OpenShift Container Platform 4.8

Release notes

Highlights of what is new and what has changed with this OpenShift Container Platform release

Last Updated: 2021-10-14

OpenShift Container Platform 4.8 Release notes

Highlights of what is new and what has changed with this OpenShift Container Platform release

Legal Notice

Abstract

The release notes for OpenShift Container Platform summarize all new features and enhancements, notable technical changes, major corrections from the previous version, and any known bugs upon general availability.

Table of Contents

CHAPTER 1. OPENSHIFT CONTAINER PLATFORM 4.8 RELEASE NOTES

Red Hat OpenShift Container Platform provides developers and IT organizations with a hybrid cloud application platform for deploying both new and existing applications on secure, scalable resources with minimal configuration and management overhead. OpenShift Container Platform supports a wide selection of programming languages and frameworks, such as Java, JavaScript, Python, Ruby, and PHP.

Built on Red Hat Enterprise Linux (RHEL) and Kubernetes, OpenShift Container Platform provides a more secure and scalable multi-tenant operating system for today's enterprise-class applications, while delivering integrated application runtimes and libraries. OpenShift Container Platform enables organizations to meet security, privacy, compliance, and governance requirements.

1.1. ABOUT THIS RELEASE

OpenShift Container Platform (RHSA-2021:2438) is now available. This release uses Kubernetes 1.21 with CRI-O runtime. New features, changes, and known issues that pertain to OpenShift Container Platform 4.8 are included in this topic.

Red Hat did not publicly release OpenShift Container Platform 4.8.0 as the GA version and, instead, is releasing OpenShift Container Platform 4.8.2 as the GA version.

OpenShift Container Platform 4.8 clusters are available at https://cloud.redhat.com/openshift. The Red Hat OpenShift Cluster Manager application for OpenShift Container Platform allows you to deploy OpenShift clusters to either on-premise or cloud environments.

OpenShift Container Platform 4.8 is supported on Red Hat Enterprise Linux (RHEL) 7.9 or later, as well as on Red Hat Enterprise Linux CoreOS (RHCOS) 4.8.

You must use RHCOS machines for the control plane, and you can use either RHCOS or Red Hat Enterprise Linux (RHEL) 7.9 or later for compute machines.

IMPORTANT

Because only RHEL 7.9 or later is supported for compute machines, you must not upgrade the RHEL compute machines to RHEL 8.

With the release of OpenShift Container Platform 4.8, version 4.5 is now end of life. For more information, see the Red Hat OpenShift Container Platform Life Cycle Policy .

1.2. MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties.

As part of that effort, with this release the following changes are in place:

- The OpenShift Docs GitHub repository **master** branch has been renamed to **main**.
- We have begun to progressively replace the terminology of "master" with "control plane". You will notice throughout the documentation that we use both terms, with "master" in parenthesis. For example "... the control plane node (also known as the master node)". In a future release, we will update this to be "the control plane node".

1.3. OPENSHIFT CONTAINER PLATFORM LAYERED AND DEPENDANT COMPONENT SUPPORT AND COMPATIBILITY

The scope of support for layered and dependant components of OpenShift Container Platform changes independently of the OpenShift Container Platform version. To determine the current support status and compatibility for an add-on, refer to its release notes. For more information, see the Red Hat OpenShift Container Platform Life Cycle Policy.

1.4. NEW FEATURES AND ENHANCEMENTS

This release adds improvements related to the following components and concepts.

1.4.1. Red Hat Enterprise Linux CoreOS (RHCOS)

1.4.1.1. RHCOS now uses RHEL 8.4

RHCOS now uses Red Hat Enterprise Linux (RHEL) 8.4 packages in OpenShift Container Platform 4.8, as well as in OpenShift Container Platform 4.7.24 and above. This enables you to have the latest fixes, features, and enhancements, as well as the latest hardware support and driver updates. OpenShift Container Platform 4.6 is an Extended Update Support (EUS) release that will continue to use RHEL 8.2 EUS packages for the entirety of its lifecycle.

1.4.1.2. Using stream metadata for improved boot image automation

Stream metadata provides a standardized JSON format for injecting metadata into the cluster during OpenShift Container Platform installation. For improved automation, the new **openshift-install coreos print-stream-json** command provides a method for printing stream metadata in a scriptable, machine-readable format.

For user-provisioned installations, the **openshift-install** binary contains references to the version of RHCOS boot images that are tested for use with OpenShift Container Platform, such as the AWS AMI. You can now parse the stream metadata from a Go program by using the official **stream-metadata-go** library at https://github.com/coreos/stream-metadata-go.

For more information, see Accessing RHCOS AMIs with stream metadata.

1.4.1.3. Butane config transpiler simplifies creation of machine configs

OpenShift Container Platform now includes the Butane config transpiler to assist with producing and validating machine configs. Documentation now recommends using Butane to create machine configs for LUKS disk encryption, boot disk mirroring, and custom kernel modules.

For more information, see Creating machine configs with Butane .

1.4.1.4. Change to custom chrony.conf default on cloud platforms

If a cloud administrator has already set a custom **/etc/chrony.conf** configuration, RHCOS no longer sets the **PEERNTP=no** option by default on cloud platforms. Otherwise, the **PEERNTP=no** option is still set by default. See BZ#1924869 for more information.

1.4.1.5. Enabling multipathing at bare metal installation time

Enabling multipathing during bare metal installation is now supported for nodes provisioned in

OpenShift Container Platform 4.8 or higher. You can enable multipathing by appending kernel arguments to the **coreos-installer install** command so that the installed system itself uses multipath beginning from the first boot. While post-installation support is still available by activating multipathing via the machine config, enabling multipathing during installation is recommended for nodes provisioned starting in OpenShift Container Platform 4.8.

For more information, see Enabling multipathing with kernel arguments on RHCOS .

1.4.2. Installation and upgrade

1.4.2.1. Installing a cluster to an existing, empty resource group on Azure

You can now define an already existing resource group to install your cluster to on Azure by defining the **platform.azure.resourceGroupName** field in the **install-config.yaml** file. This resource group must be empty and only used for a single cluster; the cluster components assume ownership of all resources in the resource group.

If you limit the service principal scope of the installation program to this resource group, you must ensure all other resources used by the installation program in your environment have the necessary permissions, such as the public DNS zone and virtual network. Destroying the cluster using the installation program deletes the user-defined resource group.

1.4.2.2. Using existing IAM roles for clusters on AWS

You can now define a pre-existing Amazon Web Services (AWS) IAM role for your machine instance profiles by setting the **compute.platform.aws.iamRole** and **controlPlane.platform.aws.iamRole** fields in the **install-config.yaml** file. This allows you to do the following for your IAM roles:

- Match naming schemes
- Include predefined permissions boundaries

1.4.2.3. Using pre-existing Route53 hosted private zones on AWS

You can now define an existing Route 53 private hosted zone for your cluster by setting the **platform.aws.hostedZone** field in the **install-config.yaml** file. You can only use a pre-existing hosted zone when also supplying your own VPC.

1.4.2.4. Increasing the size of GCP subnets within the machine CIDR

The OpenShift Container Platform installation program for Google Cloud Platform (GCP) now creates subnets as large as possible within the machine CIDR. This allows the cluster to use a machine CIDR appropriately sized to accommodate the number of nodes in the cluster.

1.4.2.5. Improved upgrade duration

With this release, the upgrade duration for cluster Operators that deploy daemon sets to all nodes is significantly reduced. For example, the upgrade duration of a 250-node test cluster is reduced from 7.5 hours to 1.5 hours, resulting in upgrade duration scaling of less than one minute per additional node.

NOTE

This change does not affect machine config pool rollout duration.

1.4.2.6. MCO waits for all machine config pools to update before reporting the update is complete

When updating, the Machine Config Operator (MCO) now reports an **Upgradeable=False** condition in the machine-config Cluster Operator if any machine config pool has not completed updating. This status blocks future minor updates, but does not block future patch updates, or the current update. Previously, the MCO reported the **Upgradeable** status based only on the state of the control plane machine config pool, even if the worker pools were not done updating.

1.4.2.7. Using Fujitsu iRMC for installation on bare metal nodes

In OpenShift Container Platform 4.8, you can use Fujitsu hardware and the Fujitsu iRMC base board management controller protocol when deploying installer-provisioned clusters on bare metal. Currently Fujitsu supports iRMC S5 firmware version **3.05P** and above for installer-provisioned installation on bare metal. Enhancements and bug fixes for OpenShift Container Platform 4.8 include:

- Supporting soft power-off on iRMC hardware.
- Stopping the provisioning services once the installer deploys the control plane on the bare metal nodes. See BZ#1949859 for more information.
- Adding an Ironic health check to the bootstrap **keepalived** checks. See BZ#1949859 for more information.
- Verifying that the unicast peers list isn't empty on the control plane nodes. See BZ#1957708 for more information.
- Updating the Bare Metal Operator to align the iRMC PowerInterface. See BZ#1957869 for more information.
- Updating the **pyghmi** library version. See BZ#1920294 for more information.
- Updating the Bare Metal Operator to address missing IPMI credentials. See BZ#1965182 for more information.
- Removing iRMC from **enabled_bios_interfaces**. See BZ#1969212 for more information.
- Adding **ironicTlsMount** and **inspectorTlsMount** to the the bare metal pod definition. See BZ#1968701 for more information.
- Disabling the RAID feature for iRMC server. See BZ#1969487 for more information.
- Disabling RAID for all drivers. See BZ#1969487 for more information.

1.4.2.8. SR-IOV network support for clusters with installer-provisioned infrastructure on RHOSP

You can now deploy clusters on RHOSP that use single-root I/O virtualization (SR-IOV) networks for compute machines.

See Installing a cluster on OpenStack that supports SR-IOV-connected compute machines for more information.

1.4.2.9. Ironic Python Agent support for VLAN interfaces

With this update, the Ironic Python Agent now reports VLAN interfaces in the list of interfaces during

introspection. Additionally, the IP address is included on the interfaces, which allows for proper creation of a CSR. As a result, a CSR can be obtained for all interfaces, including VLAN interfaces. For more information, see BZ#1888712.

1.4.2.10. Over-the-air updates with the OpenShift Update Service

The OpenShift Update Service (OSUS) provides over-the-air updates to OpenShift Container Platform, including Red Hat Enterprise Linux CoreOS (RHCOS). It was previously only accessible as a Red Hat hosted service located behind public APIs, but can now be installed locally. The OpenShift Update Service is composed of an Operator and one or more application instances and is now generally available in OpenShift Container Platform 4.6 and higher.

For more information, see Understanding the OpenShift Update Service .

1.4.3. Web console

1.4.3.1. Custom console routes now use the new CustomDomains cluster API

For **console** and **downloads** routes, custom routes functionality is now implemented to use the new **ingress** config route configuration API **spec.componentRoutes**. The Console Operator config already contained custom route customization, but for the **console** route only. The route configuration via **console-operator** config is being deprecated. Therefore, if the **console** custom route is set up in both the **ingress** config and **console-operator** config, then the new **ingress** config custom route configuration takes precedent.

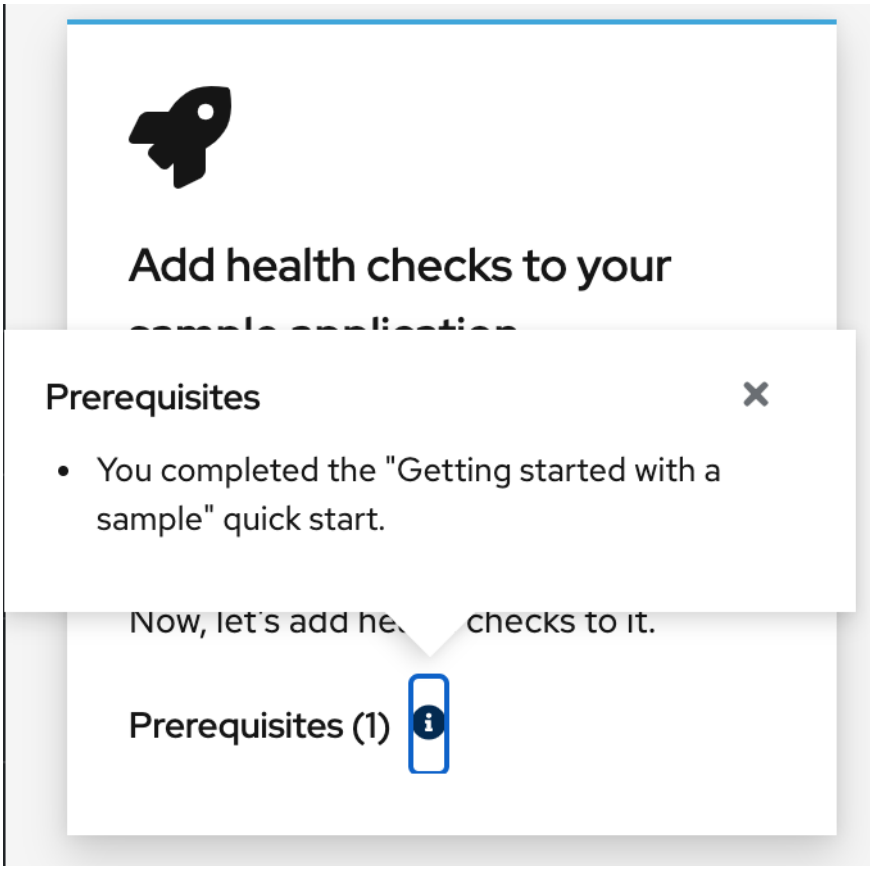
For more information, see Customizing console routes.

1.4.3.2. Access a code snippet from a quick start

You can now execute a CLI snippet when it is included in a quick start from the web console. To use this feature, you must first install the Web Terminal Operator. The web terminal and code snippet actions that execute in the web terminal are not present if you do not install the Web Terminal Operator. Alternatively, you can copy a code snippet to the clipboard regardless of whether you have the Web Terminal Operator installed or not.

1.4.3.3. Improved presentation of quick start prerequisites

Previously, quick start prerequisites were displayed as combined text instead of a list on the quick start card. With scalability in mind, the prerequisites are now presented in a popover rather than on the card.



1.4.4. IBM Z and LinuxONE

With this release, IBM Z and LinuxONE are now compatible with OpenShift Container Platform 4.8. The installation can be performed with z/VM or RHEL KVM. For installation instructions, see the following documentation:

- Installing a cluster with z/VM on IBM Z and LinuxONE
- Installing a cluster with z/VM on IBM Z and LinuxONE in a restricted network
- Installing a cluster with RHEL KVM on IBM Z and LinuxONE
- Installing a cluster with RHEL KVM on IBM Z and LinuxONE in a restricted network

Notable enhancements
The following new features are supported on IBM Z and LinuxONE with OpenShift Container Platform 4.8:

- KVM on RHEL 8.3 or later is supported as a hypervisor for user-provisioned installation of OpenShift Container Platform 4.8 on IBM Z and LinuxONE. Installation with static IP addresses as well as installation in a restricted network are now also supported.
- Encrypting data stored in etcd.
- 4K FCP block device.
- Three-node cluster support.

Supported features
The following features are also supported on IBM Z and LinuxONE:

- Multipathing
- Persistent storage using iSCSI
- Persistent storage using local volumes (Local Storage Operator)
- Persistent storage using hostPath
- Persistent storage using Fibre Channel
- Persistent storage using Raw Block
- OVN-Kubernetes with an initial installation of OpenShift Container Platform 4.8
- z/VM Emulated FBA devices on SCSI disks

These features are available only for OpenShift Container Platform on IBM Z for 4.8:

- HyperPAV enabled on IBM Z /LinuxONE for the virtual machines for FICON attached ECKD storage

Restrictions
Note the following restrictions for OpenShift Container Platform on IBM Z and LinuxONE:

- OpenShift Container Platform for IBM Z does not include the following Technology Preview features:
  - Precision Time Protocol (PTP) hardware
- The following OpenShift Container Platform features are unsupported:
  - Automatic repair of damaged machines with machine health checking
  - CodeReady Containers (CRC)
  - Controlling overcommit and managing container density on nodes
  - CSI volume cloning
  - CSI volume snapshots
  - FIPS cryptography
  - Helm command-line interface (CLI) tool
  - Multus CNI plug-in
  - NVMe
  - OpenShift Metering
  - OpenShift Virtualization
  - Tang mode disk encryption during OpenShift Container Platform deployment
- Worker nodes must run Red Hat Enterprise Linux CoreOS (RHCOS)

- Persistent shared storage must be provisioned by using either NFS or other supported storage protocols
- Persistent non-shared storage must be provisioned using local storage, like iSCSI, FC, or using LSO with DASD, FCP, or EDEV/FBA

1.4.5. IBM Power Systems

With this release, IBM Power Systems are now compatible with OpenShift Container Platform 4.8. For installation instructions, see the following documentation:

- Installing a cluster on IBM Power Systems
- Installing a cluster on IBM Power Systems in a restricted network

Notable enhancements
The following new features are supported on IBM Power Systems with OpenShift Container Platform 4.8:

- Encrypting data stored in etcd
- Three-node cluster support
- Multus SR-IOV

Supported features
The following features are also supported on IBM Power Systems:

- Currently, five Operators are supported:
  - Cluster-Logging-Operator
  - Cluster-NFD-Operator
  - Elastic Search-Operator
  - Local Storage Operator
  - SR-IOV Network Operator
- Multipathing
- Persistent storage using iSCSI
- Persistent storage using local volumes (Local Storage Operator)
- Persistent storage using hostPath
- Persistent storage using Fibre Channel
- Persistent storage using Raw Block
- OVN-Kubernetes with an initial installation of OpenShift Container Platform 4.8
- 4K Disk Support
- NVMe

Restrictions
Note the following restrictions for OpenShift Container Platform on IBM Power Systems:

- OpenShift Container Platform for IBM Power Systems does not include the following Technology Preview features:
  - Precision Time Protocol (PTP) hardware
- The following OpenShift Container Platform features are unsupported:
  - Automatic repair of damaged machines with machine health checking
  - CodeReady Containers (CRC)
  - Controlling overcommit and managing container density on nodes
  - FIPS cryptography
  - Helm command-line interface (CLI) tool
  - OpenShift Metering
  - OpenShift Virtualization
  - Tang mode disk encryption during OpenShift Container Platform deployment
- Worker nodes must run Red Hat Enterprise Linux CoreOS (RHCOS)
- Persistent storage must be of the Filesystem type that uses local volumes, Network File System (NFS), or Container Storage Interface (CSI)

1.4.6. Security and compliance

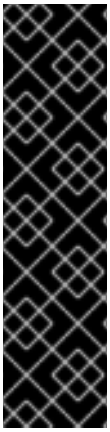
1.4.6.1. Audit logging for OAuth access token logout requests

The **Default** audit log policy now logs request bodies for OAuth access token creation (login) and deletion (logout) requests. Previously, deletion request bodies were not logged.

For more information about audit log policies, see Configuring the node audit log policy .

1.4.6.2. Wildcard subject for service serving certificates for headless services

Generating a service serving certificate for headless services now includes a wildcard subject in the format of ***.<service.name>.<service.namespace>.svc**. This allows for TLS-protected connections to individual stateful set pods without having to manually generate certificates for these pods.



IMPORTANT

Because the generated certificates contain wildcard subjects for headless services, do not use the service CA if your client must differentiate between individual pods. In this case:

- Generate individual TLS certificates by using a different CA.
- Do not accept the service CA as a trusted CA for connections that are directed to individual pods and must not be impersonated by other pods. These connections must be configured to trust the CA that was used to generate the individual TLS certificates.

For more information, see Add a service certificate.

1.4.6.3. The oc-compliance plug-in is now available

The Compliance Operator automates many of the checks and remediations for an OpenShift Container Platform cluster. However, the full process of bringing a cluster into compliance often requires administrator interaction with the Compliance Operator API and other components. The **oc-compliance** plug-in is now available and makes the process easier.

For more information, see Using the **oc-compliance** plug-in

1.4.6.4. TLS security profile for the Kubernetes control plane

The Kubernetes API server TLS security profile setting is now also honored by the Kubernetes scheduler and Kubernetes controller manager.

For more information, see Configuring TLS security profiles.

1.4.6.5. TLS security profile for the kubelet as a server

You can now set a TLS security profile for kubelet when it acts as an HTTP server for the Kubernetes API server.

For more information, see Configuring TLS security profiles.

1.4.6.6. Support for bcrypt password hashing

Previously, the **oauth-proxy** command only allowed the use of SHA-1 hashed passwords in **htpasswd** files used for authentication. **oauth-proxy** now includes support for **htpasswd** entries that use **bcrypt** password hashing. For more information, see BZ#1874322.

1.4.6.7. Enabling managed Secure Boot with installer-provisioned clusters

OpenShift Container Platform 4.8 supports automatically turning on UEFI Secure Boot mode for provisioned control plane and worker nodes and turning it back off when removing the nodes. To use this feature, set the node's **bootMode** configuration setting to **UEFISecureBoot** in the **install-config.yaml** file. Red Hat only supports installer-provisioned installation with managed Secure Boot on 10th generation HPE hardware or 13th generation Dell hardware running firmware version **2.75.75.75** or greater. For additional details, see Configuring managed Secure Boot in the install-config.yaml file .

1.4.7. Networking

1.4.7.1. Dual-stack support on installer-provisioned bare metal infrastructure with the OVN-Kubernetes cluster network provider

For clusters on installer-provisioned bare metal infrastructure, the OVN-Kubernetes cluster network provider supports both IPv4 and IPv6 address families.

For installer-provisioned bare metal clusters upgrading from previous versions of OpenShift Container Platform, you must convert your cluster to support dual-stack networking. For more information, see Converting to IPv4/IPv6 dual stack networking .

1.4.7.2. Migrate from OpenShift SDN to OVN-Kubernetes on user-provisioned infrastructure

An OpenShift SDN cluster network provider migration to the OVN-Kubernetes cluster network provider is supported for user-provisioned clusters. For more information, see Migrate from the OpenShift SDN cluster network provider.

1.4.7.3. OpenShift SDN cluster network provider egress IP feature balances across nodes

The egress IP feature of OpenShift SDN now balances network traffic roughly equally across nodes for a given namespace, if that namespace is assigned multiple egress IP addresses. Each IP address must reside on a different node. For more information, refer to Configuring egress IPs for a project for OpenShift SDN.

1.4.7.4. Network policy supports selecting host network Ingress Controllers

When using the OpenShift SDN or OVN-Kubernetes cluster network providers, you can select traffic from Ingress Controllers in a network policy rule regardless of whether an Ingress Controller runs on the cluster network or the host network. In a network policy rule, the **policy-group.network.openshift.io/ingress=""** namespace selector label matches traffic from an Ingress Controller. You can continue to use the **network.openshift.io/policy-group: ingress** namespace selector label, but this is a legacy label that can be removed in a future release of OpenShift Container Platform.

In earlier releases of OpenShift Container Platform, the following limitations existed:

- A cluster that uses the OpenShift SDN cluster network provider could select traffic from an Ingress Controller on the host network only by applying the **network.openshift.io/policy-group="ingress"** label to the **default** namespace.
- A cluster that uses the OVN-Kubernetes cluster network provider could not select traffic from an Ingress Controller on the host network.

For more information, refer to About network policy.

1.4.7.5. Network policy supports selecting host network traffic

When using either the OVN-Kubernetes cluster network provider or the OpenShift SDN cluster network provider, you can use the **policy-group.network.openshift.io/host-network: ""** namespace selector to select host network traffic in a network policy rule.

1.4.7.6. Network policy audit logs

If you use the OVN-Kubernetes cluster network provider, you can enable audit logging for network policies in a namespace. The logs are in a syslog compatible format and can be saved locally, sent over a

UDP connection, or directed to a UNIX domain socket. You can specify whether to log allowed, dropped, or both allowed and dropped connections. For more information, see Logging network policy events .

1.4.7.7. Network policy support for macvlan additional networks

You can create network policies that apply to macvlan additional networks by using the **MultiNetworkPolicy** API, which implements the **NetworkPolicy** API. For more information, see Configuring multi-network policy.

1.4.7.8. Supported hardware for SR-IOV

OpenShift Container Platform 4.8 adds support for additional Intel and Mellanox hardware.

- Intel X710 and XL710 controllers
- Mellanox ConnectX-5 Ex

For more information, see the supported devices.

1.4.7.9. Enhancements to the SR-IOV Network Operator

The Network Resources Injector that is deployed with the Operator is enhanced to expose information about huge pages requests and limits with the Downward API. When a pod specification includes a huge pages request or limit, the information is exposed in the **/etc/podnetinfo** path.

For more information, see Huge pages resource injection for Downward API .

1.4.7.10. Tracking network flows

OpenShift Container Platform 4.8 adds support for sending the metadata about network flows on the pod network to a network flows collector. The following protocols are supported:

- NetFlow
- sFlow
- IPFIX

Packet data is not sent to the network flows collector. Packet-level metadata such as the protocol, source address, destination address, port numbers, number of bytes, and other packet-level information is sent to the network flows collector.

For more information, see Tracking network flows.

1.4.7.11. CoreDNS-mDNS no longer used to resolve node names to IP addresses

OpenShift Container Platform 4.8 and later releases include functionality that uses cluster membership information to generate A/AAAA records. This resolves the node names to their IP addresses. Once the nodes are registered with the API, the cluster can disperse node information without using CoreDNS-mDNS. This eliminates the network traffic associated with multicast DNS.

1.4.7.12. Converting HTTP header names to support upgrading to OpenShift Container Platform 4.8

OpenShift Container Platform updated to HAProxy 2.2, which changes HTTP header names to

lowercase by default, for example, changing **Host: xyz.com** to **host: xyz.com**. For legacy applications that are sensitive to the capitalization of HTTP header names, use the Ingress Controller **spec.httpHeaders.headerNameCaseAdjustments** API field to accommodate legacy applications until they can be fixed. Make sure to add the necessary configuration by using **spec.httpHeaders.headerNameCaseAdjustments** before upgrading OpenShift Container Platform now that HAProxy 2.2 is available.

For more information, see Converting HTTP header case.

1.4.7.13. Configuring global access for an Ingress Controller on GCP

OpenShift Container Platform 4.8 adds support for the global access option for Ingress Controllers created on GCP with an internal load balancer. When the global access option is enabled, clients in any region within the same VPC network and compute region as the load balancer can reach the workloads running on your cluster.

For more information, see Configuring global access for an Ingress Controller on GCP .

1.4.7.14. Setting Ingress Controller thread count

OpenShift Container Platform 4.8 adds support for setting the thread count to increase the amount of incoming connections a cluster can handle.

For more information, see Setting Ingress Controller thread count.

1.4.7.15. Configuring the PROXY protocol for an Ingress Controller

OpenShift Container Platform 4.8 adds support for configuring the PROXY protocol for the Ingress Controller on non-cloud platforms, specifically for **HostNetwork** or **NodePortService** endpoint publishing strategy types.

For more information, see Configuring PROXY protocol for an Ingress Controller .

1.4.7.16. NTP servers on control plane nodes

In OpenShift Container Platform 4.8, installer-provisioned clusters can configure and deploy Network Time Protocol (NTP) servers on the control plane nodes and NTP clients on worker nodes. This enables workers to retrieve the date and time from the NTP servers on the control plane nodes, even when disconnected from a routable network. You can also configure and deploy NTP servers and NTP clients after deployment.

1.4.7.17. Changes to default API load balancer management for Kuryr

In OpenShift Container Platform 4.8 deployments on Red Hat OpenStack Platform (RHOSP) with Kuryr-Kubernetes, the API load balancer for the **default/kubernetes** service is no longer managed by the Cluster Network Operator (CNO), but instead by the kuryr-controller itself. This means that:

- When upgrading to OpenShift Container Platform 4.8, the **default/kubernetes** service will have downtime.

NOTE

In deployments where no Open Virtual Network (OVN) Octavia is available, more downtime should be expected

- The **default/kubernetes** load balancer is no longer required to use the Octavia Amphora driver. Instead, OVN Octavia will be used to implement the **default/kubernetes** service if it is available in the OpenStack cloud.

1.4.7.18. Enabling a provisioning network after installation

The assisted installer and installer-provisioned installation for bare metal clusters provide the ability to deploy a cluster without a **provisioning** network. In OpenShift Container Platform 4.8 and later, you can enable a **provisioning** network after installation by using the Cluster Baremetal Operator (CBO).

1.4.7.19. Configure network components to run on the control plane

If you need the virtual IP (VIP) addresses to run on the control plane nodes in a bare metal installation, you must configure the **apiVIP** and **ingressVIP** VIP addresses to run exclusively on the control plane nodes. By default, OpenShift Container Platform allows any node in the worker machine configuration pool to host the **apiVIP** and **ingressVIP** VIP addresses. Because many bare metal environments deploy worker nodes in separate subnets from the control plane nodes, configuring the **apiVIP** and **ingressVIP** virtual IP addresses to run exclusively on the control plane nodes prevents issues from arising due to deploying worker nodes in separate subnets. For additional details, see Configure network components to run on the control plane.

1.4.7.20. Configuring an external load balancer for apiVIP and ingressVIP traffic

In OpenShift Container Platform 4.8, you can configure an external load balancer to handle **apiVIP** and **ingressVIP** traffic to the control plane of installer-provisioned clusters. External load balancing services and the control plane nodes must run on the same L2 network, and on the same VLAN when using VLANs to route traffic between the load balancing services and the control plane nodes.

1.4.7.21. OVN-Kubernetes IPsec support for dual-stack networking

OpenShift Container Platform 4.8 adds OVN-Kubernetes IPsec support for clusters that are configured to use dual-stack networking.

1.4.7.22. Egress router CNI for OVN-Kubernetes

The egress router CNI plug-in is generally available. The Cluster Network Operator is enhanced to support an **EgressRouter** API object. The process for adding an egress router on a cluster that uses OVN-Kubernetes is simplified. When you create an egress router object, the Operator automatically adds a network attachment definition and a deployment. The pod for the deployment acts as the egress router.

For more information, see Considerations for the use of an egress router pod.

1.4.7.23. IP failover support on OpenShift Container Platform

IP failover is now supported on OpenShift Container Platform clusters on bare metal. IP failover uses **keepalived** to host a set of externally accessible VIP addresses on a set of hosts. Each VIP is only serviced by a single host at a time. **keepalived** uses the Virtual Router Redundancy Protocol (VRRP) to determine which host, from the set of hosts, services which VIP. If a host becomes unavailable, or if the service that **keepalived** is watching does not respond, the VIP is switched to another host from the set. This means a VIP is always serviced as long as a host is available.

For more information, see Configuring IP failover.

1.4.7.24. Power of Two Random Choices balancing algorithm for Ingress Controller

OpenShift Container Platform Ingress Controllers now use the Power of Two Random Choices balancing algorithm by default for more even balancing after reloading. As in earlier releases, the balancing algorithm for a specific route can still be configured using the **haproxy.router.openshift.io/balance** route annotation. For more information, see Route-specific annotations.

To revert the Ingress Controller back to use the Least Connections balancing algorithm, use the following **oc patch** command to specify **leastconn** as the default:

```
$ oc -n openshift-ingress-operator patch ingresscontroller/default --type=merge --patch='{"spec":
{"unsupportedConfigOverrides":{"loadBalancingAlgorithm":"leastconn"}}}'
```

1.4.7.25. Control DNS pod placement

In OpenShift Container Platform 4.8, you can use a custom node selector and tolerations to configure the daemon set for CoreDNS to run or not run on certain nodes.

For more information, see Controlling DNS pod placement.

1.4.7.26. Provider networks support for clusters that run on RHOSP

OpenShift Container Platform clusters on Red Hat OpenStack Platform (RHOSP) now support provider networks for all deployment types.

1.4.7.27. Configurable tune.maxrewrite and tune.bufsize for HAProxy

Cluster Administrators can now set **headerBufferMaxRewriteByte** and **headerBufferBytes** Ingress Controller tuning parameters to configure **tune.maxrewrite** and **tune.bufsize** HAProxy memory options per-Ingress Controller.

See Ingress Controller configuration parameters for more information.

1.4.8. Storage

1.4.8.1. Persistent storage using GCP PD CSI driver operator is generally available

The Google Cloud Platform (GCP) persistent disk (PD) Container Storage Interface (CSI) driver is automatically deployed and managed on GCP environments, allowing you to dynamically provision these volumes without having to install the driver manually. This feature was previously introduced as a Technology Preview feature in OpenShift Container Platform 4.7 and is now generally available and enabled by default in OpenShift Container Platform 4.8.

For more information, GCP PD CSI Driver Operator.

1.4.8.2. Persistent storage using the Azure Disk CSI Driver Operator (Technology Preview)

The Azure Disk CSI Driver Operator provides a storage class by default that you can use to create persistent volume claims (PVCs). The Azure Disk CSI Driver Operator that manages this driver is in Technology Preview.

For more information, see Azure Disk CSI Driver Operator.

1.4.8.3. Persistent storage using the vSphere CSI Driver Operator (Technology Preview)

The vSphere CSI Driver Operator provides a storage class by default that you can use to create persistent volume claims (PVCs). The vSphere CSI Driver Operator that manages this driver is in Technology Preview.

For more information, see vSphere CSI Driver Operator.

1.4.8.4. Automatic CSI migration (Technology Preview)

Starting with OpenShift Container Platform 4.8, automatic migration for the following in-tree volume plug-ins to their equivalent CSI drivers is available as a Technology Preview feature:

- Amazon Web Services (AWS) Elastic Block Storage (EBS)
- OpenStack Cinder

For more information, see Automatic CSI Migration.

1.4.8.5. External provisioner for AWS EFS (Technology Preview) feature has been removed

The Amazon Web Services (AWS) Elastic File System (EFS) Technology Preview feature has been removed and is no longer supported.

1.4.8.6. Improved control over Cinder volume availability zones for clusters that run on RHOSP

You can now select availability zones for Cinder volumes during installation. You can also use Cinder volumes in particular availability zones for your image registry.

1.4.9. Registry

1.4.10. Operator lifecycle

1.4.10.1. Enhanced error reporting for administrators

A cluster administrator using Operator Lifecycle Manager (OLM) to install an Operator can encounter error conditions that are related either to the current API or low-level APIs. Previously, there was little insight into why OLM could not fulfill a request to install or update an Operator. These errors could range from trivial issues like typos in object properties or missing RBAC, to more complex issues where items could not be loaded from the catalog due to metadata parsing.

Because administrators should not require understanding of the interaction process between the various low-level APIs or access to the OLM pod logs to successfully debug such issues, OpenShift Container Platform 4.8 introduces the following enhancements in OLM to provide administrators with more comprehensible error reporting and messages:

1.4.10.2. Retrying install plans

Install plans, defined by an **InstallPlan** object, can encounter transient errors, for example, due to API server availability or conflicts with other writers. Previously, these errors would result in the termination of partially-applied install plans that required manual cleanup. With this enhancement, the Catalog Operator now retries errors during install plan execution for up to one minute. The new **.status.message** field provides a human-readable indication when retries are occurring.

1.4.10.3. Indicating invalid Operator groups

Creating a subscription in a namespace with no Operator groups or multiple Operator groups would previously result in a stalled Operator installation with an install plan that stays in **phase=Installing** forever. With this enhancement, the install plan immediately transitions to **phase=Failed** so that the administrator can correct the invalid Operator group, and then delete and re-create the subscription again.

1.4.10.4. Specific reporting when no candidate Operators found

ResolutionFailed events, which are created when dependency resolution in a namespace fails, now provide more specific text when the namespace contains a subscription that references a package or channel that does not exist in the referenced catalog source. Previously, this message was generic:

> no candidate operators found matching the spec of subscription '<name>'

With this enhancement, the messages are more specific:

Operator does not exist

> no operators found in package <name> in the catalog referenced by subscription <name>

Catalog does not exist

> no operators found from catalog <name> in namespace openshift-marketplace referenced by subscription <name>

Channel does not exist

> no operators found in channel <name> of package <name> in the catalog referenced by subscription <name>

Cluster service version (CSV) does not exist

> no operators found with name <name>.<version> in channel <name> of package <name> in the catalog referenced by subscription <name>

1.4.11. Operator development

1.4.11.1. Migration of Operator projects from package manifest format to bundle format

Support for the legacy package manifest format for Operators is removed in OpenShift Container Platform 4.8 and later. The bundle format is the preferred Operator packaging format for Operator Lifecycle Manager (OLM) starting in OpenShift Container Platform 4.6. If you have an Operator project that was initially created in the package manifest format, which has been deprecated, you can now use the Operator SDK **pkgman-to-bundle** command to migrate the project to the bundle format.

For more information, see Migrating package manifest projects to bundle format .

1.4.11.2. Publishing a catalog containing a bundled Operator

To install and manage Operators, Operator Lifecycle Manager (OLM) requires that Operator bundles

are listed in an index image, which is referenced by a catalog on the cluster. As an Operator author, you can use the Operator SDK to create an index containing the bundle for your Operator and all of its dependencies. This is useful for testing on remote clusters and publishing to container registries.

For more information, see Publishing a catalog containing a bundled Operator .

1.4.11.3. Enhanced Operator upgrade testing

The Operator SDK's **run bundle-upgrade** subcommand automates triggering an installed Operator to upgrade to a later version by specifying a bundle image for the later version. Previously, the subcommand could only upgrade Operators that were initially installed using the **run bundle** subcommand. With this enhancement, the **run bundle-upgrade** now also works with Operators that were initially installed with the traditional Operator Lifecycle Manager (OLM) workflow.

For more information, see Testing an Operator upgrade on Operator Lifecycle Manager .

1.4.11.4. Controlling Operator compatibility with OpenShift Container Platform versions

When an API is removed from an OpenShift Container Platform version, Operators running on that cluster version that are still using removed APIs will no longer work properly. As an Operator author, you should plan to update your Operator projects to accommodate API deprecation and removal to avoid interruptions for users of your Operator.

For more details, see Controlling Operator compatibility with OpenShift Container Platform versions .

Builds

1.4.11.5. New Telemetry metric for number of builds by strategy

Telemetry includes a new **openshift:build_by_strategy:sum** gauge metric, which sends the number of builds by strategy type to the Telemeter Client. This metric gives site reliability engineers (SREs) and product managers visibility into the kinds of builds that run on OpenShift Container Platform clusters.

1.4.11.6. Mount custom PKI certificate authorities

Previously, builds could not use the cluster PKI certificate authorities that were sometimes required to access corporate artifact repositories. Now, you can configure the **BuildConfig** object to mount cluster custom PKI certificate authorities by setting **mountTrustedCA** to **true**.

1.4.12. Images

1.4.13. Machine API

1.4.13.1. Scaling machines running in vSphere to and from zero with the cluster autoscaler

When running machines in vSphere, you can now set the **minReplicas** value to **0** in the **MachineAutoscaler** resource definition. When this value is set to **0**, the cluster autoscaler scales the machine set to and from zero depending on if the machines are in use. For more information, see the MachineAutoscaler resource definition.

1.4.13.2. Automatic rotation of kubelet-ca.crt does not require node draining or reboot

The automatic rotation of the **/etc/kubernetes/kubelet-ca.crt** certificate authority (CA) no longer requires the Machine Config Operator (MCO) to drain nodes or reboot the cluster.

As part of this change, the following modifications do not require the MCO to drain nodes:

- Changes to the SSH key in the **spec.config.ignition.passwd.users.sshAuthorizedKeys** parameter of a machine config
- Changes to the global pull secret or pull secret in the **openshift-config** namespace

When the MCO detects any of these changes, it applies the changes and uncordons the node.

For more information, see Understanding the Machine Config Operator .

1.4.13.3. Machine set policy enhancement

Previously, creating machine sets required users to manually configure their CPU pinning settings, NUMA pinning settings, and CPU topology changes to get better performance from the host. With this enhancement, users can select a policy in the **MachineSet** resource to populate settings automatically. For more information, see BZ#1941334.

1.4.13.4. Machine set hugepage enhancement

Providing a **hugepages** property into the **MachineSet** resource is now possible. This enhancement creates the **MachineSet** resource's nodes with a custom property in oVirt and instructs those nodes to use the **hugepages** of the hypervisor. For more information, see BZ#1948963.

1.4.13.5. Machine Config Operator ImageContentSourcePolicy object enhancement

OpenShift Container Platform 4.8 avoids workload disruption for selected **ImageContentSourcePolicy** object changes. This feature helps users and teams add additional mirrors and registries without workload disruption. As a result, workload distruption will no longer occur for the following changes in **/etc/containers/registries.conf** files:

- Addition of a registry with **mirror-by-digest-only=true**
- Addition of a mirror in a registry with **mirror-by-digest-only=true**
- Appending items in **unqualified-search-registries** list

For any other changes in **/etc/containers/registries.conf** files, the Machine Config Operator will default to draining nodes to apply changes. For more information, see BZ#1943315.

1.4.14. Nodes

1.4.14.1. Descheduler operator.openshift.io/v1 API group is now available

The **operator.openshift.io/v1** API group is now available for the descheduler. Support for the **operator.openshift.io/v1beta1** API group for the descheduler might be removed in a future release.

1.4.14.2. Prometheus metrics for the descheduler

You can now enable Prometheus metrics for the descheduler by adding the **openshift.io/cluster-monitoring=true** label to the **openshift-kube-descheduler-operator** namespace where you installed the descheduler.

The following descheduler metrics are available:

- **descheduler_build_info** - Provides build information about the descheduler.
- **descheduler_pods_evicted** - Provides the number of pods that have been evicted for each combination of strategy, namespace, and result. There must be at least one evicted pod for this metric to appear.

1.4.14.3. Support for huge pages with the Downward API

With this release, when you set requests and limits for huge pages in a pod specification, you can use the Downward API to view the allocation for the pod from within a container. This enhancements relies on the **DownwardAPIHugePages** feature gate. OpenShift Container Platform 4.8 enables the feature gate.

For more information, see Consuming huge pages resources using the Downward API .

1.4.14.4. New labels for the Node Feature Discovery Operator

The Node Feature Discovery (NFD) Operator detects hardware features available on each node in an OpenShift Container Platform cluster. Then, it modifies node objects with node labels. This enables the NFD Operator to advertise the features of specific nodes. OpenShift Container Platform 4.8 supports three additional labels for the NFD Operator.

- **pstate intel-pstate**: When the Intel **pstate** driver is enabled and in use, the **pstate intel-pstate** label reflects the status of the Intel **pstate** driver. The status is either **active** or **passive**.
- **pstate scaling_governor**: When the Intel **pstate** driver status is **active**, the **pstate scaling_governor** label reflects the scaling governor algorithm. The algorithm is either **powersave** or **performance**.
- **cstate status**: If the **intel_idle** driver has C-states or idle states, the **cstate status** label is **true**. Otherwise, it is **false**.

1.4.14.5. Remediate unhealthy nodes with the Poison Pill Operator

You can use the Poison Pill Operator to allow unhealthy nodes to reboot automatically. This minimizes downtime for stateful applications and ReadWriteOnce (RWO) volumes, and restores compute capacity if transient failures occur.

The Poison Pill Operator works with all cluster and hardware types.

For more information, see Remediating nodes with the Poison Pill Operator.

1.4.14.6. Automatic rotation of kubelet-ca.crt does not require reboot

The automatic rotation of the **/etc/kubernetes/kubelet-ca.crt** certificate authority (CA) no longer requires the Machine Config Operator (MCO) to drain nodes or reboot the cluster.

As part of this change, the following modifications do not require the MCO to drain nodes:

- Changes to the SSH key in the **spec.config.ignition.passwd.users.sshAuthorizedKeys** parameter of a machine config
- Changes to the global pull secret or pull secret in the **openshift-config** namespace

When the MCO detects any of these changes, it applies the changes and uncordons the node.

For more information, see Understanding the Machine Config Operator.

1.4.14.7. Vertical pod autoscaling is generally available

The OpenShift Container Platform vertical pod autoscaler (VPA) is now generally available. The VPA automatically reviews the historic and current CPU and memory resources for containers in pods and can update the resource limits and requests based on the usage values it learns.

You can also use the VPA with pods that require only one replica by modifying the **VerticalPodAutoscalerController** object as described below. Previously, the VPA worked only with pods that required two or more replicas.

For more information, see Automatically adjust pod resource levels with the vertical pod autoscaler .

1.4.14.8. Vertical pod autoscaling minimum can be configured

By default, workload objects must specify a minimum of two replicas in order for the VPA to automatically update pods. As a result, workload objects that specify fewer than two replicas are not acted upon by the VPA. You can change this cluster-wide minimum value by modifying the **VerticalPodAutoscalerController** object to add the **minReplicas** parameter.

For more information, see Automatically adjust pod resource levels with the vertical pod autoscaler .

1.4.14.9. Automatically allocate CPU and memory resources for nodes

OpenShift Container Platform can automatically determine the optimal sizing value of the **system-reserved** setting when a node starts. Previously, the CPU and memory allocations in the **system-reserved** setting were fixed limits that you needed to manually determine and set.

When automatic resource allocation is enabled, a script on each node calculates the optimal values for the respective reserved resources based on the installed CPU and memory capacity on the node.

For more information, see Automatically allocating resources for nodes.

1.4.14.10. Adding specific repositories to pull images

You can now specify an individual repository within a registry when creating lists of allowed and blocked registries for pulling and pushing images. Previously, you could specify only a registry.

For more information, see Adding specific registries and Blocking specific registries.

1.4.14.11. Cron jobs are generally available

The cron job custom resource is now generally available. As part of this change, a new controller has been implemented that substantially improves the performance of cron jobs. For more information on cron jobs, see Understanding jobs and cron jobs.

1.4.15. Red Hat OpenShift Logging

In OpenShift Container Platform 4.7, *Cluster Logging* became *Red Hat OpenShift Logging*. For more information, see Release notes for Red Hat OpenShift Logging .

1.4.16. Monitoring

1.4.16.1. Alerting rule changes

OpenShift Container Platform 4.8 includes the following alerting rule changes:

Example 1.1. Alerting rule changes

- The **ThanosSidecarPrometheusDown** alert severity is updated from *critical* to *warning*.
- The **ThanosSidecarUnhealthy** alert severity is updated from *critical* to *warning*.
- The **ThanosQueryHttpRequestQueryErrorRateHigh** alert severity is updated from *critical* to *warning*.
- The **ThanosQueryHttpRequestQueryRangeErrorRateHigh** alert severity is updated from *critical* to *warning*.
- The **ThanosQueryInstantLatencyHigh** critical alert is removed. This alert fired if Thanos Querier had a high latency for instant queries.
- The **ThanosQueryRangeLatencyHigh** critical alert is removed. This alert fired if Thanos Querier had a high latency for range queries.
- For all Thanos Querier alerts, the **for** duration is increased to 1 hour.
- For all Thanos sidecar alerts, the **for** duration is increased to 1 hour.

NOTE

Red Hat does not guarantee backward compatibility for metrics, recording rules, or alerting rules.

1.4.16.2. Alerts and information on APIs in use that will be removed in the next release

OpenShift Container Platform 4.8 introduces two new alerts that fire when an API that will be removed in the next release is in use:

- **APIRemovedInNextReleaseInUse** - for APIs that will be removed in the next OpenShift Container Platform release.
- **APIRemovedInNextEUSReleaseInUse** - for APIs that will be removed in the next OpenShift Container Platform Extended Update Support (EUS) release.

You can use the new **APIRequestCount** API to track what is using the deprecated APIs. This allows you to plan whether any actions are required in order to upgrade to the next release.

1.4.16.3. Version updates to monitoring stack components and dependencies

OpenShift Container Platform 4.8 includes version updates to the following monitoring stack components and dependencies:

- The Prometheus Operator is now on version 0.48.1.

- Prometheus is now on version 2.26.1.
- The **node-exporter** agent is now on version 1.1.2.
- Thanos is now on version 0.20.2.
- Grafana is now on version 7.5.5.

1.4.16.4. kube-state-metrics upgraded to version 2.0.0

kube-state-metrics is upgraded to version 2.0.0. The following metrics were deprecated in **kube-state-metrics** version 1.9 and are effectively removed in version 2.0.0:

- Non-generic resource metrics for pods:
  - kube_pod_container_resource_requests_cpu_cores
  - kube_pod_container_resource_limits_cpu_cores
  - kube_pod_container_resource_requests_memory_bytes
  - kube_pod_container_resource_limits_memory_bytes
- Non-generic resource metrics for nodes:
  - kube_node_status_capacity_pods
  - kube_node_status_capacity_cpu_cores
  - kube_node_status_capacity_memory_bytes
  - kube_node_status_allocatable_pods
  - kube_node_status_allocatable_cpu_cores
  - kube_node_status_allocatable_memory_bytes

1.4.16.5. Removed Grafana and Alertmanager UI links

The link to the third-party Alertmanager UI is removed from the **Monitoring → Alerting** page in the OpenShift Container Platform web console. Additionally, the link to the third-party Grafana UI is removed from the **Monitoring → Dashboards** page. You can still access the routes to the Grafana and Alertmanager UIs in the web console in the **Administrator** perspective by navigating to the **Networking → Routes** page in the **openshift-monitoring** project.

1.4.16.6. Monitoring dashboard enhancements in the web console

New enhancements are available on the **Monitoring → Dashboards** page in the OpenShift Container Platform web console:

- When you zoom in on a single graph by selecting an area with the mouse, all other graphs now update to reflect the same time range.
- Dashboard panels are now organized into groups, which you can expand and collapse.
- Single-value panels now support changing color depending on their value.

- Dashboard labels now display in the **Dashboard** drop-down list.
- You can now specify a custom time range for a dashboard by selecting **Custom time range** in the **Time Range** drop-down list.
- When applicable, you can now select the **All** option in a dashboard filter drop-down menu to display data for all of the options in that filter.

1.4.17. Metering

The Metering Operator is deprecated as of OpenShift Container Platform 4.6, and is scheduled to be removed in the next OpenShift Container Platform release.

1.4.18. Scale

1.4.18.1. Running on a single node cluster

Running tests on a single node cluster causes longer timeouts for certain tests, including SR-IOV and SCTP tests, and tests requiring control plane and worker nodes are skipped. Reconfiguration requiring node reboots causes a reboot of the entire environment, including the OpenShift control plane, and therefore takes longer to complete. All PTP tests requiring a control plane node and a worker node are skipped. No additional configuration is needed because the tests check for the number of nodes at startup and adjust test behavior accordingly.

PTP tests can run in Discovery mode. The tests look for a PTP control plane configured outside of the cluster. The following parameters are required:

- **ROLE_WORKER_CNF=master** - Required because the control plane (**master**) is the only machine pool to which the node will belong.
- **XT_U32TEST_HAS_NON_CNF_WORKERS=false** - Required to instruct the **xt_u32** negative test to skip because there are only nodes where the module is loaded.
- **SCTPTEST_HAS_NON_CNF_WORKERS=false** - Required to instruct the SCTP negative test to skip because there are only nodes where the module is loaded.

1.4.18.2. Reducing NIC using the Performance Addon Operator

The Performance Addon Operator allows you to adjust the Network Interface Card (NIC) queue count for each network device by configuring the performance profile. Device network queues allow packets to be distributed among different physical queues, and each queue gets a separate thread for packet processing.

For Data Plane Development Kit (DPDK) based workloads, it is important to reduce the NIC queues to only the number of reserved or housekeeping CPUs to ensure the desired low latency is achieved.

For more information, see Reducing NIC queues using the Performance Addon Operator .

1.4.18.3. Cluster maximums

Updated guidance around cluster maximums for OpenShift Container Platform 4.8 is now available.

IMPORTANT

No large scale testing for performance against OVN-Kubernetes testing was executed for this release.

Use the OpenShift Container Platform Limit Calculator to estimate cluster limits for your environment.

1.4.18.4. Creating a performance profile

You can now create a performance profile using the Performance Profile Creator (PPC) tool. The tool consumes **must-gather** data from the cluster and several user-supplied profile arguments, and using this information it generates a performance profile that is appropriate for your hardware and topology.

For more information, see Creating a Performance Profile .

1.4.18.5. Node Feature Discovery Operator

The Node Feature Discovery (NFD) Operator is now available. Use it to expose node-level information by orchestrating Node Feature Discovery, a Kubernetes add-on for detecting hardware features and system configuration.

1.4.18.6. The Driver Toolkit (Technology Preview)

You can now use the Driver Toolkit as a base image for driver containers so that you can enable special software and hardware devices on Kubernetes. This is currently a Technology Preview feature.

1.4.19. Backup and restore

1.4.19.1. etcd snapshot enhancement

A new enhancement validates the status of the etcd snapshot after backup and before restore. Previously, the backup process did not validate that the snapshot taken was complete, and the restore process did not verify that the snapshot being restored was valid, not corrupted. Now, if the disk is corrupted during backup or restore, the error is clearly reported to the admin. For more information, see BZ#1965024.

1.4.20. Insights Operator

1.4.20.1. Insights Advisor recommendations for restricted networks

In OpenShift Container Platform 4.8, users operating in restricted networks can gather and upload Insights Operator archives to Insights Advisor to diagnose potential issues. Additionally, users can obfuscate sensitive data contained in the Insights Operator archive before upload.

For more information, see Using remote health reporting in a restricted network .

1.4.20.2. Insights Advisor improvements

Insights Advisor in the OpenShift Container Platform web console now correctly reports 0 issues found. Previously, Insights Advisor gave no information.

1.4.20.3. Insights Operator data collection enhancements

In OpenShift Container Platform 4.8, the Insights Operator collects the following additional information:

- Non-identifiable cluster workload information to find known security and version issues.
- The **MachineHealthCheck** and **MachineAutoscaler** definitions.
- The **virt_platform** and **vsphere_node_hw_version_total** metrics.
- Information about unhealthy SAP pods to assist in the installation of SAP Smart Data Integration.
- The **datahubs.installers.datahub.sap.com** resources to identfy SAP clusters.
- A summary of failed **PodNetworkConnectivityChecks** to enhance networking.
- Information about the **cluster-version** pods and events from the **openshift-cluster-operator** namespace to debug issues with the **cluster-version** Operator.

With this additional information, Red Hat can provide improved remediation steps in Insights Advisor.

1.4.20.4. Insights Operator enhancement for unhealthy SAP pods

The Insights Operator can now gather data for unhealthy SAP pods. When the SDI installation fails, it is possible to detect the problem by looking at which of the initialization pods have failed. The Insights Operator now gathers information about failed pods in the SAP/SDI namespaces. For more information, see BZ#1930393.

1.4.20.5. Insights Operator enhancement for gathering SAP pod data

The Insights Operator can now gather **Datahubs** resources from SAP clusters. This data allows SAP clusters to be distinguished from non-SAP clusters in the Insights Operator archives, even in situations in which all of the data gathered exclusively from SAP clusters is missing and it would otherwise be impossible to determine if a cluster has an SDI installation. For more information, see BZ#1940432.

1.4.21. Authentication and authorization

1.4.21.1. Running OpenShift Container Platform using AWS Security Token Service (STS) for credentials is generally available

You can now use the Cloud Credential Operator (CCO) utility (**ccoctl**) to configure the CCO to use the Amazon Web Services Security Token Service (AWS STS). When the CCO is configured to use STS, it assigns IAM roles that provide short-term, limited-privilege security credentials to components.

This feature was previously introduced as a Technology Preview feature in OpenShift Container Platform 4.7, and is now generally available in OpenShift Container Platform 4.8.

For more information, see Using manual mode with STS.

1.4.22. OpenShift sandboxed containers

1.4.22.1. OpenShift sandboxed containers support on OpenShift Container Platform (Technology Preview)

OpenShift sandboxed containers 1.0.0 Technology Preview release introduces built-in support for

running Kata Containers as an additional runtime. OpenShift sandboxed containers enables users to choose Kata Containers as an additional runtime to provide additional isolation for their workloads. The OpenShift sandboxed containers Operator automates the tasks of installing, removing, and updating Kata Containers. It allows for tracking the state of those tasks by describing the **KataConfig** custom resource.

OpenShift sandboxed containers are only supported on bare metal. Red Hat Enterprise Linux CoreOS (RHCOS) is the only supported operating system for OpenShift sandboxed containers 1.0.0. Disconnected environments are not supported in OpenShift Container Platform 4.8.

For more information, see Understanding OpenShift sandboxed containers

1.5. NOTABLE TECHNICAL CHANGES

OpenShift Container Platform 4.8 introduces the following notable technical changes.

Kuryr service subnet creation Changes
New installations of OpenShift Container Platform on Red Hat OpenStack Platform (RHOSP) with Open Virtual Network configured to use Kuryr no longer create a **services** subnet that is twice the size requested in **networking.serviceCIDR**. The subnet created is now the same as the requested size. For more information, see BZ#1955548.

OAuth tokens without a SHA-256 prefix can no longer be used
Prior to OpenShift Container Platform 4.6, OAuth access and authorize tokens used secret information for the object names.

Starting with OpenShift Container Platform 4.6, OAuth access token and authorize token object names are stored as non-sensitive object names, with a SHA-256 prefix. OAuth tokens that do not contain a SHA-256 prefix can no longer be used or created in OpenShift Container Platform 4.8.

The Federal Risk and Authorization Management Program (FedRAMP) moderate controls
In OpenShift Container Platform 4.8, the **rhcos4-moderate** profile is now complete. The **ocp4-moderate** profile will be completed in a future release.

Ingress Controller upgraded to HAProxy 2.2.13
The OpenShift Container Platform Ingress Controller is upgraded to HAProxy version 2.2.13.

CoreDNS update to version 1.8.1
In OpenShift Container Platform 4.8, CoreDNS uses version 1.8.1, which has several bug fixes, renamed metrics, and dual-stack IPv6 enablement.

etcd now uses the zap logger
In OpenShift Container Platform 4.8, etcd now uses zap as the default logger instead of capnslog. Zap is a structured logger that provides machine consumable JSON log messages. You can use **jq** to easily parse these log messages.

If you have a log consumer that is expecting the capnslog format, you might need to adjust it for the zap logger format.

Example capnslog format (OpenShift Container Platform 4.7)

```
2021-06-03 22:40:16.984470 W | etcdserver: read-only range request
"key:\"/kubernetes.io/operator.openshift.io/clustercsidrivers/\"
range_end:\"/kubernetes.io/operator.openshift.io/clustercsidrivers0\" count_only:true " with result
"range_response_count:0 size:8" took too long (100.498102ms) to execute
```

Example zap format (OpenShift Container Platform 4.8)


```
{"level":"warn","ts":"2021-06-14T13:13:23.243Z","caller":"etcdserver/util.go:163","msg":"apply request took too long","took":"163.262994ms","expected-duration":"100ms","prefix":"read-only range ","request":"key:\"/kubernetes.io/namespaces/default\" serializable:true keys_only:true ","response":"range_response_count:1 size:53"}
```


Multiple daemon sets merged for LSO
In OpenShift Container Platform 4.8, multiple daemon sets are merged for Local Storage Object (LSO). When you create a local volume custom resource, only **daemonset.apps/diskmaker-manager** is created.

Bound service account token volumes are enabled
Previously, service account tokens were secrets that were mounted into pods. Starting with OpenShift Container Platform 4.8, projected volumes are used instead. As a result of this change, service account tokens no longer have an underlying corresponding secret.

Bound service account tokens are audience-bound and time-bound. For more information, see Using bound service account tokens.

Additionally, the kubelet refreshes tokens automatically after they reach 80% of duration, and **client-go** watches for token changes and reloads automatically. The combination of these two behaviors means that most usage of bound tokens is no different from usage of legacy tokens that never expire. Non-standard usage outside of **client-go** might cause issues.

Operator SDK v1.8.0
OpenShift Container Platform 4.8 supports Operator SDK v1.8.0. See Installing the Operator SDK CLI to install or update to this latest version.

NOTE

Operator SDK v1.8.0 supports Kubernetes 1.20.

If you have any Operator projects that were previously created or maintained with Operator SDK v1.3.0, see Upgrading projects for newer Operator SDK versions to ensure your projects are upgraded to maintain compatibility with Operator SDK v1.8.0.

1.6. DEPRECATED AND REMOVED FEATURES

Some features available in previous releases have been deprecated or removed.

Deprecated functionality is still included in OpenShift Container Platform and continues to be supported; however, it will be removed in a future release of this product and is not recommended for new deployments. For the most recent list of major functionality deprecated and removed within OpenShift Container Platform 4.8, refer to the table below. Additional details for more fine-grained functionality that has been deprecated and removed are listed after the table.

In the table, features are marked with the following statuses:

- **GA**: *General Availability*
- **TP**: *Technology Preview*
- **DEP**: *Deprecated*

- **REM**: *Removed*

Table 1.1. Deprecated and removed features tracker

Feature	OCP 4.6	OCP 4.7	OCP 4.8
OperatorSource objects	REM	REM	REM
Package manifest format (Operator Framework)	DEP	DEP	REM
oc adm catalog build	DEP	DEP	REM
--filter-by-os flag for **oc adm catalog mirror**	GA	DEP	REM
v1beta1 CRDs	DEP	DEP	DEP
Docker Registry v1 API	DEP	DEP	DEP
Metering Operator	DEP	DEP	DEP
Scheduler policy	GA	DEP	DEP
ImageChangesInProgress condition for Cluster Samples Operator	GA	DEP	DEP
MigrationInProgress condition for Cluster Samples Operator	GA	DEP	DEP
Use of **v1** in **apiVersion** for OpenShift Container Platform resources	GA	DEP	DEP
Use of **dhclient** in Red Hat Enterprise Linux CoreOS (RHCOS)	DEP	DEP	DEP
Cluster Loader	GA	GA	DEP
Bring your own RHEL 7 compute machines	DEP	DEP	DEP
External provisioner for AWS EFS	REM	REM	REM
lastTriggeredImageID field in the **BuildConfig** spec for Builds	GA	GA	DEP
Jenkins Operator	TP	TP	DEP
HPA custom metrics adapter based on Prometheus	TP	TP	REM

1.6.1. Deprecated features

1.6.1.1. Descheduler operator.openshift.io/v1beta1 API group is deprecated

The **operator.openshift.io/v1beta1** API group for the descheduler is deprecated and might be removed in a future release. Use the **operator.openshift.io/v1** API group instead.

1.6.1.2. Use of dhclient in Red Hat Enterprise Linux CoreOS (RHCOS) is deprecated

Starting with OpenShift Container Platform 4.6, Red Hat Enterprise Linux CoreOS (RHCOS) switched to using **NetworkManager** in the **initramfs** to configure networking during early boot. As part of this change, the use of the **dhclient** binary for DHCP was deprecated. Use the **NetworkManager** internal DHCP client for networking configuration instead. The **dhclient** binary will be removed from Red Hat Enterprise Linux CoreOS (RHCOS) in a future release. See BZ#1908462 for more information.

1.6.1.3. Cluster Loader is deprecated

Cluster Loader is now deprecated and will be removed in a future release.

1.6.1.4. The lastTriggeredImageID parameter in builds is deprecated

This release deprecates the **lastTriggeredImageID** in the **ImageChangeTrigger** object, which is one of the **BuildTriggerPolicy** types that can be set on a **BuildConfig** spec.

OpenShift Container Platform next release will remove support for **lastTriggeredImageID** and ignore it. Then, image change triggers will not start a build based on changes to the **lastTriggeredImageID** field in the **BuildConfig** spec. Instead, the image IDs that trigger a build will be recorded in the status of the **BuildConfig** object, which most users cannot change.

Therefore, update scripts and jobs that inspect **buildConfig.spec.triggers[i].imageChange.lastTriggeredImageID** accordingly. (BUILD-213)

1.6.1.5. The Jenkins Operator (Technology Preview) is deprecated

This release deprecates the Jenkins Operator, which was a Technology Preview feature. A future version of OpenShift Container Platform will remove the Jenkins Operator from OperatorHub in the OpenShift Container Platform web console interface. Then, upgrades for the Jenkins Operator will no longer be available, and the Operator will not be supported.

Customers can continue to deploy Jenkins on OpenShift Container Platform using the templates provided by the Samples Operator.

1.6.2. Removed features

External provisioner for AWS EFS (Technology Preview) feature has been removed

The Amazon Web Services (AWS) Elastic File System (EFS) Technology Preview feature has been removed and is no longer supported.

1.6.2.1. Images removed from samples imagestreams

The following images are no longer included in the samples imagestreams provided with OpenShift Container Platform:

```
registry.redhat.io/rhscl/nodejs-10-rhel7
registry.redhat.io/ubi7/nodejs-10
registry.redhat.io/rhscl/perl-526-rhel7
```

```
registry.redhat.io/rhscl/postgresql-10-rhel7
registry.redhat.io/rhscl/ruby-25-rhel7
registry.redhat.io/ubi7/ruby-25
registry.redhat.io/rhdm-7/rhdm-decisioncentral-rhel8:7.9.0
registry.redhat.io/rhdm-7/rhdm-kieserver-rhel8:7.9.0
registry.redhat.io/rhpam-7/rhpam-businesscentral-monitoring-rhel8:7.9.0
registry.redhat.io/rhpam-7/rhpam-businesscentral-rhel8:7.9.0
registry.redhat.io/rhpam-7/rhpam-smartrouter-rhel8:7.9.0
```

1.6.2.2. Package manifest format for Operators no longer supported

Support for the legacy package manifest format for Operators is removed in OpenShift Container Platform 4.8 and later. This removal of support includes custom catalogs that were built with the legacy format and Operator projects initially created in the legacy format with the Operator SDK. The bundle format is the preferred Operator packaging format for Operator Lifecycle Manager (OLM) starting in OpenShift Container Platform 4.6.

For more information about working with the bundle format, see Managing custom catalogs and Migrating package manifest projects to bundle format .

In addition, the following commands related to the format have been removed from OpenShift CLI (**oc**) and the Operator SDK CLI:

- **oc adm catalog build**
- **operator-sdk generate packagemanifest**
- **operator-sdk run packagemanifest**

1.6.2.3. Support for HPA custom metrics adapter based on Prometheus is removed

This release removes the Prometheus Adapter, which was a Technology Preview feature.

1.6.2.4. Secure token storage annotation recognition is removed

The **authentication** and **openshift-apiserver** Operators now ignore the **oauth-apiserver.openshift.io/secure-token-storage** annotation when picking the audit policy of a cluster. Audit policies now use **secure-** by default. For more information, see BZ#1879182.

1.7. BUG FIXES

assisted-installer

- Previously, the **assisted-service** container did not wait for **postgres** to start up and be ready to accept connections. The **assisted-service** container attempted to establish a database connection, failed, and the **assisted-service** container failed and restarted. This issue has been fixed by the **assisted-service** container attempting to connect to the database for up to 10 seconds. If **postgres** starts and is ready to accept connection within 10 seconds, the **assisted-service** container connects without going into an error state. If the **assisted-service** container is unable to connect to **postgres** within 10 seconds, it goes into an error state, restarts, and tries again. (BZ#1941859)

Bare Metal Hardware Provisioning

- Previously, Ironic failed to download an image for installation because Ironic uses HTTPS by default and did not have the correct certificate bundle available. This issue is fixed by setting the image download as **Insecure** to request a transfer without the certificate. (BZ#1953795)

- Previously, when using dual-stack networking, worker node host names sometimes did not match the host name that Ironic inspected before deployment. This caused nodes to need manual approval. This has been fixed. (BZ#1955114)

- Previously, in UEFI mode, the **ironic-python-agent** created a UEFI bootloader entry after downloading the RHCOS image. When using an RHCOS image based on RHEL 8.4, the image could fail to boot using this entry. If the entry installed by Ironic was used when booting the image, the boot could fail and output a BIOS error screen. This is fixed by the **ironic-python-agent** configuring the boot entry based on a CSV file located in the image, instead of using a fixed boot entry. The image boots properly without error. (BZ#1972213)

- Previously, a node sometimes selected the incorrect IP version upon startup (IPv6 instead of IPv4, or vice versa). The node would fail to start because it did not receive an IP address. This is fixed by the Cluster Bare Metal Operator passing the IP option to the downloader (**ip=dhcp** or **ip=dhcp6**), so this is set correctly at startup and the node starts as expected. (BZ#1946079)

- Previously, the image caching mechanism in Ironic was disabled to enable a direct connection to the HTTP server that hosts the virtualmedial iso to prevent local storage issues. Non-standard compliant HTTP clients and redfish implementations caused failures on BMC connections. This has been fixed by reverting to the default Ironic behavior where the virtualmedia iso is cached and served from the Ironic conductor node. Issues caused by non-standard compliant HTTP clients and redfish implementations have been fixed. (BZ#1962905)

- Previously, the machine instance **state** annotation was not set. Consequently, the **STATE** column was empty. With this update, the machine instance **state** annotation is now set, and information in the **STATE** column automatically populates. (BZ#1857008)

- Because newer ipmitool packages default to using cipher suite 17, older hardware that does not support cipher suite 17 fails during deployment. When cipher suite 17 is not supported by the hardware, Ironic now uses cipher suite 3 so that deployments on older hardware using ipmitool should succeed. (BZ#1897415)

- Previously, in some cases, adoption occurred before the image cache was populated, which resulted in permanent adoption failure, and no retry was attempted. This caused the control plane bare metal hosts to report **adoption failed.** With this update, adoption of externally provisioned hosts automatically retries after adoption failure until control plane hosts are correctly adopted. (BZ#1905577)

- Previously, the Custom Resource (CR) required the Baseboard Management Controller (BMC) details. However, with assisted installers this information was not provided. This update allows the CR to bypass the BMC details when the operator is not creating the nodes. (BZ#1913112)

- When provisioning an image to nodes, qemu-image was restricted to 1G of RAM, which could cause the qemu-img to crash. This fix increases the limit to 2G so that qemu-img now completes provisioning reliably. (BZ#1917482)

- Because the redfish/v1/SessionService URL requires authentication, Ironic would generate an authentication error when accessing the site. Because there was no functional problem when this error message was reported by Ironic, it has been removed. (BZ#1924816)

- For some drives, the partition, for example /**dev**/**sda1**, did not have a read-only file. The base device, for example, /**dev**/**sda** had this file, however. Therefore, Ironic could not determine that the partition was read-only, which could cause metadata cleaning to fail on that drive. This

update ensures that the partition is detected as read-only, and includes an additional check for the base device. As a result, metadata cleaning is not performed on the read-only partition and metadata cleaning no longer fails. (BZ#1935419)

- When a Baremetal IPI was deployed with a proxy configured, the internal machine-os image download was directed through the proxy. This corrupted the image and prevented it from being downloaded. This update fixes the internal image traffic to **no_proxy**, so that the image download no longer uses a proxy. (BZ#1962592)

- Previously, bare metal deployments failed if large packet transfers between Ironic and the RAM disk resulted in connection failures. With this update, Ironic queries the RAM disk for information to work around the connection error, allowing deployments to succeed. (BZ#1957976)

Builds

- Previously, after CVE-2021-3344 was fixed, builds did not automatically mount entitlement keys on the OpenShift Container Platform node. As a result, when the entitlement certificates were stored on the host or node, the fix prevented entitled builds from working seamlessly. The failure to bring in entitlement certificates stored on the host or node was fixed with BZ#1945692 in 4.7.z and BZ#1946363 in 4.6.z; however, those fixes introduced a benign warning message for builds running on Red Hat Enterprise Linux CoreOS (RHCOS) worker nodes. The current release fixes this issue by allowing builds to automatically mount entitlements only on RHEL worker nodes, and avoiding mount attempts on RHCOS worker nodes. Now, there will not be any benign warnings around entitlement mounts when running builds on RHCOS nodes. (BZ#1951084)

- Some users pulling images from Docker Hub can encounter the following error:

  > container image registry lookup failed...toomanyrequests: You have reached your pull rate limit

  This error happens because the **docker.io** login they used to call the **oc new-app** does not have sufficient paid support with **docker.io**. The resulting application is subject to image pull throttling, which can produce failures. The current release updates the **oc new-app** help to remind users how defaulting works for the image registry and repository specs, so users can, when possible, use non-default image references to avoid similar errors. (BZ#1928850)

- Previously, builds did not perform an error check to see if an image push had failed. As a result, builds always logged the **Successfully pushed** message. Now, builds check if an error has occurred, and only log the **Successfully pushed** message after an image push has succeeded. (BZ#1947164)

- Previously, the documentation and **oc explain** help text did not convey that the **buildArgs** field in the **BuildConfig** object does not support the **valueFrom** field of its underlying Kubernetes **EnvVar** type. As a result, users believed it was supported and tried to use it. The current release updates the documentation and help text, so it is more apparent that the **BuildConfig** object's **buildArgs** field does not support the **valueFrom** field. (BZ#1956826)

- When builds interact with image registries, such as pulling base images, intermittent communications issues can produce build failures. The current release increases the number of retries to these interactions. Now, OpenShift Container Platform builds are more resilient when they encounter intermittent communication issues with image registries. (BZ#1937535)

Cloud Compute

- Previously, **Cluster Image Registry Operator** considered **user_domain_name** an immutable field and would not modify it after installation. This resulted in a refusal to accept changes to

user_domain_name and resulting credentials. This update marks **user_domain_name** as mutable and does not store it in the image registry config. This allows **user_domain_name** and all other **auth** parameters to be modified after installation. (BZ#1937464)

- Previously, a proxy update caused a full cluster configuration update, including API server restart, during continuous integration (CI) runs. Consequently, some clusters in the Machine API Operator would time out because of unexpected API server outages. This update separates proxy tests and adds postconditions so that clusters in the Machine API Operator become stable again during CI runs. (BZ#1913341)

- Previously, deleting a machine stuck in **Insufficient disk space on datastore** took longer than expected because there was no distinction between various vCenter task types. With this update, the machine controller deletion procedure checks the vCenter task type, and no longer blocks the deletion of the machine controller. As a result, the machine controller is deleted quickly. (BZ#1918101)

- Previously, scaling from zero annotations requeued even if the instance type was missing. Consequently, there were constant requeue and error space messages in the MachineSet controller logs. With this update, users can set the annotation manually if the instance type is not resolved automatically. As a result, scaling from zero for unknown instance types works if users manually provide the annotation. (BZ#1918910)

- Previously, HTTP responses were not closed properly by the Machine API termination handler. Consequently, goroutines were leaked in **net.http** read and write loops, which led to high memory usage. This update ensures that HTTP responses are always closed properly. As a result, memory usage is now stable. (BZ#1934021)

- Previously, multiple client sets created inside of the MachineSet controller caused slow startup times, which resulted in pods failing readiness checks in some large clusters. Consequently, the MachineSet controller would get stuck in an endless loop. This update fixes the MachineSet controller so that it uses a single client. As a result, the MachineSet controller behaves as expected. (BZ#1934216)

- Previously, instances took longer to boot when an upgrade was performed by the Machine Config Daemon on the first boot. Consequently, worker nodes were stuck in restart loops, and machine healthchecks (MCHs) removed the worker nodes because they did not properly start. With this update, MHCs no longer remove nodes that have not started correctly. Instead, MHCs only remove nodes when explicitly requested. (BZ#1939054)

- Previously, a certificate signing request (CSR) approval was delayed for an unknown reason. Consequently, new machines appearing in the cluster during installation were not approved quickly enough, prolonging cluster installation. To mitigate occasional API server unavailability in early installation phases, this update changes the cache resync period from 10 hours to 10 minutes. As a result, control plane machines are now approved faster so that cluster installation is no longer prolonged. (BZ#1940972)

- Previously, the default Google Cloud Platform (GCP) image was out of date and referenced a version from the OpenShift Container Platform 4.6 release that did not support newer Ignition versions. Consequently, new machines in clusters that used the default GCP image were not able to boot OpenShift Container Platform 4.7 and later. With this update, the GCP image is updated to match the release version. As a result, new machines can now boot with the default GCP image. (BZ#1954597)

- Previously, due to a strict check of the virtual machine's (VM) ProvisioningState value, the VM would sometimes fail during an existence check. With this update, the check is more lenient so that only deleted machines go into a **Failed** phase during an existence check. (BZ#1957349)

- Previously, if you deleted a control plane machine using **oc delete machine** in an AWS cluster, the machine was not removed from the load balancers. As a result, the load balancer continued to serve requests to the removed control plane machine. With this fix, when you remove a control plane machine, the load balancer no longer serves requests to the machine. (BZ#1880757)

- Previously, when deleting an unreachable machine, the vSphere Virtual Machine Disk (VMDK) that is created for persistent volumes and attached to the node was incorrectly deleted. As a result, the data on the VMDK was unrecoverable. With this fix, the vSphere cloud provider checks for and detaches these disks from the node if the kubelet is not reachable. As a result, you can delete an unreachable machine without losing the VMDK. (BZ#1883993)

- Previously, because a generated list of AWS instance types was out of date, some newer Amazon Web Services (AWS) instance types were not able to scale from zero when using the Cluster Autoscaler Operator and machine sets with zero replicas. The list of AWS instance types is now updated to include newer instance types. With this fix, more instance types are available to the Cluster Autoscaler Operator for scaling from zero replicas. (BZ#1896321)

- Previously, pod disruption budgets did not drain pods on an unreachable node due to missing upstream eviction API features. As a result, machines on unreachable nodes could take excessive amounts of time to be removed after being deleted. Now, the grace period timeout is changed to 1 second when deleting machines on unreachable nodes. With this fix, the Machine API can successfully drain and delete nodes that are unreachable. (BZ#1905709)

Cloud Credential Operator

- Previously, the Cloud Credential Operator repeated an **unsupported platform type: BareMetal** warning on bare metal platforms. With this update, bare metal platforms are no longer treated as unknown platforms. As a result, misleading logging messages are reduced. (BZ#1864116)

- Previously, a recurring error message stored in the **credentialsRequest** custom resources (CRs) of the Cloud Credential Operator led to excessive CPU usage and logging in some error scenarios, such as Amazon Web Services (AWS) rate limiting. This update removes the request ID coming back from the cloud provider so that error messages are stored in conditions where users can more easily find them, and eliminates recurring error messages in the **credentialsRequest** CR. (BZ#1910396)

- Previously, both the Cloud Credential Operator (CCO) and the Cluster Version Operator (CVO) reported if the CCO deployment was unhealthy. This resulted in double reporting if there was an issue. With this release, the CCO no longer reports if its deployment is unhealthy. (BZ#1957424)

Cluster Version Operator

- Previously, the Cluster Version Operator evaluated both the **Available** and **Degraded** parameters when setting the **cluster_operator_up** metric, which caused the **ClusterOperatorDown** alert to be displayed for Operators with **Available=True** or **Degraded=True**, even though **Available=True** did not match the alert description of "has not been available". With this fix, the Cluster Version Operator now ignores the **Degraded** parameter when setting the **cluster_operator_up** metric. (BZ#1834551)

- Previously, when Prometheus was installed on the cluster, important platform topology metrics were not available and a CI error would occur if the installer metric that was generated with the invoker was set to **""**. The possible race condition in which informers were not synced before metrics were served that was causing the error has now been fixed. (BZ#1871303)

- Previously, manifests with multiple tolerations for the same key, such as the Cluster Version Operator's own deployment), would accept only the last entry read and overwrite prior entries. This caused **in-cluster tolerations** to diverge from the manifest's listed tolerations. With this update, the Cluster Version Operator now considers tolerations matching when they are completely equal. This allows the Cluster Version Operator to keep all tolerations present in the manifest for the **in-cluster resource**. (BZ#1941901)

- Previously, the Cluster Version Operator did not reconcile **env** and **envFrom** for manifests that did not set those properties. This meant the Cluster Version Operator did not properly manage container environments. This update improves the Cluster Version Operator so that it now clears **env** and **envFrom** if they are unset in the manifest. This allows the cluster to automatically recover from invalid **cluster-admin** changes to these properties. (BZ#1951339)

- Previously, manifests with multiple tolerations for the same key, such as the **cluster-version-operator** deployment object, would accept only the last entry read and overwrite prior entries. This caused in-cluster tolerations to diverge from the manifest's listed tolerations. With this update, the Cluster Version Operator now considers tolerations to match when they are equal. This allows the Cluster Version Operator to keep all tolerations present in the manifest for the in-cluster resource. (BZ#1941901)

- Previously, the Cluster Version Operator reported a **ClusterOperatorDegraded** alert when **ClusterOperator** resources were degraded for 10 minutes. This alert was sometimes occurred prematurely during installation as resources were still being created. This update changes the 10-minute period to 30 minutes, providing enough time for the installation to progress without premature **ClusterOperatorDegraded** alerts. (BZ#1957991)

Compliance Operator

- Previously, when users ran a compliance check, **NON-COMPLIANT** results were given with no indication of required remediation steps for the user to act upon. This release provides an **instructions** key that allows users to review the steps needed to verify a rule. This allows users and auditors to verify that an Operator is checking a correct value. (BZ#1919367)

Console Kubevirt Plug-in

- Previously, in a web console form that helped users add boot sources to virtualization templates, an explanatory text gave information only for Fedora, regardless of what operating system the template used. This update adds a fix that provides examples that are specific to the template's operating system so users have relevant guidance. (BZ#1908169)

- Previously, in a web console wizard that helped users create virtual machine templates, imprecise language made it unclear whether an operation applied to the template or to a virtual machine. This fix clarifies the description so users can make an informed decision. (BZ#1922063)

- Previously, a vague error message in the web console caused unnecessary confusion for some users who tried to add a network interface to a virtual machine they were creating from a template. This update adds detail to the error message so users can troubleshoot the error more easily. (BZ#1924788)

- Previously, when you tried to create a virtual machine from a Red Hat Enterprise Linux (RHEL) 6 template in the web console, a pop-up window gave information about how to define the support level, even through RHEL 6 is not supported. This fix changes the text in this window to make it clear that RHEL 6 is not supported. (BZ#1926776)

- Previously, a drop-down list in the web console was obscured by button elements, leaving users unable to select certain operating systems when creating a virtual machine. This fix includes an

adjustment to the button elements' **z-index** value, which corrects the error and lets users select any of the available operating systems. (BZ#1930015)

- Previously, if you used the web console's new virtual machine wizard on a cluster with no defined storage classes, the web console got stuck in an infinite loop and crashed. This fix removes the storage class drop-down list in instances where no storage classes are defined. As a result, the web console does not crash. (BZ#1930064)

- Previously, text in a button element did not clearly describe the button's function, which is to remove a VM template from a list of favorites. This fix updates the text to clarify what the button does. (BZ#1937941)

- Previously, for virtual machines that have a **RerunOnFailure** run strategy, stopping the virtual machine resulted in several UI elements becoming unresponsive, preventing users from reading status information or restarting the virtual machines. This update fixes the unresponsive elements so users can use those features. (BZ#1951209)

- Previously, for clusters that were configured to have a separate /**var** partition, querying the file system only returned the size of disks mounted in the root directory, excluding the size of the /**var** partition. This fix changes how that query runs, and users can now determine the total size of the file system on a cluster. (BZ#1960612)

Console Storage Plug-in

- Previously, the OpenShift Container Storage Operator displayed an error message when the correct storage class was not available. This update removes the error message and disables the **Next** button until the correct storage class is available. (BZ#1924641)

- Previously, when a user clicked the browser's back button while creating an internal-attached storage cluster, the installation wizard restarted the process. This update fixes the issue. (BZ#1928008)

- When you add a node to local volume discovery, you can now see a list of existing nodes, which reduces unnecessary navigation. (BZ#1947311)

- Previously, the **Create Storage Cluster** wizard let you enable an arbiter zone that had an undefined value. The fix in this update filters out undefined values so arbiter zones can be created only with defined values. (BZ#1926798)

- Previously, quick start cards were displayed incorrectly in the web console because of inconsistencies in how product titles were spelled and how the registered trade mark symbol was used. In this update, the product names are spelled correctly, and the registered trademark symbol appears consistently in the first card only. (BZ#1931760)

DNS

- Previously, BZ#1936587 set the global CoreDNS cache max TTL to 900 seconds. Consequently, NXDOMAIN records received from upstream resolvers were cached for 900 seconds. This update explicitly caches negative DNS response records for a maximum of 30 seconds. As a result, resolving NXDOMAINs are no longer cached for 900 seconds. (BZ#1943578)

- The fix for BZ#1953097 enabled the CoreDNS **bufsize** plug-in with a size of 1232 bytes. Some primitive DNS resolvers are not capable of receiving DNS response messages over UDP that are greater than 512 bytes. Consequently, some DNS resolvers, such as Go's internal DNS library, are

unable to receive verbose DNS responses from the DNS Operator. This update sets the CoreDNS **bufsize** to 512 bytes for all servers. As a result, UDP DNS messages are now properly received. (BZ#1966116)

- Previously, cluster upstream resolver returned DNS responses that exceeded 512 bytes via UDP. Consequently, coreDNS returned **SERVFAIL** or other error messages and forced the client to retry over TCP. This update enabled the coreDNS bufisze plug-in with a UDP buffer size of 1232 bytes. (BZ#1949361)

etcd

- Previously, there was a transport leak in the etcd Operator, which caused memory usage to grow over time. The memory leak has been fixed. (BZ#1925586)
- Previously, the **etcdInsufficientMembers** alert fired incorrectly. With this release, the alert is updated to include the pod label in addition to the instance label, so that the alert only fires when quorum is lost. (BZ#1929944)
- Previously, the readiness probe was not reporting the correct readiness due to the introduction of SO_REUSEADDR socket options, which caused the etcd pod to show as ready even though the etcd-quorum-guard failed. The readiness probe checks were updated to account for these options, and the etcd readiness probe now properly reflects the readiness of the operand. (BZ#1946607)
- Previously, the **spec.loglevel** field did not set the **log-level** flag on the etcd operand, so users could not change the etcd log level. Users can now set the log levels as follows:
  - **Debug**, **Trace**, and **TraceAll** log levels map to the etcd **debug** log level
  - **Default** or **Normal** log levels map to the etcd **info** log level

  For more information, see BZ#1948553.
- Previously, following an etcd process, the next process did not start until the relevant ports were released. With the addition of **SO_REUSEADDR** to this process, the ports can be reused immediately. For more information, see BZ#1927942.
- Previously, the etcd-endpoint's ConfigMap was left empty if the **network.Status.ServiceNetwork** field was unpopulated. As a result, the etcd Operator failed to scale up. A new feature in OpenShift Container Platform 4.8 allows the etcd Operator to scale up when the **network.Status.ServiceNetwork** field is unpopulated. (BZ#1902247)

Image Registry

- Previously, the image pruner stopped when it failed to delete an image. As a result, when two image pruners were trying to delete an image concurrently, one of them failed with the **not found** error. With this update, **not found** errors are ignored, which allows the image pruner to tolerate concurrent deletions. (BZ#1890828)
- Previously, a lack of route status inclusion during the Image Registry Operator status assessment meant that the Image Registry Operator was not degraded, even with routes in the **degraded** state. With this fix, the Image Registry Operator now fetches all configured routes and evaluates their statuses when assessing its own status. With this update, if any of the routes are **degraded**, the Image Registry Operator reports itself as **degraded** with an error message. (BZ#1902076)
- Previously, an automatically created Docker config secret did not include credentials for

integrated internal registry routes. Because no credentials were present for accessing the registry through any of its routes, pods attempting to reach the registry failed due to lack of authentication. This fix includes all configured registry routes in the default Docker credential secret. Now, pods can reach the integrated registry by any of its routes, as credentials now contain an entry for each route. (BZ#1911470)

- Previously the image registry was ignoring cluster wide **ImageContentSourcePolicy** (ICSP) rules. During pull-through, images mirrors were ignored, which caused pull failures in disconnected clusters. With this update, the registry pulls from mirrors if ICSP rules exist for the target repository. As a result, pulling an image from configured mirrors does not fail. (BZ#1918376)

- Previously, the Image Registry Operator did not update the **.status.readyReplicas** field of the config resource, so its value was always **0**. With this fix, the Image Registry Operator writes the number of ready image registry replicas from the deployment into the config. Now, this field shows how many image registry replicas are ready. (BZ#1923811)

- Azure recommends users use Storage Accounts **v2** instead of **v1**. Under certain security profiles, administrators can force Azure to not accept the creation of **v1** Storage Accounts. Because the image registry depends on **v1** Storage Accounts, a cluster install would fail in such environments. With this fix, during cluster bootstrap, the Image Registry Operator now attempts to create and use **V2** Storage Accounts. Clusters running on **v1** continue using **V1** Storage Accounts. Installation succeeds and Image Registry Operator now reports **Available**. (BZ#1929654)

ImageStreams

- Previously, performance was sometimes slow when importing multiple images from a stream. With this release, the number of concurrent requests to the image registry is increased from five to 50, resulting in improved performance. (BZ#1954715)

Insights Operator

- Previously, the Insights Operator did not collect Cluster Version Operator (CVO) pods or events in the **openshift-cluster-version** namespace. As a result, the Insights Operator did not display information about any problems that the CVO might experience, and users could not get diagnostic information about the CVO. The Insights Operator is now updated to gather the CVO pods and events from the **openshift-cluster-operator** namespace so that issues with the the CVO are reported by the Insights Operator. (BZ#1942271)

Installer

- Previously, DNSmasq required specifying the prefix length when an IPv6 network was anything other than /64. Consequently, control plane hosts failed to PXE boot. This update includes the subnet prefix length in the DNSmasq configuration. As a result, control plane hosts will now DHCP and PXE boot on IPv6 networks of any prefix length. (BZ#1927068)

- When installing to vSphere, the bootstrap machine sometimes did not correctly update the name servers in the **/etc/resolv.conf** file. As a result, the bootstrap machine could not access the temporary control plane, and installations failed. This fix includes changes that makes finding the right line to update more reliable. With the bootstrap manager now able to access its temporary control plane, installations are able to succeed. (BZ#1967355)

- Previously, the installer did not take into account the region where the bootstrap Ignition config should be located when generating its URL. Consequently, the bootstrap machine could not fetch the config from the provided URL because it was incorrect. This update takes the user's

region into account when generating the URL and selects the correct public endpoint. As a result, the installer always generates correct bootstrap Ignition URLs. (BZ#1934123)

- Previously, the default version of Azure's Minimum TLS was 1.0 when creating a storage account. Consequently, policy checks would fail. This update configures the openshift-installer Azure client to set the Minimum TLS version to 1.2 when creating a storage account. As a result, policy checks now pass. (BZ#1943157)

- Previously, private clusters deployed using IPI on Azure had an inbound NSG rule that allowed SSH to the bootstrap node. This allowance could trigger Azure's security policy. With this update, that NSG rule has been removed. (BZ#1943219)

- Previously, the installer did not recognize the **ap-northeast-3** AWS region. With this update, the installer allows installs to unknown regions that fit the pattern for known partitions, which allows users to create infrastructure in the **ap-northeast-3** AWS region. (BZ#1944268)

- Previously, on-premise platforms lacked the capability to create internal load balancers. With this update, a check has been added when users create manfiests to ensure that this strategy is only used on cloud platforms such as AWS, Azure, and GCP. (BZ#1953035)

- Previously, when naming Google Cloud Platform resources, a filter prevented certain names using the word **Google** from being used. This update adds a check in the installer on the cluster name that allows some variations of the word **Google** to be used when setting a name. (BZ#1955336)

- Previously, a bare metal installation with installer-provisioned infrastructure required that the installer process was able to communicate with the provisioning network. Now, the installer process can communicate with the virtual IP for the API server. This change enables cases when the provisioning network is not routable and the installer process is run from a remote location, such as Hive for Red Hat OpenStack Platform (RHOSP) or Red Hat Advanced Cluster Management. You might need to adjust your firewall rules to allow communication with TCP ports **6385** and **5050** on virtual IP for the API server. (BZ#1932799)

- Previously, when the installation on Red Hat OpenStack Platform (RHOSP) was provided an ID of a subnet that did not exist in the **platform.openstack.machinesSubnet** field, the **openshift-install** command produced a SIGSEGV and backtrace. Now, the **openshift-install** command is revised so that it produces an error like the following message:

  ```
  FATAL failed to fetch Metadata: failed to load asset "Install Config":
  platform.openstack.machinesSubnet: Not found: "<network-ID>"
  ```

  (BZ#1957809)

- Previously, the installation on Red Hat OpenStack Platform (RHOSP) failed unless the RHOSP HTTPS certificate was imported to the hosting device. Now, a successful installation occurs when the **cacert** value in **cloud.yaml** is set to the RHOSP HTTPS certificate. Importing the certificate to the host is no longer required. (BZ#1786314)

- Previously, an installation might fail due to inaccurate external network entries in **proxy.config.openshift.io**. A validation check now identifies these inaccuracies to enable correction. (BZ#1873649)

- Previously vague or confusing Terraform component descriptions are now replaced with clearer information. (BZ#1880758)

- A previous change to gophercloud/utils introduced a custom HTTP client that used a self-signed certificate. Because this change removed settings from **DefaultTransport**, including

those for proxy environment variables, this caused failures for installations that used both self-signed certificates and proxies. In this update, the custom HTTP client inherits settings from **DefaultTransport**, so now OpenShift Container Platform can be installed with self-signed certificates and proxies. (BZ#1925216)

- Previously, the installer did not take into account **defaultMachineSet** values in the install config during its validation which caused the installer to fail. This update applies the default values to the install config and starts validating empty fields. (BZ#1903055)

- Previously, **soft-anti-affinity** required the client to set a minimum Nova microversion. Most versions of Ansible OS server module did not automatically require the client to set a minimum. As a result, the soft-anti-affinity commands were likely to fail. This update fixes the use of the Python OpenStack client to set a Nova microversion when dealing with soft-anti-affinity. (BZ#1910067)

- Previously, OpenStack UPI playbooks did not tag all resources that were created. Consequently, the **openshift-install destroy** command failed to properly identify all of the cluster resources and looped over resource deletion until it reached a timeout, which left resources behind. This update adds missing tag instructions to OpenStack UPI playbooks. (BZ#1916593)

- Previously, **e2e-gcp-upi** did not succeed because of a Python package error which resulted in a failure. With this update, you can set the correct Python version, pip version, and the **CLOUDSDK_PYTHON** for gsutil to resolve the package error. (BZ#1917931)

- Previously, pip version 21 did not support installed Python version 2. As a consequence, this lead to an error resolving all dependent packages required to setup the container. In this update, the pip version is fixed to a value less than 21 to avoid the problem. (BZ#1922235)

- Previously, the installer collected information about the cloud twice. Consequently, there were double the number of requests to OpenStack API, which created an additional load on the cloud and increased the installation time. This update fixes the issue by collecting information about the cloud before it checks for quota and then reuses the same information for validations. (BZ#1923038)

- Previously, when deploying with IPv6 provision network with subnet other than /64, the DNSmasq required specifying the prefix length. Consequently, the hosts failed to PXE boot when using a non-/64 network. This update includes the prefix length in the DNSmasq configuration. As a result, the hosts succeed on DHCP and PXE boot on IPv6 networks of any prefix length. (BZ#1925291)

- Previously, the OpenShift Container Platform installer was not reporting IAM permission issues when removing the **Shared Subnet** tag even though the logging indicated that they were removed. This update checks the results for untagging and logging errors. The logs now indicate the status of untagging shared resources. (BZ#1926547)

- Previously, the Azure clusters were created with the disk type of Premium_LRS and with an instance type that did not support PremiumIO capabilities which caused the cluster to fail. This update checks to see if the instance type picked has the PremiumIO capabilities only if the disk type is Premium_LRS which is the default disk type. The code queries the Azure subscription and region to get the information required and returns an error if the conditions are not met. (BZ#1931115)

- Previously, the API VIP could become unavailable on the bootstrap when the API server restarted, which made the provisioning services unavailable and caused provisioning to fail. The provisioning services (Ironic) are now included in the VIP health checks, and the API VIP remains available. (BZ#1949859)

kube-apiserver

- Previously, Google Cloud Platform (GCP) load balancer health checkers left stale conntrack entries on the host, which caused network interruptions to the API server traffic that used the GCP load balancers. The health check traffic no longer loops through the host, so there is no longer network disruption against the API server. (BZ#1925698)

Machine Config Operator

- Previously, the **drain timeout** and pool degrading period was too short and would cause alerts prematurely on a normal cluster that needed more time. With this update, the time needed before a timeout reports a failure is extended. This provides the Cluster Operator with more realistic and useful alerts without prematurely degrading performance of a normal cluster. (BZ#1968019)

- Previously, while creating a new virtual machine from VMware vSphere with the OpenShift Installer Provisioned Infrastructure (IPI), the node failed to join the cluster. This occurred when Dynamic Host Configuration Protocol (DHCP), entered a host name in place of the name provided by IDI. This has been resolved. (BZ#1920807)

- Previously, installation might fail if the network was enabled before the host name was set. This prevented the node from joining the cluser and forced a five minute delay before another attempt could be made. This is now resolved and the node automatically joins the cluster during the first attempt. (BZ#1899187)

- Previously, users were able to delete the core user and related SSH keys, although the keys remained. With this update, users cannot delete the core user. (BZ#1885186)

- When upgrading from 4.6 to 4.7, the host name set by the **vsphere-hostname** service was applied only when a node was installed. If the host name was not statically set prior to upgrading, the host name could be lost. This update removes the condition which allowed the **vsphere-hostname** service to run only when a node is installed. As a result, vSphere host names are no longer lost when upgrading. (BZ#1942207)

- Due to a bug in **keepalived** 2.0.10, if a liveness probe killed a **keepalived** container, any virtual IP addresses (VIPs) that were assigned to the system remained and were not cleaned up when **keepalived** restarted. As a result, multiple nodes could hold the same VIP. Now, the VIPs are removed when **keepalived** is started. As a result, VIPs are held by a single node. (BZ#1931505)

- Previously, rpm-ostree related operations were not handled properly on non-CoreOS nodes such as Red Hat Enterprise Linux CoreOS (RHCOS). As a result, RHEL nodes were degraded when an operation, such as kernel switching, was applied in the pool that contained RHEL nodes. With this update, the Machine Config Daemon logs a message whenever a non-supported operation is performed on non-CoreOS nodes. After logging the message, it returns nil instead of an error. RHEL nodes in the pool now proceed as expected when an unsupported operation is performed by the Machine Config Daemon. (BZ#1952368)

- Previously, empty static pod files were being written to the **/etc/kubernetes/manifests** directory. As a result, the kubelet log was reporting errors that could cause confusion with some users. Empty manifests are now moved to a different location when they are not needed. As a result, the errors do not appear in the kubelet log. (BZ#1927042)

Metering Operator

- Previously, the Reporting Operator incorrectly handled **Report** custom resources (CRs) that contained a user-provided retention period when reconciling events. Consequently, an expired **Report** CR would cause the Reporting Operator to continually loop, as the affected custom

resources are requeued indefinitely. This update avoids requeueing expired **Report** CRs that have specified a retention period. As a result, the Reporting Operator correctly handles events for expired **Report** CRs. (BZ#1926984)

Monitoring

- Previously, the **mountstats** collector for the **node-exporter** daemontset caused high memory usage on nodes with NFS mount points. With this update, users can now disable the **mountstats** collector to reduce memory usage. (BZ#1955467)

Networking

- Previously, an incorrect **keepalived** setting sometimes resulted in the VIP ending up on an incorrect system and unable to move back to the correct system. With this update, the incorrect **keepalived** setting is removed so that the VIP ends up on the correct system. (BZ#1916890)

- Due to iptables rewriting rules, clients that used a fixed source port to connect to a service via both the service IP and a pod IP might have encountered problems with port conflicts. With this update, an additional OVS rule is inserted to notice when port conflicts occur and to do an extra SNAT to avoid said conflicts. As a result, there are no longer port conflicts when connecting to a service. (BZ#1910378)

- Previously, IP port 9 between control plane nodes and egress-assigned nodes was blocked by the internal firewall. This caused the assignment of IP addresses to egress nodes to fail. This update enables access between control plane and egress nodes via IP port 9. As a result, the assignment of IP addresses to egress nodes is now successfully permitted. (BZ#1942856)

- Previously, UDP services traffic could be blocked because of stale connection tracking entries that were no longer valid. This pevented access to a server pod after it was cycled for **NodePort** service. With this update, the connection tracking entries are purged in the case of **NodePort** service cycling, which allows new network traffic to reach cycled endpoints. (BZ#1949063)

- Previously, OVN-Kubernetes network provider ignored network policies with multiple **ipBlocks**. Every ipBlock after the first one was ignored, resulting in pods being unable to reach all of the configured IP addresses. The code for generating OVN ACLs from Kubernetes network policies has been corrected. As a result, network policies with multiple **ipBlocks** now work correctly. (BZ#1953680)

- Previously, when using the OVN-Kubernetes cluster network provider, a Kubernetes service without any endpoints erroneously accepted connections. With this update, a load balancer is no longer created for services without endpoints and therefore traffic is no longer accepted. (BZ#1918442)

- Previously, the Container Network Interface (CNI) plug-in for Multus did not understand IPv6 addresses that started with any number of zeros. With this update, the CNI plug-in works with IPv6 that start with values greater than zero. (BZ#1919048)

- Previously, a race condition might be triggered if the SR-IOV Network Operator initiated a reboot when a change in the machine config policy also triggered a reboot. If this occurred, the node was left in an indeterminate state. With this update, that situation is avoided. (BZ#1921321)

- Previously, when creating a new user-provisioned cluster with the Kuryr cluster network provider, the OpenStack subset used by the cluster nodes might be undetected, which caused the cluster installation to time out. With this update, the subnet is correctly detected and a user-provisioned installation succeeds. (BZ#1927244)

- Previously, when upgrading from OpenShift Container Platform 4.6 to OpenShift Container Platform 4.7, the Cluster Network Operator (CNO) incorrectly marked itself as having completed its upgrade to the next version. If the upgrade subsequently failed then the CNO reported itself as **degraded**, but erroneously as being at version 4.7. With this update, the CNO waits for the cluster network provider images to upgrade successfully before reporting the CNO upgrade as successful. (BZ#1928157)

- Previously, when using the OVN-Kubernetes cluster network provider, the endpoint slice controller might not run if the Kubernetes version included a minor version that contained non-numeric characters. With this update, the endpoint slice controller is enabled by default. (BZ#1929314)

- When using the Kuryr cluster network provider, Neutron Ports created subsequent to the installation were named with a different pattern than Neutron Ports created during installation. As a result, Neutron Ports created after installation were not added to the default load balancer. With this update, Kuryr detects Neutron Ports created with either naming convention. (BZ#1933269)

- Previously, Open Virtual Network (OVN) changed the source IP addresses of hairpin traffic packets to the IP address of the load balancer, which sometimes blocked traffic when a network policy was in use. With this update, Kuryr opens traffic to the IP addresses of all services in a network policy's namespace, and hairpin traffic flows freely. (BZ#1920532)

- Previously, when starting a single-stack IPv6 cluster on nodes with IPv4 address, the kubelet might have used the IPv4 IP instead of the IPv6 IP for the node IP. Consequently, host network pods would have IPv4 IPs rather than IPv6 IPs, which made them unreachable from IPv6-only pods. This update fixes the node-IP-picking code, which results in the kubelet using the IPv6 IPs. (BZ#1939740)

- Previously, and for unknown reasons, a kubelet could register the wrong IP address for a node. As a consequence, the node would be in a **NotReady** state until it was rebooted. Now, the systemd manager configuration is reloaded with the valid IP address as an environment variable, meaning that nodes no longer enter a **NotReady** state because a kubelet registered the wrong IP address. (BZ#1940939)

- Previously, refactoring for a shadowed variable caused a regression related to the use of the checkpoint file, and SR-IOV pod sandboxes would not start. A check for the path of the kubelet socket was not properly accounted for during the refactor. The fix properly restores the check for the kubelet socket path, and now the SR-IOV pod sandboxes are properly created. (BZ#1968625)

Node

- Previously, Reliable Autonomic Distributed Object Store (RADOS) Block Devices (RBDs) were visible in unprivileged container pods running **lsblk**. This has been fixed and RBDs no longer are visible in unprivileged container pods running **lsblk**. (BZ#1772993).

- Previously, during a cluster upgrade, the **/etc/hostname** file was altered by CRI-O, which caused the nodes to fail and to return when rebooting. This update adds special handling in CRI-O to leave the **/etc/hosts** file alone during upgrade, which allows upgraded nodes to boot without trouble. (BZ#1921937)

- Previously, CRI-O was taking too long to create a pod after the network had been provisioned. This would trigger a bug in the network cleanup code, causing network resources not to properly clean up after network resources have been provisioned. This update changes the code to

properly clean up networking resources, even if a command has timed out. This allows the cluster to continue normal network operation even if pod creation was taking too long. (BZ#1957224)

- Previously, node reboots using a **CNI** plug-in would not complete successfully. CRI-O was modified to call **CNI DEL** on all containers that were running before reboot. This update cleans up **CNI** resources and allows a successful reboot. (BZ#1948137)

- Previously, a **CNI DEL** request would not be recalled if it failed, because **CNI** cleanup operations would not check for cleanup failure. Now, CRI-O recalls **CNI DEL** requests until they succeed, correctly cleaning up **CNI** resources. (BZ#1948047)

- Previously, a reboot request to a container or image could cause failure if the reboot occurred while the container or image was being committed to the disk. This caused apparent corruption of storage for the container and created failures to pull the image or recreate containers from the image. This update detects when a node has rebooted and clears the container storage if true. (BZ#1942536)

- Previously, **runc** took on the permissions of the entity that ran it. However, permissions on the **workdir** are set by the **container** user. When those permissions differ, container creation errors occured and caused failure of the container startup. This patch updates **runc** to **chdir** to the **workdir** multiple times, in case only one time fails. This ensures that creation of the container will succeed. (BZ#1934177)

- Previously, the CRI-O logs did not contain information about the source where images were pulled from. With this fix, the log pull source is added to the info level of the CRI-O logs. (BZ#1881694)

- Previously, when pods were created and deleted rapidly, a pod might not have enough time to complete the pod sandbox creation before the pod started deletion. As a result, pod deletion could fail with a `ErrCreatePodSandbox`error. This error is now ignored if a pod is terminating. As a result, pod termination no longer fails if a pod could not complete the pod sandbox creation. (BZ#1908378)

- Previously, the Machine Config Operator (MCO) did not accept **trace** as a valid log level. As a result, the MCO could not provide a method to enable trace-level logging, even though CRI-O supports it. The MCO is now updated to support the **trace** log level. As a result, users can see a trace log level through the MCO configurations. (BZ#1930620)

- Previously, the kubelet tried to get the status of images that are not completely pulled. As a result, **crictl** reports a **error locating item named "manifest"** error for these images. CRI-O is now updated to not list images that do not have a manifest. As a result, **crictl** no longer reports these errors. (BZ#1942608)

- Previously, outdated status messages were not removed. Because of this, the Machine Config Operator (MCO) was sometimes unable to locate the proper machine config pool. With this release, a cleanup function is added to limit the number of statuses. As a result, the MCO keeps at most 3 different kubeletConfig status. (BZ#1950133)

- Previously, when upgrading from OpenShift Container Platform version 4.6.25, in clusters with more than one **kubeletconfig** CR or **ContainerRuntimeConfig** CR, the Machine Config Operator (MCO) could generate duplicate machine configs for the same configuration. Consequently, the upgrade failed because the MCO would use the old controller version (IGNITIONVERSION 3.1.0). This update cleans up outdated duplicate machine configs and allows proper upgrading from version 4.6.25. (BZ#1955517)

oauth-apiserver

- Previously, some OAuth server metrics were not initialized properly and did not appear in searches in the Prometheus UI. The missing OAuth server metrics are now initialized properly and appear in the Prometheus UI metrics searches. (BZ#1892642)

- Previously, if a custom security context constraint (SCC) contained a combination of the **defaultAllowPrivilegeEscalation: false** and **allowPrivilegedContainer: true** fields, the security context mutator mutated the privileged **openshift-apiserver** and **oauth-apiserver** pods to a state that failed API validation. The pods failed to start, which sometimes caused an OpenShift API outage. The security context mutator now ignores the **defaultAllowPrivilegeEscalation** field for containers that are already privileged, and custom SCCs that include those fields do not prevent the pods from starting. (BZ#1934400)

oc

- Previously, when running the **oc explain** command, the resource group name was not automatically detected if it was provided as part of the resource string. If two resources in different groups had the same resource name, the highest priority definition was returned unless the group was stated through the **--api-version** parameter. Now, if the **--api-version** parameter is not included, a prefix check is run against the resource string to detect the group name. The explanation returned by the command relates to the matching resource in the stated group. (BZ#1725981)

- Previously, the **oc image extract** command did not extract files from the root directory of an image. The command has been updated and can now be used to extract files from the image root directory. (BZ#1919032)

- Previously, the **oc apply** command would fetch the OpenAPI specification on each invocation. The OpenAPI specification is now cached when the command is first run. The cached OpenAPI specification is reused when the **oc apply** command is run multiple times and the network load is reduced. (BZ#1921885)

- Previously, the authorization header created during image mirroring could exceed the header size limit for some registries. This would cause an error during the mirroring operation. Now, the **--skip-multiple-scopes** option is set to **true** for the **oc adm catalog mirror** command, to help prevent the authorization header from exceeding the header size limits. (BZ#1946839)

- Previously, the **storageClassName** attribute was not added to a **PersistentVolumeClaim** object when the **oc volume set** command included the **--claim-class** option. The value of the **--claim-class** option was added to the **volume.beta.kubernetes.io/storage-class** annotation instead. This would cause snapshots for the volume to fail due to a dependency on the **storageClassName** attribute. Now, the **oc volume set** command applies the value of the **--claim-class** option to the **storageClassName** attribute in the **PersistentVolumeClaim** object, and volume snapshots can reference the attribute value. (BZ#1954124)

- Previously, the output of **oc adm top --help** stated that the **oc adm top** command could display CPU, memory, and storage resource usage for pods and nodes. The **oc adm top** command does not display storage resource usage. Now, the storage reference is not included in the **oc adm top --help** output. (BZ#1959648)

Operator Lifecycle Manager (OLM)

- Previously, **CustomResourceDefinition** (CRD) objects applied as part of an Operator installation could sometimes satisfy the installation requirements of a newer version of the same Operator. Consequently, during an Operator upgrade, the version being replaced could be prematurely removed. In some cases, the upgrade would stop. With this update, the CRDs that are created or updated as part of the Operator bundle installation are annotated to indicate

their bundle of origin. These annotations are used by the **ClusterServiceVersion** (CSV) object to distinguish between pre-existing CRDs and same-bundle CRDs. As a result, upgrades will not complete until the current version's CRDs have been applied. (BZ#1947946)

- Previously, pods that ran an index referenced by a **CatalogSource** object did not have **readOnlyRootFileSystem: false** explicitly set in their **securityContext** field. Consequently, if a security context constraint (SCC) existed that enforced **readOnlyRootFileSystem: true** and matched the **securityContext** of the that pod, it would be assigned to that pod and cause it to fail repeatedly. This update explicitly sets **readOnlyRootFileSystem: false** in the **securityContext** field. As a result, pods that are referenced by a **CatalogSource** object are no longer matched to SCCs that enforce a read-only root file system and no longer fail. (BZ#1961472)

- Operator Lifecycle Manager (OLM) previously did not allow skipped versions to be installed if the version was specified in the **startingCSV** field during initial installations. This caused those skipped versions to be unable to be installed, even if users wanted to install them regardless of reasons why they are skipped. This fix updates OLM to allow users to install skipped versions only during initial installation by using the **startingCSV** specification in **Subscription** objects; users still cannot upgrade to skipped versions, as expected. (BZ#1906056)

- Because **k8s.io/apiserver** was not handling context errors for the webhook authorizer, context errors, such as timeouts, caused the authorizer to panic. This fix increments the API server version to include an upstream fix for the issue, and as a result, the authorizer can gracefully handle context errors. (BZ#1913525)

- Previously, the **oc adm catalog mirror** command could not be easily used to mirror Operator catalogs across an airgapped environment. With this enhancement, the contents of a catalog can be mirrored to a file system, placed onto removable media, and then mirrored back from the file system to a registry for usage by an airgapped cluster. (BZ#1919168)

- The Catalog Operator previously created bundle unpacking jobs for an install plan without setting a timeout. In the case of a non-existent or deleted bundle image, this caused the job to run forever and the install plan would stay in the **Installing** phase with no indication of the job's pod failing to resolve the image. With this fix, the Catalog Operator now sets a default **10m** timeout on bundle unpacking jobs, which can be configured by using **--bundle-unpack-timeout** flag. As a result, bundle unpacking jobs fail after the configured timeout, and the installation also transitions to a **Failed** phase with the reason visible in **status.conditions** and **status.bundleLookups.conditions** properties. (BZ#1921264)

- Operators that were installed on clusters prior to OpenShift Container Platform 4.6 were previously not identified as coming from a given Operator package for the purposes of dependency resolution and upgrade selection. This caused existing Operator installations to conflict with the criteria of their own subscription, which blocked upgrades and dependency resolution within the namespace. This fix updates OLM to infer the package name and version for Operators that are referenced by a subscription. As a result, upgrades and dependency resolution proceed as expected. (BZ#1921953)

- The **Info** log level used for transient errors caused verbose OLM Operator logs for the default configuration. This fix changes the transient error log level to **debug**. As a result, fewer non-critical logs are visible for the **debug** configuration. (BZ#1925614)

- Previously, the **spec.config.resources** section of a **Subscription** object was always applied to the installed deployment, even when it was unset or empty. This caused resources defined in the cluster service version (CSV) to be ignored, and only the resources defined in the **spec.config.resources** section of the **Subscription** object were used. This fix updates OLM to override deployment-specific resources only when the **spec.config.resources** section is set to a non-nil or non-empty value. (BZ#1926893)

- During dependency and upgrade resolution, subscription uniqueness was previously based on the subscribed package name. If two subscriptions in a namespace subscribe to the same package, they are treated as a single subscription internally, resulting in unexpected behaviors. With this fix, subscriptions are now uniquely identified internally within a namespace by **.metadata.name** instead of **.spec.name**. As a result, upgrade and dependency resolution behavior is consistent for namespaces containing multiple **Subscription** objects with the same **.spec.name.** (BZ#1932001)

- When less than one minute remains before an upcoming catalog update polling attempt, the interval jitter function truncates the resync interval down to zero. This caused the Operator Catalog to enter a hot-loop, wasting CPU cycles. This fix increases precision of the jitter function used to calculate resync delays. As a result, the Catalog Operator remains mostly idle until the next catalog update poll. (BZ#1932182)

- During an Operator upgrade, the owner reference of any associated **ServiceAccount** object was updated to point to the new **ClusterServiceVersion** (CSV) object instead of the old one. This could cause a race condition to occur between the OLM Operator, which reconciles CSVs, and the Catalog Operator, executes install plans, marking the old CSV as **Pending**/**RequirementsNotMet** due to the service account ownership change. This blocked upgrade completion while the new CSV waited indefinitely for the old CSV to indicate a healthy status. With this fix, instead of updating owner references in one step, the second owner is now appended to any existing owners. As a result, the same service account can satisfy requirements for both the old and the new CSV. (BZ#1934080)

- Cluster service versions (CSV) previously required associated service accounts to either have no **ownerReferences** values set or to have an **ownerReferences** value set to the related CSV. This caused **default** service account, which is not created as part of Operator installation, to be unsatisfied as a CSV requirement if its **metadata.ownerReferences** field was not empty. With this fix, CSVs now require associated service accounts to either have no **ownerReferences** values set to CSVs or to have an **ownerReference** value set to the related CSV. As a result, service accounts with only non-CSV **ownerReferences** values can satisfy the requirements of any CSV. (BZ#1935909)

- Prior to OpenShift Container Platform 4.5, the default catalogs deployed and managed by the Marketplace Operator in the **openshift-marketplace** namespace were created by **OperatorSource** objects, which was the API exposed by the Marketplace Operator. Appropriate metrics and alerting were instrumented to indicate an error encountered by the Operator sources. In OpenShift Container Platform 4.6, the **OperatorSource** resource was removed after being deprecated for several releases, and the Marketplace Operator instead directly created OLM's **CatalogSource** resource. However, the same metrics and alerting instrumentation was not done for catalog sources deployed in the **openshift-marketplace** namespace. Therefore, errors encountered by the default catalog sources were not highlighted with Prometheus alerting. This fix introduces the new **catalogsource_ready** metric in OLM, which is used by the Marketplace Operator to fire alerts whenever the metric for a default catalog source indicates that a catalog source is in an unready state. As a result, Prometheus alerts are now provided for unready default catalog sources in the **openshift-marketplace** namespace. (BZ#1936585)

- Previously, when a candidate Operator dependency was available from its default channel and a non-default channel, the Operator Lifecycle Manager (OLM) could generate a subscription that arbitrarily specified either of the two channels. Now, Operator dependencies are satisfied by candidates from the default channel first and then from other channels. (BZ#1945261)

- Previously, it was possible that a cluster service version (CSV) was copied as a component of multiple Operators. This could happen when a namespace was added to an Operator group after the Operator was installed. This behavior affected memory use and CPU load. Now, CSVs do not appear in the **status.components** field of an Operator with a reason of **Copied**, and performance is not affected. (BZ#1946838)

Operator SDK

- Previously, some resources were caught in infinite loops because **ManagedFields** was being processed during reconciliation. This fix updates **operator-lib** to ignore **ManagedFields**, resulting in consistently reconciled loops. (BZ#1856714)

- A minimum help message was being printed for the Operator SDK because the default plug-in was not invoked when **--help** was passed on the command line interface (CLI). This fix invokes the default plug-in and prints a more useful help message when a user runs the **operator-sdk init --help** command. (BZ#166222)

- Previously, if ran with missing optional validators, **operator-sdk bundle** would fail rather than issue warnings. This has been corrected. (BZ#1921727)

openshift-apiserver

- Previously, custom security context constraints (SCCs) could have a higher priority than others in a default set. Consequently, those SSCs were sometimes matched to **openshift-apiserver** pods, which broke their ability to write in their root file system. This bug also caused an outage of some OpenShift APIs. This fix explicitly mentions in the **openshift-apiserver** pods that the root file system should be writable. As a result, custom SCCs should not prevent **openshift-apiserver** pods from running. (BZ#1942725)

Performance Addon Operator

- Previously, when configuring a container to provide low latency response, dynamic interrupt masks with CRI-O did not match the interrupt mask set by **irqbalance** system service. Each one set different masks and compromised container latency. This update changes the interrupt mask set by setting CRI-O to match the **irqbalance** system service. As a result, dynamic interrupt mask handling now works as expected. (BZ#1934630)

RHCOS

- Previously, multipath was enabled too late in the boot process. Consequently, Red Hat Enterprise Linux CoreOS (RHCOS) would return I/O errors in some multipath environments. With this update, multipath is now enabled earlier in the boot process. As a result, RHCOS no longer returns I/O errors in some multipath environments. (BZ#1954025)

- Previously, a potential race condition could cause a fetch of the rootfs in a Red Hat Enterprise Linux CoreOS (RHCOS) PXE deployment to fail in some environments. With this fix, a connectivity check has been added that retries before an attempt to pull the rootfs so that access to the remote server and rootfs file is verified before continuing to the point where the **coreos-livepxe-rootfs** script used to sometimes fail. (BZ#1871303)

- Previously, user presets for **MachineConfig** were ignored. This meant users could not change the configuration of **kdump.service**. Now, the priority level of default presets are lower than user configured defaults, so the user configuration can properly override vendor configuration. (BZ#1969208)

- Previously, the **coreos-installer** would refuse to install onto a disk with a corrupted GUID Partition Table (GPT) because it would try to read the GPT of the target disk before overwriting it with the install image. With this fix, the **coreos-installer** now successfully installs onto a disk with a corrupt GPT by only reading the GPT of the target disk when it is instructed to preserve existing partitions. (BZ#1914976)

- Previously, installation of clusters on unformatted direct-access storage devices (DASD) would result in the creation of disk sectors written incorrectly by **coreos-installer**. Now **coreos-**

installer correctly formats new, unformatted DASD drives to 4096 byte sectors. This allows **coreos-installer** to complete the installation of the OS image to the disk drive. (BZ#1905159)

- Previously, hardware-assisted **zlib** decompression on s390x z15 systems caused the mounting of the RHEL rootfs image to fail, which resulted in boot failure for REHL s390x z15 nodes using the RHEL 8.3 kernel. The kernel has now been updated to correctly handle **zlib**-compressed squashfs files when hardware-assisted **zlib** compression is available. (BZ#1903383)

- Previously, the **zipl** command configured the disk geometry by assuming a sector size of 512 bytes. As a result, on SCSI disks with 4k sectors, the **zipl** bootloader configuration contained incorrect offsets and zVM was unable to boot. With this fix, **zipl** now takes the disk sector size into account so that zVM boots successfully. (BZ#1918723)

- Previously, **chrony.config** might automatically run multiple time and fail each time but the first. This caused issues because **chrony.config** configuration is set during the initial run and cannot be changed. These errors are now avoided by limiting the configuration setup process to the first time **chrony.config** runs. (BZ#1924869)

- Previously, nodes appeared unhealthy and did not operate as expected during periods of high workloads. This resulted from workloads using memory faster than the memory could be reclaimed. With this update, memory reclamation and out-of-memory situations were addressed and these conditions no longer occur during high workload situations. (BZ#1931467)

- Previously, the maximum transmission unit (MTU) specification for a bond interface using kernal arguments did not get assigned properly. This has been corrected. (BZ#1932502)

- Previously, the **clevis-luks-askpass.path** unit was not enabled by default. This caused non-root **LUKS Clevis** devices to fail to unlock automatically on reboot. This update enables the **clevis-luks-askpass.path** unit by default and allows non-root **LUKS Clevis** devices to unlock automatically on reboot. (BZ#1947490)

- Previously, systemd was excessively reading **mountinfo** and over-consuming CPU resources, which caused containers to fail to start. This update enables limits when **systemd** reads **mountinfo**, which allows containers to start successfully. (BZ#1957726)

- Previously, when the Machine Config Operator (MCO) invoked Ignition at startup to check the Ignition version, Ignition would crash. Consequently, the MCO would fail start. With this update, the MCO no longer queries the Ignition version, and the MCO starts successfully. (BZ#1927731)

Routing

- Previously, the HAProxyDown alert message was vague. Consequently, end users thought the alert meant that router pods, instead of just HAProxy pods, were unavailable. This update makes the HAProxyDown alert message clearer. (BZ#1941592)

- Previously, HAProxy's helper function template that was responsible for generating a file for whitelist IPs expected a wrong argument type. Consequently, no whitelist ACL was applied for the backend in long IP lists. With this update, argument types of the helper function template are changed so that whitelist ACL is applied to the backend of long IP lists. (BZ#1964486)

- Previously when creating an Ingress with a custom domain, the Ingress' status was updated by the OpenShift Container Platform Ingress controller with router canonical host name, and used **external-dns** to sync with Route 53. The problem was that the canonical router host name did not exist in the DNS and was not created by OpenShift Container Platform. OpenShift Container Platform creates the ***.apps.<cluster_name>.<base_domain>** DNS record and not the **apps.<cluster_name>.<base_domain>** DNS record. So the canonical router host name was not right. This fix sets the canonical router host name to **router-default.apps.**

<cluster_name>.<base_domain>. Cluster administrators that have automation that takes the canonical host name and prepends a wildcard or a subdomain should be aware that the canonical Ingress host name is set as **<ingress-controller-name>.apps.<cluster_name>.<base_domain>**. (BZ#1901648)

- Previously, the fix for BZ#1932401 overrode the default Go HTTP client transport. Consequently, cluster-wide proxy settings were not plumbed through to the Ingress Operator pod, which resulted in the failure of canary checks on a cluster with a cluster-wide egress proxy. This update explicitly sets proxy settings in the canary client's HTTP transport. As a result, canary checks work with all cluster-wide proxies. (BZ#1935528)

- Previously, the canary DaemonSet did not specify a node selector, so it used the default node selector for the canary namespace. Consequently, the canary DaemonSet could not schedule to infra nodes and in some cases would throw alerts. This update explicitly schedules the canary DaemonSet to infra nodes and tolerates tainted infra nodes. This allows the canary DaemonSet to safely roll out to worker and infra nodes without issues or alerts. (BZ#1933102)

- Previously, when upgrading a cluster from a prior version with an idled workload, the idled workload would not wake on HTTP request once upgraded to OpenShift Container Platform 4.6 or 4.7 due to **oc idle** feature fixups and reworks. With this update, idling changes are mirrored from endpoints to services on Ingress Operator startup. As a result, unidling workloads after upgrades works as expected. (BZ#1925245)

- Previously, exposing the default Ingress Controller through an external load balancer that redirected all HTTP traffic to HTTPS caused Ingress Canary endpoint checks performed by the Ingress Operator to fail, which would ultimately cause the Ingress Operator to become **degraded**. This fix converts the cleartext canary route to an edge encrypted route. Now the canary route works though HTTPS only load balancers when insecure traffic is redirected by the load balancer. (BZ#1932401)

- Previously, the Ingress Operator Canary Check Client sent canary requests over HTTP to load balancers that dropped HTTP traffic. This caused the Ingress Operator to become become **degraded** after canary checks failed. With this fix, instead of relying on a redirect from the router, the Ingress Operator Canary Check Client sends canary check requests over HTTPS from the start. Now, canary checks work for clusters that expose the default Ingress Controller through a load balancer that drops insecure HTTP traffic. (BZ#1934773)

- Previously, the HAProxy template used by **openshift-router** made repeated calls to a **firstMatch()** function. That function would parse and recompile a regular expression every time. Parsing and recompiling the regular expression on each call to **firstMatch()** is expensive, particularly for configurations that have many thousands of routes. With this fix, if the regular expression in the call to **firstMatch()** has already been seen, then an already compiled version is reused and cached. Now, there is a 60 percent reduction in run time when parsing and evaluating the **haproxy-config.template**. (BZ#1937972)

- Previously, users could name a route with an invalid host name by using an override annotation. This update fixes the issue. (BZ#1925697)

- Previously, removing **selector** from a service exposed via a route resulted in the duplication of **endpointslices** that would be created for the service's pods, which would trigger HAProxy reload errors due to duplicate server entries. This update filters out accidental duplicate server lines when writing out the HAProxy config file, so that deleting the selector from a service no longer causes the router to fail. (BZ#1961550)

Samples

- Previously, the Cluster Samples Operator could make changes to the controller cache for

objects it was watching, which caused errors when Kubernetes managed the controller cache. This update fixes how the Cluster Samples Operator uses information in the controller cache. As a result, the Cluster Samples Operator does not cause errors by modifying controller caches. (BZ#1949481)

service-ca

- OpenShift Container Platform 4.8 allows users to run **service-ca-operator** pods as a non-root user to suit their organization's needs. When run as a non-root user, the **service-ca-operator** runs as the following UID and GID:

  ```
  uid=1001(1001) gid=1001 groups=1001
  ```

  (BZ#1914446)

Storage

- Previously, metrics for **block type PVC** filesystems were not being reported when requesting a **capacity breakdown**. This meant users received an inaccurate reporting of metrics across all their filesystems. With this update, **block type PVC** are included when requested by Kubelet. This provides an accurate reporting of all filesystem metrics. (BZ#1927359)

- Previously, **/var/lib/kubelet** was mounted twice in the **Cinder CSI Node Controller** container. This caused the **CSI Node Controller** to fail to start with an error indicating **/var/lib/kubelet/pods** is out of space. The fix removes the duplicate mount of **/var/lib/kubelet** and **/var/lib/kubelet/pods**, which allows the **CSI Node Controller** to run successfully. (BZ#1952211)

- Previously, during Cinder CSI Driver resizing of a persistent volume (PV), the **findmnt** command received multiple volume mounts and could not choose the correct one, thereby causing resizing to stop. As a result, users would have to extend the file system manually. With this fix, the command now uses the first mount so that the file system is resized along with the PV. (BZ#1919291)

- The Cinder CSI Driver Operator now automatically provisions a default **VolumeSnapshotClass** object for Cinder CSI when creating a default storage class, rather than having to create the **VolumeSnapshotClass** object manually. (BZ#1905849)

- Previously, the recycler-pod template was incorrectly placed in the kubelet static manifest directory. This incorrect location produced static pod log messages that indicated a recycler static pod start failure. With this update, the misplaced recycler-pod template has been removed from the static pod manifest directory. As a result, the error messages no longer appear. (BZ#1896226)

- Previously, the Local Storage Operator (LSO) could claim disks belonging to other provisioners because busy disks were erroneously detected as free. Disks are now checked for bind-mounts so that the LSO cannot claim those disks. (BZ#1929175)

- Previously, the Local Storage Operator (LSO) would attempt to create a persistent volume (PV) with an invalid label value, because the device-id contained unsupported characters such as **:**. This has been corrected by moving the device information from labels to annotations. (BZ#1933630)

- Previously, the Local Storage Operator (LSO) was not cleaning up persistent volumes (PVs) since the deleter was not being enqueued correctly. This caused PVs to remain in the **released** state. PVs are now enqueued correctly so that they delete properly. (BZ#1937145)

- Previously, the Fibre Channel volume was incorrectly unmounted from a node when a pod was deleted. This happened when a different pod that used the volume was deleted in the API server when the kubelet on the node was not running. With this update, Fibre Channel volumes correctly unmount when a new kubelet starts. Additionally, the volume cannot be mounted to multiple nodes until the new kubelet fully starts and cofnrims that the volume is unmounted, which ensures that Fibre Channel volumes are uncorrupted. (BZ#1954509)

Web console (Administrator perspective)

- Previously, when attempting to delete a custom resource within the CNV namespace in the console UI in developer mode, clicking **Delete** in resulted in the **Delete** button hanging in a stuck state. Additionally, an error message that appears when performing the same action in the CLI was not displaying. With this update, the error message displays as expected and the **Delete** button does not stick. (BZ#1939753)

- Previously, OperatorHub Provider Type **filter** property did not clearly show the relationship to **CatalogSource**. Because of this problem, users could not tell what the **filter** criteria meant. This patch updates the Provider Type **filter** to **Source**. This more clearly shows the relationship between **filter** and **CatalogSource**. (BZ#1919406)

- Previously, the **ResourceListDropdown** component in the **Resources** menu was not internationalized for some languages. With this update, the **Resources** menu is updated to better the user experience for non-English speakers. (BZ#1921267)

- Previously, some menu items, such as **Delete Persistent Volume Claim** were not internationalized correctly. Now, more menu items are correctly internationalized. (BZ#1926126)

- Previously, some text and warning messages for the **Add HorizontalPodAutoscaler** page were not internationalized. The text is now internationalized. (BZ#1926131)

- Previously, when users created an Operator with the Operator SDK and specified an annotation like **xDescriptors={"urn:alm:<...>:hidden"}** to hide a field from the Operator instance creation page, the field might still be visible on the page. Now, the hidden fields are omitted from the Operator instance creation page. (BZ#1966077)

- Previously, tables displayed incorrectly on mobile devices. With this update, tables now display correctly. (BZ#1927013)

- Previously, launching the OpenShift Container Platform web console may be slow. With this update, the web console launches quicker. (BZ#1927310)

- Previously, a lack of internationalized notifications to OpenShift Container Platform administrators detracted from the user experience. Now, internationalization is possible. (BZ#1927898)

- Previously, a lack of internationalized duration times on the **Cluster Utilization** dashboard detracted from the user experience. Now, internationalization is possible. (BZ#1927902)

- Previously, errors occurred when Operator Lifecycle Manager (OLM) status descriptors in the OpenShift Container Platform web console were assigned incompatible data types. Validation has been added, eliminating incompatible data types from processing, thus avoiding errors. Logged warnings also identify the incompatible status types. (BZ#1927941)

- The following OpenShift Container Platform web console views now support multi-faceted filtering:

  - **Home → Search** (the **Resources** tab)
  - **Home → Events** (the **Resources** tab)
  - **Workloads → Pods** (the **Filter** tab)

  For more information, see BZ#1930007.

- The following bug fixes address various translation issues for the OpenShift Container Platform web console:
  - BZ#1921780
  - BZ#1921781
  - BZ#1922992
  - BZ#1924585
  - BZ#1924747
  - BZ#1925083
- Previously, the web console relied on hard-coded channel strings to populate the channel modal dropdown. As a result, users could see channel values that may not be correct for their current version. Now, if the Cluster Version Operator does not supply the correct channels for a given version, the channel modal dropdown changes to a text input field and suggests channels and help text for the user. The console no longer relies on hard coded channel string. (BZ#1932281)
- Previously, timestamps were not correctly formatted for Chinese or Japanese languages. As a result, timestamps were harder to read, which provided a bad user experience. With this update, default timestamp formats are used for Chinese and Japanese in **Moment.js**, which provides a better user experience. (BZ#1932453)
- Previously, the **rowFilters** prop in the FilterToolbar component did not accept the **null** value. So if the **rowFilters** prop was undefined, the uncaught exception was thrown. Now, when the **rowFilters** prop is referenced in the FilterToolbar component, the **null** value is accepted. As a result, the FilterToolbar does not throw exceptions when **rowFilters** prop is undefined. (BZ#1937018)
- Previously, the wrong style of help text was applied to the field level help instances. Now, the correct style of help text is shown for the field level help instances and is consistent across the console. (BZ#1942749).
- Previously, the Operator Lifecycle Manamgent (OLM) status conditions descriptors were rendered as normal detail items on the resource details page. As a result, the **Conditions** table was rendered at half width. With this update, conditions descriptors are rendered as a full-width table below the normal **Conditions** table on the **Operand** details page. (BZ#1943238)
- Previously, the word "Ingresses" was translated for Chinese users, but the user experience was bad. Now the word "Ingress" is not translated. (BZ#1945816)
- Previously, the word "Operators" was translated for Chinese users, but the plural translation resulted in a bad user experience. Now the word "Operators" is not translated. (BZ#1945818)

- Previously, an incorrect code was causing the **User** and **Group** details to show unrelated subjects. Now, a code has been added to filter by **User** or **Group**, so the **User** and **Group** details show related subjects. (BZ#1951212)

- Previously, the pod Containers text was not internationalized, so there was a poor user experience. Now the pod Containers text has been internationalized, so the user experience is improved. (BZ#1937102)

- Previously, the **PackageManifest** list page items did not link to the details page, so users could not easily drill down into individual **PackageManifest** items from the list page. Now, each **PackageManifest** item is linked to the details page that matches the convention of the other list pages. Users can easily access the PackageManifest details page from the list page. (BZ#1938321)

- The **Completions** column of the **Jobs** table sorted by the number of **Desired** completions instead of the number of **Succeeded** completions. The data is presented as **# Succeeded of # Desired**, so when sorting by that column the results looked confusing because the data was sorted by the second number. The **Jobs Completions** column now sorts on the **# Succeeded** for better understanding. (BZ#1902003)

- The input labels in the **Manage Columns** modal were not clickable buttons, so you could not click them to manage the columns. With this bug fix, the labels are now clickable buttons that you can use to manage columns. (BZ#1908343)

- CSI provisioners were not listed when creating a storage class on the Google Cloud Platform. With this bug fix, the issue is resolved. (BZ#1910500)

- Previously, if the user clicked into a **Cluster Role** from the **User Management → Roles** list view, the back link from the details page is **Cluster Roles**, which provides a generic list view of **Cluster Roles**. This caused backward web console navigation to redirect to the incorrect page. With this release, the back link directs the user to the **Role/Bindings** list view from the **Cluster Role/RoleBinding** details page. This allows the user to correctly navigate backward in the web console. (BZ#1915971)

- Previously, **Created date time** was not displayed in a readable format, which made it difficult to understand and use the time shown in UTC. With this release, the displayed time is reformatted so that UTC is readable and understandable. (BZ#1917241)

- Previously, pod requests and limit calculations in the web console were incorrect. This was a result of not excluding completed pods or init containers. With this release, pods that are not needed in the calculation are excluded, which improves the accuracy of the results of the web console calculation for pod requests. (BZ#1918785)

- Previously, parsing undefined values resulted in a not a number (NaN) exception and the **Chart** tooltip showed a box with no values. With this release, a start date is specified when fetching data so that the **Chart** toolip shows correct values. This change ensures that the results are synced and that undefined values are not parsed. (BZ#1906304)

- During a previous bug fix, the download link for pod logs was changed to a standard HTML anchor element with an empty download attribute. Consequently, the download file lost the default file name format. This update adds a file name to the anchor element download attribute so that a default file name, formatted as **<pod-name>-<container-name>.log**, is used when downloading pod logs. (BZ#1945630)

- Previously, when a user had permission to create a resource but not permission to edit it, the web console YAML editor was incorrectly set to read-only mode. The editor content is now editable by users with create access for the resource. (BZ#1824911)

- Previously, the web console showed times in the 12-hour format in most places, and the 24-hour format in others. Additionally, the year was not displayed for dates more than one year in the past. With this release, dates and times are formatted consistently and match the user locale and language preference settings, and the year is displayed for dates more than one year in the past. (BZ#1862084)

- Previously, the web console was polling the **ClusterVersion** resource for users who didn't have the authority to view those events. This would output large numbers of errors in the console pod log. To avoid this, checking the user's persmissions before polling the resource is required, which eliminates unnecessary errors in the console pod log. (BZ#1848151)

- Previously, keyboard users of the YAML editor were unable to exit the editor. The **view shortcuts** popover outside of the editor was unavailable inside the editor for access by the user. With this update, users can display **Accessibility help** above the editor using the **opt + F1** keystrokes. This change allows keyboard users of the YAML editor to exit the editor using the correct keystrokes. (BZ#1874931)

- After the 4.x release of OpenShift Container Platform (OCP), binary secret files uploaded to the OCP 4 web console failed to load. This caused the installation to fail. With OpenShift Container Platform 4.8, this capability has been restored to the OCP 4 web console. Input of the required secret can now be accomplished using binary file format. (BZ#1879638)

- Previously, the fix for BZ#1871996 to properly create RoleBinding links consistently resulted in the inability to select the binding type when a namespace was selected. Consequently, users with an active namespace could not create a cluster RoleBinding without changing the active namespace to **All namespaces**. This update reverts part of the changes for BZ#1871996 so that users can create a cluster role binding regardless of active namespace. (BZ#1927882)

Web console (Developer perspective)

- Previously, when the label changed for making a service cluster local on the Developer Console, users were not able to create a Knative service. This update to the Knative service uses the latest supported label for **cluster-local** in order to enable users to create a Knative service as cluster-local from Developer Console. (BZ#1969951)

- Previously, the colors for the **Low** and **Medium** severity issues of the Image Manifest Vulnerabilities (IMVs) did not match the color representation shown in the (Quay.io) interface. As a result, when the user changed the severity order of vulnerabilities to **High**, the IMVs ordered the issues incorrectly. This created confusion when reviewing the IMVs. The current release fixes this issue. (BZ#1942716)

- Previously, the **Topology** view in the Developer perspective did not load if OpenShift namespace templates were not available because the Samples Operator was not installed. This update fixes the issue. (BZ#1949810)

- Previously, when you imported a devfile, the web console ignored the **build guidance** placeholder container which provided the configuration for environment variables, ports, and limits. The new deployment had a second container which could not start because the placeholder image could not be fetched and it missed the required configurations. Now, the **build guidance** container is dropped from the new deployment, and the container adds the environment variables, ports, and limit configurations. (BZ#1952214)

- Previously, when switching to the **Developer** perspective in another tab and reloading the project details, the routes tied to the perspective were not rendered and resulted in a **404** error. This update loads all inactive routes and switches to the correct perspective. (BZ#1929769)

- Previously, when an error occurred due to a user not having the required access for a

namespace, the **Workload** drop-down menu in the **Monitoring** dashboard page continuously displayed a loading-in-progress icon. The current release fixes this issue. Now, the **Monitoring** dashboard page displays an error message indicating that a **Forbidden** error has occurred. (BZ#1930546)

- Previously, an API server could fail to create a resource, which would return a 409 status code when there was a conflict updating a **resource quota** resource. Consequently, the resource would fail to create, and you might have had to retry the API request. With this update, the **OpenShift Console** web application attempts to retry the request 3 times when receiving a 409 status code, which is often sufficient for completing the request. In the event that a 409 status code continues to occur, an error will be displayed in the console. (BZ#1920699)

- Previously, when selecting the **YAML** tab, the **metadata.managedFields** section did not collapse immediately. This was due to an issue with the **Form** or **YAML** switcher for pages such as **Pipeline Builder** and **Edit HorizontalPodAutoscaler** (HPA). As a result, the part of the document where you tried to type collapsed. The **metadata.managedFields** section remained as is, and the cursor was reset to the starting position to the top left of the **YAML** editor. The current release fixes this issue. Now, upon loading the **YAML**, the **metadata.managedFields** section collapses immediately. (BZ#1932472)

- Previously, pipelines created in the **Git Import** flow for private repositories failed to run. This happened because the pipeline **ServiceAccount** object did not use secrets created by the **Git Import** flow for private Git repositories. With this update, you can add a secret name to the annotations of the pipeline **ServiceAccount** object, and add pipeline-specific annotations to the provided secret. As a result, pipelines for private Git repositories run successfully. (BZ#1970470)

- Previously, when users inserted a formatted **YAML** snippet in the **YAML** editor, the new selection did not match the new content in the snippet. The indentation was removed, and some random letters were seen in the selection. The current release fixes this issue. Now, the cursor remains in the position where it started and adds the missing indentation for the cursor end position. After inserting the **YAML** snippet, the new selection matches the new content. (BZ#1952545)

- Previously, annotations were passed to the specification of the Knative service as well as to the metadata. As a result, decorators were shown for associated revisions of Knative service in **Topology**. This release fixes this issue by passing annotations only to the Knative service metadata. Now, decorators are shown only for the Knative service in **Topology** and not associated revisions. (BZ#1954959)

- Previously, if you created a pipeline with parameters that had empty strings, for example, '', the fields in the OpenShift Container Platform web console would not accept the empty strings. The current release fixes this issue. Now, '' is supported as a valid default property within the parameters section. (BZ#1951043)

- Previously, users were not able to create a Knative service as a private service from the **Developer** perspective. This issue has now been fixed, by updating the label **'networking.knative.dev/visibility': 'cluster-local'**. (BZ#1970796)

- Previously, the Kamelets of type sink were shown in the catalog for event sources along with the type source. This issue has now been fixed by filtering Kamelets resources to list only the source type. (BZ#1972258)

Windows Containers

- Previously, the load balancer service became unstable when users would scale additional Windows nodes. With this update, the load balancer service is stabilized, which allows users to add multiple Windows nodes without erratic performance. (BZ#1905950)
- Previously, the **kube-proxy** service crashed unexpectedly after the load balancer was created if it was created after the Windows pods were running. With this update, the kube-proxy service does not crash when recreating the load balancer service. (BZ#1939968)
- Previously, empty IP address values in the load balancer's Ingress broke the data path. As a result, the Windows service was unreachable. With this update, the Windows service is reachable even if the IP address value is empty. (BZ#1952914)
- Previously, when users created a Windows pod with a projected volume, the pod would remain stuck in the **ContainerCreating** phase. With this update, the Windows pod creation successfully proceeds to the **Running** phase. (BZ#1973580)

1.8. TECHNOLOGY PREVIEW FEATURES

Some features in this release are currently in Technology Preview. These experimental features are not intended for production use. Note the following scope of support on the Red Hat Customer Portal for these features:

Technology Preview Features Support Scope

In the table below, features are marked with the following statuses:

- **TP**: *Technology Preview*
- **GA**: *General Availability*
- **-**: *Not Available*
- **DEP**: *Deprecated*

Table 1.2. Technology Preview tracker

Feature	OCP 4.6	OCP 4.7	OCP 4.8
Precision Time Protocol (PTP)	TP	TP	TP
oc CLI plug-ins	TP	TP	GA
Descheduler	TP	GA	GA
OVN-Kubernetes Pod network provider	GA	GA	GA
HPA for memory utilization	TP	GA	GA
Service Binding	TP	TP	TP
Log forwarding	GA	GA	GA
Monitoring for user-defined projects	GA	GA	GA

Feature	OCP 4.6	OCP 4.7	OCP 4.8
Raw Block with Cinder	TP	TP	GA
CSI volume snapshots	TP	GA	GA
CSI volume cloning	GA	GA	GA
CSI volume expansion	TP	TP	TP
vSphere Problem Detector Operator	-	GA	GA
CSI Azure Disk Driver Operator	-	-	TP
CSI GCP PD Driver Operator	-	TP	GA
CSI OpenStack Cinder Driver Operator	-	TP	TP
CSI AWS EBS Driver Operator	TP	TP	TP
CSI automatic migration	-	-	TP
Red Hat Virtualization (oVirt) CSI Driver Operator	GA	GA	GA
CSI inline ephemeral volumes	TP	TP	TP
CSI vSphere Driver Operator	-	-	TP
Automatic device discovery and provisioning with Local Storage Operator	TP	TP	TP
OpenShift Pipelines	TP	GA	GA
OpenShift GitOps	TP	GA	GA
OpenShift sandboxed containers	-	-	TP
Vertical Pod Autoscaler	TP	TP	GA
Cron jobs	TP	TP	GA
PodDisruptionBudget	TP	TP	GA
Operator API	GA	GA	GA
Adding kernel modules to nodes with kvc	TP	TP	TP

Feature	OCP 4.6	OCP 4.7	OCP 4.8
Egress router CNI plug-in	-	TP	GA
Scheduler profiles	-	TP	TP
Non-preempting priority classes	-	TP	TP
Kubernetes NMState Operator	-	TP	TP
Assisted Installer	-	TP	TP
AWS Security Token Service (STS)	-	TP	GA
Kdump	-	TP	TP
OpenShift Serverless	-	-	GA
Serverless functions	-	-	TP
Jenkins Operator	TP	TP	DEP
Driver Toolkit	-	-	TP

1.9. KNOWN ISSUES

- In OpenShift Container Platform 4.1, anonymous users could access discovery endpoints. Later releases revoked this access to reduce the possible attack surface for security exploits because some discovery endpoints are forwarded to aggregated API servers. However, unauthenticated access is preserved in upgraded clusters so that existing use cases are not broken.
  If you are a cluster administrator for a cluster that has been upgraded from OpenShift Container Platform 4.1 to 4.8, you can either revoke or continue to allow unauthenticated access. It is recommended to revoke unauthenticated access unless there is a specific need for it. If you do continue to allow unauthenticated access, be aware of the increased risks.

WARNING

If you have applications that rely on unauthenticated access, they might receive HTTP **403** errors if you revoke unauthenticated access.

Use the following script to revoke unauthenticated access to discovery endpoints:

```
## Snippet to remove unauthenticated group from all the cluster role bindings
$ for clusterrolebinding in cluster-status-binding discovery system:basic-user
system:discovery system:openshift:discovery ;
```

```
do
### Find the index of unauthenticated group in list of subjects
index=$(oc get clusterrolebinding ${clusterrolebinding} -o json | jq 'select(.subjects!=null) |
.subjects | map(.name=="system:unauthenticated") | index(true)');
### Remove the element at index from subjects array
oc patch clusterrolebinding ${clusterrolebinding} --type=json --patch "[{'op': 'remove','path':
'/subjects/$index'}]";
done
```

This script removes unauthenticated subjects from the following cluster role bindings:

- **cluster-status-binding**
- **discovery**
- **system:basic-user**
- **system:discovery**
- **system:openshift:discovery**

(BZ#1821771)

- The **oc annotate** command does not work for LDAP group names that contain an equal sign (**=**), because the command uses the equal sign as a delimiter between the annotation name and value. As a workaround, use **oc patch** or **oc edit** to add the annotation. (BZ#1917280)

- When powering on a virtual machine on vSphere with user-provisioned infrastructure, the process of scaling up a node might not work as expected. A known issue in the hypervisor configuration causes machines to be created within the hypervisor but not powered on. If a node appears to be stuck in the **Provisioning** state after scaling up a machine set, you can investigate the status of the virtual machine in the vSphere instance itself. Use the VMware commands **govc tasks** and **govc events** to determine the status of the virtual machine. Check for a similar error message to the following:

  ```
  Invalid memory setting: memory reservation (sched.mem.min) should be equal to
  memsize(8192).
  ```

  You can attempt to resolve the issue with the steps in this VMware KBase article. For more information, see the Red Hat Knowledgebase solution UPI vSphere Node scale-up doesn't work as expected. (**BZ#1918383**)

- The installation of RHCOS on a RHEL KVM installation on IBM Z fails when using an ECKD type DASD as a VirtIO block device. (BZ#1960485)

- An Open Virtual Network (OVN) bug causes persistent connectivity issues with Octavia load balancers. When Octavia load balancers are created, OVN might not plug them into some Neutron subnets. These load balancers might be unreachable for some of the Neutron subnets. This problem affects Neutron subnets, which are created for each OpenShift namespace, at random when Kuryr is configured. As a result, when this problem occurs the load balancer that implements OpenShift **Service** objects will be unreachable from OpenShift namespaces affected by the issue. Because of this bug, OpenShift Container Platform 4.8 deployments that use Kuryr SDN are not recommended on Red Hat OpenStack Platform (RHOSP) 16.1 with OVN and OVN Octavia configured until the bug is fixed. (BZ#1937392)

- The Console Operator does not properly update the **Ingress** resource with the **componentRoutes** conditions for either of the console's routes (**console** or **downloads**). (BZ#1954148)

- If you are using OpenShift sandboxed containers, you cannot use the **hostPath** volume in a OpenShift Container Platform cluster to mount a file or directory from the host node's file system into your pod. (BZ#1904609)

- If you are running Fedora on OpenShift sandboxed containers, you need a workaround to install some packages. Some packages, like **iputils**, require file access permission changes that OpenShift Container Platform does not grant to containers by default. To run containers that require such special permissions, it is necessary to add an annotation to the YAML file describing the workload, which tells **virtiofsd** to accept such file permissions for that workload. The required annotations are:

  ```
  io.katacontainers.config.hypervisor.virtio_fs_extra_args: [ "-o", "modcaps=+sys_admin", "-o", "xattr" ]
  ```

  (BZ#1915377)

- In the 4.8 release, adding a value to **kataConfgPoolSelector** by using the OpenShift Container Platform web console causes **scheduling.nodeSelector** to be populated with an empty value. Pods that use **RuntimeClass** with the value of **kata** might be scheduled to nodes that do not have the Kata Containers runtime installed.
  To work around this issue, specify the **nodeSelector** value manually in the **RuntimeClass kata** by running the following command:

  ```
  $ oc edit runtimeclass kata
  ```

  The following is an example of a **RuntimeClass** with the correct **nodeSelector** statement.

  ```
  apiVersion: node.k8s.io/v1
  handler: kata
  kind: RuntimeClass
  metadata:
    creationTimestamp: "2021-06-14T12:54:19Z"
    name: kata
  overhead:
    podFixed:
      cpu: 250m
      memory: 350Mi
  scheduling:
    nodeSelector:
      custom-kata-pool: "true"
  ```

  (KATA-764)

- The OpenShift sandboxed containers Operator details page on Operator Hub contains a few missing fields. The missing fields do not prevent you from installing the OpenShift sandboxed containers Operator in 4.8.
  (KATA-826)

- Creating multiple **KataConfig** custom resources results in a silent failure. The OpenShift Container Platform web console does not provide a prompt to notify the user that creating more than one custom resource has failed.

(KATA-725)

- Sometimes the Operator Hub in the OpenShift Container Platform web console does not display icons for an Operator.
(KATA-804)

- The OVN-Kubernetes network provider does not support the **externalTrafficPolicy** feature for **NodePort**- and **LoadBalancer**-type services. The **service.spec.externalTrafficPolicy** field determines whether traffic for a service is routed to node-local or cluster-wide endpoints. Currently, such traffic is routed by default to cluster-wide endpoints, and there is no way to limit traffic to node-local endpoints. This will be resolved in a future release. (BZ#1903408)

- Currently, a Kubernetes port collision issue can cause a breakdown in pod-to-pod communication, even after pods are redeployed. For detailed information and a workaround, see the Red Hat Knowledge Base solution Port collisions between pod and cluster IPs on OpenShift 4 with OVN-Kubernetes. (BZ#1939676, BZ#1939045)

- For clusters that use the OVN-Kubernetes network provider and whose compute nodes run RHEL 7.9, upgrading from OpenShift Container Platform 4.7 to OpenShift Container Platform 4.8 is blocked by BZ#1976232. To upgrade to release 4.8, you must wait for the 4.8 patch that includes the fix for this bug. (BZ#1976232)

- For clusters that use the OVN-Kubernetes network provider and upgrade from OpenShift Container Platform 4.7 to OpenShift Container Platform 4.8, a bug in OVN-Kubernetes can sometimes cause the pod IP address to become stale. The bug is a rarely experienced race conditition. As a consequence, during the upgrade to the 4.8 release, nodes fail to drain and some Operators report a status of **Degraded**. As a workaround, identify the pods that are stuck in the **CrashLoopBackOff** state and that did not complete the upgrade. Delete each pod with the **oc delete <pod-name>** command. (BZ#1974403)

- The description for the **tlsSecurityProfile** field of the **kubeletconfig** resource (for example when using the **oc explain** command) does not list the correct ciphers for the TLS security profiles. As a workaround, review the list of ciphers in the **/etc/kubernetes/kubelet.conf** file of an affected node. (BZ#1971899)

- When running CNF tests in regular mode on a single node, the logic in place to understand if the cluster is ready is missing details. In particular, creating an SR-IOV network will not create a network attachment definition until at least one minute elapses. All the DPDK tests fail in cascade. Run the CNF tests in regular mode skipping the DPDK feature when running against an installation on a single node, with the **-ginkgo.skip** parameter. Run CNF tests in Discovery mode to execute tests against an installation on a single node. (BZ#1970409)

- Currently, CNF-tests does not support secure boot with MLX NICs for SR-IOV and DPDK tests. You can run the CNF tests skipping the SR-IOV feature when running against a secure boot enabled environment in regular mode, with the **-ginkgo.skip** parameter. Running in Discovery mode is the recommended way to execute tests against a secure boot enabled environment with MLX cards. This will be resolved in a future release. (BZ#1975708)

- When the **ArgoCD** Operator is subscribed to and an ArgoCD and AppProject are started, launching the example application named **guestbook** fails because the image does not work in more restrictive OpenShift Container Platform environments. As a temporary workaround, users can ensure the **ArgoCD** Operator works properly by deploying the following example:

```
PROJ=younamespace
cat > $PROJ-app.yaml <<EOF
apiVersion: argoproj.io/v1alpha1
kind: Application
```

```
metadata:
  name: simple-restricted-webserver
  namespace: $PROJ
spec:
  destination:
    namespace: $PROJ
    server: https://kubernetes.default.svc
  project: default
  source:
    path: basic-nginx
    repoURL: 'https://github.com/opdev/argocd-example-restricted-apps.git'
    targetRevision: HEAD
EOF
oc create -f $PROJ-app.yaml
```

For more information, see BZ#1812212.

- If you have the console open in multiple tabs, some sidebar links in the **Developer** perspective do not directly link to the project, and there is an unexpected change in the selected project. This will be resolved in a future release. (BZ#1839101)

- Creating a passthrough route using Ingress fails when using **pathType: Prefix**. Instead, you can create a passthrough route by setting **pathType** to **ImplementationSpecific** and setting **path** to **''**:

  Sample Ingress YAML file

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress7
  namespace: test-ingress
  annotations:
    route.openshift.io/termination: passthrough
spec:
  rules:
  - host: <ingress-psql-example-test-ingress.apps>
    http:
      paths:
      - path: ''
        pathType: ImplementationSpecific
        backend:
          service:
            name: <ingress-psql-example>
            port:
              number: 8080
```

  For more information, see BZ#1878685.

- Currently, in the **Search** page, the **Pipelines** resources table is not immediately updated after you apply or remove the **Name** filter. However, if you refresh the page and expand the **Pipelines** section, the **Name** filter is applied. The same behavior is seen when you remove the **Name** filter. This will be resolved in a future release. (BZ#1901207).

- Documentation now describes that the **ProvisioningNetworkCIDR** value in the **Provisioning** custom resource. This limits the IPv6 provisioning networks to a limit of /64 due to **dnsmasq**. (BZ#1947293)

- To assist with troubleshooting, logs collected on boostratp failures by the installer now include IP addresses and routes for the control plane and bootstrap hosts. (BZ#1956079)

- When using a self-signed Amazon Commercial Cloud Services cluster, you cannot pull from or push to an internal image registry. As a workaround, you must set **spec.disableRedirect** to **true** in the **configs.imageregistry/cluster** resource. This lets you pull the image layers from the image registry rather than directly from S3 storage. (BZ#1924568)

- Previously, the topology URLs created for deployments using Bitbucket repository in the OpenShift Container Platform web console did not work if they included a branch name that contained a slash character. This was due to an issue with the Bitbucket API (BCLOUD-9969). The current release mitigates this issue. If a branch name contains a slash, the topology URLs point to the default branch page for the repository. This issue will be fixed in a future release of OpenShift Container Platform. (BZ#1969535).

- Installing OpenShift Container Platform (OCP) version 4.6 on Red Hat Virtualization (RHV) requires RHV version 4.4. If you are running an earlier version of OCP on RHV 4.3, do not update it to OCP version 4.6. Red Hat has not tested running OCP version 4.6 on RHV version 4.3 and does not support this combination. For additional information about tested integrations, see OpenShift Container Platform 4.x Tested Integrations (for x86_x64) .

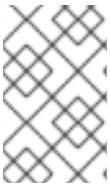
- The **operator-sdk pkgman-to-bundle** command exits with an error when run with the **--build-cmd** flag. For more information, see (BZ#1967369).

- Currently, the prerequisites in the web console quick start cards appear as a paragraph instead of a list. This will be resolved in a future release. (BZ#1905147)

1.10. ASYNCHRONOUS ERRATA UPDATES

Security, bug fix, and enhancement updates for OpenShift Container Platform 4.8 are released as asynchronous errata through the Red Hat Network. All OpenShift Container Platform 4.8 errata is available on the Red Hat Customer Portal . See the OpenShift Container Platform Life Cycle for more information about asynchronous errata.

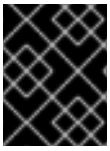
Red Hat Customer Portal users can enable errata notifications in the account settings for Red Hat Subscription Management (RHSM). When errata notifications are enabled, users are notified via email whenever new errata relevant to their registered systems are released.



NOTE

Red Hat Customer Portal user accounts must have systems registered and consuming OpenShift Container Platform entitlements for OpenShift Container Platform errata notification emails to generate.

This section will continue to be updated over time to provide notes on enhancements and bug fixes for future asynchronous errata releases of OpenShift Container Platform 4.8. Versioned asynchronous releases, for example with the form OpenShift Container Platform 4.8.z, will be detailed in subsections. In addition, releases in which the errata text cannot fit in the space provided by the advisory will be detailed in subsections that follow.



IMPORTANT

For any OpenShift Container Platform release, always review the instructions on updating your cluster properly.

1.10.1. RHSA-2021:2438 - OpenShift Container Platform 4.8.2 image release, bug fix, and security update advisory

Issued: 2021-07-27

OpenShift Container Platform release 4.8.2, which includes security updates, is now available. The bug fixes that are included in the update are listed in the RHSA-2021:2438 advisory. The RPM packages that are included in the update are provided by the RHSA-2021:2437 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.2 container image list

1.10.2. RHBA-2021:2896 - OpenShift Container Platform 4.8.3 bug fix update

Issued: 2021-08-02

OpenShift Container Platform release 4.8.3 is now available. The bug fixes that are included in the update are listed in the RHBA-2021:2896 advisory. The RPM packages that are included in the update are provided by the RHBA-2021:2899 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.3 container image list

1.10.2.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.3. RHSA-2021:2983 - OpenShift Container Platform 4.8.4 security and bug fix update

Issued: 2021-08-09

OpenShift Container Platform release 4.8.4 is now available. The bug fixes that are included in the update are listed in the RHSA-2021:2983 advisory. The RPM packages that are included in the update are provided by the RHSA-2021:2984 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.4 container image list

1.10.3.1. Bug fixes

- Previously, **BZ#1954309** and **BZ#1960446** were listed as fixed bugs in the OpenShift Container Platform 4.8.3 release notes, but were omitted from the version 4.8.3 release. With

this release, the bug fix summary for **BZ#1960446** is moved to the "Bug fixes" section of the OpenShift Container Platform 4.8.4 release notes, and the bug fix summary for **BZ#1954309** is removed.

- Previously, there was an incorrect toleration setting on the **nmstate-handler** pod, which made network configuration on nodes with the **nmstate** Operator impossible. With this update, the handler pod allows toleration on all nodes. (BZ#1960446)

- Previously, the web console showed **The operator is running in openshift-operators but is managing this namespace** for failed copied **ClusterServiceVersion** objects (CSVs). This message was not specific and did not help the user troubleshoot the failed CSV. With this release, the message for copied CSVs directs the user to the original CSV to find the cause of the failure, and provides a link to the original CSV. (BZ#1972478)

- Previously, the Operator to check if the registry should use custom tolerations checked **spec.nodeSelector** instead of **spec.tolerations**, but the custom tolerations from **spec.tolerations** are applied only when **spec.nodeSelector** is set. With this release, the **spec.tolerations** is checked and the Operator uses custom tolerations if **spec.tolerations** is set. (BZ#1973662)

- Previously, if a deployment was created without an image stream and no **image.openshift.io/triggers** annotation, the deployment controller created replica sets in infinite loop. The issue has been resolved in this release. (BZ#1981770)

- With this release, Manila CSI logs are added to the **must-gather** load. (BZ#1986026)

- Previously, when using the automatic pinning for a VM, the name of the property was **disabled**, **existing**, or **adjust**. With this release, the name better describes each policy, and **existing** was removed because it is blocked on oVirt. The new property names are **none** and **resize_and_pin**, which align with the oVirt user interface. (BZ#1987182)

1.10.3.2. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.4. RHBA-2021:3121 - OpenShift Container Platform 4.8.5 bug fix update

Issued: 2021-08-16

OpenShift Container Platform release 4.8.5 is now available. The bug fixes that are included in the update are listed in the RHBA-2021:3121 advisory. The RPM packages that are included in the update are provided by the RHBA-2021:3122 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.5 container image list

1.10.4.1. Features

1.10.4.1.1. Egress IP enhancement

A new enhancement adds egress IP address support to the OpenShift Container Platform 4.8 Anonymizer. For more information, see BZ#1974877.

1.10.4.2. Bug fixes

- Previously, **oc logs** did not work against **BuildConfig** objects with **JenkinsPipelineStrategy** defined. With this update, **oc logs** now works with pipeline builds. (BZ#1974267)
- Previously, when a **Keepalived** container that held virtual IP (VIP) was denoted as **SIGTERMed**, no VRRP proactive message was sent. Consequently, the VIP migrated to another node after it timed out. With this update, a **Keepalived** container that holds VIP denoted as **SIGTERMed** sends a **VRRP priority 0** advertisement message. As a result, there is now a faster VIP migration. (BZ#1920670)
- Previously, **Kameletbinding** could be used to create **action** and **sink** Kamelets, however, only Kamelets of the **source** type should have been listed. This update removes the option to select **sink** and **action** type Kamelets. As a result, **source** Kamelets are the only type shown in the catalog for event sources. (BZ#1972258)

1.10.4.3. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.5. RHBA-2021:3247 - OpenShift Container Platform 4.8.9 security and bug fix update

Issued: 2021-08-31

OpenShift Container Platform release 4.8.9 is now available. The bug fixes that are included in the update are listed in the RHBA-2021:3247 advisory. The RPM packages that are included in the update are provided by the RHSA-2021:3248 advisory.

IMPORTANT

The SHA-256 image digest information in the RHBA-2021:3247 advisory is incorrect. The correct information is as follows:

To inspect release image metadata, download the **oc** tool and run the following command:

- For x86_64 architecture:

```
$ oc adm release info quay.io/openshift-release-dev/ocp-release:4.8.9-x86_64
```

The image digest is
sha256:5fb4b4225498912357294785b96cde6b185eaed20bbf7a4d008c462134a4edfd

- For s390x architecture:

```
$ oc adm release info quay.io/openshift-release-dev/ocp-release:4.8.9-s390x
```

The image digest is
sha256:2665dcca917890b3d06c339bb03dac65b84485fef36c90f219f2773393ba291d

- For ppc64le architecture:

```
$ oc adm release info quay.io/openshift-release-dev/ocp-release:4.8.9-ppc64le
```

The image digest is
sha256:ded5e8d61915f74d938668cf58cdc9f37eb4172bc24e80c16c7fe1a6f84eff43

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.9 container image list

1.10.5.1. Bug fixes

- This release adds additional localized content in Chinese, Japanese, and Korean. (**BZ#1972987**)
- The address set naming convention used in OVN-Kubernetes for OpenShift Container Platform 4.5 was changed in OpenShift Container Platform 4.6, but the migration of existing address sets to the new naming convention was not handled as part of the upgrade. Network policies that were created in version 4.5 with namespace selector criteria for their ingress or egress sections rely on matching old address sets that were not kept up-to-date with the pod IP addresses within such namespaces. These policies might not work correctly in 4.6 or later releases and might allow or drop unexpected traffic.
  Previously, the workaround was to remove and recreate these policies. With this release, address sets with the old naming convention are removed, and policy ACLs referencing the old address sets are updated to reference the address sets following the new naming convention during the OVN-Kubernetes upgrade. Affected network policies created in version 4.5 work correctly again after upgrade. (**BZ#1976241**)

1.10.5.2. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.6. RHBA-2021:3299 - OpenShift Container Platform 4.8.10 bug fix update

Issued: 2021-09-06

OpenShift Container Platform release 4.8.10 is now available. The bug fixes that are included in the update are listed in the RHBA-2021:3299 advisory. The RPM packages that are included in the update are provided by the RHBA-2021:3300 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.10 container image list

1.10.6.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.7. RHBA-2021:3429 - OpenShift Container Platform 4.8.11 bug fix update

Issued: 2021-09-14

OpenShift Container Platform release 4.8.11 is now available. The bug fixes that are included in the update are listed in the RHBA-2021:3429 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.11 container image list

1.10.7.1. Bug fixes

- Previously, **Event Sources** could be found in the **Developer Catalog Group**. With this update, the **Serverless** add group has been renamed to **Eventing** and **Event Sources** can now be found within the **Eventing** add group. (BZ#1999931)

1.10.7.2. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.8. RHBA-2021:3511 - OpenShift Container Platform 4.8.12 bug fix update

Issued: 2021-09-21

OpenShift Container Platform release 4.8.12 is now available. The bug fixes that are included in the update are listed in the RHBA-2021:3511 advisory. The RPM packages that are included in the update are provided by the RHBA-2021:3512 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.12 container image list

1.10.8.1. Features

1.10.8.1.1. New minimum storage requirement for clusters

The minimum storage required to install an OpenShift Container Platform cluster has decreased from 120 GB to 100 GB. This update applies to all supported platforms.

1.10.8.2. Bug fixes

- Previously, the **oc** tool sent headers that were too big for some registries, which caused those registries to reject large mirroring requests. This update puts a limit on header size for the **oc adm catalog mirror** command, allowing mirroring to work as expected. (BZ#1874106)

- Before this update, the cluster autoscaler was unable to access the **csidrivers.storage.k8s.io** or **csistoragecapacities.storage.k8s.io** resources, which resulted in permissions errors. This fix updates the role assigned to the cluster autoscaler so that it includes permissions for these resources. (BZ#1995595)

1.10.8.3. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.9. RHBA-2021:3632 - OpenShift Container Platform 4.8.13 bug fix and security update

Issued: 2021-09-27

OpenShift Container Platform release 4.8.13, which includes security updates, is now available. The bug fixes that are included in the update are listed in the RHBA-2021:3632 advisory. The RPM packages that are included in the update are provided by the RHSA-2021:3631 advisory.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.13 container image list

1.10.9.1. Features

- Kubernetes 1.21.4 is now available. More information can be found in the following changelogs: 1.21.4, 1.21.3, and 1.21.2.

1.10.9.2. Bug fixes

- Previously, there was an unchecked index operation on a slice when using the **--max components** argument. Consequently, the **oc** client returned a panic error and crashed. This update adds a check to ensure that a value is not requested for an index that is out of range. As a result, when using the **--max-components** argument, the **oc** client no longer crashes. (**BZ#2004193**)

1.10.9.3. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.10.10. RHBA-2021:3682 - OpenShift Container Platform 4.8.14 bug fix update

Issued: 2021-10-11

OpenShift Container Platform release 4.8.14 is now available. The bug fixes that are included in the update are listed in the RHBA-2021:3682 advisory. There are no RPM packages for this release.

Space precluded documenting all of the container images for this release in the advisory. See the following article for notes on the container images in this release:

OpenShift Container Platform 4.8.14 container image list

1.10.10.1. Preparing to upgrade to the next OpenShift Container Platform release

OpenShift Container Platform 4.8.14 introduces a check that impacts upgrading to the next OpenShift Container Platform release, which is currently planned to be OpenShift Container Platform 4.9. This is because Kubernetes 1.22, which OpenShift Container Platform 4.9 is expected to use, removed a significant number of deprecated **v1beta1** APIs.

This check requires an administrator to provide a manual acknowledgment before the cluster can be upgraded from OpenShift Container Platform 4.8 to 4.9. This is to help prevent issues after upgrading to OpenShift Container Platform 4.9, where APIs that have been removed are still in use by the cluster. Administrators must evaluate their cluster for any removed APIs in use and migrate them to use the appropriate new API version. After this evaluation and migration is complete, the administrator can provide the acknowledgment.

All clusters require this administrator acknowledgment before they can be upgraded to OpenShift Container Platform 4.9.

For more information on the list of removed Kubernetes APIs, tips for how to evaluate your cluster for removed APIs in use, and how to provide the administrator acknowledgment, see Preparing to upgrade to OpenShift Container Platform 4.9.

1.10.10.2. Bug fixes

- Previously, when **provisioningHostIP** was set, it was being assigned to the Metal3 pod even in cases where the provisioning network was disabled. This no longer happens. (**BZ#1975711**)

- Previously, when using IPv6 DHCP, node interface addresses might be leased with a **/128** prefix. Consequently, OVN-Kubernetes uses the same prefix to infer the node's network and routes any other address traffic, including traffic to other cluster nodes, through the gateway. With this update, OVN-Kubernetes inspects the node's routing table and checks for the wider routing entry for the node's interface address and uses that prefix to infer the node's network. As a result, traffic to other cluster nodes is no longer routed through the gateway. (**BZ#1994624**)

1.10.10.3. Upgrading

To upgrade an existing OpenShift Container Platform 4.8 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

CHAPTER 2. OPENSHIFT CONTAINER PLATFORM VERSIONING POLICY

OpenShift Container Platform provides strict backwards compatibility guarantees for all supported APIs, excluding alpha APIs (which may be changed without notice) and beta APIs (which may occasionally be changed in a non-backwards compatible manner).

Red Hat did not publicly release OpenShift Container Platform 4.0 and, instead, released OpenShift Container Platform 4.1 directly after version 3.11.

The OpenShift Container Platform version must match between master and node hosts, excluding temporary mismatches during cluster upgrades. For example, in a 4.8 cluster, all masters must be 4.8 and all nodes must be 4.8. If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.8. You must download and install the new version of **oc**.

Changes of APIs for non-security related reasons will involve, at minimum, two minor releases (4.1 to 4.2 to 4.3, for example) to allow older **oc** to update. Using new capabilities may require newer **oc**. A 4.3 server may have additional capabilities that a 4.2 **oc** cannot use and a 4.3 **oc** may have additional capabilities that are not supported by a 4.2 server.

Table 2.1. Compatibility Matrix

	X.Y (**oc** Client)	**X.Y+N** footnote:versionpolicyn[Where **N** is a number greater than 1.] (**oc** Client)
X.Y (Server)	1	3
X.Y+N footnote:versionpolicyn[] (Server)	2	1

1 Fully compatible.

2 **oc** client may not be able to access server features.

3 **oc** client may provide options and features that may not be compatible with the accessed server.