positional arguments

POLICY

Firewall policy id

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--name NAME

Name for the firewall

--description DESCRIPTION

Description for the firewall rule

--shared

Set shared to True (default False)

--admin-state-down

Set admin state up to false

26. neutron firewall-delete command

```
usage: neutron firewall-delete [-h] [--request-format {json,xml}]  
FIREWALL
```

Delete a given firewall.

Positional arguments

FIREWALL

ID or name of firewall to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

27. neutron firewall-list command

```
usage: neutron firewall-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List firewalls that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

28. neutron firewall-policy-create command

-

```
usage: neutron firewall-policy-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [--tenant-id TENANT_ID] [--description DESCRIPTION] [--shared] [--firewall-rules FIREWALL_RULES] [--audited] NAME
```

Create a firewall policy.

Positional arguments

NAME

Name for the firewall policy

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--description DESCRIPTION

Description for the firewall policy

--shared

To create a shared policy

--firewall-rules FIREWALL_RULES

Ordered list of whitespace-delimited firewall rule names or IDs; e.g., --firewall-rules "rule1 rule2"

--audited

To set audited to True

29. neutron firewall-policy-delete command

```
usage: neutron firewall-policy-delete [-h] [--request-format {json,xml}] FIREWALL_POLICY
```

Delete a given firewall policy.

Positional arguments

FIREWALL_POLICY

ID or name of firewall_policy to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

30. neutron firewall-policy-insert-rule command

```
usage: neutron firewall-policy-insert-rule [-h] [--request-format
{json,xml}] [--insert-before FIREWALL_RULE] [--insert-after
FIREWALL_RULE] FIREWALL_POLICY FIREWALL_RULE
```

Insert a rule into a given firewall policy.

Positional arguments

FIREWALL_POLICY

ID or name of firewall_policy to update

FIREWALL_RULE

New rule to insert

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--insert-before FIREWALL_RULE

Insert before this rule

--insert-after FIREWALL_RULE

Insert after this rule

31. neutron firewall-policy-list command

```
usage: neutron firewall-policy-list [-h] [-f {csv,table}] [-c COLUMN]
[--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List firewall policies that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

32. neutron firewall-policy-remove-rule command

```
usage: neutron firewall-policy-remove-rule [-h] [--request-format {json,xml}] FIREWALL_POLICY FIREWALL_RULE
```

Remove a rule from a given firewall policy.

Positional arguments

FIREWALL_POLICY

ID or name of firewall_policy to update

FIREWALL_RULE

Firewall rule to remove from policy

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

33. neutron firewall-policy-show command

```
usage: neutron firewall-policy-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] FIREWALL_POLICY
```

Show information of a given firewall policy.

Positional arguments

FIREWALL_POLICY

ID or name of firewall_policy to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

34. neutron firewall-policy-update command

```
usage: neutron firewall-policy-update [-h] [--request-format
{json,xml}] FIREWALL_POLICY
```

Update a given firewall policy.

Positional arguments

FIREWALL_POLICY

ID or name of firewall_policy to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

35. neutron firewall-rule-create command

```
usage: neutron firewall-rule-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [--tenant-id TENANT_ID] [--name NAME] [--description DESCRIPTION] [--shared] [--source-ip-address SOURCE_IP_ADDRESS] [--destination-ip-address DESTINATION_IP_ADDRESS] [--source-port SOURCE_PORT] [--destination-port DESTINATION_PORT] [--disabled] --protocol {tcp,udp,icmp,any} --action {allow,deny}
```

Create a firewall rule.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--name NAME

Name for the firewall rule

--description DESCRIPTION

Description for the firewall rule

--shared

Set shared to True (default False)

--source-ip-address SOURCE_IP_ADDRESS

Source ip address or subnet

--destination-ip-address DESTINATION_IP_ADDRESS

Destination ip address or subnet

--source-port SOURCE_PORT

Source port (integer in [1, 65535] or range in a:b)

--destination-port DESTINATION_PORT

Destination port (integer in [1, 65535] or range in a:b)

--disabled

To disable this rule

--protocol {tcp,udp,icmp,any}

Protocol for the firewall rule

--action {allow,deny}

Action for the firewall rule

36. neutron firewall-rule-delete command

```
usage: neutron firewall-rule-delete [-h] [--request-format
{json,xml}] FIREWALL_RULE
```

Delete a given firewall rule.

Positional arguments

FIREWALL_RULE

ID or name of firewall_rule to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

37. neutron firewall-rule-list command

```
usage: neutron firewall-rule-list [-h] [-f {csv,table}] [-c COLUMN]
[--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List firewall rules that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

38. neutron firewall-rule-show command

```
usage: neutron firewall-rule-show [-h] [-f {shell,table}] [-c COLUMN]
  [--variable VARIABLE] [--prefix PREFIX] [--request-format
  {json,xml}] [-D] [-F FIELD] FIREWALL_RULE
```

Show information of a given firewall rule.

Positional arguments

FIREWALL_RULE

ID or name of firewall_rule to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

39. neutron firewall-rule-update command

```
usage: neutron firewall-rule-update [-h] [--request-format
  {json,xml}] [--protocol {tcp,udp,icmp,any}] FIREWALL_RULE
```

Update a given firewall rule.

Positional arguments

FIREWALL_RULE

ID or name of firewall_rule to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--protocol {tcp,udp,icmp,any}

Protocol for the firewall rule

40. neutron firewall-show command

```
usage: neutron firewall-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] FIREWALL
```

Show information of a given firewall.

Positional arguments

FIREWALL

ID or name of firewall to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

41. neutron firewall-update command

```
usage: neutron firewall-update [-h] [--request-format {json,xml}] FIREWALL
```

Update a given firewall.

Positional arguments

FIREWALL

ID or name of firewall to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

42. neutron floatingip-associate command

```
usage: neutron floatingip-associate [-h] [--request-format
{json,xml}] [--fixed-ip-address FIXED_IP_ADDRESS] FLOATINGIP_ID PORT
```

Create a mapping between a floating ip and a fixed ip.

Positional arguments**FLOATINGIP_ID**

ID of the floating IP to associate

PORT

ID or name of the port to be associated with the floatingip

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--fixed-ip-address FIXED_IP_ADDRESS

IP address on the port (only required if port has multiple IPs)

43. neutron floatingip-create command

```
usage: neutron floatingip-create [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--port-id PORT_ID] [--fixed-ip-
address FIXED_IP_ADDRESS] FLOATING_NETWORK
```

Create a floating ip for a given tenant.

Positional arguments

FLOATING_NETWORK

Network name or id to allocate floating IP from

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--port-id PORT_ID ID

of the port to be associated with the floatingip

--fixed-ip-address FIXED_IP_ADDRESS

IP address on the port (only required if port has multiple IPs)

44. neutron floatingip-delete command

```
usage: neutron floatingip-delete [-h] [--request-format {json,xml}]
FLOATINGIP
```

Delete a given floating ip.

Positional arguments

FLOATINGIP

ID of floatingip to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

45. neutron floatingip-disassociate command

```
usage: neutron floatingip-disassociate [-h] [--request-format
{json,xml}] FLOATINGIP_ID
```

Remove a mapping from a floating ip to a fixed ip.

Positional arguments

FLOATINGIP_ID

ID of the floating IP to associate

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

46. neutron floatingip-list command

```
usage: neutron floatingip-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List floating ips that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

47. neutron floatingip-show command

-

```
usage: neutron floatingip-show [-h] [-f {shell,table}] [-c COLUMN] [-
-variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[-D] [-F FIELD] FLOATINGIP
```

Show information of a given floating ip.

Positional arguments

FLOATINGIP

ID of floatingip to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

48. neutron ipsec-site-connection-create command

```
usage: neutron ipsec-site-connection-create [-h] [-f {shell,table}]
[-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--admin-state-down] [--name
NAME] [--description DESCRIPTION] [--mtu MTU] [--initiator {bi-
directional,response-only}] [--dpd
action=ACTION,interval=INTERVAL,timeout=TIMEOUT] --vpnservice-id
VPNSERVICE --ikepolicy-id IKEPOLICY --ipsecpolicy-id IPSECPOLICY --
peer-address PEER_ADDRESS --peer-id PEER_ID --peer-cidr PEER_CIDRS
--psk PSK
```

Create an IPsecSiteConnection.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--admin-state-down

Set admin state up to false

--name NAME

Set friendly name for the connection

--description DESCRIPTION

Set a description for the connection

--mtu MTU MTU

size for the connection, default:1500

--initiator {bi-directional,response-only}

Initiator state in lowercase, default:bi-directional

--dpd

action=ACTION,interval=INTERVAL,timeout=TIMEOUT Ipsec connection Dead Peer Detection Attributes. 'action'-hold,clear,disabled,restart,restart-by-peer. 'interval' and 'timeout' are non negative integers. 'interval' should be less than 'timeout' value. 'action', default:hold 'interval', default:30, 'timeout', default:120.

--vpnservice-id VPNSERVICE

VPNService instance id associated with this connection

--ikepolicy-id IKEPOLICY

IKEPolicy id associated with this connection

--ipsecpolicy-id IPSECPOLICY

IPsecPolicy id associated with this connection

--peer-address PEER_ADDRESS

Peer gateway public IPv4/IPv6 address or FQDN.

--peer-id PEER_ID

Peer router identity for authentication. Can be IPv4/IPv6 address, e-mail address, key id, or FQDN.

--peer-cidr PEER_CIDRS

Remote subnet(s) in CIDR format

--psk PSK

Pre-Shared Key string

49. neutron ipsec-site-connection-delete command

```
usage: neutron ipsec-site-connection-delete [-h] [--request-format {json,xml}] IPSEC_SITE_CONNECTION
```


Delete a given IPsecSiteConnection.

Positional arguments

IPSEC_SITE_CONNECTION

ID or name of ipsec_site_connection to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

50. neutron ipsec-site-connection-list command

```
usage: neutron ipsec-site-connection-list [-h] [-f {csv,table}] [-c
COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List IPsecSiteConnections that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

51. neutron ipsec-site-connection-show command

```
usage: neutron ipsec-site-connection-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] IPSEC_SITE_CONNECTION
```

Show information of a given IPsecSiteConnection.

Positional arguments

IPSEC_SITE_CONNECTION

ID or name of ipsec_site_connection to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

52. neutron ipsec-site-connection-update command

```
usage: neutron ipsec-site-connection-update [-h] [--request-format {json,xml}] [--dpd action=ACTION,interval=INTERVAL,timeout=TIMEOUT] IPSEC_SITE_CONNECTION
```

Update a given IPsecSiteConnection.

Positional arguments

IPSEC_SITE_CONNECTION

ID or name of ipsec_site_connection to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--dpd

action=ACTION,interval=INTERVAL,timeout=TIMEOUT Isec connection Dead Peer Detection Attributes. 'action'-hold,clear,disabled,restart,restart-by-peer. 'interval' and 'timeout' are non negative integers. 'interval' should be less than 'timeout' value. 'action', default:hold 'interval', default:30, 'timeout', default:120.

53. neutron l3-agent-list-hosting-router command

```
usage: neutron l3-agent-list-hosting-router [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] router
```

List L3 agents hosting a router.

Positional arguments**router**

Router to query

Optional arguments**-h, --help**

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

54. neutron l3-agent-router-add command

```
usage: neutron l3-agent-router-add [-h] [--request-format {json,xml}] l3_agent router
```

Add a router to a L3 agent.

Positional arguments**l3_agent**

ID of the L3 agent

router

Router to add

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

55. neutron l3-agent-router-remove command

```
usage: neutron l3-agent-router-remove [-h] [--request-format {json,xml}] l3_agent router
```

Remove a router from a L3 agent.

Positional arguments

l3_agent

ID of the L3 agent

router

Router to remove

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

56. neutron lb-agent-hosting-pool command

```
usage: neutron lb-agent-hosting-pool [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] pool
```

Get loadbalancer agent hosting a pool. Deriving from ListCommand though server will return only one agent to keep common output format for all agent schedulers

Positional arguments

pool

Pool to query

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

57. neutron lb-healthmonitor-associate command

```
usage: neutron lb-healthmonitor-associate [-h] [--request-format
{json,xml}] HEALTH_MONITOR_ID POOL
```

Create a mapping between a health monitor and a pool.

Positional arguments

HEALTH_MONITOR_ID

Health monitor to associate

POOL

ID of the pool to be associated with the health monitor

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

58. neutron lb-healthmonitor-create command

```
usage: neutron lb-healthmonitor-create [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--admin-state-down] [--
expected-codes EXPECTED_CODES] [--http-method HTTP_METHOD] [--url-
path URL_PATH] --delay DELAY --max-retries MAX_RETRIES --timeout
TIMEOUT --type {PING,TCP,HTTP,HTTPS}
```

Create a healthmonitor.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--admin-state-down

Set admin state up to false

--expected-codes EXPECTED_CODES

The list of HTTP status codes expected in response from the member to declare it healthy. This attribute can contain one value, or a list of values separated by comma, or a range of values (e.g. "200-299"). If this attribute is not specified, it defaults to "200".

--http-method HTTP_METHOD

The HTTP method used for requests by the monitor of type HTTP.

--url-path URL_PATH

The HTTP path used in the HTTP request used by the monitor to test a member health. This must be a string beginning with a / (forward slash)

--delay DELAY

The time in seconds between sending probes to members.

--max-retries MAX_RETRIES

Number of permissible connection failures before changing the member status to INACTIVE. [1..10]

--timeout TIMEOUT

Maximum number of seconds for a monitor to wait for a connection to be established before it times out. The value must be less than the delay value.

--type {PING,TCP,HTTP,HTTPS}

One of predefined health monitor types

59. neutron lb-healthmonitor-delete command

```
usage: neutron lb-healthmonitor-delete [-h] [--request-format {json,xml}] HEALTH_MONITOR
```

Delete a given healthmonitor.

Positional arguments

HEALTH_MONITOR

ID or name of health_monitor to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

60. neutron lb-healthmonitor-disassociate command

```
usage: neutron lb-healthmonitor-disassociate [-h] [--request-format
{json,xml}] HEALTH_MONITOR_ID POOL
```

Remove a mapping from a health monitor to a pool.

Positional arguments

HEALTH_MONITOR_ID

Health monitor to associate

POOL

ID of the pool to be associated with the health monitor

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

61. neutron lb-healthmonitor-list command

```
usage: neutron lb-healthmonitor-list [-h] [-f {csv,table}] [-c
COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List healthmonitors that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

62. neutron lb-healthmonitor-show command

```
usage: neutron lb-healthmonitor-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] HEALTH_MONITOR
```

Show information of a given healthmonitor.

Positional arguments

HEALTH_MONITOR

ID or name of health_monitor to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

63. neutron lb-healthmonitor-update command


```
usage: neutron lb-healthmonitor-update [-h] [--request-format
{json,xml}] HEALTH_MONITOR
```

Update a given healthmonitor.

Positional arguments

HEALTH_MONITOR

ID or name of health_monitor to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

64. neutron lb-member-create command

```
usage: neutron lb-member-create [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--admin-state-down] [--weight
WEIGHT] --address ADDRESS --protocol-port PROTOCOL_PORT POOL
```

Create a member.

Positional arguments

POOL

Pool id or name this vip belongs to

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--admin-state-down

Set admin state up to false

--weight WEIGHT

Weight of pool member in the pool (default:1, [0..256])

--address ADDRESS IP

address of the pool member on the pool network.

--protocol-port PROTOCOL_PORT

Port on which the pool member listens for requests or connections.

65. neutron lb-member-delete command

```
usage: neutron lb-member-delete [-h] [--request-format {json,xml}]
MEMBER
```

Delete a given member.

Positional arguments

MEMBER

ID or name of member to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

66. neutron lb-member-list command

```
usage: neutron lb-member-list [-h] [-f {csv,table}] [-c COLUMN] [--
quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}]
[-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List members that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

67. neutron lb-member-show command

```
usage: neutron lb-member-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] MEMBER
```

Show information of a given member.

Positional arguments

MEMBER

ID or name of member to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

68. neutron lb-member-update command

```
usage: neutron lb-member-update [-h] [--request-format {json,xml}] MEMBER
```

Update a given member.

Positional arguments

MEMBER

ID or name of member to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

69. neutron lb-pool-create command

```
usage: neutron lb-pool-create [-h] [-f {shell,table}] [-c COLUMN] [-  
-variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]  
[--tenant-id TENANT_ID] [--admin-state-down] [--description  
DESCRIPTION] --lb-method {ROUND_ROBIN,LEAST_CONNECTIONS,SOURCE_IP}  
--name NAME --protocol {HTTP,HTTPS,TCP} --subnet-id SUBNET [--  
provider PROVIDER]
```

Create a pool.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--admin-state-down

Set admin state up to false

--description DESCRIPTION

Description of the pool

--lb-method {ROUND_ROBIN,LEAST_CONNECTIONS,SOURCE_IP}

The algorithm used to distribute load between the members of the pool

--name NAME

The name of the pool

--protocol {HTTP,HTTPS,TCP}

Protocol for balancing

--subnet-id SUBNET

The subnet on which the members of the pool will be located

--provider PROVIDER

Provider name of loadbalancer service

70. neutron lb-pool-delete command

```
usage: neutron lb-pool-delete [-h] [--request-format {json,xml}]
POOL
```

Delete a given pool.

Positional arguments

POOL

ID or name of pool to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

71. neutron lb-pool-list command

```
usage: neutron lb-pool-list [-h] [-f {csv,table}] [-c COLUMN] [--
quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}]
[-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List pools that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

72. neutron lb-pool-list-on-agent command

```
usage: neutron lb-pool-list-on-agent [-h] [-f {csv,table}] [-c COLUMN]
[--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}]
[-D] [-F FIELD] lbaas_agent
```

List the pools on a loadbalancer agent.

Positional arguments

lbaas_agent

ID of the loadbalancer agent to query

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

73. neutron lb-pool-show command

```
usage: neutron lb-pool-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE]
[--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] POOL
```

Show information of a given pool.

Positional arguments

POOL

ID or name of pool to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

74. neutron lb-pool-stats command

```
usage: neutron lb-pool-stats [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] POOL
```

Retrieve stats for a given pool.

Positional arguments

POOL

ID or name of pool to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

75. neutron lb-pool-update command

```
usage: neutron lb-pool-update [-h] [--request-format {json,xml}]
POOL
```

Update a given pool.

Positional arguments

POOL

ID or name of pool to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

76. neutron lb-vip-create command

```
usage: neutron lb-vip-create [-h] [-f {shell,table}] [-c COLUMN] [--
variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[--tenant-id TENANT_ID] [--address ADDRESS] [--admin-state-down] [--
connection-limit CONNECTION_LIMIT] [--description DESCRIPTION] --
name NAME --protocol-port PROTOCOL_PORT --protocol {TCP,HTTP,HTTPS}
--subnet-id SUBNET POOL
```

Create a vip.

Positional arguments

POOL

Pool id or name this vip belongs to

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--address ADDRESS IP

address of the vip

--admin-state-down

Set admin state up to false

--connection-limit CONNECTION_LIMIT

The maximum number of connections per second allowed for the vip. Positive integer or -1 for unlimited (default)

--description DESCRIPTION

Description of the vip

--name NAME

Name of the vip

--protocol-port PROTOCOL_PORT

TCP port on which to listen for client traffic that is associated with the vip address

--protocol {TCP,HTTP,HTTPS}

Protocol for balancing

--subnet-id SUBNET

The subnet on which to allocate the vip address

77. neutron lb-vip-delete command

```
usage: neutron lb-vip-delete [-h] [--request-format {json,xml}] VIP
```

Delete a given vip.

Positional arguments

VIP

ID or name of vip to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

78. neutron lb-vip-list command

```
usage: neutron lb-vip-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}]
  [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List vips that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

79. neutron lb-vip-show command

```
usage: neutron lb-vip-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] VIP
```

Show information of a given vip.

Positional arguments

VIP

ID or name of vip to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

80. neutron lb-vip-update command

```
usage: neutron lb-vip-update [-h] [--request-format {json,xml}] VIP
```

Update a given vip.

Positional arguments

VIP

ID or name of vip to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

81. neutron meter-label-create command

```
usage: neutron meter-label-create [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[--tenant-id TENANT_ID] [--description DESCRIPTION] NAME
```

Create a metering label for a given tenant.

Positional arguments

NAME

Name of metering label to create

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--description DESCRIPTION

Description of metering label to create

82. neutron meter-label-delete command

```
usage: neutron meter-label-delete [-h] [--request-format {json,xml}]
METERING_LABEL
```

Delete a given metering label.

Positional arguments

METERING_LABEL

ID or name of metering_label to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

83. neutron meter-label-list command

```
usage: neutron meter-label-list [-h] [-f {csv,table}] [-c COLUMN] [--
quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}]
[-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List metering labels that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

84. neutron meter-label-rule-create command

```
usage: neutron meter-label-rule-create [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--direction {ingress,egress}]
[--excluded] LABEL REMOTE_IP_PREFIX
```

Create a metering label rule for a given label.

Positional arguments**LABEL**

Id or Name of the label

REMOTE_IP_PREFIX

CIDR to match on

Optional arguments**-h, --help**

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--direction {ingress,egress}

Direction of traffic, default:ingress

--excluded

Exclude this cidr from the label, default:not excluded

85. neutron meter-label-rule-delete command

```
usage: neutron meter-label-rule-delete [-h] [--request-format
{json,xml}] METERING_LABEL_RULE
```

Delete a given metering label.

Positional arguments

METERING_LABEL_RULE

ID or name of metering_label_rule to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

86. neutron meter-label-rule-list command

```
usage: neutron meter-label-rule-list [-h] [-f {csv,table}] [-c
COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List metering labels that belong to a given label.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

87. neutron meter-label-rule-show command

```
usage: neutron meter-label-rule-show [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[-D] [-F FIELD] METERING_LABEL_RULE
```

Show information of a given metering label rule.

Positional arguments

METERING_LABEL_RULE

ID or name of metering_label_rule to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

88. neutron meter-label-show command

```
usage: neutron meter-label-show [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[-D] [-F FIELD] METERING_LABEL
```

Show information of a given metering label.

Positional arguments

METERING_LABEL

ID or name of metering_label to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

89. neutron net-create command

```
usage: neutron net-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
                        [--tenant-id TENANT_ID] [--admin-state-down] [--shared] NAME
```

Create a network for a given tenant.

Positional arguments

NAME

Name of network to create

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--admin-state-down

Set Admin State Up to false

--shared

Set the network as shared

90. neutron net-delete command

```
usage: neutron net-delete [-h] [--request-format {json,xml}] NETWORK
```

Delete a given network.

Positional arguments

NETWORK

ID or name of network to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

91. neutron net-external-list command

```
usage: neutron net-external-list [-h] [-f {csv,table}] [-c COLUMN] [-
quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}]
[-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List external networks that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

92. neutron net-gateway-connect command

```
usage: neutron net-gateway-connect [-h] [--request-format
{json,xml}] [--segmentation-type SEGMENTATION_TYPE] [--segmentation-
id SEGMENTATION_ID] NET-GATEWAY-ID NETWORK-ID
```

Add an internal network interface to a router.

Positional arguments

NET - GATEWAY - ID

ID of the network gateway

NETWORK - ID

ID of the internal network to connect on the gateway

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--segmentation-type SEGMENTATION_TYPE

L2 segmentation strategy on the external side of the gateway (e.g.: VLAN, FLAT)

--segmentation-id SEGMENTATION_ID

Identifier for the L2 segment on the external side of the gateway

93. neutron net-gateway-create command

```
usage: neutron net-gateway-create [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--device DEVICE] NAME
```

Create a network gateway.

Positional arguments

NAME

Name of network gateway to create

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--device DEVICE

Device info for this gateway device_id=<device identifier>,interface_name=<name_or_identifier> It can be repeated for multiple devices for HA gateways

94. neutron net-gateway-delete command

```
usage: neutron net-gateway-delete [-h] [--request-format {json,xml}]
NETWORK_GATEWAY
```

Delete a given network gateway.

Positional arguments**NETWORK_GATEWAY**

ID or name of network_gateway to delete

Optional arguments**-h, --help**

show this help message and exit

--request-format {json,xml}

The xml or json request format

95. neutron net-gateway-disconnect command

```
usage: neutron net-gateway-disconnect [-h] [--request-format
{json,xml}] [--segmentation-type SEGMENTATION_TYPE] [--segmentation-
id SEGMENTATION_ID] NET-GATEWAY-ID NETWORK-ID
```

Remove a network from a network gateway.

Positional arguments**NET - GATEWAY - ID**

ID of the network gateway

NETWORK - ID

ID of the internal network to connect on the gateway

Optional arguments**-h, --help**

show this help message and exit

--request-format {json,xml}

The xml or json request format

--segmentation-type SEGMENTATION_TYPE

L2 segmentation strategy on the external side of the gateway (e.g.: VLAN, FLAT)

--segmentation-id SEGMENTATION_ID

Identifier for the L2 segment on the external side of the gateway

96. neutron net-gateway-list command

```
usage: neutron net-gateway-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD]
```

List network gateways for a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

97. neutron net-gateway-show command

```
usage: neutron net-gateway-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] NETWORK_GATEWAY
```

Show information of a given network gateway.

Positional arguments

NETWORK_GATEWAY

ID or name of network_gateway to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

98. neutron net-gateway-update command

```
usage: neutron net-gateway-update [-h] [--request-format {json,xml}]
NETWORK_GATEWAY
```

Update the name for a network gateway.

Positional arguments

NETWORK_GATEWAY

ID or name of network_gateway to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

99. neutron net-list command

```
usage: neutron net-list [-h] [-f {csv,table}] [-c COLUMN] [--quote
{all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-
F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List networks that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

100. neutron net-list-on-dhcp-agent command

```
usage: neutron net-list-on-dhcp-agent [-h] [-f {csv,table}] [-c COLUMN]
    [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}]
    [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
    dhcp_agent
```

List the networks on a DHCP agent.

Positional arguments

dhcp_agent

ID of the DHCP agent

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

101. neutron net-show command

```
usage: neutron net-show [-h] [-f {shell,table}] [-c COLUMN] [--
variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[-D] [-F FIELD] NETWORK
```

Show information of a given network.

Positional arguments**NETWORK**

ID or name of network to look up

Optional arguments**-h, --help**

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

102. neutron net-update command

```
usage: neutron net-update [-h] [--request-format {json,xml}] NETWORK
```

Update network's information.

Positional arguments**NETWORK**

ID or name of network to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

103. neutron port-create command

```
usage: neutron port-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[--tenant-id TENANT_ID] [--name NAME] [--admin-state-down] [--mac-address MAC_ADDRESS] [--device-id DEVICE_ID] [--fixed-ip
subnet_id=SUBNET,ip_address=IP_ADDR] [--security-group SECURITY_GROUP | --no-security-groups] [--extra-dhcp-opt
EXTRA_DHCP_OPTS] NETWORK
```

Create a port for a given tenant.

Positional arguments

NETWORK

Network id or name this port belongs to

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--name NAME

Name of this port

--admin-state-down

Set admin state up to false

--mac-address MAC_ADDRESS

MAC address of this port

--device-id DEVICE_ID

Device id of this port

--fixed-ip

subnet_id=SUBNET,ip_address=IP_ADDR Desired IP and/or subnet for this port:
 subnet_id=<name_or_id>,ip_address=<ip>, (This option can be repeated.)

--security-group SECURITY_GROUP

Security group associated with the port (This option can be repeated)

--no-security-groups

Associate no security groups with the port

--extra-dhcp-opt EXTRA_DHCP_OPTS

Extra dhcp options to be assigned to this port: opt_name=
 <dhcp_option_name>,opt_value=<value>, (This option can be repeated.)

104. neutron port-delete command

```
usage: neutron port-delete [-h] [--request-format {json,xml}] PORT
```

Delete a given port.

Positional arguments

PORT

ID or name of port to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

105. neutron port-list command

```
usage: neutron port-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}]
  [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List ports that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

106. neutron port-show command

```
usage: neutron port-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] PORT
```

Show information of a given port.

Positional arguments

PORT

ID or name of port to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

107. neutron port-update command

```
usage: neutron port-update [-h] [--request-format {json,xml}] [--security-group SECURITY_GROUP | --no-security-groups] [--extra-dhcp-opt EXTRA_DHCP_OPTS] PORT
```

Update port's information.

Positional arguments

PORT

ID or name of port to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--security-group SECURITY_GROUP

Security group associated with the port (This option can be repeated)

--no-security-groups

Associate no security groups with the port

--extra-dhcp-opt EXTRA_DHCP_OPTS

Extra dhcp options to be assigned to this port: opt_name=<dhcp_option_name>,opt_value=<value>, (This option can be repeated.)

108. neutron queue-create command

```
usage: neutron queue-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [--tenant-id TENANT_ID] [--min MIN] [--max MAX] [--qos-marking QOS_MARKING] [--default DEFAULT] [--dscp DSCP] NAME
```

Create a queue.

Positional arguments

NAME

Name of queue

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--min MIN

min-rate

--max MAX

max-rate

--qos-marking QOS_MARKING

QOS marking untrusted/trusted

--default DEFAULT

If true all ports created with be the size of this queue if queue is not specified

--dscp DSCP

Differentiated Services Code Point

109. neutron queue-delete command

```
usage: neutron queue-delete [-h] [--request-format {json,xml}]
QOS_QUEUE
```

Delete a given queue.

Positional arguments

QOS_QUEUE

ID or name of qos_queue to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

110. neutron queue-list command

```
usage: neutron queue-list [-h] [-f {csv,table}] [-c COLUMN] [--quote
{all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-
F FIELD]
```

List queues that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

111. neutron queue-show command

```
usage: neutron queue-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] QOS_QUEUE
```

Show information of a given queue.

Positional arguments

QOS_QUEUE

ID or name of qos_queue to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

112. neutron quota-delete command

```
usage: neutron quota-delete [-h] [--request-format {json,xml}] [--tenant-id tenant-id]
```

Delete defined quotas of a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id

tenant-id The owner tenant ID

113. neutron quota-list command

```
usage: neutron quota-list [-h] [-f {csv,table}] [-c COLUMN] [--quote
{all,minimal,none,nonnumeric}] [--request-format {json,xml}]
```

List quotas of all tenants who have non-default quota values.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

114. neutron quota-show command

```
usage: neutron quota-show [-h] [-f {shell,table}] [-c COLUMN] [--
variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[--tenant-id tenant-id]
```

Show quotas of a given tenant

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id

tenant-id The owner tenant ID

115. neutron quota-update command

```
usage: neutron quota-update [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[--tenant-id tenant-id] [--network networks] [--subnet subnets] [--port ports]
[--router routers] [--floatingip floatingips] [--security-group security_groups]
[--security-group-rule security_group_rules]
```

Define tenant's quotas not to use defaults.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id

tenant-id The owner tenant ID

--network

networks The limit of networks

--subnet

subnets The limit of subnets

--port

ports The limit of ports

--router

routers The limit of routers

--floatingip

floatingips The limit of floating IPs

--security-group

security_groups The limit of security groups

--security-group-rule

security_group_rules The limit of security groups rules

116. neutron router-create command

```
usage: neutron router-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE]
[--prefix PREFIX] [--request-format {json,xml}] [--tenant-id TENANT_ID]
[--admin-state-down] NAME
```

Create a router for a given tenant.

Positional arguments

NAME

Name of router to create

distributed

Create a distributed router (VMware NSX plugin only)

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--admin-state-down

Set Admin State Up to false

117. neutron router-delete command

```
usage: neutron router-delete [-h] [--request-format {json,xml}]  
ROUTER
```

Delete a given router.

Positional arguments

ROUTER

ID or name of router to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

118. neutron router-gateway-clear command


```
usage: neutron router-gateway-clear [-h] [--request-format
{json,xml}] router-id
```

Remove an external network gateway from a router.

Positional arguments

router-id

ID of the router

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

119. neutron router-gateway-set command

```
usage: neutron router-gateway-set [-h] [--request-format {json,xml}]
[--disable-snat] router-id external-network-id
```

Set the external network gateway for a router.

Positional arguments

router-id

ID of the router

external-network-id

ID of the external network for the gateway

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--disable-snat

Disable Source NAT on the router gateway

120. neutron router-interface-add command

```
usage: neutron router-interface-add [-h] [--request-format
{json,xml}] router-id INTERFACE
```

Add an internal network interface to a router.

Positional arguments

router-id

ID of the router

INTERFACE

The format is "SUBNET|subnet=SUBNET|port=PORT". Either a subnet or port must be specified. Both ID and name are accepted as SUBNET or PORT. Note that "subnet=" can be omitted when specifying subnet.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

121. neutron router-interface-delete command

```
usage: neutron router-interface-delete [-h] [--request-format
{json,xml}] router-id INTERFACE
```

Remove an internal network interface from a router.

Positional arguments

router-id

ID of the router

INTERFACE

The format is "SUBNET|subnet=SUBNET|port=PORT". Either a subnet or port must be specified. Both ID and name are accepted as SUBNET or PORT. Note that "subnet=" can be omitted when specifying subnet.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

122. neutron router-list command

```
usage: neutron router-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List routers that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

123. neutron router-list-on-l3-agent command

```
usage: neutron router-list-on-l3-agent [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] l3_agent
```

List the routers on a L3 agent.

Positional arguments

l3_agent

ID of the L3 agent to query

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

124. neutron router-port-list command

```
usage: neutron router-port-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}] router
```

List ports that belong to a given tenant, with specified router.

Positional arguments

router

ID or name of router to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

125. neutron router-show command

```
usage: neutron router-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] ROUTER
```

Show information of a given router.

Positional arguments

ROUTER

ID or name of router to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

126. neutron router-update command

```
usage: neutron router-update [-h] [--request-format {json,xml}] ROUTER
```

Update router's information.

Positional arguments

ROUTER

ID or name of router to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

127. neutron security-group-create command

```
usage: neutron security-group-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [--tenant-id TENANT_ID] [--description DESCRIPTION] NAME
```

Create a security group.

Positional arguments

NAME

Name of security group

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--description DESCRIPTION

Description of security group

128. neutron security-group-delete command

```
usage: neutron security-group-delete [-h] [--request-format {json,xml}] SECURITY_GROUP
```

Delete a given security group.

Positional arguments

SECURITY_GROUP

ID or name of security_group to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

129. neutron security-group-list command

```
usage: neutron security-group-list [-h] [-f {csv,table}] [-c COLUMN]
[--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List security groups that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

130. neutron security-group-rule-create command

```
usage: neutron security-group-rule-create [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--direction {ingress,egress}]
[--ethertype ETHERTYPE] [--protocol PROTOCOL] [--port-range-min
PORT_RANGE_MIN] [--port-range-max PORT_RANGE_MAX] [--remote-ip-
prefix REMOTE_IP_PREFIX] [--remote-group-id REMOTE_GROUP]
SECURITY_GROUP
```

Create a security group rule.

Positional arguments

SECURITY_GROUP

Security group name or id to add rule.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--direction {ingress,egress}

Direction of traffic: ingress/egress

--ethertype ETHERTYPE

IPv4/IPv6

--protocol PROTOCOL

Protocol of packet

--port-range-min PORT_RANGE_MIN

Starting port range

--port-range-max PORT_RANGE_MAX

Ending port range

--remote-ip-prefix REMOTE_IP_PREFIX

CIDR to match on

--remote-group-id REMOTE_GROUP

Remote security group name or id to apply rule

131. neutron security-group-rule-delete command

```
usage: neutron security-group-rule-delete [-h] [--request-format {json,xml}] SECURITY_GROUP_RULE
```

Delete a given security group rule.

Positional arguments

SECURITY_GROUP_RULE

ID of security_group_rule to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

132. neutron security-group-rule-list command

```
usage: neutron security-group-rule-list [-h] [-f {csv,table}] [-c
COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}] [--no-nameconv]
```

List security group rules that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

--no-nameconv

Do not convert security group ID to its name

133. neutron security-group-rule-show command

```
usage: neutron security-group-rule-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] SECURITY_GROUP_RULE
```

Show information of a given security group rule.

Positional arguments

SECURITY_GROUP_RULE

ID of security_group_rule to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

134. neutron security-group-show command

```
usage: neutron security-group-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] SECURITY_GROUP
```

Show information of a given security group.

Positional arguments

SECURITY_GROUP

ID or name of security_group to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

135. neutron security-group-update command

```
usage: neutron security-group-update [-h] [--request-format
{json,xml}] [--name NAME] [--description DESCRIPTION] SECURITY_GROUP
```

Update a given security group.

Positional arguments

SECURITY_GROUP

ID or name of security_group to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--name NAME

Name of security group

--description DESCRIPTION

Description of security group

136. neutron service-provider-list command

```
usage: neutron service-provider-list [-h] [-f {csv,table}] [-c
COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List service providers.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

137. neutron subnet-create command

```
usage: neutron subnet-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [--tenant-id TENANT_ID] [--name NAME] [--ip-version {4,6}] [--gateway GATEWAY_IP] [--no-gateway] [--allocation-pool start=IP_ADDR,end=IP_ADDR] [--host-route destination=CIDR,nexthop=IP_ADDR] [--dns-nameserver DNS_NAMESERVER] [--disable-dhcp] NETWORK CIDR
```

Create a subnet for a given tenant.

Positional arguments

NETWORK

Network id or name this subnet belongs to

CIDR

CIDR of subnet to create

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--name NAME

Name of this subnet

--ip-version {4,6} IP

version with default 4

--gateway GATEWAY_IP

Gateway ip of this subnet

--no-gateway

No distribution of gateway

--allocation-pool

start=IP_ADDR,end=IP_ADDR Allocation pool IP addresses for this subnet (This option can be repeated)

--host-route

destination=CIDR,nexthop=IP_ADDR Additional route (This option can be repeated)

--dns-nameserver DNS_NAMESERVER

DNS name server for this subnet (This option can be repeated)

--disable-dhcp

Disable DHCP for this subnet

138. neutron subnet-delete command

```
usage: neutron subnet-delete [-h] [--request-format {json,xml}]
SUBNET
```

Delete a given subnet.

Positional arguments

SUBNET

ID or name of subnet to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

139. neutron subnet-list command

```
usage: neutron subnet-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List subnets that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

140. neutron subnet-show command

```
usage: neutron subnet-show [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [-D] [-F FIELD] SUBNET
```

Show information of a given subnet.

Positional arguments

SUBNET

ID or name of subnet to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

141. neutron subnet-update command

```
usage: neutron subnet-update [-h] [--request-format {json,xml}]
SUBNET
```

Update subnet's information.

Positional arguments

SUBNET

ID or name of subnet to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

142. neutron vpn-ikepolicy-create command

```
usage: neutron vpn-ikepolicy-create [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [--tenant-id TENANT_ID] [--description DESCRIPTION] [--
auth-algorithm {sha1}] [--encryption-algorithm {3des,aes-128,aes-
192,aes-256}] [--phase1-negotiation-mode {main}] [--ike-version
{v1,v2}] [--pfs {group2,group5,group14}] [--lifetime
units=UNITS,value=VALUE] NAME
```

Create an IKEPolicy.

Positional arguments

NAME

Name of the IKE Policy

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--description DESCRIPTION

Description of the IKE policy

--auth-algorithm {sha1}

Authentication algorithm in lowercase. default:sha1

--encryption-algorithm {3des,aes-128,aes-192,aes-256}

Encryption Algorithm in lowercase, default:aes-128

--phase1-negotiation-mode {main}

IKE Phase1 negotiation mode in lowercase, default:main

--ike-version {v1,v2}

IKE version in lowercase, default:v1

--pfs {group2,group5,group14}

Perfect Forward Secrecy in lowercase, default:group5

--lifetime

units=UNITS,value=VALUE IKE Lifetime Attributes.'units'-seconds,default:seconds.
'value'-non negative integer, default:3600.

143. neutron vpn-ikepolicy-delete command

```
usage: neutron vpn-ikepolicy-delete [-h] [--request-format {json,xml}] IKEPOLICY
```

Delete a given IKE Policy.

Positional arguments

IKEPOLICY

ID or name of ikepolicy to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

144. neutron vpn-ikepolicy-list command

```
usage: neutron vpn-ikepolicy-list [-h] [-f {csv,table}] [-c COLUMN]
[--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List IKEPolicies that belong to a tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

145. neutron vpn-ikepolicy-show command

```
usage: neutron vpn-ikepolicy-show [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] IKEPOLICY
```

Show information of a given IKEPolicy.

Positional arguments

IKEPOLICY

ID or name of ikepolicy to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

146. neutron vpn-ikepolicy-update command

```
usage: neutron vpn-ikepolicy-update [-h] [--request-format {json,xml}]
[--lifetime units=UNITS,value=VALUE] IKEPOLICY
```

Update a given IKE Policy.

Positional arguments

IKEPOLICY

ID or name of ikepolicy to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--lifetime

units=UNITS,value=VALUE IKE Lifetime Attributes.'units'-seconds,default:seconds.
'value'-non negative integer, default:3600.

147. neutron vpn-ipsecpolicy-create command

```
usage: neutron vpn-ipsecpolicy-create [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}]
[--tenant-id TENANT_ID] [--description DESCRIPTION] [--transform-protocol {esp,ah,ah-esp}]
[--auth-algorithm {sha1}] [--
```

```
encryption-algorithm {3des,aes-128,aes-192,aes-256}] [- -
encapsulation-mode {tunnel,transport}] [- -pfs
{group2,group5,group14}] [- -lifetime units=UNITS,value=VALUE] NAME
```

Create an ipsecpolicy.

Positional arguments

NAME

Name of the IPsecPolicy

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--description DESCRIPTION

Description of the IPsecPolicy

--transform-protocol {esp,ah,ah-esp}

Transform Protocol in lowercase, default:esp

--auth-algorithm {sha1}

Authentication algorithm in lowercase, default:sha1

--encryption-algorithm {3des,aes-128,aes-192,aes-256}

Encryption Algorithm in lowercase, default:aes-128

--encapsulation-mode {tunnel,transport}

Encapsulation Mode in lowercase, default:tunnel

--pfs {group2,group5,group14}

Perfect Forward Secrecy in lowercase, default:group5

--lifetime

units=UNITS,value=VALUE IPsec Lifetime Attributes.'units'-seconds,default:seconds.
'value'-non negative integer, default:3600.

148. neutron vpn-ipsecpolicy-delete command

```
usage: neutron vpn-ipsecpolicy-delete [-h] [--request-format
{json,xml}] IPSECPOLICY
```

Delete a given ipsecpolicy.

Positional arguments

IPSECPOLICY

ID or name of ipsecpolicy to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

149. neutron vpn-ipsecpolicy-list command

```
usage: neutron vpn-ipsecpolicy-list [-h] [-f {csv,table}] [-c COLUMN]
[--quote {all,minimal,none,nonnumeric}] [--request-format
{json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir
{asc,desc}]
```

List ipsecpolicies that belongs to a given tenant connection.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

150. neutron vpn-ipsecpolicy-show command

```
usage: neutron vpn-ipsecpolicy-show [-h] [-f {shell,table}] [-c
COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] IPSECPOLICY
```

Show information of a given ipsecpolicy.

Positional arguments

IPSECPOLICY

ID or name of ipsecpolicy to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

151. neutron vpn-ipsecpolicy-update command

```
usage: neutron vpn-ipsecpolicy-update [-h] [--request-format
{json,xml}] [--lifetime units=UNITS,value=VALUE] IPSECPOLICY
```

Update a given ipsec policy.

Positional arguments

IPSECPOLICY

ID or name of ipsecpolicy to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--lifetime

units=UNITS,value=VALUE IPsec Lifetime Attributes.'units'-seconds,default:seconds.
'value'-non negative integer, default:3600.

152. neutron vpn-service-create command

```
usage: neutron vpn-service-create [-h] [-f {shell,table}] [-c COLUMN] [--variable VARIABLE] [--prefix PREFIX] [--request-format {json,xml}] [--tenant-id TENANT_ID] [--admin-state-down] [--name NAME] [--description DESCRIPTION] ROUTER SUBNET
```

Create a VPNService.

Positional arguments

ROUTER

Router unique identifier for the vpnservice

SUBNET

Subnet unique identifier for the vpnservice deployment

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

--tenant-id TENANT_ID

The owner tenant ID

--admin-state-down

Set admin state up to false

--name NAME

Set a name for the vpnservice

--description DESCRIPTION

Set a description for the vpnservice

153. neutron vpn-service-delete command

```
usage: neutron vpn-service-delete [-h] [--request-format {json,xml}] VPNSERVICE
```

Delete a given VPNService.

Positional arguments

VPNSERVICE

ID or name of vpnservice to delete

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

154. neutron vpn-service-list command

```
usage: neutron vpn-service-list [-h] [-f {csv,table}] [-c COLUMN] [--quote {all,minimal,none,nonnumeric}] [--request-format {json,xml}] [-D] [-F FIELD] [-P SIZE] [--sort-key FIELD] [--sort-dir {asc,desc}]
```

List VPNService configurations that belong to a given tenant.

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

-P SIZE, --page-size SIZE

Specify retrieve unit of each request, then split one request to several requests

--sort-key FIELD

Sort list by specified fields (This option can be repeated), The number of sort_dir and sort_key should match each other, more sort_dir specified will be omitted, less will be filled with asc as default direction

--sort-dir {asc,desc}

Sort list in specified directions (This option can be repeated)

155. neutron vpn-service-show command

-

```
usage: neutron vpn-service-show [-h] [-f {shell,table}] [-c COLUMN]
[--variable VARIABLE] [--prefix PREFIX] [--request-format
{json,xml}] [-D] [-F FIELD] VPNSERVICE
```

Show information of a given VPNService.

Positional arguments

VPNSERVICE

ID or name of vpnservice to look up

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

-D, --show-details

Show detailed info

-F FIELD, --field FIELD

Specify the field(s) to be returned by server, can be repeated

156. neutron vpn-service-update command

```
usage: neutron vpn-service-update [-h] [--request-format {json,xml}]
VPNSERVICE
```

Update a given VPNService.

Positional arguments

VPNSERVICE

ID or name of vpnservice to update

Optional arguments

-h, --help

show this help message and exit

--request-format {json,xml}

The xml or json request format

Chapter 8. neutron-debug command-line client

The **neutron-debug** client is an extension to the **neutron** command-line interface (CLI) for the OpenStack neutron-debug tool. This chapter documents **neutron-debug** version 2.3.0.

For help on a specific **neutron-debug** command, enter:

```
$ neutron-debug help COMMAND
```

1. neutron-debug usage

```
[--os-password <auth-password>] [--os-tenant-name <auth-tenant-name>] [--os-tenant-id <auth-tenant-id>] [--os-auth-url <auth-url>]
[--os-region-name <region-name>] [--service-type <service-type>] [--service-name <service-name>] [--volume-service-name <volume-service-name>]
[--endpoint-type <endpoint-type>] [--os-volume-api-version <volume-api-ver>] [--os-cacert <ca-certificate>] [--retries <retries>] <subcommand> ...
```

Subcommands

probe-create

Create probe port - create port and interface within a network namespace.

probe-list

List all probes.

probe-clear

Clear all probes.

probe-delete

Delete probe - delete port then delete the namespace.

probe-exec

Execute commands in the namespace of the probe.

ping-all

ping-all is all-in-one command to ping all fixed IP's in a specified network.

2. neutron-debug optional arguments

--version

Show version number and exit.

-v, --verbose, --debug

Increase verbosity of output and show tracebacks on errors. Can be repeated.

-q, --quiet

Suppress output except warnings and errors

-h, --help

Show this help message and exit

--os-auth-strategy <auth-strategy>

Authentication strategy (Env: OS_AUTH_STRATEGY, default keystone). For now, any other value will disable the authentication

--os-auth-url <auth-url>

Authentication URL (Env: OS_AUTH_URL)

--os-tenant-name <auth-tenant-name>

Authentication tenant name (Env: OS_TENANT_NAME)

--os-tenant-id <auth-tenant-id>

Authentication tenant name (Env: OS_TENANT_ID)

--os-username <auth-username>

Authentication username (Env: OS_USERNAME)

--os-password <auth-password>

Authentication password (Env: OS_PASSWORD)

--os-region-name <auth-region-name>

Authentication region name (Env: OS_REGION_NAME)

--os-token <token>

Defaults to **env[OS_TOKEN]**

--endpoint-type <endpoint-type>

Defaults to **env[OS_ENDPOINT_TYPE]** or public URL.

--os-url <url>

Defaults to **env[OS_URL]**

--os-cacert <ca-certificate>

Specify a CA bundle file to use in verifying a TLS (https) server certificate. Defaults to **env[OS_CACERT]**

--insecure

Explicitly allow neutron-debug to perform "insecure" SSL (https) requests. The server's certificate will not be verified against any certificate authorities. This option should be used with caution.

--config-file CONFIG_FILE

Config file for interface driver (You may also use l3_agent.ini)

3. neutron-debug probe-create command

```
usage: neutron-debug probe-create NET
```

Create probe port - create port and interface, then place it into the created network namespace.

Positional arguments

NET ID

ID of the network in which the probe will be created.

4. neutron-debug probe-list command

```
usage: neutron-debug probe-list
```

List probes.

5. neutron-debug probe-clear command

```
usage: neutron-debug probe-clear
```

Clear all probes.

6. neutron-debug probe-delete command

```
usage: neutron-debug probe-delete <port-id>
```

Remove a probe.

Positional arguments

<port-id>

ID of the probe to delete.

7. neutron-debug probe-exec command

```
usage: neutron-debug probe-exec <port-id> <command>
```

Execute commands in the namespace of the probe

8. neutron-debug ping-all command

```
usage: neutron-debug ping-all <port-id> --timeout <number>
```

All-in-one command to ping all fixed IP's in a specified network.

Positional arguments

<port-id>

ID of the port to use.

Optional arguments

--timeout <timeout in seconds>

Optional ping timeout.

9. neutron-debug example

```
usage: neutron-debug create-probe < NET_ID>
```

Create a probe namespace within the network identified by NET_ID. The namespace will have the name of qprobe-<UUID of the probe port>

Note

For the following examples to function, the security group rules may need to be modified to allow the SSH (TCP port 22) or ping (ICMP) traffic into network.

```
usage: neutron-debug probe-exec <probe ID> "ssh <IP of instance>"
```

SSH to an instance within the network.

```
usage: neutron-debug ping-all <network ID>"
```

Ping all instances on this network to verify they are responding.

```
usage: neutron-debug probe-exec <probe_ID> dhcping <VM_MAC address>  
-s <IP of DHCP server>"
```

Ping the DHCP server for this network using dhcping to verify it is working.

Chapter 9. Object Storage command-line client

The **swift** client is the command-line interface (CLI) for the OpenStack Object Storage API and its extensions. This chapter documents **swift** version 2.0.3.

For help on a specific **swift** command, enter:

```
$ swift help COMMAND
```

1. swift usage

```
[--debug] [--info] [--quiet] [--auth <auth_url>] [--auth-version
<auth_version>] [--user <username>] [--key <api_key>] [--retries
<num_retries>] [--os-username <auth-user-name>] [--os-password
<auth-password>] [--os-tenant-id <auth-tenant-id>] [--os-tenant-name
<auth-tenant-name>] [--os-auth-url <auth-url>] [--os-auth-token
<auth-token>] [--os-storage-url <storage-url>] [--os-region-name
<region-name>] [--os-service-type <service-type>] [--os-endpoint-
type <endpoint-type>] [--os-cacert <ca-certificate>] [--insecure] [-
-no-ssl-compression] <subcommand> ...
```

Subcommands

delete

Delete a container or objects within a container

download

Download objects from containers

list

Lists the containers for the account or the objects for a container

post

Updates meta information for the account, container, or object; creates containers if not present

stat

Displays information for the account, container, or object

upload

Uploads files or directories to the given container

capabilities

List cluster capabilities

2. swift examples

```
swift -A https://auth.api.rackspacecloud.com/v1.0 -U user -K
```

```
api_key stat -v swift --os-auth-url https://api.example.com/v2.0 --  
os-tenant-name tenant \ --os-username user --os-password password  
list swift --os-auth-token 6ee5eb33efad4e45ab46806eac010566 \ --os-  
storage-url https://10.1.5.2:8080/v1/AUTH_ced809b6a4baea7aeab61a \  
list swift list --lh
```

3. swift optional arguments

--version

show program's version number and exit

-h, --help

show this help message and exit

-s, --snet

Use SERVICENET internal network

-v, --verbose

Print more info

--debug

Show the curl commands and results of all http queries regardless of result status.

--info

Show the curl commands and results of all http queries which return an error.

-q, --quiet

Suppress status output

-A AUTH, --auth=AUTH URL

for obtaining an auth token

-V AUTH_VERSION, --auth-version=AUTH_VERSION

Specify a version for authentication. Defaults to 1.0.

-U USER, --user=USER

User name for obtaining an auth token.

-K KEY, --key=KEY

Key for obtaining an auth token.

-R RETRIES, --retries=RETRIES

The number of times to retry a failed connection.

--os-username=<auth-user-name>

OpenStack username. Defaults to `env[OS_USERNAME]`.

--os-password=<auth-password>

OpenStack password. Defaults to **env[OS_PASSWORD]**.

--os-tenant-id=<auth-tenant-id>

OpenStack tenant ID. Defaults to **env[OS_TENANT_ID]**

--os-tenant-name=<auth-tenant-name>

OpenStack tenant name. Defaults to **env[OS_TENANT_NAME]**.

--os-auth-url=<auth-url>

OpenStack auth URL. Defaults to **env[OS_AUTH_URL]**.

--os-auth-token=<auth-token>

OpenStack token. Defaults to **env[OS_AUTH_TOKEN]**. Used with **--os-storage-url** to bypass the usual username/password authentication.

--os-storage-url=<storage-url>

OpenStack storage URL. Defaults to **env[OS_STORAGE_URL]**. Overrides the storage url returned during auth. Will bypass authentication when used with **--os-auth-token**.

--os-region-name=<region-name>

OpenStack region name. Defaults to **env[OS_REGION_NAME]**

--os-service-type=<service-type>

OpenStack Service type. Defaults to **env[OS_SERVICE_TYPE]**

--os-endpoint-type=<endpoint-type>

OpenStack Endpoint type. Defaults to **env[OS_ENDPOINT_TYPE]**

--os-cacert=<ca-certificate>

Specify a CA bundle file to use in verifying a TLS (https) server certificate. Defaults to **env[OS_CACERT]**

--insecure

Allow swiftclient to access servers without having to verify the SSL certificate. Defaults to **env[SWIFTCLIENT_INSECURE]** (set to 'true' to enable).

--no-ssl-compression

This option is deprecated and not used anymore. SSL compression should be disabled by default by the system SSL library

4. swift delete command

Usage: `swift delete`

Delete a container or objects within a container

Positional arguments

<container>

Name of container to delete from

[object]

Name of object to delete. Specify multiple times for multiple objects

Optional arguments

--all

Delete all containers and objects

--leave-segments

Do not delete segments of manifest objects

--object-threads <threads>

Number of threads to use for deleting objects. Default is 10

--container-threads <threads>

Number of threads to use for deleting containers. Default is 10

5. swift download command

Usage: swift download

Download objects from containers

Positional arguments

<container>

Name of container to download from. To download a whole account, omit this and specify --all.

[object]

Name of object to download. Specify multiple times for multiple objects. Omit this to download all objects from the container.

Optional arguments

--all

Indicates that you really want to download everything in the account

--marker

Marker to use when starting a container or account download

--prefix <prefix>

Only download items beginning with <prefix>

--output <out_file>

For a single file download, stream the output to <out_file>. Specifying "-" as <out_file> will redirect to stdout

--object-threads <threads>

Number of threads to use for downloading objects. Default is 10

--container-threads <threads>

Number of threads to use for downloading containers. Default is 10

--no-download

Perform download(s), but don't actually write anything to disk

--header <header_name:header_value>

Adds a customized request header to the query, like "Range" or "If-Match". This argument is repeatable. Example --header "content-type:text/plain"

--skip-identical

Skip downloading files that are identical on both sides

6. swift list command

Usage: swift list

Lists the containers for the account or the objects for a container

Positional arguments

[container]

Name of container to list object in

Optional arguments

--long

Long listing format, similar to ls -l

--lh

Report sizes in human readable format similar to ls -lh

--totals

Used with -l or --lh, only report totals

--prefix

Only list items beginning with the prefix

--delimiter

Roll up items with the given delimiter. For containers only. See OpenStack Swift API documentation for what this means.

7. swift post command

Usage: `swift post`

Updates meta information for the account, container, or object. If the container is not found, it will be created automatically.

Positional arguments

[container]

Name of container to post to

[object]

Name of object to post. Specify multiple times for multiple objects

Optional arguments

--read-acl <acl>

Read ACL for containers. Quick summary of ACL syntax: `.r:*`, `.r:-example.com`, `.r:www.example.com`, `account1`, `account2:user2`

--write-acl <acl>

Write ACL for containers. Quick summary of ACL syntax: `account1` `account2:user2`

--sync-to <sync-to>

Sync To for containers, for multi-cluster replication

--sync-key <sync-key>

Sync Key for containers, for multi-cluster replication

--meta <name:value>

Sets a meta data item. This option may be repeated. Example: `-m Color:Blue -m Size:Large`

--header <header>

Set request headers. This option may be repeated. Example `-H "content-type:text/plain"`

8. swift stat command

Usage: `swift stat`

Displays information for the account, container, or object

Positional arguments

[container]

Name of container to stat from

[object]

Name of object to stat. Specify multiple times for multiple objects

Optional arguments**--lh**

Report sizes in human readable format similar to `ls -lh`

9. swift upload command

Usage: `swift upload`

Uploads specified files and directories to the given container

Positional arguments**<container>**

Name of container to upload to

<file_or_directory>

Name of file or directory to upload. Specify multiple times for multiple uploads

Optional arguments**--changed**

Only upload files that have changed since the last upload

--skip-identical

Skip uploading files that are identical on both sides

--segment-size <size>

Upload files in segments no larger than `<size>` and then create a "manifest" file that will download all the segments as if it were the original file

--segment-container <container>

Upload the segments into the specified container. If not specified, the segments will be uploaded to a `<container>_segments` container so as to not pollute the main `<container>` listings.

--leave-segments

Indicates that you want the older segments of manifest objects left alone (in the case of overwrites)

--object-threads <threads>

Number of threads to use for uploading full objects. Default is 10.

--segment-threads <threads>

Number of threads to use for uploading object segments. Default is 10.

--header <header>

Set request headers with the syntax header:value. This option may be repeated.
Example -H "content-type:text/plain".

--use-slo

When used in conjunction with --segment-size will create a Static Large Object instead of the default Dynamic Large Object.

--object-name <object-name>

Upload file and name object to <object-name> or upload dir and use <object-name> as object prefix instead of folder name

Chapter 10. Orchestration command-line client

The **heat** client is the command-line interface (CLI) for the Orchestration API and its extensions. This chapter documents **heat** version 0.2.9.

For help on a specific **heat** command, enter:

```
$ heat help COMMAND
```

1. heat usage

```
usage: heat [--version] [-d] [-v] [-k] [--cert-file CERT_FILE] [--key-file KEY_FILE] [--ca-file CA_FILE] [--api-timeout API_TIMEOUT]
[--os-username OS_USERNAME] [--os-password OS_PASSWORD] [--os-tenant-id OS_TENANT_ID] [--os-tenant-name OS_TENANT_NAME] [--os-auth-url OS_AUTH_URL]
[--os-region-name OS_REGION_NAME] [--os-auth-token OS_AUTH_TOKEN] [--os-no-client-auth] [--heat-url HEAT_URL] [--heat-api-version HEAT_API_VERSION]
[--os-service-type OS_SERVICE_TYPE] [--os-endpoint-type OS_ENDPOINT_TYPE] [--include-password] <subcommand> ...
```

Subcommands

action-resume

Resume the stack.

action-suspend

Suspend the stack.

build-info

Retrieve build information.

create

DEPRECATED! Use stack-create instead.

delete

DEPRECATED! Use stack-delete instead.

describe

DEPRECATED! Use stack-show instead.

event

DEPRECATED! Use event-show instead.

event-list

List events for a stack.

event-show

Describe the event.

gettemplate

DEPRECATED! Use template-show instead.

list

DEPRECATED! Use stack-list instead.

output-list

Show available outputs.

output-show

Show a specific stack output.

resource

DEPRECATED! Use resource-show instead.

resource-list

Show list of resources belonging to a stack.

resource-metadata

List resource metadata.

resource-show

Describe the resource.

resource-signal

Send a signal to a resource.

resource-template

Generate a template based on a resource.

resource-type-list

List the available resource types.

resource-type-show

Show the resource type.

stack-abandon

Abandon the stack.

stack-adopt

Adopt a stack.

stack-create

Create the stack.

stack-delete

Delete the stack(s).

stack-list

List the user's stacks.

stack-preview

Preview the stack.

stack-show

Describe the stack.

stack-update

Update the stack.

template-show

Get the template for the specified stack.

template-validate

Validate a template with parameters.

update

DEPRECATED! Use stack-update instead.

validate

DEPRECATED! Use template-validate instead.

bash-completion

Prints all of the commands and options to stdout.

help

Display help about this program or one of its subcommands.

2. heat optional arguments

--version

Shows the client version and exits.

-d, --debug

Defaults to `env[HEATCLIENT_DEBUG]`.

-v, --verbose

Print more verbose output.

-k, --insecure

Explicitly allow the client to perform "insecure" SSL (https) requests. The server's certificate will not be verified against any certificate authorities. This option should be used with caution.

--cert-file CERT_FILE

Path of certificate file to use in SSL connection. This file can optionally be prepended with the private key.

--key-file KEY_FILE

Path of client key to use in SSL connection. This option is not necessary if your key is prepended to your cert file.

--ca-file CA_FILE

Path of CA SSL certificate(s) used to verify the remote server's certificate. Without this option the client looks for the default system CA certificates.

--api-timeout API_TIMEOUT

Number of seconds to wait for an API response, defaults to system socket timeout

--os-username OS_USERNAME

Defaults to `env[OS_USERNAME]`.

--os-password OS_PASSWORD

Defaults to `env[OS_PASSWORD]`.

--os-tenant-id OS_TENANT_ID

Defaults to `env[OS_TENANT_ID]`.

--os-tenant-name OS_TENANT_NAME

Defaults to `env[OS_TENANT_NAME]`.

--os-auth-url OS_AUTH_URL

Defaults to `env[OS_AUTH_URL]`.

--os-region-name OS_REGION_NAME

Defaults to `env[OS_REGION_NAME]`.

--os-auth-token OS_AUTH_TOKEN

Defaults to `env[OS_AUTH_TOKEN]`.

--os-no-client-auth

Do not contact keystone for a token. Defaults to `env[OS_NO_CLIENT_AUTH]`.

--heat-url HEAT_URL

Defaults to `env[HEAT_URL]`.

--heat-api-version HEAT_API_VERSION

Defaults to `env[HEAT_API_VERSION]` or 1.

--os-service-type OS_SERVICE_TYPE

Defaults to `env[OS_SERVICE_TYPE]`.

--os-endpoint-type OS_ENDPOINT_TYPE

Defaults to `env[OS_ENDPOINT_TYPE]`.

--include-password

Send os-username and os-password to heat.

3. heat action-resume command

usage: `heat action-resume <NAME or ID>`

Resume the stack.

Positional arguments

<NAME or ID>

Name or ID of stack to resume.

4. heat action-suspend command

usage: `heat action-suspend <NAME or ID>`

Suspend the stack.

Positional arguments

<NAME or ID>

Name or ID of stack to suspend.

5. heat build-info command

usage: `heat build-info`

Retrieve build information.

6. heat event-list command

usage: `heat event-list [-r <RESOURCE>] <NAME or ID>`

List events for a stack.

Positional arguments

<NAME or ID>

Name or ID of stack to show the events for.

Optional arguments

-r <RESOURCE>, --resource <RESOURCE>

Name of the resource to filter events by.

7. heat event-show command

usage: heat event-show <NAME or ID> <RESOURCE> <EVENT>

Describe the event.

Positional arguments

<NAME or ID>

Name or ID of stack to show the events for.

<RESOURCE>

Name of the resource the event belongs to.

<EVENT>

ID of event to display details for.

8. heat output-list command

usage: heat output-list <NAME or ID>

Show available outputs.

Positional arguments

<NAME or ID>

Name or ID of stack to query.

9. heat output-show command

usage: heat output-show <NAME or ID> <OUTPUT NAME>

Show a specific stack output.

Positional arguments

<NAME or ID>

Name or ID of stack to query.

<OUTPUT NAME>

Name of an output to display.

10. heat resource-list command

```
usage: heat resource-list <NAME or ID>
```

Show list of resources belonging to a stack.

Positional arguments

<NAME or ID>

Name or ID of stack to show the resources for.

11. heat resource-metadata command

```
usage: heat resource-metadata <NAME or ID> <RESOURCE>
```

List resource metadata.

Positional arguments

<NAME or ID>

Name or ID of stack to show the resource metadata for.

<RESOURCE>

Name of the resource to show the metadata for.

12. heat resource-show command

```
usage: heat resource-show <NAME or ID> <RESOURCE>
```

Describe the resource.

Positional arguments

<NAME or ID>

Name or ID of stack to show the resource for.

<RESOURCE>

Name of the resource to show the details for.

13. heat resource-signal command

```
usage: heat resource-signal [-D <DATA>] [-f <FILE>] <NAME or ID>  
<RESOURCE>
```

Send a signal to a resource.

Positional arguments

<NAME or ID>

Name or ID of stack the resource belongs to.

<RESOURCE>

Name of the resource to signal.

Optional arguments

-D <DATA>, --data <DATA>

JSON Data to send to the signal handler.

-f <FILE>, --data-file <FILE>

File containing JSON data to send to the signal handler.

14. heat resource-template command

```
usage: heat resource-template [-F <FORMAT>] <RESOURCE>
```

Generate a template based on a resource.

Positional arguments

<RESOURCE>

Name of the resource to generate a template for.

Optional arguments

-F <FORMAT>, --format <FORMAT>

The template output format, one of: yaml, json.

15. heat resource-type-list command

```
usage: heat resource-type-list
```

List the available resource types.

16. heat resource-type-show command

```
usage: heat resource-type-show <RESOURCE_TYPE>
```

Show the resource type.

Positional arguments

<RESOURCE_TYPE>

Resource type to get the details for.

17. heat stack-abandon command

```
usage: heat stack-abandon <NAME or ID>
```

Abandon the stack.

Positional arguments

<NAME or ID>

Name or ID of stack to abandon.

18. heat stack-adopt command

```
usage: heat stack-adopt [-f <FILE>] [-e <FILE or URL>] [-u <URL>] [-o
<URL>] [-c <TIMEOUT>] [-t <TIMEOUT>] [-a <FILE or URL>] [-r] [-P
<KEY1=VALUE1;KEY2=VALUE2...>] <STACK_NAME>
```

Adopt a stack.

Positional arguments

<STACK_NAME>

Name of the stack to adopt.

Optional arguments

-f <FILE>, --template-file <FILE>

Path to the template.

-e <FILE or URL>, --environment-file <FILE or URL>

Path to the environment.

-u <URL>, --template-url <URL>

URL of template.

-o <URL>, --template-object <URL>

URL to retrieve template object (e.g from swift).

-c <TIMEOUT>, --create-timeout <TIMEOUT>

Stack creation timeout in minutes. *DEPRECATED* use --timeout instead.

-t <TIMEOUT>, --timeout <TIMEOUT>

Stack creation timeout in minutes.

-a <FILE or URL>, --adopt-file <FILE or URL>

Path to adopt stack data file.

-r, --enable-rollback

Enable rollback on create/update failure.

**-P <KEY1=VALUE1;KEY2=VALUE2...>, --parameters
<KEY1=VALUE1;KEY2=VALUE2...>**

Parameter values used to create the stack. This can be specified multiple times, or once with parameters separated by a semicolon.

19. heat stack-create command

```
usage: heat stack-create [-f <FILE>] [-e <FILE or URL>] [-u <URL>] [-o <URL>] [-c <TIMEOUT>] [-t <TIMEOUT>] [-r] [-P <KEY1=VALUE1;KEY2=VALUE2...>] <STACK_NAME>
```

Create the stack.

Positional arguments

<STACK_NAME>

Name of the stack to create.

Optional arguments

-f <FILE>, --template-file <FILE>

Path to the template.

-e <FILE or URL>, --environment-file <FILE or URL>

Path to the environment.

-u <URL>, --template-url <URL>

URL of template.

-o <URL>, --template-object <URL>

URL to retrieve template object (e.g. from swift).

-c <TIMEOUT>, --create-timeout <TIMEOUT>

Stack creation timeout in minutes. *DEPRECATED* use --timeout instead.

-t <TIMEOUT>, --timeout <TIMEOUT>

Stack creation timeout in minutes.

-r, --enable-rollback

Enable rollback on create/update failure.

**-P <KEY1=VALUE1;KEY2=VALUE2...>, --parameters
<KEY1=VALUE1;KEY2=VALUE2...>**

Parameter values used to create the stack. This can be specified multiple times, or once with parameters separated by a semicolon.

20. heat stack-delete command

```
usage: heat stack-delete <NAME or ID> [<NAME or ID> ...]
```

Delete the stack(s).

Positional arguments

<NAME or ID>

Name or ID of stack(s) to delete.

21. heat stack-list command

```
usage: heat stack-list [-f <KEY1=VALUE1;KEY2=VALUE2...>] [-l <LIMIT>]
                        [-m <ID>]
```

List the user's stacks.

Optional arguments

-f <KEY1=VALUE1;KEY2=VALUE2...>, --filters <KEY1=VALUE1;KEY2=VALUE2...>

Filter parameters to apply on returned stacks. This can be specified multiple times, or once with parameters separated by a semicolon.

-l <LIMIT>, --limit <LIMIT>

Limit the number of stacks returned.

-m <ID>, --marker <ID>

Only return stacks that appear after the given stack ID.

22. heat stack-preview command

```
usage: heat stack-preview [-f <FILE>] [-e <FILE or URL>] [-u <URL>]
                        [-o <URL>] [-c <TIMEOUT>] [-r] [-P <KEY1=VALUE1;KEY2=VALUE2...>]
                        <STACK_NAME>
```

Preview the stack.

Positional arguments

<STACK_NAME>

Name of the stack to preview.

Optional arguments

-f <FILE>, --template-file <FILE>

Path to the template.

-e <FILE or URL>, --environment-file <FILE or URL>

Path to the environment.

-u <URL>, --template-url <URL>

URL of template.

-o <URL>, --template-object <URL>

URL to retrieve template object (e.g from swift)

-c <TIMEOUT>, --create-timeout <TIMEOUT>

Stack timeout in minutes. Default: 60

-r, --enable-rollback

Enable rollback on failure

**-P <KEY1=VALUE1;KEY2=VALUE2...>, --parameters
<KEY1=VALUE1;KEY2=VALUE2...>**

Parameter values used to preview the stack. This can be specified multiple times, or once with parameters separated by semicolon.

23. heat stack-show command

```
usage: heat stack-show <NAME or ID>
```

Describe the stack.

Positional arguments

<NAME or ID>

Name or ID of stack to describe.

24. heat stack-update command

```
usage: heat stack-update [-f <FILE>] [-e <FILE or URL>] [-u <URL>] [-o <URL>] [-t <TIMEOUT>] [-P <KEY1=VALUE1;KEY2=VALUE2...>] <NAME or ID>
```

Update the stack.

Positional arguments

<NAME or ID>

Name or ID of stack to update.

Optional arguments

-f <FILE>, --template-file <FILE>

Path to the template.

-e <FILE or URL>, --environment-file <FILE or URL>

Path to the environment.

-u <URL>, --template-url <URL>

URL of template.

-o <URL>, --template-object <URL>

URL to retrieve template object (e.g. from swift).

-t <TIMEOUT>, --timeout <TIMEOUT>

Stack update timeout in minutes.

**-P <KEY1=VALUE1;KEY2=VALUE2...>, --parameters
<KEY1=VALUE1;KEY2=VALUE2...>**

Parameter values used to create the stack. This can be specified multiple times, or once with parameters separated by a semicolon.

25. heat template-show command

usage: heat template-show <NAME or ID>

Get the template for the specified stack.

Positional arguments

<NAME or ID>

Name or ID of stack to get the template for.

26. heat template-validate command

usage: heat template-validate [-u <URL>] [-f <FILE>] [-e <FILE or
URL>] [-o <URL>] [-P <KEY1=VALUE1;KEY2=VALUE2...>]

Validate a template with parameters.

Optional arguments

-u <URL>, --template-url <URL>

URL of template.

-f <FILE>, --template-file <FILE>

Path to the template.

-e <FILE or URL>, --environment-file <FILE or URL>

Path to the environment.

-o <URL>, --template-object <URL>

URL to retrieve template object (e.g. from swift).

**-P <KEY1=VALUE1;KEY2=VALUE2 . . .>, --parameters
<KEY1=VALUE1;KEY2=VALUE2 . . .>**

Parameter values to validate. This can be specified multiple times, or once with parameters separated by a semicolon.

Chapter 11. Telemetry command-line client

The **ceilometer** client is the command-line interface (CLI) for the Telemetry API and its extensions. This chapter documents **ceilometer** version 1.0.10.

For help on a specific **ceilometer** command, enter:

```
$ ceilometer help COMMAND
```

1. ceilometer usage

```
usage: ceilometer [--version] [-d] [-v] [-k] [--cert-file CERT_FILE]
  [--key-file KEY_FILE] [--os-cacert <ca-certificate-file>] [--ca-file
  OS_CACERT] [--timeout TIMEOUT] [--os-username OS_USERNAME] [--os-
  password OS_PASSWORD] [--os-tenant-id OS_TENANT_ID] [--os-tenant-
  name OS_TENANT_NAME] [--os-auth-url OS_AUTH_URL] [--os-region-name
  OS_REGION_NAME] [--os-auth-token OS_AUTH_TOKEN] [--ceilometer-url
  CEILOMETER_URL] [--ceilometer-api-version CEILOMETER_API_VERSION] [-
  -os-service-type OS_SERVICE_TYPE] [--os-endpoint-type
  OS_ENDPOINT_TYPE] <subcommand> ...
```

Subcommands

alarm-combination-create

Create a new alarm based on state of other alarms.

alarm-combination-update

Update an existing alarm based on state of other alarms.

alarm-create

Create a new alarm (Deprecated). Use alarm-threshold- create instead.

alarm-delete

Delete an alarm.

alarm-history

Display the change history of an alarm.

alarm-list

List the user's alarms.

alarm-show

Show an alarm.

alarm-state-get

Get the state of an alarm.

alarm-state-set

Set the state of an alarm.

alarm-threshold-create

Create a new alarm based on computed statistics.

alarm-threshold-update

Update an existing alarm based on computed statistics.

alarm-update

Update an existing alarm (Deprecated).

event-list

List events.

event-show

Show a particular event.

event-type-list

List event types.

meter-list

List the user's meters.

query-alarm-history

Query Alarm History.

query-alarms

Query Alarms.

query-samples

Query samples.

resource-list

List the resources.

resource-show

Show the resource.

sample-create

Create a sample.

sample-list

List the samples for a meter.

statistics

List the statistics for a meter.

trait-description-list

List trait info for an event type.

trait-list

List trait all traits with name <trait_name> for Event Type

bash-completion

Prints all of the commands and options to stdout.

help

Display help about this program or one of its subcommands.

2. ceilometer optional arguments

--version

show program's version number and exit

-d, --debug

Defaults to `env[CEILOMETERCLIENT_DEBUG]`.

-v, --verbose

Print more verbose output.

-k, --insecure

Explicitly allow ceilometerclient to perform "insecure" SSL (https) requests. The server's certificate will not be verified against any certificate authorities. This option should be used with caution.

--cert-file CERT_FILE

Path of certificate file to use in SSL connection. This file can optionally be prepended with the private key.

--key-file KEY_FILE

Path of client key to use in SSL connection. This option is not necessary if your key is prepended to your cert file.

--os-cacert <ca-certificate-file>

Path of CA TLS certificate(s) used to verify the remote server's certificate. Without this option ceilometer looks for the default system CA certificates.

--ca-file OS_CACERT

DEPRECATED! Use `--os-cacert`.

--timeout TIMEOUT

Number of seconds to wait for a response.

--os-username OS_USERNAME

Defaults to `env[OS_USERNAME]`.

--os-password OS_PASSWORD
Defaults to `env[OS_PASSWORD]`.

--os-tenant-id OS_TENANT_ID
Defaults to `env[OS_TENANT_ID]`.

--os-tenant-name OS_TENANT_NAME
Defaults to `env[OS_TENANT_NAME]`.

--os-auth-url OS_AUTH_URL
Defaults to `env[OS_AUTH_URL]`.

--os-region-name OS_REGION_NAME
Defaults to `env[OS_REGION_NAME]`.

--os-auth-token OS_AUTH_TOKEN
Defaults to `env[OS_AUTH_TOKEN]`.

--ceilometer-url CEILOMETER_URL
Defaults to `env[CEILOMETER_URL]`.

--ceilometer-api-version CEILOMETER_API_VERSION
Defaults to `env[CEILOMETER_API_VERSION]` or 2.

--os-service-type OS_SERVICE_TYPE
Defaults to `env[OS_SERVICE_TYPE]`.

--os-endpoint-type OS_ENDPOINT_TYPE
Defaults to `env[OS_ENDPOINT_TYPE]`.

3. ceilometer alarm-combination-create command

```
usage: ceilometer alarm-combination-create --name <NAME> [--project-id <PROJECT_ID>] [--user-id <USER_ID>] [--description <DESCRIPTION>] [--state <STATE>] [--enabled {True|False}] [--alarm-action <Webhook URL>] [--ok-action <Webhook URL>] [--insufficient-data-action <Webhook URL>] [--time-constraint <Time Constraint>] --alarm_ids <ALARM_IDS> [--operator <OPERATOR>] [--repeat-actions {True|False}]
```

Create a new alarm based on state of other alarms.

Optional arguments

--name <NAME>
Name of the alarm (must be unique per tenant). Required.

--project-id <PROJECT_ID>

Tenant to associate with alarm (only settable by admin users).

--user-id <USER_ID>

User to associate with alarm (only settable by admin users).

--description <DESCRIPTION>

Free text description of the alarm.

--state <STATE>

State of the alarm, one of: ['ok', 'alarm', 'insufficient_data']

--enabled {True|False}

True if alarm evaluation/actioning is enabled.

--alarm-action <Webhook URL>

URL to invoke when state transitions to alarm. May be used multiple times. Defaults to None.

--ok-action <Webhook URL>

URL to invoke when state transitions to OK. May be used multiple times. Defaults to None.

--insufficient-data-action <Webhook URL>

URL to invoke when state transitions to insufficient_data. May be used multiple times. Defaults to None.

--time-constraint <Time Constraint>

Only evaluate the alarm if the time at evaluation is within this time constraint. Start point(s) of the constraint are specified with a cron expression , whereas its duration is given in seconds. Can be specified multiple times for multiple time constraints, format is: name=<CONSTRAINT_NAME>;start=< CRON>;duration=<SECONDS>;[description=<DESCRIPTION>;[t imezone=<IANA Timezone>]] Defaults to None.

--alarm_ids <ALARM_IDS>

List of alarm ids. Required.

--operator <OPERATOR>

Operator to compare with, one of: ['and', 'or'].

--repeat-actions {True|False}

True if actions should be repeatedly notified while alarm remains in target state. Defaults to False.

4. ceilometer alarm-combination-update command

```
usage: ceilometer alarm-combination-update -a <ALARM_ID> [--name
<NAME>] [--project-id <PROJECT_ID>] [--user-id <USER_ID>] [--
description <DESCRIPTION>] [--state <STATE>] [--enabled
{True|False}] [--alarm-action <Webhook URL>] [--ok-action <Webhook
```

```
URL>] [--insufficient-data-action <Webhook URL>] [--time-constraint  
<Time Constraint>] [--remove-time-constraint <Constraint names>] [--  
alarm_ids <ALARM_IDS>] [--operator <OPERATOR>] [--repeat-actions  
{True|False}]
```

Update an existing alarm based on state of other alarms.

Optional arguments

-a <ALARM_ID>, --alarm-id <ALARM_ID>

ID of the alarm to update. Required.

--name <NAME>

Name of the alarm (must be unique per tenant).

--project-id <PROJECT_ID>

Tenant to associate with alarm (only settable by admin users).

--user-id <USER_ID>

User to associate with alarm (only settable by admin users).

--description <DESCRIPTION>

Free text description of the alarm.

--state <STATE>

State of the alarm, one of: ['ok', 'alarm', 'insufficient_data']

--enabled {True|False}

True if alarm evaluation/actioning is enabled.

--alarm-action <Webhook URL>

URL to invoke when state transitions to alarm. May be used multiple times. Defaults to None.

--ok-action <Webhook URL>

URL to invoke when state transitions to OK. May be used multiple times. Defaults to None.

--insufficient-data-action <Webhook URL>

URL to invoke when state transitions to insufficient_data. May be used multiple times. Defaults to None.

--time-constraint <Time Constraint>

Only evaluate the alarm if the time at evaluation is within this time constraint. Start point(s) of the constraint are specified with a cron expression , whereas its duration is given in seconds. Can be specified multiple times for multiple time constraints, format is: name=<CONSTRAINT_NAME>;start=< CRON>;duration=<SECONDS>;[description=<DESCRIPTION>;[t imezone=<IANA Timezone>]] Defaults to None.

--remove-time-constraint <Constraint names>

Name or list of names of the time constraints to remove.

--alarm_ids <ALARM_IDS>

List of alarm id.

--operator <OPERATOR>

Operator to compare with, one of: ['and', 'or'].

--repeat-actions {True|False}

True if actions should be repeatedly notified while alarm remains in target state.

5. ceilometer alarm-delete command

```
usage: ceilometer alarm-delete -a <ALARM_ID>
```

Delete an alarm.

Optional arguments

-a <ALARM_ID>, --alarm_id <ALARM_ID>

ID of the alarm to delete. Required.

6. ceilometer alarm-history command

```
usage: ceilometer alarm-history -a <ALARM_ID> [-q <QUERY>]
```

Display the change history of an alarm.

Optional arguments

-a <ALARM_ID>, --alarm_id <ALARM_ID>

ID of the alarm for which history is shown. Required.

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float, or boolean.

7. ceilometer alarm-list command

```
usage: ceilometer alarm-list [-q <QUERY>]
```

List the user's alarms.

Optional arguments

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string,

integer, float, or boolean.

8. ceilometer alarm-show command

```
usage: ceilometer alarm-show -a <ALARM_ID>
```

Show an alarm.

Optional arguments

-a <ALARM_ID>, --alarm_id <ALARM_ID>

ID of the alarm to show. Required.

9. ceilometer alarm-state-get command

```
usage: ceilometer alarm-state-get -a <ALARM_ID>
```

Get the state of an alarm.

Optional arguments

-a <ALARM_ID>, --alarm_id <ALARM_ID>

ID of the alarm state to show. Required.

10. ceilometer alarm-state-set command

```
usage: ceilometer alarm-state-set -a <ALARM_ID> --state <STATE>
```

Set the state of an alarm.

Optional arguments

-a <ALARM_ID>, --alarm_id <ALARM_ID>

ID of the alarm state to set. Required.

--state <STATE>

State of the alarm, one of: ['ok', 'alarm', 'insufficient_data']. Required.

11. ceilometer alarm-threshold-create command

```
usage: ceilometer alarm-threshold-create --name <NAME> [--project-id <PROJECT_ID>] [--user-id <USER_ID>] [--description <DESCRIPTION>] [--state <STATE>] [--enabled {True|False}] [--alarm-action <Webhook URL>] [--ok-action <Webhook URL>] [--insufficient-data-action
```

```
<Webhook URL>] [--time-constraint <Time Constraint>] -m <METRIC> [--
period <PERIOD>] [--evaluation-periods <COUNT>] [--statistic
<STATISTIC>] [--comparison-operator <OPERATOR>] --threshold
<THRESHOLD> [-q <QUERY>] [--repeat-actions {True|False}]
```

Create a new alarm based on computed statistics.

Optional arguments

--name <NAME>

Name of the alarm (must be unique per tenant). Required.

--project-id <PROJECT_ID>

Tenant to associate with alarm (only settable by admin users).

--user-id <USER_ID>

User to associate with alarm (only settable by admin users).

--description <DESCRIPTION>

Free text description of the alarm.

--state <STATE>

State of the alarm, one of: ['ok', 'alarm', 'insufficient_data']

--enabled {True|False}

True if alarm evaluation/actioning is enabled.

--alarm-action <Webhook URL>

URL to invoke when state transitions to alarm. May be used multiple times. Defaults to None.

--ok-action <Webhook URL>

URL to invoke when state transitions to OK. May be used multiple times. Defaults to None.

--insufficient-data-action <Webhook URL>

URL to invoke when state transitions to insufficient_data. May be used multiple times. Defaults to None.

--time-constraint <Time Constraint>

Only evaluate the alarm if the time at evaluation is within this time constraint. Start point(s) of the constraint are specified with a cron expression , whereas its duration is given in seconds. Can be specified multiple times for multiple time constraints, format is: name=<CONSTRAINT_NAME>;start=< CRON>;duration=<SECONDS>;[description=<DESCRIPTION>;[t imezone=<IANA Timezone>]] Defaults to None.

-m <METRIC>, --meter-name <METRIC>

Metric to evaluate against. Required.

--period <PERIOD>

Length of each period (seconds) to evaluate over.

--evaluation-periods <COUNT>

Number of periods to evaluate over.

--statistic <STATISTIC>

Statistic to evaluate, one of: ['max', 'min', 'avg', 'sum', 'count'].

--comparison-operator <OPERATOR>

Operator to compare with, one of: ['lt', 'le', 'eq', 'ne', 'ge', 'gt'].

--threshold <THRESHOLD>

Threshold to evaluate against. Required.

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float, or boolean.

--repeat-actions {True|False}

True if actions should be repeatedly notified while alarm remains in target state.
Defaults to False.

12. ceilometer alarm-threshold-update command

```
usage: ceilometer alarm-threshold-update -a <ALARM_ID> [--name
<NAME>] [--project-id <PROJECT_ID>] [--user-id <USER_ID>] [--
description <DESCRIPTION>] [--state <STATE>] [--enabled
{True|False}] [--alarm-action <Webhook URL>] [--ok-action <Webhook
URL>] [--insufficient-data-action <Webhook URL>] [--time-constraint
<Time Constraint>] [--remove-time-constraint <Constraint names>] [-m
<METRIC>] [--period <PERIOD>] [--evaluation-periods <COUNT>] [--
statistic <STATISTIC>] [--comparison-operator <OPERATOR>] [--
threshold <THRESHOLD>] [-q <QUERY>] [--repeat-actions {True|False}]
```

Update an existing alarm based on computed statistics.

Optional arguments

-a <ALARM_ID>, --alarm_id <ALARM_ID>

ID of the alarm to update. Required.

--name <NAME>

Name of the alarm (must be unique per tenant).

--project-id <PROJECT_ID>

Tenant to associate with alarm (only settable by admin users).

--user-id <USER_ID>

User to associate with alarm (only settable by admin users).

--description <DESCRIPTION>

Free text description of the alarm.

--state <STATE>

State of the alarm, one of: ['ok', 'alarm', 'insufficient_data']

--enabled {True|False}

True if alarm evaluation/actioning is enabled.

--alarm-action <Webhook URL>

URL to invoke when state transitions to alarm. May be used multiple times. Defaults to None.

--ok-action <Webhook URL>

URL to invoke when state transitions to OK. May be used multiple times. Defaults to None.

--insufficient-data-action <Webhook URL>

URL to invoke when state transitions to insufficient_data. May be used multiple times. Defaults to None.

--time-constraint <Time Constraint>

Only evaluate the alarm if the time at evaluation is within this time constraint. Start point(s) of the constraint are specified with a cron expression , whereas its duration is given in seconds. Can be specified multiple times for multiple time constraints, format is: name=<CONSTRAINT_NAME>;start=< CRON>;duration=<SECONDS>;[description=<DESCRIPTION>;[t imezone=<IANA Timezone>]] Defaults to None.

--remove-time-constraint <Constraint names>

Name or list of names of the time constraints to remove.

-m <METRIC>, --meter-name <METRIC>

Metric to evaluate against.

--period <PERIOD>

Length of each period (seconds) to evaluate over.

--evaluation-periods <COUNT>

Number of periods to evaluate over.

--statistic <STATISTIC>

Statistic to evaluate, one of: ['max', 'min', 'avg', 'sum', 'count'].

--comparison-operator <OPERATOR>

Operator to compare with, one of: ['lt', 'le', 'eq', 'ne', 'ge', 'gt'].

--threshold <THRESHOLD>

Threshold to evaluate against.

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float, or boolean.

--repeat-actions {True|False}

True if actions should be repeatedly notified while alarm remains in target state.

13. ceilometer alarm-update command

```
usage: ceilometer alarm-update -a <ALARM_ID> [--name <NAME>] [--project-id <PROJECT_ID>] [--user-id <USER_ID>] [--description <DESCRIPTION>] [--state <STATE>] [--enabled {True|False}] [--alarm-action <Webhook URL>] [--ok-action <Webhook URL>] [--insufficient-data-action <Webhook URL>] [--time-constraint <Time Constraint>] [--remove-time-constraint <Constraint names>] [--period <PERIOD>] [--evaluation-periods <COUNT>] [-m <METRIC>] [--statistic <STATISTIC>] [--comparison-operator <OPERATOR>] [--threshold <THRESHOLD>] [--matching-metadata <Matching Metadata>] [--repeat-actions {True|False}]
```

Update an existing alarm (Deprecated).

Optional arguments

-a <ALARM_ID>, --alarm_id <ALARM_ID>

ID of the alarm to update. Required.

--name <NAME>

Name of the alarm (must be unique per tenant).

--project-id <PROJECT_ID>

Tenant to associate with alarm (only settable by admin users).

--user-id <USER_ID>

User to associate with alarm (only settable by admin users).

--description <DESCRIPTION>

Free text description of the alarm.

--state <STATE>

State of the alarm, one of: ['ok', 'alarm', 'insufficient_data']

--enabled {True|False}

True if alarm evaluation/actioning is enabled.

--alarm-action <Webhook URL>

URL to invoke when state transitions to alarm. May be used multiple times. Defaults to None.

--ok-action <Webhook URL>

URL to invoke when state transitions to OK. May be used multiple times. Defaults to None.

--insufficient-data-action <Webhook URL>

URL to invoke when state transitions to insufficient_data. May be used multiple times. Defaults to None.

--time-constraint <Time Constraint>

Only evaluate the alarm if the time at evaluation is within this time constraint. Start point(s) of the constraint are specified with a cron expression, whereas its duration is given in seconds. Can be specified multiple times for multiple time constraints, format is: name=<CONSTRAINT_NAME>;start=<CRON>;duration=<SECONDS>;[description=<DESCRIPTION>;[timezone=<IANA Timezone>]] Defaults to None.

--remove-time-constraint <Constraint names>

Name or list of names of the time constraints to remove.

--period <PERIOD>

Length of each period (seconds) to evaluate over.

--evaluation-periods <COUNT>

Number of periods to evaluate over.

-m <METRIC>, --meter-name <METRIC>

Metric to evaluate against.

--statistic <STATISTIC>

Statistic to evaluate, one of: ['max', 'min', 'avg', 'sum', 'count']

--comparison-operator <OPERATOR>

Operator to compare with, one of: ['lt', 'le', 'eq', 'ne', 'ge', 'gt'].

--threshold <THRESHOLD>

Threshold to evaluate against.

--matching-metadata <Matching Metadata>

A meter should match this resource metadata (key=value) additionally to the meter_name. Defaults to None.

--repeat-actions {True|False}

True if actions should be repeatedly notified while alarm remains in target state.

14. ceilometer event-list command

```
usage: ceilometer event-list [-q <QUERY>]
```

List events.

Optional arguments

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float or datetime.

15. ceilometer event-show command

usage: ceilometer event-show -m <message_id>

Show a particular event.

Optional arguments

-m <message_id>, --message_id <message_id>

The id of the event. Should be a UUID Required.

16. ceilometer event-type-list command

usage: ceilometer event-type-list

List event types.

17. ceilometer meter-list command

usage: ceilometer meter-list [-q <QUERY>]

List the user's meters.

Optional arguments

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float, or boolean.

18. ceilometer query-alarm-history command

**usage: ceilometer query-alarm-history [-f <FILTER>] [-o <ORDERBY>]
[-l <LIMIT>]**

Query Alarm History.

Optional arguments

-f <FILTER>, --filter <FILTER>

{complex_op: [{simple_op: {field_name: value}}]} The complex_op is one of: ['and', 'or'], simple_op is one of: ['=', '!=', '<', '<=', '>', '>='].

-o <ORDERBY>, --orderby <ORDERBY>

{field_name: direction}, {field_name: direction}} The direction is one of: ['asc', 'desc'].

-l <LIMIT>, --limit <LIMIT>

Maximum number of alarm history items to return.

19. ceilometer query-alarms command

```
usage: ceilometer query-alarms [-f <FILTER>] [-o <ORDERBY>] [-l
<LIMIT>]
```

Query Alarms.

Optional arguments

-f <FILTER>, --filter <FILTER>

{complex_op: [{simple_op: {field_name: value}}]} The complex_op is one of: ['and', 'or'], simple_op is one of: ['=', '!=', '<', '<=', '>', '>='].

-o <ORDERBY>, --orderby <ORDERBY>

{field_name: direction}, {field_name: direction}} The direction is one of: ['asc', 'desc'].

-l <LIMIT>, --limit <LIMIT>

Maximum number of alarms to return.

20. ceilometer query-samples command

```
usage: ceilometer query-samples [-f <FILTER>] [-o <ORDERBY>] [-l
<LIMIT>]
```

Query samples.

Optional arguments

-f <FILTER>, --filter <FILTER>

{complex_op: [{simple_op: {field_name: value}}]} The complex_op is one of: ['and', 'or'], simple_op is one of: ['=', '!=', '<', '<=', '>', '>='].

-o <ORDERBY>, --orderby <ORDERBY>

{field_name: direction}, {field_name: direction}} The direction is one of: ['asc', 'desc'].

-l <LIMIT>, --limit <LIMIT>

Maximum number of samples to return.

21. ceilometer resource-list command

```
usage: ceilometer resource-list [-q <QUERY>]
```

List the resources.

Optional arguments

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float, or boolean.

22. ceilometer resource-show command

```
usage: ceilometer resource-show -r <RESOURCE_ID>
```

Show the resource.

Optional arguments

-r <RESOURCE_ID>, --resource_id <RESOURCE_ID>

ID of the resource to show. Required.

23. ceilometer sample-create command

```
usage: ceilometer sample-create [--project-id <PROJECT_ID>] [--user-id <USER_ID>] -r <RESOURCE_ID> -m <METER_NAME> --meter-type <METER_TYPE> --meter-unit <METER_UNIT> --sample-volume <SAMPLE_VOLUME> [--resource-metadata <RESOURCE_METADATA>] [--timestamp <TIMESTAMP>]
```

Create a sample.

Optional arguments

--project-id <PROJECT_ID>

Tenant to associate with sample (only settable by admin users).

--user-id <USER_ID>

User to associate with sample (only settable by admin users).

-r <RESOURCE_ID>, --resource-id <RESOURCE_ID>

ID of the resource. Required.

-m <METER_NAME>, --meter-name <METER_NAME>

The meter name. Required.

--meter-type <METER_TYPE>

The meter type. Required.

--meter-unit <METER_UNIT>

The meter unit. Required.

--sample-volume <SAMPLE_VOLUME>

The sample volume. Required.

--resource-metadata <RESOURCE_METADATA>

Resource metadata.

--timestamp <TIMESTAMP>

The sample timestamp.

24. ceilometer sample-list command

```
usage: ceilometer sample-list [-q <QUERY>] -m <NAME> [-l <NUMBER>]
```

List the samples for a meter.

Optional arguments

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float, or boolean.

-m <NAME>, --meter <NAME>

Name of meter to show samples for. Required.

-l <NUMBER>, --limit <NUMBER>

Maximum number of samples to return.

25. ceilometer statistics command

```
usage: ceilometer statistics [-q <QUERY>] -m <NAME> [-p <PERIOD>] [-g <FIELD>] [-a <FUNC>[<-<PARAM>]]
```

List the statistics for a meter.

Optional arguments

-q <QUERY>, --query <QUERY>

key[op]data_type::value; list. data_type is optional, but if supplied must be string, integer, float, or boolean.

-m <NAME>, --meter <NAME>

Name of meter to show samples for. Required.

-p <PERIOD>, --period <PERIOD>

Period in seconds over which to group samples.

-g <FIELD>, --groupby <FIELD>

Field for group by.

-a <FUNC>[<-<PARAM>], --aggregate <FUNC>[<-<PARAM>]

Function for data aggregation. Available aggregates are: count, cardinality, min, max, sum, stddev, avg. Defaults to [].

26. ceilometer trait-description-list command

usage: ceilometer trait-description-list -e <EVENT_TYPE>

List trait info for an event type.

Optional arguments

-e <EVENT_TYPE>, --event_type <EVENT_TYPE>

Type of the event for which traits will be shown. Required.

27. ceilometer trait-list command

usage: ceilometer trait-list -e <EVENT_TYPE> -t <TRAIT_NAME>

List trait all traits with name <trait_name> for Event Type <event_type>.

Optional arguments

-e <EVENT_TYPE>, --event_type <EVENT_TYPE>

Type of the event for which traits will listed. Required.

-t <TRAIT_NAME>, --trait_name <TRAIT_NAME>

The name of the trait to list. Required.

Chapter 12. Database Service command-line client

The **trove** client is the command-line interface (CLI) for the Database API and its extensions. This chapter documents **trove** version 1.0.5.

For help on a specific **trove** command, enter:

```
$ trove help COMMAND
```

1. trove usage

```
usage: trove [--version] [--debug] [--os-username <auth-user-name>]
  [--os-password <auth-password>] [--os-tenant-name <auth-tenant-
  name>] [--os-tenant-id <auth-tenant-id>] [--os-auth-url <auth-url>]
  [--os-region-name <region-name>] [--service-type <service-type>] [--
  service-name <service-name>] [--bypass-url <bypass-url>] [--
  database-service-name <database-service-name>] [--endpoint-type
  <endpoint-type>] [--os-database-api-version <database-api-ver>] [--
  os-cacert <ca-certificate>] [--retries <retries>] [--json]
  <subcommand> ...
```

Subcommands

backup-create

Creates a backup.

backup-delete

Deletes a backup.

backup-list

Lists available backups.

backup-list-instance

Lists available backups for an instance.

backup-show

Shows details of a backup.

configuration-attach

Attaches a configuration group to an instance.

configuration-create

Creates a configuration group.

configuration-default

Shows the default configuration of an instance.

configuration-delete

Deletes a configuration group.

configuration-detach

Detaches a configuration group from an instance.

configuration-instances

Lists all instances associated with a configuration group.

configuration-list

Lists all configuration groups.

configuration-parameter-list

Lists available parameters for a configuration group.

configuration-parameter-show

Shows details of a configuration parameter.

configuration-patch

Patches a configuration group.

configuration-show

Shows details of a configuration group.

configuration-update

Updates a configuration group.

create

Creates a new instance.

database-create

Creates a database on an instance.

database-delete

Deletes a database.

database-list

Lists available databases on an instance.

datastore-list

Lists available datastores.

datastore-show

Shows details of a datastore.

datastore-version-list

Lists available versions for a datastore.

datastore-version-show

Shows details of a datastore version.

delete

Deletes an instance.

flavor-list

Lists available flavors.

flavor-show

Shows details of a flavor.

limit-list

Lists the limits for a tenant.

list

Lists all the instances.

resize-flavor

Resizes the flavor of an instance.

resize-volume

Resizes the volume size of an instance.

restart

Restarts the instance.

root-enable

Enables root for a instance.

root-show

Gets root enabled status for a instance.

secgroup-add-rule

Creates a security group rule.

secgroup-delete-rule

Deletes a security group rule.

secgroup-list

Lists all security groups.

secgroup-show

Shows details about a security group.

show

Shows details of an instance.

user-create

Creates a user.

user-delete

Deletes a user from the instance.

user-grant-access

Grants access to a database(s) for a user.

user-list

Lists the users for a instance.

user-revoke-access

Revokes access to a database for a user.

user-show

Gets a user from the instance.

user-show-access

Gets a users access from the instance.

user-update-attributes

Updates a users attributes from the instance.

bash-completion

Prints arguments for bash_completion.

help

Displays help about this program or one of its subcommands.

2. trove optional arguments

--version

show program's version number and exit

--debug

Print debugging output.

--os-username <auth-user-name>

Defaults to `env[OS_USERNAME]`.

--os-password <auth-password>

Defaults to `env[OS_PASSWORD]`.

--os-tenant-name <auth-tenant-name>

Defaults to `env[OS_TENANT_NAME]`.

--os-tenant-id <auth-tenant-id>

Defaults to `env[OS_TENANT_ID]`.

--os-auth-url <auth-url>

Defaults to `env[OS_AUTH_URL]`.

--os-region-name <region-name>

Defaults to `env[OS_REGION_NAME]`.

--service-type <service-type>

Defaults to database for most actions.

--service-name <service-name>

Defaults to `env[TROVE_SERVICE_NAME]`.

--bypass-url <bypass-url>

Defaults to `env[TROVE_BYPASS_URL]`.

--database-service-name <database-service-name>

Defaults to `env[TROVE_DATABASE_SERVICE_NAME]`.

--endpoint-type <endpoint-type>

Defaults to `env[TROVE_ENDPOINT_TYPE]` or `publicURL`.

--os-database-api-version <database-api-ver>

Accepts 1, defaults to `env[OS_DATABASE_API_VERSION]`.

--os-cacert <ca-certificate>

Specify a CA bundle file to use in verifying a TLS (https) server certificate. Defaults to `env[OS_CACERT]`.

--retries <retries>

Number of retries.

--json, --os-json-output

Output json instead of prettyprint. Defaults to `env[OS_JSON_OUTPUT]`.

3. trove backup-create command

```
usage: trove backup-create <name> <instance> [--description
<description>] [--parent <parent>]
```

Creates a backup.

Positional arguments

<name>

Name of the backup.

<instance>

UUID of the instance.

Optional arguments

--description <description>

An optional description for the backup.

--parent <parent>

Optional UUID of the parent backup to perform an incremental backup from.

4. trove backup-delete command

```
usage: trove backup-delete <backup>
```

Deletes a backup.

Positional arguments

<backup>

ID of the backup.

5. trove backup-list command

```
usage: trove backup-list [--limit <limit>]
```

Lists available backups.

Optional arguments

--limit <limit>

Return up to N number of the most recent backups.

6. trove backup-list-instance command

```
usage: trove backup-list-instance [--limit <limit>] <instance>
```

Lists available backups for an instance.

Positional arguments

<instance>

ID of the instance.

Optional arguments

--limit <limit>

Return up to N number of the most recent backups.

7. trove backup-show command

usage: trove backup-show <backup>

Shows details of a backup.

Positional arguments

<backup>

ID of the backup.

8. trove configuration-attach command

usage: trove configuration-attach <configuration> <instance>

Attaches a configuration group to an instance.

Positional arguments

<configuration>

UUID of the configuration group to attach to the instance.

<instance>

UUID of the instance.

9. trove configuration-create command

usage: trove configuration-create <name> <values> [--datastore
<datastore>] [--datastore_version <datastore_version>] [--
description <description>]

Creates a configuration group.

Positional arguments

<name>

Name of the configuration group.

<values>

Dictionary of the values to set.

Optional arguments

--datastore <datastore>

Datastore assigned to the configuration group.

--datastore_version <datastore_version>

Datastore version UUID assigned to the configuration group.

--description <description>

An optional description for the configuration group.

10. trove configuration-default command

```
usage: trove configuration-default <instance>
```

Shows the default configuration of an instance.

Positional arguments

<instance>

UUID of the instance.

11. trove configuration-delete command

```
usage: trove configuration-delete <configuration_group>
```

Deletes a configuration group.

Positional arguments

<configuration_group>

ID of the configuration group.

12. trove configuration-detach command

```
usage: trove configuration-detach <instance>
```

Detaches a configuration group from an instance.

Positional arguments

<instance>

UUID of the instance.

13. trove configuration-instances command

```
usage: trove configuration-instances <configuration_group>
```

Lists all instances associated with a configuration group.

Positional arguments

<configuration_group>

ID of the configuration group.

14. trove configuration-list command

```
usage: trove configuration-list
```

Lists all configuration groups.

15. trove configuration-parameter-list command

```
usage: trove configuration-parameter-list <datastore_version> [--  
datastore <datastore>]
```

Lists available parameters for a configuration group.

Positional arguments

<datastore_version>

Datastore version name or UUID assigned to the configuration group.

Optional arguments

--datastore <datastore>

UUID or name of the datastore to list configuration parameters for. Optional if UUID of the datastore_version is provided.

16. trove configuration-parameter-show command

```
usage: trove configuration-parameter-show <datastore_version>  
<parameter> [--datastore <datastore>]
```

Shows details of a configuration parameter.

Positional arguments

<datastore_version>

Datastore version name or UUID assigned to the configuration group.

<parameter>

Name of the configuration parameter.

Optional arguments

--datastore <datastore>

UUID or name of the datastore to list configuration parameters for. Optional if UUID of the `datastore_version` is provided.

17. trove configuration-patch command

```
usage: trove configuration-patch <configuration_group> <values>
```

Patches a configuration group.

Positional arguments

<configuration_group>

ID of the configuration group.

<values>

Dictionary of the values to set.

18. trove configuration-show command

```
usage: trove configuration-show <configuration_group>
```

Shows details of a configuration group.

Positional arguments

<configuration_group>

ID of the configuration group.

19. trove configuration-update command

```
usage: trove configuration-update <configuration_group> <values> [--name <name>] [--description <description>]
```

Updates a configuration group.

Positional arguments

<configuration_group>

ID of the configuration group.

<values>

Dictionary of the values to set.

Optional arguments

--name <name>

Name of the configuration group.

--description <description>

An optional description for the configuration group.

20. trove create command

```
usage: trove create <name> <flavor_id> [--size <size>] [--databases
<databases> [<databases> ...]] [--users <users> [<users> ...]] [--
backup <backup>] [--availability_zone <availability_zone>] [--
datastore <datastore>] [--datastore_version <datastore_version>] [--
nic <net-id=net-uuid,v4-fixed-ip=ip-addr,port-id=port-uuid>] [--
configuration <configuration>]
```

Creates a new instance.

Positional arguments

<name>

Name of the instance.

<flavor_id>

Flavor of the instance.

Optional arguments

--size <size>

Size of the instance disk in GB.

--databases <databases> [<databases> ...]

Optional list of databases.

--users <users> [<users> ...]

Optional list of users in the form user:password.

--backup <backup> A

backup UUID.

--availability_zone <availability_zone>

The Zone hint to give to nova.

--datastore <datastore>

A datastore name or UUID.

--datastore_version <datastore_version>

A datastore version name or UUID.

--nic <net-id=net-uuid,v4-fixed-ip=ip-addr,port-id=port-uuid>

Create a NIC on the instance. Specify option multiple times to create multiple NICs.
net-id: attach NIC to network with this UUID (required, if no port-id specified), v4-fixed-ip: IPv4 fixed address for NIC (optional), port-id: attach NIC to port with this UUID (required, if no net-id specified).

--configuration <configuration>

UUID of the configuration group to attach to the instance.

21. trove database-create command

```
usage: trove database-create <instance> <name> [--character_set
<character_set>] [--collate <collate>]
```

Creates a database on an instance.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of the database.

Optional arguments

--character_set <character_set>

Optional character set for database.

--collate <collate>

Optional collation type for database.

22. trove database-delete command

```
usage: trove database-delete <instance> <database>
```

Deletes a database.

Positional arguments

<instance>

UUID of the instance.

<database>

Name of the database.

23. trove database-list command


```
usage: trove database-list <instance>
```

Lists available databases on an instance.

Positional arguments

<instance>

UUID of the instance.

24. trove datastore-list command

```
usage: trove datastore-list
```

Lists available datastores.

25. trove datastore-show command

```
usage: trove datastore-show <datastore>
```

Shows details of a datastore.

Positional arguments

<datastore>

ID of the datastore.

26. trove datastore-version-list command

```
usage: trove datastore-version-list <datastore>
```

Lists available versions for a datastore.

Positional arguments

<datastore>

ID of the datastore.

27. trove datastore-version-show command

```
usage: trove datastore-version-show <datastore_version> [--datastore  
<datastore>]
```

Shows details of a datastore version.

Positional arguments

<datastore_version>

ID of the datastore version.

Optional arguments

--datastore <datastore>

ID or name of the datastore. Optional if UUID of the datastore_version is provided.

28. trove delete command

usage: trove delete <instance>

Deletes an instance.

Positional arguments

<instance>

ID of the instance.

29. trove flavor-list command

usage: trove flavor-list

Lists available flavors.

30. trove flavor-show command

usage: trove flavor-show <flavor>

Shows details of a flavor.

Positional arguments

<flavor>

ID of the flavor.

31. trove limit-list command

usage: trove limit-list

Lists the limits for a tenant.

32. trove list command

usage: trove list

Lists all the instances.

33. trove resize-flavor command

```
usage: trove resize-flavor <instance> <flavor_id>
```

Resizes the flavor of an instance.

Positional arguments

<instance>

UUID of the instance.

<flavor_id>

Flavor of the instance.

34. trove resize-volume command

```
usage: trove resize-volume <instance> <size>
```

Resizes the volume size of an instance.

Positional arguments

<instance>

UUID of the instance.

<size>

Size of the instance disk in GB.

35. trove restart command

```
usage: trove restart <instance>
```

Restarts the instance.

Positional arguments

<instance>

UUID of the instance.

36. trove root-enable command

```
usage: trove root-enable <instance>
```

Enables root for a instance.

Positional arguments

<instance>

UUID of the instance.

37. trove root-show command

```
usage: trove root-show <instance>
```

Gets root enabled status for a instance.

Positional arguments

<instance>

UUID of the instance.

38. trove secgroup-add-rule command

```
usage: trove secgroup-add-rule <security_group> <cidr>
```

Creates a security group rule.

Positional arguments

<security_group>

Security group name.

<cidr>

CIDR address.

39. trove secgroup-delete-rule command

```
usage: trove secgroup-delete-rule <security_group_rule>
```

Deletes a security group rule.

Positional arguments

<security_group_rule>

Name of security group rule.

40. trove secgroup-list command

```
usage: trove secgroup-list
```

Lists all security groups.

41. trove secgroup-show command

```
usage: trove secgroup-show <security_group>
```

Shows details about a security group.

Positional arguments

<security_group>

ID of the security group.

42. trove show command

```
usage: trove show <instance>
```

Shows details of an instance.

Positional arguments

<instance>

ID of the instance.

43. trove user-create command

```
usage: trove user-create <instance> <name> <password> [--host
<host>] [--databases <databases> [<databases> ...]]
```

Creates a user.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of user.

<password>

Password of user.

Optional arguments

--host <host>

Optional host of user.

--databases <databases> [<databases> ...]

Optional list of databases.

44. trove user-delete command

```
usage: trove user-delete [--host <host>] <instance> <name>
```

Deletes a user from the instance.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of user.

Optional arguments

--host <host>

Optional host of user.

45. trove user-grant-access command

```
usage: trove user-grant-access <instance> <name> <databases>
[<databases> ...] [--host <host>]
```

Grants access to a database(s) for a user.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of user.

<databases>

List of databases.

Optional arguments

--host <host>

Optional host of user.

46. trove user-list command

```
usage: trove user-list <instance>
```

Lists the users for a instance.

Positional arguments

<instance>

UUID of the instance.

47. trove user-revoke-access command

```
usage: trove user-revoke-access [--host <host>] <instance> <name>
<database>
```

Revokes access to a database for a user.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of user.

<database>

A single database.

Optional arguments

--host <host>

Optional host of user.

48. trove user-show command

```
usage: trove user-show [--host <host>] <instance> <name>
```

Gets a user from the instance.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of user.

Optional arguments

--host <host>

Optional host of user.

49. trove user-show-access command

```
usage: trove user-show-access [--host <host>] <instance> <name>
```

Gets a users access from the instance.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of user.

Optional arguments

--host <host>

Optional host of user.

50. trove user-update-attributes command

```
usage: trove user-update-attributes <instance> <name> [--host  
<host>] [--new_name <new_name>] [--new_password <new_password>] [--  
new_host <new_host>]
```

Updates a users attributes from the instance.

Positional arguments

<instance>

UUID of the instance.

<name>

Name of user.

Optional arguments

--host <host>

Optional host of user.

--new_name <new_name>

Optional new name of user.

--new_password <new_password>

Optional new password of user.

--new_host <new_host>

Optional new host of user.

Revision History

Revision 5.0.0-9	Mon Jul 7 2014	Martin Lopes
Final version for Red Hat Enterprise Linux OpenStack Platform 5.		
Revision 5.0.0-5	Wed Jun 25 2014	Martin Lopes
BZ#1110080 - Applied updated brand package.		
Revision 5.0.0-4	Tue Jun 17 2014	Martin Lopes
BZ#1110080 - Applied Docbook 5 compatibility settings.		
Revision 5.0.0-2	Fri May 20 2014	Martin Lopes
Initial publication for Red Hat Enterprise Linux OpenStack Platform 5.		