OpenShift Container Platform 4.4

Release notes

Highlights of what is new and what has changed with the OpenShift Container Platform release
OpenShift Container Platform 4.4 Release notes

Highlights of what is new and what has changed with the OpenShift Container Platform release
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. OPENSШFT CONTAINER PLATFORM 4.4 RELEASE NOTES ........................................ 4
1.1. ABOUT THIS RELEASE .................................................. 4
1.2. NEW FEATURES AND ENHANCEMENTS ................................... 4
   1.2.1. Installation and upgrade .............................................. 4
      1.2.1.1. Installing a cluster on Microsoft Azure using user-provisioned infrastructure .... 4
      1.2.1.2. Installing a cluster on Red Hat Virtualization using installer-provisioned infrastructure .. 5
      1.2.1.3. Installing a cluster on OpenStack using user-provisioned infrastructure ............... 5
      1.2.1.4. Installing a cluster on OpenStack no longer requires the Swift object storage service .... 5
      1.2.1.5. Clusters installed on OpenStack support self-signed certificates ...................... 5
      1.2.1.6. OpenStack validates RHCOS images by checkingsha256 checksum .................... 5
      1.2.1.7. Support for east-west traffic with OVN load balancing on OpenStack with Kuryr .... 6
      1.2.1.8. Using upgrade channels for 4.4 release ........................................... 6
   1.2.2. Security ................................................................. 6
      1.2.2.1. Support for bound service account tokens ........................................... 6
      1.2.2.2. The oauth-proxy imagestream is now available ....................................... 6
      1.2.2.3. kube-apiserver checks client certificates before tokens ................................... 6
   1.2.3. Nodes ................................................................. 6
      1.2.3.1. Evicting Pods using the descheduler (Technology Preview) ......................... 6
      1.2.3.2. Controlling overcommit and managing container density on nodes ..................... 7
   1.2.4. Cluster monitoring ................................................... 7
      1.2.4.1. Monitoring Dashboards in web console ............................................. 7
      1.2.4.2. hwmon collector disabled in node-exporter ........................................ 7
      1.2.4.3. cluster-reader can read node metrics .................................................. 7
      1.2.4.4. Cluster alert for when multiple containers are killed ..................................... 7
      1.2.4.5. New API server alerts ................................................................. 7
      1.2.4.6. Permission updates for Prometheus Operator ........................................ 7
      1.2.4.7. Cluster monitoring component version updates ........................................ 8
   1.2.5. Web console .......................................................... 8
      1.2.5.1. IBM Marketplace integration in OperatorHub ........................................... 8
      1.2.5.2. Edit applications in the Topology view ............................................... 8
      1.2.5.3. Create Helm releases ................................................................. 8
   1.2.6. Networking ............................................................ 8
      1.2.6.1. Stream Control Transmission Protocol (SCTP) on OpenShift Container Platform .... 8
      1.2.6.2. Using DNS forwarding ................................................................. 8
      1.2.6.3. HAProxy upgraded to version 2.0 .................................................... 8
      1.2.6.4. Ingress Enhancements ................................................................. 9
   1.2.7. Storage ............................................................... 9
      1.2.7.1. Persistent storage using CSI snapshots (Technology Preview) ...................... 9
      1.2.7.2. Persistent storage using CSI cloning (Technology Preview) .......................... 9
   1.2.8. Scale ................................................................. 9
      1.2.8.1. Cluster maximums ................................................................. 9
   1.2.9. Developer experience ............................................... 9
      1.2.9.1. Automatic image pruning .......................................................... 9
      1.2.9.2. Build objects report conditions in status ........................................... 9
      1.2.9.3. Recreate rollouts for image registry .................................................. 10
      1.2.9.4. odo enhancements ................................................................. 10
      1.2.9.5. OpenShift Pipelines (Technology Preview) ....................................... 10
      1.2.9.6. Helm 3 GA support ................................................................. 10
   1.2.10. Operators .......................................................... 10
      1.2.10.1. etcd cluster Operator .............................................................. 10
      1.2.10.2. Insights Operator now collects anonymized CSRs .................................... 11
1.2.10.3. Remove Samples Operator if it cannot connect to registry.redhat.io

1.2.11. Documentation updates and conventions

1.2.11.1. OpenShift documentation licensed under Apache license 2.0

1.2.11.2. Copy button for docs.openshift.com site

1.2.11.3. OpenShift Container Engine renamed to OpenShift Kubernetes Engine

1.2.11.4. Documentation is now available for the 4.3 version of Azure Red Hat OpenShift

1.3. NOTABLE TECHNICAL CHANGES

Sending cluster logs using the Fluentd syslog plug-in (RFC 3164)

Operator SDK v0.15.0

1.4. DEPRECATED AND REMOVED FEATURES

1.4.1. Deprecated features

1.4.1.1. OpenShift CLI config flag

1.4.1.2. OpenShift CLI timeout flag

1.4.1.3. OpenShift editor

1.4.1.4. machineCIDR network parameter

1.4.1.5. Service Catalog, Template Service Broker, Ansible Service Broker, and their Operators

1.4.1.6. Deprecation of OperatorSources, CatalogSourceConfigs, and packaging format

1.4.1.6.1. Converting custom OperatorSources and CatalogSourceConfigs

1.4.2. Removed features

1.4.2.1. OpenShift CLI secrets subcommands

1.4.2.2. OpenShift CLI build-logs command

1.4.2.3. Deprecated upstream Kubernetes metrics have been removed

Kubelet metrics

Scheduler metrics

API server metrics

Docker metrics

Reflector metrics

etcd metrics

Transformation metrics

Other metrics

1.4.2.4. High granularity request duration buckets in Prometheus

1.5. BUG FIXES

1.6. TECHNOLOGY PREVIEW FEATURES

1.7. KNOWN ISSUES

1.8. ASYNCHRONOUS ERRATA UPDATES


1.8.2. RHSA-2020:1936 - Moderate: OpenShift Container Platform 4.4 Security Update

1.8.3. RHSA-2020:1937 - Moderate: OpenShift Container Platform 4.4 Security Update

1.8.4. RHSA-2020:1938 - Moderate: OpenShift Container Platform 4.4 Security Update

1.8.5. RHSA-2020:1939 - Moderate: OpenShift Container Platform 4.4 Security Update

1.8.6. RHSA-2020:1940 - Moderate: OpenShift Container Platform 4.4 Security Update

1.8.7. RHSA-2020:1942 - Moderate: OpenShift Container Platform 4.4 Security Update

1.8.8. RHBA-2020:2133 - OpenShift Container Platform 4.4.4 Bug Fix Update

1.8.8.1. Upgrading

1.8.9. RHSA-2020:2136 - Important: OpenShift Container Platform 4.4 Security Update

CHAPTER 2. OPENSHIFT CONTAINER PLATFORM VERSIONING POLICY
CHAPTER 1. OPENSShift CONTAINER PLATFORM 4.4
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 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

Red Hat OpenShift Container Platform (RHBA-2020:0581) is now available. This release uses Kubernetes 1.17 with CRI-O runtime. New features, changes, and known issues that pertain to OpenShift Container Platform 4.4 are included in this topic.

Red Hat did not publicly release OpenShift Container Platform 4.4.0 as the GA version and, instead, is releasing OpenShift Container Platform 4.4.3 as the GA version.

OpenShift Container Platform 4.4 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 4.4 is supported on Red Hat Enterprise Linux 7.6 or later, as well as Red Hat Enterprise Linux CoreOS (RHCOS) 4.4.

You must use RHCOS for the control plane, which are also known as master machines, and can use either RHCOS or Red Hat Enterprise Linux 7.6 or later for compute machines, which are also known as worker machines.

**IMPORTANT**

Because only Red Hat Enterprise Linux version 7.6 or later is supported for compute machines, you must not upgrade the Red Hat Enterprise Linux compute machines to version 8.

With the release of OpenShift Container Platform 4.4, version 4.1 is now end of life. For more information, see the Red Hat OpenShift Container Platform Life Cycle Policy.

1.2. NEW FEATURES AND ENHANCEMENTS

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

1.2.1. Installation and upgrade

1.2.1.1. Installing a cluster on Microsoft Azure using user-provisioned infrastructure

OpenShift Container Platform 4.4 introduces support for installing a cluster on Azure using user-provisioned infrastructure. Running user-provisioned infrastructure on Azure lets you use customizations your environment might require, like regulatory, security, and operational control.
You can incorporate example Azure Resource Manager (ARM) templates provided by Red Hat to assist in the deployment process, or create your own. You are also free to create the required resources through other methods; the ARM templates are just an example.

See Installing a cluster on Azure using ARM templates for details.

1.2.1.2. Installing a cluster on Red Hat Virtualization using installer-provisioned infrastructure

OpenShift Container Platform 4.4 introduces support for installing a cluster in a Red Hat Virtualization (RHV) environment using installer-provisioned infrastructure.

For more information, see Installing a cluster quickly on RHV.

1.2.1.3. Installing a cluster on OpenStack using user-provisioned infrastructure

OpenShift Container Platform 4.4 introduces support for installing a cluster on Red Hat OpenStack Platform (RHOSP) that runs on infrastructure that you provide. Using your own infrastructure allows you to integrate your cluster with existing infrastructure and modifications. For example, you must create all RHOSP resources, like Nova servers, Neutron ports, and security groups. Red Hat provides Ansible playbooks to help you with the deployment process.

You can also install a cluster on RHOSP with Kuryr using your own infrastructure.

For more information, see Installing a cluster on OpenStack on your own infrastructure or Installing a cluster on OpenStack with Kuryr on your own infrastructure.

1.2.1.4. Installing a cluster on OpenStack no longer requires the Swift object storage service

Beginning with version 4.4, OpenShift Container Platform no longer requires that the Swift object storage service be present on the RHOSP cloud where it is installed. If Swift is not available for the OpenShift Container Platform installation, the installer uses the Cinder block storage and Glance image registry services in its place.

For more information, see Installing a cluster on OpenStack using your own infrastructure.

1.2.1.5. Clusters installed on OpenStack support self-signed certificates

OpenShift Container Platform 4.4 can now be installed on RHOSP clouds that use self-signed certificates for authorization.

For more information, see Installing a cluster on OpenStack using your own infrastructure.

1.2.1.6. OpenStack validates RHCOS images by checking sha256 checksum

On RHOSP, the installer now performs automatic checksum validation of Red Hat Enterprise Linux CoreOS (RHCOS) images.

1.2.1.7. Support for east–west traffic with OVN load balancing on OpenStack with Kuryr

OpenShift Container Platform installations that use Kuryr on RHOSP 16 can now use the OVN load-balancing provider driver instead of the Amphora driver. If OVN and the OVN Octavia driver are present in the environment, OVN is used automatically. As a result, load balancer performance and resource utilization are improved. The need for a load balancer VM for each service is also removed.
1.2.1.8. Using upgrade channels for 4.4 release

Upgrades from the latest OpenShift Container Platform 4.3.z release to 4.4 will be available at GA for clusters that have switched to the fast-4.4 channel. Telemetry data from early adopters in the fast-4.4 channel will be monitored to inform when the upgrade is promoted into the stable-4.4 channel. This monitoring is above and beyond our extensive enterprise-grade testing and may take several weeks.

1.2.2. Security

1.2.2.1. Support for bound service account tokens

OpenShift Container Platform 4.4 provides support for bound service account tokens, which improves the ability to integrate with cloud provider identity access management (IAM) services, such as AWS IAM.

For more information, see Using bound service account tokens.

1.2.2.2. The oauth-proxy imagestream is now available

OpenShift Container Platform 4.4 introduces the oauth-proxy imagestream for third party authentication integration. You should no longer use the oauth-proxy image from the Red Hat Registry. You should instead use the openshift/oauth-proxy:v4.4 imagestream if you target OpenShift Container Platform 4.4 clusters and newer. This guarantees backwards compatibility and allows you to add imagestream triggers to get critical fixes. The v4.4 tag will be available for at least the next three OpenShift Container Platform minor releases without breaking changes. Each minor release will also introduce its own tag.

1.2.2.3. kube-apiserver checks client certificates before tokens

In previous versions of OpenShift Container Platform, the kube-apiserver checked tokens before client certificates for authentication. Now kube-apiserver checks client certificates before tokens.

For example, if you had a system:admin kubeconfig and ran the oc --token=foo get pod command in previous versions of OpenShift Container Platform, it would authenticate as a user with token foo. Now it authenticates as system:admin. The recommendation for past releases was to impersonate a user with the parameter --as in such cases instead of overriding the token when using a client certificate; this is no longer necessary.

1.2.3. Nodes

1.2.3.1. Evicting Pods using the descheduler (Technology Preview)

The descheduler provides the ability to evict a running Pod so that the Pod can be rescheduled onto a more suitable node.

You can benefit from descheduling Pods in situations such as the following:

- Nodes are underutilized or overutilized.
- Pod and node affinity requirements, such as taints or labels, have changed and the original scheduling decisions are no longer appropriate for certain nodes.
- Node failure requires Pods to be moved.
• New nodes are added to clusters.

See Evicting Pods using the descheduler for more information.

1.2.3.2. Controlling overcommit and managing container density on nodes

OpenShift Container Platform administrators can now control the level of overcommit and manage container density on nodes. You can configure cluster-level overcommit using the Cluster Resource Override Operator to override the ratio between requests and limits set on developer containers.

For more information, see Configuring your cluster to place Pods on overcommitted nodes.

1.2.4. Cluster monitoring

1.2.4.1. Monitoring Dashboards in web console

The Dashboards view is now available from the Monitoring section in the web console. This lets you view metrics that bring transparency to the OpenShift Container Platform cluster and its dependent components.

1.2.4.2. hwmon collector disabled in node-exporter

The hwmon collector has been disabled in the node-exporter monitoring component because it is no longer used to collect cluster metrics.

1.2.4.3. cluster-reader can read node metrics

The cluster-reader role can now read node metrics by default.

1.2.4.4. Cluster alert for when multiple containers are killed

You are notified with a MultipleContainersOOMKilled alert when multiple containers are killed within 15 minutes due to memory outages.

1.2.4.5. New API server alerts

There are two new API server alerts available for OpenShift Container Platform 4.4:

• ErrorBudgetBurn: fires when the API server issues 5xx request responses.

• AggregatedAPIErrors: fires when the number of errors have increased for the aggregated API servers.

1.2.4.6. Permission updates for Prometheus Operator

The custom resource definitions managed by the Prometheus Operator now have more restrictive permissions.

The custom resources the Prometheus Operator manages include:

• Prometheus

• ServiceMonitor
1.2.4.7. Cluster monitoring component version updates

The following monitoring components have been upgraded:

- Prometheus: version upgrade from 2.14.0 to 2.15.2
- Alertmanager: version upgrade from 0.19.0 to 0.20.0
- Prometheus Operator: version upgrade from 0.34.0 to 0.35.1
- kube-state-metrics: version upgrade from 1.8.0 to 1.9.5
- Grafana: version upgrade from 6.4.3 to 6.5.3

1.2.5. Web console

1.2.5.1. IBM Marketplace integration in OperatorHub

IBM Marketplace is now integrated with the OperatorHub, which is located in the OpenShift Container Platform web console. This integration allows you to install and manage Operators hosted on the IBM Marketplace from within the OperatorHub interface.

1.2.5.2. Edit applications in the Topology view

You can now edit applications from the Developer perspective by using the Topology view.

1.2.5.3. Create Helm releases

You can now create Helm releases from the Helm charts that are provided in the Developer Catalog.

1.2.6. Networking

1.2.6.1. Stream Control Transmission Protocol (SCTP) on OpenShift Container Platform

SCTP is a reliable message based protocol that runs on top of an IP network. When enabled, you can use SCTP as a protocol with both Pods and Services. For more information, see Using SCTP.

1.2.6.2. Using DNS forwarding

You can use DNS forwarding to override the forwarding default configuration on a per-zone basis by specifying which name server should be used for a given zone.

For more information, see Using DNS forwarding.

1.2.6.3. HAProxy upgraded to version 2.0

The HAProxy used for Ingress has been upgraded from version 1.8.17 to 2.0.13. This upgrade introduces no new APIs or supported user-facing capabilities to OpenShift Container Platform. The upgrade does
provide significant performance improvements and many bug fixes. HAProxy 2.0 also adds native Prometheus metrics and provides full IPv6 support when other OpenShift Container Platform components are configured to support it.

1.2.6.4. Ingress Enhancements

There are two noteworthy Ingress enhancements introduced in OpenShift Container Platform 4.4:

- Ingress gains a Route admission policy API: allows you to run applications in multiple namespaces with the same domain name.
- Ingress can be exposed by a NodePort service: facilitates integration of your existing load balancer so that you can have granular control over your load balancing solution.

1.2.7. Storage

1.2.7.1. Persistent storage using CSI snapshots (Technology Preview)

You can now use the Container Storage Interface (CSI) to create, restore, and delete a volume snapshot. This feature is enabled by default in Technology Preview.

For more information, see Using CSI volume snapshots.

1.2.7.2. Persistent storage using CSI cloning (Technology Preview)

You can now use the Container Storage Interface (CSI) to clone storage volumes after they have already been created. This feature is enabled by default in Technology Preview.

For more information, see Using CSI volume cloning.

1.2.8. Scale

1.2.8.1. Cluster maximums

Updated guidance around Cluster maximums for OpenShift Container Platform 4.4 is now available.

The 4.4 tested maximum for the number of Pods per node is 500.

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

1.2.9. Developer experience

1.2.9.1. Automatic image pruning

You can now enable automatic image pruning. This is not enabled by default; you will be notified of this option after installing or upgrading to OpenShift Container Platform 4.4. This automation is managed by the Image Registry Operator, which creates a CronJob to run periodic image pruning.

1.2.9.2. Build objects report conditions in status

Build conditions have been added for each existing OpenShift Container Platform build phase. These conditions contain information about the build during its build lifecycle. You can also use commands like oc wait to wait for a specific build phase to be reached.
1.2.9.3. Recreate rollouts for image registry

You can now use the **Recreate** rollout strategy when deploying the image registry. This lets you use **ReadWriteOnce** persistent volumes, such as AWS Elastic Block Store. When using these storage types, you must use the **Recreate** rollout strategy to successfully upgrade an OpenShift Container Platform cluster.

1.2.9.4. odo enhancements

**odo** has several enhancements and improvements that focus on the user experience:

- An **odo debug info** command is now available.
- The **odo url** command now has a **--secure** flag to specify HTTPS URLs.
- The **odo create**, **odo url**, and **odo config** commands now have a **--now** flag to apply changes on the cluster immediately.
- The **odo debug port-forward** command now selects a port automatically if the default port is occupied.
- The output of the **odo storage** and **odo push** commands is restructured for better readability.
- Experimental mode is now available, in which you may use Technology Preview features, such as creating applications using devfiles.
- Technology Preview feature - support of devfiles is now available. To learn more, see the odo [Release Notes](#).

1.2.9.5. OpenShift Pipelines (Technology Preview)

OpenShift Pipelines use Tekton Custom Resources (CR) to create extensible CI/CD solutions for automating deployments. These CRs serve as the building blocks to assemble the Pipeline. OpenShift Pipelines provide a catalog of reusable Tasks that can be used to easily build Pipelines. Each Pipeline runs in an isolated container without requiring the maintenance of a CI server and is portable across multiple platforms.

1.2.9.6. Helm 3 GA support

Helm is a package manager for Kubernetes and OpenShift Container Platform applications. It uses a packaging format called Helm charts to simplify defining, installing, and upgrading of applications and Services.

Helm CLI is built and shipped with OpenShift Container Platform and is available to download from the web console’s CLI menu.

1.2.10. Operators

1.2.10.1. etcd cluster Operator

OpenShift Container Platform 4.4 introduces the etcd cluster Operator, which handles the scaling of etcd and provisioning etcd dependencies such as TLS certificates. The etcd cluster Operator simplifies the disaster recovery procedure to restore to a previous cluster state, automates the addition of etcd members, provides more accurate etcd member health reporting, and reports events to assist with debugging the etcd cluster.
With this update, the names of the following disaster recovery scripts were changed:

- `etcd-snapshot-backup.sh` is now `cluster-backup.sh`.
- `etcd-snapshot-restore.sh` is now `cluster-restore.sh`.

For more information, see About disaster recovery.

### 1.2.10.2. Insights Operator now collects anonymized CSRs

With this enhancement, the Insights Operator periodically collects anonymized certificate signing requests (CSR) to identify CSRs that are not verified in Kubernetes or have not been approved. Additionally, the Insights Operator collects data if certificates are valid. As a result, this helps improve the OpenShift Container Platform customer support experience.

### 1.2.10.3. Remove Samples Operator if it cannot connect to registry.redhat.io

Sample imagestreams are not created if the Samples Operator cannot connect to `registry.redhat.io` during installation. This ensures that sample content installation does not fail OpenShift Container Platform cluster installation.

You can configure alternate or mirrored registries to bypass this issue if it arises during your cluster installation.

### 1.2.11. Documentation updates and conventions

#### 1.2.11.1. OpenShift documentation licensed under Apache license 2.0

The OpenShift documentation is now licensed under Apache license 2.0. It was previously licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported license.

#### 1.2.11.2. Copy button for docs.openshift.com site

All code blocks on docs.openshift.com now provide a Copy button that lets you copy all text in a code block to your machine’s clipboard. This capability is not available on the Customer Portal version of OpenShift documentation.

#### 1.2.11.3. OpenShift Container Engine renamed to OpenShift Kubernetes Engine

Red Hat has decided to rename Red Hat OpenShift Container Engine to Red Hat OpenShift Kubernetes Engine in order to better communicate what value the product offering delivers. For more information, see About OpenShift Kubernetes Engine.

#### 1.2.11.4. Documentation is now available for the 4.3 version of Azure Red Hat OpenShift

This new version is jointly managed, supported, and documented by both Red Hat and Microsoft:

- Microsoft documentation
- Red Hat documentation

### 1.3. NOTABLE TECHNICAL CHANGES

OpenShift Container Platform 4.4 introduces the following notable technical changes.
Sending cluster logs using the Fluentd syslog plug-in (RFC 3164)
Due to changes introduced with the Log Forwarding feature in OpenShift Container Platform 4.3, you could no longer use the Fluentd syslog plug-in to forward logs to an external syslog server. In OpenShift Container Platform 4.4, this functionality is restored and you can use the syslog plug-in. The procedure to configure the plug-in is different in OpenShift Container Platform version 4.4 than it was in version 4.2. For more information, see Sending logs using the Fluentd syslog plug-in (RFC 3164).

Operator SDK v0.15.0
OpenShift Container Platform 4.4 supports Operator SDK v0.15.0, which introduces the following notable technical changes:

- The `olm-catalog gen-csv` subcommand is now moved to the `generate csv` subcommand.
- The `up local` subcommand is now moved to the `run --local` subcommand.

1.4. 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 4.4, 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
- **DEP**: Deprecated
- **-**: Removed

Table 1.1. Deprecated and removed features tracker

<table>
<thead>
<tr>
<th>Feature</th>
<th>OCP 4.2</th>
<th>OCP 4.3</th>
<th>OCP 4.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Catalog</td>
<td>DEP</td>
<td>DEP</td>
<td>DEP</td>
</tr>
<tr>
<td>Template Service Broker</td>
<td>DEP</td>
<td>DEP</td>
<td>DEP</td>
</tr>
<tr>
<td>OpenShift Ansible Service Broker</td>
<td>DEP</td>
<td>DEP</td>
<td>-</td>
</tr>
<tr>
<td>OperatorSources</td>
<td>DEP</td>
<td>DEP</td>
<td>DEP</td>
</tr>
<tr>
<td>CatalogSourceConfigs</td>
<td>DEP</td>
<td>DEP</td>
<td>DEP</td>
</tr>
<tr>
<td>Operator Framework's Package Manifest Format</td>
<td>GA</td>
<td>GA</td>
<td>DEP</td>
</tr>
<tr>
<td>System Containers for Docker, CRI-O</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Hawkular Agent</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
### 1.4.1. Deprecated features

#### 1.4.1.1. OpenShift CLI config flag

The `--config` flag used with `oc` is deprecated. You should start using the `--kubeconfig` flag instead.

#### 1.4.1.2. OpenShift CLI timeout flag

The `--timeout` flag used with `oc rsh` is deprecated. You should start using the `--request-timeout` flag instead.

#### 1.4.1.3. OpenShift editor

The `OS_EDITOR` is deprecated. Users should start using `KUBE_EDITOR` or `EDITOR` instead.

#### 1.4.1.4. machineCIDR network parameter

The `machineCIDR` network parameter used in the `install-config.yaml` file is now deprecated. You should use `machineNetwork.cidr` instead.

#### 1.4.1.5. Service Catalog, Template Service Broker, Ansible Service Broker, and their Operators

**NOTE**

Service Catalog is not installed by default in OpenShift Container Platform 4.

Service Catalog, Template Service Broker, Ansible Service Broker, and their associated Operators were deprecated starting in OpenShift Container Platform 4.2.

Ansible Service Broker, the Ansible Service Broker Operator, and the following APBs are now removed in OpenShift Container Platform 4.4:

- APB base image
- APB tools container
- PostgreSQL APB
- MySQL APB
- MariaDB APB

The following related APIs have also been removed:

- `.automationbroker.io/v1alpha1`
- `.osb.openshift.io/v1`

Service Catalog and Template Service Broker will be removed in a future OpenShift Container Platform release, as well as the following related API:

- `.servicecatalog.k8s.io/v1beta1`

If they are enabled in 4.4, the web console warns cluster administrators that these features are still enabled. The following alerts can be viewed from the Monitoring → Alerting page and have a Warning severity:

- `ServiceCatalogAPIServerEnabled`
- `ServiceCatalogControllerManagerEnabled`
- `TemplateServiceBrokerEnabled`

The `service-catalog-controller-manager` and `service-catalog-apiserver` cluster Operators are also now set to `Upgradeable=false` in 4.4. This means that they will block future cluster upgrades to the next minor version, for example 4.5, if they are still installed at that time. Upgrading to z-stream releases such as 4.4.z, however, are still permitted in this state.

If Service Catalog is installed, cluster administrators can see Uninstalling Service Catalog to uninstall it before the next minor version of OpenShift Container Platform is released.

1.4.1.6. Deprecation of OperatorSources, CatalogSourceConfigs, and packaging format

OperatorSources and CatalogSourceConfigs are deprecated from OperatorHub. The following related APIs will be removed in a future release:

- `operatorsources.operators.coreos.com/v1`
- `catalogsourceconfigs.operators.coreos.com/v2`
- `catalogsourceconfigs.operators.coreos.com/v1`

The Operator Framework’s current packaging format, the Package Manifest Format, is also deprecated in this release, to be replaced by the new Bundle Format in a future release. As a result, the `oc adm catalog build` command, which builds catalogs in the Package Manifest Format, is also deprecated.

For more information on the upcoming Bundle Format and Operator Package Manager CLI (`opm`), see the upstream OKD documentation.

1.4.1.6.1. Converting custom OperatorSources and CatalogSourceConfigs
If there are any custom OperatorSources or CatalogSourceConfigs objects present on the cluster in OpenShift Container Platform 4.4, the marketplace cluster Operator now sets an Upgradeable=false condition and issues a Warning alert. This means that it will block future cluster upgrades to the next minor version, for example 4.5, if they are still installed at that time. Upgrades to z-stream releases such as 4.4.z, however, are still permitted in this state.

Cluster administrators can convert custom OperatorSources or CatalogSourceConfigs to using CatalogSources directly to clear this alert:

Procedure

1. Remove your custom OperatorSources or CatalogSourceConfigs objects.
   a. Search for OperatorSources or CatalogSourceConfigs objects across all namespaces:
      
      $ oc get opsrc --all-namespaces
      $ oc get csc --all-namespaces
   
   b. Remove all custom objects from all relevant namespaces:
      
      $ oc delete opsrc <custom_opsrc_name> -n <namespace>
      $ oc delete csc <custom_csc_name> -n <namespace>

   IMPORTANT

   Do not remove the default OperatorSources in the openshift-marketplace namespace: redhat-operators, community-operators, certified-operators, and redhat-marketplace. They are bootstrapped if accidentally removed, however.

2. Use the procedure as described in Building an Operator catalog image from the restricted network documentation to create and push a new catalog image, making the following changes at the oc adm catalog build command step:

   • Change --appregistry-org to your namespace on the App Registry instance, for example on Quay.io.

   • Change --to to your image repository tag, which is applied to the built catalog image and pushed.

   For example:

   $ oc adm catalog build \
     --appregistry-org <namespace> \ 
     --from=registry.redhat.io/openshift4/ose-operator-registry:v4.4 \ 
     --to=quay.io/<namespace>/<catalog_name>:<tag> \ 
     [-a ${REG_CREDS}]

   NOTE

   The oc adm catalog build command is deprecated; however, deprecated features are still supported.
3. Apply a CatalogSource to your cluster that references your new catalog image:

```bash
cat <<EOF | oc apply -f -
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: my-operator-catalog
  namespace: openshift-marketplace
spec:
  sourceType: grpc
  image: quay.io/<namespace>/<catalog_name>:<tag>
  displayName: My Operator Catalog
  updateStrategy:
    registryPoll:
      interval: 30m
EOF
```

1. Specify your custom Operator catalog image.
2. CatalogSources can now automatically check for new versions to keep up to date.

1.4.2. Removed features

1.4.2.1. OpenShift CLI secrets subcommands

The following `oc secrets` subcommands that were deprecated in OpenShift Container Platform 3.9 are no longer available:

- `new`
- `new-basicauth`
- `new-dockercfg`
- `new-sshauth`

You must use the `oc create secret` command instead.

1.4.2.2. OpenShift CLI build-logs command

The `oc build-logs` command was deprecated in OpenShift Container Platform 3.11 and has been removed. You must use `oc logs` instead.

1.4.2.3. Deprecated upstream Kubernetes metrics have been removed

All deprecated upstream Kubernetes metrics have been removed. The complete list of removed metrics is next.

Kubelet metrics

- `kubelet_pod_worker_latency_microseconds`
- `kubelet_pod_start_latency_microseconds`
- kubelet_cgroup_manager_latency_microseconds
- kubelet_pod_worker_start_latency_microseconds
- kubelet_pleg_relist_latency_microseconds
- kubelet_pleg_relist_interval_microseconds
- kubelet_runtime_operations
- kubelet_runtime_operations_latency_microseconds
- kubelet_runtime_operations_errors
- kubelet_eviction_stats_age_microseconds
- kubelet_device_plugin_registration_count
- kubelet_device_plugin_alloc_latency_microseconds
- kubelet_network_plugin_operations_latency_microseconds

Scheduler metrics

- scheduler_e2e_scheduling_latency_microseconds
- scheduler_scheduling_algorithm_predicate_evaluation
- scheduler_scheduling_algorithm_priority_evaluation
- scheduler_scheduling_algorithm_preemption_evaluation
- scheduler_scheduling_algorithm_latency_microseconds
- scheduler_binding_latency_microseconds
- scheduler_scheduling_latency_seconds

API server metrics

- apiserver_request_count
- apiserver_request_latencies
- apiserver_request_latencies_summary
- apiserver_dropped_requests
- apiserver_storage_data_key_generation_latencies_microseconds
- apiserver_storage_transformation_failures_total
- apiserver_storage_transformation_latencies_microseconds
- apiserver_proxy_tunnel_sync_latency_secs

Docker metrics

- kubelet_docker_operations
- `kubelet_docker_operations_latency_microseconds`
- `kubelet_docker_operations_errors`
- `kubelet_docker_operations_timeout`

Reflector metrics
- `reflector_items_per_list`
- `reflector_items_per_watch`
- `reflector_list_duration_seconds`
- `reflector_lists_total`
- `reflector_short_watches_total`
- `reflector_watch_duration_seconds`
- `reflector_watches_total`

etcd metrics
- `etcd_helper_cache_hit_count`
- `etcd_helper_cache_miss_count`
- `etcd_helper_cache_entry_count`
- `etcd_request_cache_get_latencies_summary`
- `etcd_request_cache_add_latencies_summary`
- `etcd_request_latencies_summary`

Transformation metrics
- `transformation_latencies_microseconds`
- `transformation_failures_total`

Other metrics
- `admission_quota_controller_adds`
- `crd_autoregistration_controller_work_duration`
- `APIServiceOpenAPIAggregationControllerQueue1_adds`
- `AvailableConditionController_retries`
- `crd_openapi_controllerunfinished_work_seconds`
- `APIServiceRegistrationController_retries`
- `admission_quota_controller_longest_running_processor_microseconds`
- `crdEstablishing_longest_running_processor_microseconds`
- crdEstablishing_unfinished_work_seconds
- crd_openapi_controller_adds
- crd_autoregistration_controller_retries
- crd_finalizer_queue_latency
- AvailableConditionController_work_duration
- non_structural_schema_condition_controller_depth
- crd_autoregistration_controller_unfinished_work_seconds
- AvailableConditionController_adds
- DiscoveryController_longest_running_processor_microseconds
- autoregister_queue_latency
- crd_autoregistration_controller_adds
- non_structural_schema_condition_controller_work_duration
- APIServiceRegistrationController_adds
- crd_finalizer_work_duration
- crd_naming_condition_controller_unfinished_work_seconds
- crd_openapi_controller_longest_running_processor_microseconds
- DiscoveryController_adds
- crd_autoregistration_controller_longest_running_processor_microseconds
- autoregister_unfinished_work_seconds
- crd_naming_condition_controller_queue_latency
- crd_naming_condition_controller_retries
- non_structural_schema_condition_controller_queue_latency
- crd_naming_condition_controller_depth
- AvailableConditionController_longest_running_processor_microseconds
- crdEstablishing_depth
- crd_finalizer_longest_running_processor_microseconds
- crd_naming_condition_controller_adds
- APIServiceOpenAPIAggregationControllerQueue1_longest_running_processor_microseconds
- DiscoveryController_queue_latency
- DiscoveryController_unfinished_work_seconds
- crd_openapi_controller_depth
- APIServiceOpenAPIAggregationControllerQueue1_queue_latency
- APIServiceOpenAPIAggregationControllerQueue1_unfinished_work_seconds
- DiscoveryController_work_duration
- autoregister_adds
- crd_autoregistration_controller_queue_latency
- crd_finalizer_retries
- AvailableConditionController_unfinished_work_seconds
- autoregister_longest_running_processor_microseconds
- non_structural_schema_condition_controller_unfinished_work_seconds
- APIServiceOpenAPIAggregationControllerQueue1_depth
- AvailableConditionController_depth
- DiscoveryController_retries
- admission_quota_controller_depth
- crdEstablishing_adds
- APIServiceOpenAPIAggregationControllerQueue1_retries
- crdEstablishing_queue_latency
- non_structural_schema_condition_controller_longest_running_processor_microseconds
- autoregister_work_duration
- crd_openapi_controller_retries
- APIServiceRegistrationController_work_duration
- crdEstablishing_work_duration
- crd_finalizer_adds
- crd_finalizer_depth
- crd_openapi_controller_queue_latency
- APIServiceOpenAPIAggregationControllerQueue1_work_duration
- APIServiceRegistrationController_queue_latency
- crd_autoregistration_controller_depth
- AvailableConditionController_queue_latency
- admission_quota_controller_queue_latency
- crd_naming_condition_controller_work_duration
- crd_openapi_controller_work_duration
- DiscoveryController_depth
- crd_naming_condition_controller_longest_running_processor_microseconds
- APIServiceRegistrationController_depth
- APIServiceRegistrationController_longest_running_processor_microseconds
- crd_finalizer_unfinished_work_seconds
- crdEstablishing_retries
- admission_quota_controller_unfinished_work_seconds
- non_structural_schema_condition_controller_adds
- APIServiceRegistrationController_unfinished_work_seconds
- admission_quota_controller_work_duration
- autoregister_depth
- autoregister_retries
- kubeproxy_sync_proxy_rules_latency_microseconds
- rest_client_request_latency_seconds
- non_structural_schema_condition_controller_retries

### 1.4.2.4. High granularity request duration buckets in Prometheus

High granularity request duration buckets were dropped in Prometheus, which were tracked with the `apiserver_request_duration_seconds_bucket` metric. This leaves enough buckets for meaningful alerting from other monitoring components, but drastically reduces cardinality.

### 1.5. BUG FIXES

**apiserver-auth**

- Previously, if a user attempted to log in from the CLI when only browser-based login was configured, they were prompted for a username and password. Now, if a user attempts to log in from the CLI when only browser-based login is configured, a message is shown that instructs users how to retrieve the login token. ([BZ#1671604](http://bz.redhat.com/bz_bug/1671604))

- Previously, due to a race condition, it went unnoticed when the mounted serving certificate changed or appeared, so the serving certificate was not trusted by metrics scrapers on the HTTPS endpoint. The race condition was removed and Operators based on library-go are now able to reload the serving certificate correctly. ([BZ#1779438](http://bz.redhat.com/bz_bug/1779438))
- Previously, the Kubernetes API server service network address was not handled properly if an IPv6 address was used. The OAuth proxy can now properly connect to the Kubernetes API server if it serves on an IPv6 address. (BZ#1789462)

**Build**

- Before starting a build, the OpenShift builder would parse the supplied Dockerfile and reconstruct a modified version of it to use for the build, to add labels and handle substitutions of the images named in `FROM` instructions. The generated Dockerfile did not always correctly reconstruct `ENV` and `LABEL` instructions. The generated Dockerfile would sometimes include `=` characters where the original did not, and the build would fail with a syntax error. With this bug fix, when generating the modified Dockerfile, the original text for `ENV` and `LABEL` instructions is now used verbatim. As a result, the build process no longer introduces syntax errors in `ENV` and `LABEL` instructions. (BZ#1821860)

- The JenkinsPipeline build strategy is deprecated as of OpenShift Container Platform 4.3.0. Use Jenkinsfiles directly on Jenkins or OpenShift Pipelines instead. (BZ#1804976)

- Build label generation and validation was not fully conforming to Kubernetes expectations. Builds could fail with certain BuildConfig names with invalid label errors. This bug fix updates the build controller and build API server to now use complete Kubernetes validation routines to ensure any added build labels will meet Kubernetes label criteria. As a result, builds with any valid BuildConfig name will not fail because of invalid build label values. (BZ#1804934)

- Previously, if the Samples Operator’s `samplesRegistry` field was changed and it still led to an imagestream import error, it appeared that the Samples Operator did not take the configuration change when viewing the imagestream status. Now, if a change to the Samples Operator’s `samplesRegistry` field still leads to an imagestream import error, the new failure reasons now properly appear in the imagestream status. (BZ#1795705)

- Previously, after a `RUN` instruction, the OpenShift builder attempted to unmount each of the bind mounts that were created, and logged any errors that were encountered in the process. The builder now only unmounts the top-level directory, and has the kernel unmount the bind mounts. The errors are no longer encountered and therefore no longer reported. (BZ#1772179)

- Previously, setting both `incremental` and `forcePull` flags to `true` on a build strategy could result in builds using push image credentials to pull images. As a result, image pulls from private registries would fail. Now, the build image properly manages registry push and pull credentials when both `incremental` and `forcePull` are set to `true`. (BZ#1774492)

- The command `oc new-build` did not have the same `--insecure-registries` flag available with the `oc new-app` command to allow for the sure of insecure image reference URLs as their source. As a result, `oc new-build` invocations would receive errors when attempting to make HTTPS connections using HTTP-based image references were supplied as the base image for the build. Now, The `--insecure-registries` option has been added to the `oc new-build` command and users can now create builds that reference insecure registries as their base image. (BZ#1780714)

**Cloud Credential Operator**

- The Cloud Credential Operator (CCO) would report on CredentialsRequests with conditions even when the CCO has been disabled. Alerts would show even when the Operator has been configured to be disabled. With this bug fix, conditions are no longer reported when the CCO is set to disabled. (BZ#1794536)

- Reconciling a CredentialsRequest would attempt to create a role assignment that already exists, and Microsoft Azure logs would show `create role assignment` errors. This bug fix checks for...
existing role assignments to avoid creating one that already exists. As a result, there are less error messages in Azure logs. (BZ#1776079)

### Console Kubevirt Plugin

- When selecting a VM template without annotations, the VM wizard closed unexpectedly. The VM wizard now works with templates that do not have any annotations. (BZ#1776190)

- Previously, when creating a VM template that uses a URL as a disk image source, a persistent volume claim (PVC) was not created for VMs created when using the template. Now when creating a new VM from such a template, the PVC is cloned and used for the disk image. (BZ#1779116)

- Previously, different units were used when interpreting values for memory and storage in a template, causing requests to create a VM to fail on some occasions. The value for memory and storage in a VM template now use Gi units consistently. (BZ#1792101)

- Previously, the VM wizard ignored any disk configuration in a VM template. The VM wizard now uses the disk configuration if specified in a template. (BZ#1782434)

- The UI previously reported that failed VM migrations had succeeded. When migrating a VM, the UI now correctly reports when a VM migration fails. (BZ#1785344)

- Previously, if a VM had multiple CD-ROM drives defined, it was not possible to remove each CD-ROM drive without saving, and then reopening the dialog for each CD-ROM. Now multiple CD-ROM drives can be removed without saving and reopening the dialog. (BZ#1786070)

- Previously, it was not possible to create a VM with the default YAML used by the VM wizard because the YAML was invalid. It is now possible to use the default VM template when creating a VM with the wizard. (BZ#1808304)

- Previously, it was not possible to modify the boot order by using the visual editor because the boot order in the template YAML was unrecognized. It is now possible to modify the boot order when using the visual editor. (BZ#1789269)

### Image

- The `oc tag` command did not update ImageStreams when ImageStreamTags are inaccessible. The command would report that the new tag was created, but it was not. This bug fix updates the `oc tag` command so that it actually creates the tag even if there are no permissions for the ImageStreamTagAPI. (BZ#1746149)

- To detect whether base64 was padded or unpadded, the decoder was relying on the string length. This made the decoder unable to handle pull secrets that contain whitespaces. This bug fix checks if the string has trailing padding symbols instead. As a result, pull secrets with whitespaces can be used to pull images. (BZ#1776599)

### Image Registry

- Previously, the Image Registry Operator did not report a new version if it was in the unmanaged state, which blocked upgrades. With this bug fix the Image Registry Operator now reports the accurate version when it is the unmanaged state, which results in successful upgrades. (BZ#1791934)

- Previously, the `nodeca` daemonset did not tolerate `NoSchedule` taints, which caused missing pods on nodes. This bug fix adds toleration so tainted node receive updates from the `nodeca` daemonset. (BZ#1785115)
The Image Registry Operator was using a rolling update strategy that was not compatible with RWO volumes, which meant that RWO volumes could not be used. This bug fix allows the Image Registry Operator to pick up the rolling update strategy, and can now be deployed with RWO volumes in non-high-available configurations (i.e., configurations with only one replica). (BZ#1798759)

Previously, the Image Registry Operator did not clean up the storage status when it was removing storage. When the registry was switched back to the Managed state, it was not able to detect that storage needed to be bootstrapped. With this bug fix the Image Registry Operator cleans up the storage status, allowing the Operator to create storage when it is switched back to the Managed state. (BZ#1787488)

Installer

Earlier versions of clusters that you provisioned on AWS with installer-provisioned infrastructure do not include security group rules that allow traffic from control plane hosts to workers on TCP and UDP ports 30000-32767. Because of this, new OVN Networking components cannot work as intended. Now, the required security group rules will be added to these clusters to allow communication between control plane and worker machines on TCP and UDP ports 30000-32767, and OVN Networking components work as intended. (BZ#1763936)

Previously, the upgrade process on Red Hat Enterprise Linux (RHEL) nodes was blocked. An unnecessary machine config apply step failed when it could not pull images from behind the proxy. The unnecessary step was removed, and upgrades to RHEL nodes behind a proxy can succeed. (BZ#1786297)

Previously, when you upgraded RHEL 7 nodes from version 4.2.12, the machine config was not properly updated by the MCO. Because package installs updated files on the local disk, the MCO did not process config updates on RHEL nodes. Now, the machine config apply step has been restored, and the image pull process can complete behind a proxy. Machine configs are correctly applied after package updates, and upgrades on RHEL 7 nodes succeed. (BZ#1792139)

kube-apiserver

Although metrics for aggregated API server status existed, alerts for them did not. Now, an error displays when an aggregated API reports too many errors in a short period of time because that indicates that the availability of the services changes too often. (BZ#1772564)

kube-controller-manager

Previously, certificates were not propagated correctly, so the cloud provider was unable to initialize behind a man-in-the-middle proxy. Now, the certificates are correctly propagated to the kube-controller-manager, and the cloud provider works as expected with a man-in-the-middle proxy. (BZ#1772756)

Logging

Previously, you could use Fluentd plug-ins to forward logs to external systems using syslog protocol (RFC 3164). The Log Forwarding API added to OpenShift Container Platform 4.3 changed the process for configuring log forwarding using syslog. As a result, you could not forward logs using the same method as in OpenShift Container Platform 4.2. To address this change, a new process was devised to allow log forwarding using the syslog protocol. This change was backported to OpenShift Container Platform 4.3.7. As a result, you can continue to forward logs to an external syslog server. (BZ#1799024)

Machine Config Operator
Because some applications are sensitive to HAProxy timeout values, such as Kuryr, 24 hour timeout values were used for the API LB. If an HAProxy reload operation was triggered many times in a short period, it was possible for many HAProxy processes to accumulate. This bug fix forces sending SIGTERM after a default timeout of 120 seconds to old HAProxy processes which have not yet terminated. As a result, the number of long lived duplicate HAProxy processes is reduced. (BZ#1771566)

Metering Operator

- Metering does not manage or delete any S3 bucket data. When deleting a report, any S3 buckets used to store metering data must be manually cleaned up. If reporting data stored in an S3 bucket is not manually cleaned up and the same report is re-created, the original reporting data will still exist and will cause duplicate row entries. (BZ#1728350)

Monitoring

- Because the OAuth proxy container readiness probe was misconfigured, the container logs were flooded by error messages every 10 seconds. The readiness probe has been configured with the proper settings. As a result, the error messages are not appearing in the logs. (BZ#1658899)

- Because the cluster-reader role does not have the permission to view node or Pod metrics, users bound to that role could not access metrics using commands such as oc top node. The cluster-reader role has been updated to include permissions to allow viewing metrics. (BZ#1723662)

- A new experimental Prometheus user interface was introduced upstream. The new experimental interface that was not fully tested and not stable. As a result, switching from the experimental interface to the default interface returned an empty page. To prevent this issue, the link to the experimental UI has been hidden. As a result, it is no longer possible to access the experimental Prometheus interface. (BZ#1781415)

- OpenShift Container Platform was evaluating some recording rules incorrectly, As a result, the metrics generated from the recording rules are missing. The recording rules have been fixed. All recording rules now evaluate successfully. (BZ#1807843, BZ#1779324)

Networking

- A previous Egress IP bug fix did not fully clean up after removed Egress IPs resulting in the possibility that harmless extra iptables rules could be left behind on a node. With this bug fix, the extra rules are now removed if they are no longer being used. (BZ#1787488)

- Previously, using the httpProxy or httpsProxy hostname with uppercase letters makes the the CNO fatal, which is a violation of RFC 3986. As a result, CNO was not operational. This bug fix parses it with golangs url.ParseRequestURI, which implemented RFC 3986 correctly and a few more RFCs. As a result, capital case is now allowed in httpProxy and httpsProxy. (BZ#1802606)

- Previously, the Kubeconfig used by the kubelet, which is used on the SDN, changed its path causing SDN to have a null deference trying to parse the empty file. This bug fix made SDN able to handle both old and new paths. (BZ#1781707)

Node

- Previously, no alerts were issued when kubelet certificates expired. This caused kubelets to stop working without administrators realizing it. This fix adds a server_expiration_renew_errors metric to report expired certificates. (BZ#1767523)
- Common was timing out on kubectl exec liveness probes. This caused some exec probes to fail, forcing the container to be killed and restarted. The exec liveness probes now work as expected. (BZ#1817568)

- On node reboot, CRI-O was not cleaning up IP addresses correctly when Pods could not be restored. This led to node IP exhaustion, causing Pods to not start. Now if a Pod cannot be restored after restart, the Pod network is destroyed, releasing the IP addresses for future use. (BZ#1781824)

- RHEL 7 could not be added to an OpenShift Container Platform cluster because CRI-O would not start. This was caused by a Common package issue. This bug fixes Common and RHEL 7 can now join an OpenShift Container Platform cluster. (BZ#1809906)

- The horizontal Pod autoscaler (HPA) was not receiving metrics from exited init containers. This is fixed by submitting zero-based metrics for exited init containers, allowing the HPA to perform analysis on Pods with an init container. (BZ#1814283)

- The kubelet metrics endpoint was periodically returning a 500 status code, which prevented Prometheus from gathering metrics for the kubelet endpoint and node. The 500 code was caused by dead containers mixing into the metrics stream, causing duplicate metrics to be injected. This bug is fixed and metrics are now correctly reported from the kubelet. (BZ#1748073)

oc

- The `oc logs` command was returning errors for some resources due to OpenShift CLI's internal code not accounting for new versions of APIs. OpenShift CLI now supports all known API types and versions, allowing `oc logs` to work for all resources. (BZ#1774366)

- When running the `oc adm node-logs` command with the `--since` argument, an error occurred. This was caused by a typo in the expected timestamp format. The typo has been corrected and the `oc adm node-logs` now works with the `--since` argument. (BZ#1779563)

openshift-apiserver

- Previously, Helm 3.0.0+ did not work with OpenShift objects. This resulted in an error when trying to deploy valid Helm charts. With this update it is now possible to deploy Helm charts with OpenShift objects. (BZ#1773682)

openshift-controller-manager

- Previously, openshift-controller-manager metrics were not properly registered with the 1.16 Kubernetes Prometheus registry. This caused missing metrics for the OpenShift control plane. With this update the openshift-controller-manager metrics are now properly registered and the missing OpenShift control plane metrics have been restored. (BZ#1810304)

- Previously, pull secrets were sometimes not deleted when their associated token was deleted. This caused pull secrets to remain associated with Kubernetes service accounts. With this update references to the owner of a pull secret and the associated token secret have been established. Now, when a pull secret is deleted the associated token is deleted as well. (BZ#1806792)

- Previously, the rate limit for creating pull secrets in the internal registry was low. This caused long wait times when a large number of namespaces were created in a short time period. With this update the rate limit for creating pull secrets in the internal registry has been increased. Pull secrets can now be created quickly even during heavy load. (BZ#1819849)
RHCOS

- Network teaming is now supported in RHEL CoreOS. The teamd and NetworkManager-team RPMs have been added to RHEL CoreOS, enabling the setup and management of teamed network devices. (BZ#1758162)

Samples

- IPv6 was not supported by registry.redhat.io, which meant that the Samples Operator was not supposed to attempt to install imagestreams, as all Red Hat samples are hosted on registry.redhat.io. Because IPv6 is a key initiative for Red Hat and for OpenShift Container Platform, the Samples Operator will no longer break OpenShift Container Platform installs on IPv6. (BZ#1788676)

- The Samples Operator was previously failing to report its version when running on s390x or ppc64le, so installs on those architectures would not complete successfully. With this fix, the Samples Operator now reports its version correctly and no longer prevents installs on s390x or ppc64le. (BZ#1779933)

- Previously, the latest Java 11 imagestream tag did not correctly link to the version on the Imagestream details page, which meant that that imagestreamtag could not be inspected from the OpenShift Container Platform web console. With this fix, the correct imagestreamtag specification for Java 11 can now be properly inspected from the web console. (BZ#1778613)

- Previously, local reference settings on imagestreams could be ignored if the imagestream was accessed by controller manager shortly after startup but before the imagestream metadata was updated. This resulted in failed requests to imagestreams backed by private registries. With this update the controller manager refreshes its imagestream cache after metadata initialization is complete. This results in accurate local reference imagestream policies even immediately after startup. (BZ#1775973)

Storage

- Pods could not be scheduled and PVCs remained pending if the storage-class annotation was used instead of storeClassName for the Local Storage Operator. With this fix, the Kubernetes scheduler now checks both volume.beta.kubernetes.io/storage-class and the PVC.Spec.StorageClassName when evaluating a Pod and its PVCs. Pods that use the beta annotation to refer to a StorageClass can now be scheduled and will now run. (BZ#1791786)

- Previously, Kubernetes did not unmount a CSI volume when a Pod was deleted while the volume mount timed out before eventually being completed by the CSI driver. This caused a volume to be mounted on a node without Kubernetes knowing about it. As a result, the volume could not be mounted anywhere else. With this fix, Kubernetes waits for a final success or error after the CSI driver returns a timeout or other similar transient error. Now, Kubernetes knows if a volume was mounted or unmounted and cleans up possible stray mounts when a Pod is deleted. (BZ#1745776)

Templates

- When using the --param-file option with template processing commands like oc new-app or oc process, if the file was greater than 64K in size, it would not be fully read in. This caused the the oc-based template processing with --param-file to fail. OpenShift Container Platform now checks the size of the file specified by --process-file and augments the parameters used to read the entire file. The oc-based template processing with --param-file pointing to files greater than 64K now works. (BZ#1748061)

- Previously, one of the new-app/new-build examples used for project creation would cause an
error in a FIPS environment because the example was not FIPS-compliant. Now, only FIPS-compliant new-app/new-build examples are shown on new project creation and users can use any of the examples in a FIPS environment. (BZ#1774318)

Web console (Administrator perspective)

- Previously, the Rebuild action on the OpenShift Console build page was incorrectly disabled for users who were did not have cluster-admin privileges. Now, this issue has been resolved and the action is now correctly enabled for normal users who should have permission to clone builds. (BZ#1774842)

- Previously, only cluster administrators could see example YAML templates created using the ConsoleYAMLSample resource in console YAML editor. Now, all users can see these templates. (BZ#1783163)

- On the CronJobs list page, sorting by Starting Deadlines Seconds or Concurrency Policy fields were not sorted correctly. Now, the sortField has been updated and you can now sort CronJobs correctly. (BZ#1787096)

- The Local Volumes page would display duplicate content due to conditions that created content duplication. Now, the conditions are removed from the status descriptors and content is not duplicated. (BZ#1776131)

- The Role Bindings page had a non-clickable Create Binding button for users who did not have any projects. Now, the Create Binding button is hidden for users who do not have any projects. (BZ#1785487)

- Previously, some required fields that have OLM descriptors were missing the required red asterisk in the OpenShift Container Platform console Create Operand form. Now all of the required fields are correctly labeled. (BZ#1779858)

- Previously, you could set a negative value for the machine or replica count values in the OpenShift Container Platform console. Now, you cannot set a value that is less than 0. (BZ#1780367)

- Previously, the PodRing GUI component did not reflect the correct Pod count when the count was updated on the Deployment Config Page and then updated again on the same page by clicking Actions then Edit Count. For example, if the Pod count was increased from 5 to 10 Pods by using Edit Count on the Deployment Config Page and the user then increased the count on the PodRing component by using the “up” arrow, the PodRing counter incorrectly increased from 5 to 6 Pods instead of from 10 to 11. With this update, when a change is made using Edit Count on the Deployment Config Page, the PodRing component now shows the correct, updated Pod count. Additionally, when the user clicks the up or down arrow on the PodRing component, the Pod count accurately updates. (BZ#1787210)

- Previously, a user could not edit a cluster-scoped operand on the Installed Operators page in the web console. Instead, an HTTP 404 Not Found client error response code occurred in the web browser for the operand YAML editor. With this update, the web console correctly opens a new web browser for the operand YAML editor, in which a user can update a cluster-scoped operand. (BZ#1781246)

- Previously, the OpenShift web console did not replace all instances of a variable when it occurred multiple times in a ConsoleExternalLogLink URL template. Instead, only the first expression of the variable was replaced. With this update, the console correctly replaces every instance of the variable in the template with the correct value. (BZ#1781827)

- Previously on the Operator Details page in the web console, the link to the InstallPlan resource
from the Subscription Overview was broken. This made it difficult for users to approve InstallPlans in the web console. The link to approve InstallPlans (for example, a link showing output that **1 requires approval**) now works as expected. (BZ#1783651)

- Previously, an error occurred when a user filtered by source name in the Source tab on the OperatorHub Details page in the web console. The filter has been fixed and now works as expected when a user inputs a source name. (BZ#1786418)

- Previously, API documentation was missing for the Endpoints resource on the Explore page in the web console. API documentation, such as descriptions and schema information, is now available for the Endpoints resource. (BZ#1794754)

- Previously, the web console failed to show an operand if an invalid OLM descriptor was set by an Operator. As a result, an error occurred on the expected web console page. With this update, invalid OLM descriptors are tolerated and the console correctly displays the operand details. (BZ#1797727)

- Previously, some status values did not have an icon associated with them. As a result, some values appeared with an icon and others did not. Icons are now defined for and will appear with all values. (BZ#1780629)

- Previously, the console did not check for special characters in routing labels, which could result in the following error:

  AlertmanagerFailedReload Alert: Reloading Alertmanager's configuration has failed for openshift-monitoring/alertmanager-main-x.

The Create Receiver form now restricts label names to valid characters only. (BZ#1784725)

- Previously, the OpenShift console would show a blank page if an Operator declared an invalid K8sResourceLink OLM descriptor. The console now tolerates incorrect K8sResourceLink descriptors, which prevents the appearance of blank pages. (BZ#1795407)

- Previously, modifying required Operator resources from the web UI would not update those Operators' YAML files. The YAML files now update as expected. (BZ#1797769)

- Previously, alerts remained in the notifications drawer after they were silenced. Silenced alerts no longer remain in the notifications drawer. (BZ#1808062)

- A bug in OpenShift console builds occasionally resulted in runtime errors on certain pages. The bug was fixed. (BZ#1818978)

- The query browser results in the web console are rendered with a hard-coded sort, rendering a different result than might be intended with a custom sort. The hard-coded sort has been removed, allowing for the query results to reflect any custom sort. (BZ#1808437)

- Creating a new MachineConfigPool using the console’s **Compute → Machine Config Pools → Create Machine Config Pool** button results in a MachineConfigPool that does not match the node. This is caused by the template using the `spec.machineSelector` key for selecting the nodes to match. However, this key is not recognized by the API; the correct one for selecting a node is `spec.nodeSelector`. The key for selecting nodes has been updated, allowing the GUI to display a Machine Selector which now matches the appropriate node. (BZ#1818944)

- Previously, the OpenShift console Pod terminal did not correctly handle Unicode characters. This problem has been fixed, and Unicode characters now display correctly. (BZ#1821285)
• Previously, the volumes table on the OpenShift console workload pages could partially disappear when scrolling the page. This bug has been fixed. (BZ#1822195)

• Previously, the default templates used to create reports and report queries were using apiVersion v1alpha instead of v1. Although the alpha versioned templates still worked, their case support could be dropped at any time. The templates have been updated to use apiVersion: metering.openshift.io/v1. (BZ#1772694)

• When clicking the web console’s Dashboard and selecting a navigation item within the Dashboard, both navigation items that were selected remain highlighted. This bug fix applies new CSS rules to eliminate multiple navigation items from being highlighted at the same time, ensuring only the active navigation item is highlighted. (BZ#1774702)

Web console (Developer perspective)

• The Microsoft Edge browser did not recognize the functionality used for scrolling. The Log screen was unable to load and would result in an error. Screen reader support was enabled and the logs now render. (BZ#1777980)

• Serverless resources like the Service YAML file are listed as v1beta1 instead of v1. However, v1beta1 is deprecated. With this bug fix, the apiVersion is updated to v1. (BZ#1796421)

• A service binding request (SBR) in the topology is associated with Revisions in the topology view. Therefore, no new revision will get the associated secrets. The SBR should go through the knative service; secrets would be injected and new revisions would be created. (BZ#1798288)

• The API group of KnativeServing resources serving.knative.dev is deprecated and it has changed to operator.knative.dev in the Serverless Operator version 1.4. In the next release of the Serverless Operator, serving.knative.dev will be obsolete. (BZ#1800598)

• In container image deployment, if an internal image stream is selected for knative, then knative tries to create a new image stream, which might fail. You are unable to deploy the internal image stream as a knative service. Do not create a new image stream for internal image selection; it already exists. (BZ#1808280)

• In a Kubernetes deployment, if the images from the external image registry had tags such as openshift/hello-world:1.0, then the tags were not being applied. The user was unable to import external images with tags. With this big fix, the proper tag for the deployment is now passed. (BZ#1801736)

• Previously, when two consecutive rollouts failed, the Topology view showed failed pods instead of displaying the last active revision. With this bug fix, when a rollout fails, the last active revision is displayed. (BZ#1760828)

• Previously, existing imagestreams in a namespace were not detected when creating an application. This occurred when users with limited cluster-wide permissions used the Container Image → Image name from the internal registry* options in the Add page. Now, the imagestream fetching logic has been moved from the cluster level to the namespace level, which enables a user with permission to a namespace to see imagestreams in that namespace. (BZ#1784264)

• Knative service and revision resources did not have binding or visual connectors in the Topology view and therefore Knative workloads could not connect to other workloads. These resources now have connectors in the Topology view and can connect to other workloads. (BZ#1779201)

• When the Eclipse Che Operator was installed and configured, the Topology view showed the Git icon instead of the Che icon. This provided no indication to the user that they could click the
icon to access the Che workspace. The Topology view now correctly displays the Che icon when Che is configured, making it easier for the user to access the Che workspace. (BZ#1780338)

- Creating a Knative service using the CLI while the Topology view was open caused a GUI error. Checks have been added to handle this workload without causing a GUI error. (BZ#1781188)

- The internal registry feature had an unclear error message when there was a server-side error. Improved error messaging helps the user identify the cause of the problem. (BZ#1787492)

### 1.6. 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

#### Table 1.2. Technology Preview tracker

<table>
<thead>
<tr>
<th>Feature</th>
<th>OCP 4.2</th>
<th>OCP 4.3</th>
<th>OCP 4.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prometheus Cluster Monitoring</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Precision Time Protocol (PTP)</td>
<td>-</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>CRI-O for runtime Pods</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td><strong>oc</strong> CLI Plug-ins</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Network Policy</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Multus</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>New Add Project Flow</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Search Catalog</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Cron Jobs</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Kubernetes Deployments</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>StatefulSets</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Feature</td>
<td>OCP 4.2</td>
<td>OCP 4.3</td>
<td>OCP 4.4</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>Explicit Quota</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Mount Options</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>experimental-qos-reserved</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Pod sysctls</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Static IPs for External Project Traffic</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Template Completion Detection</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td><strong>replicaSet</strong></td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>ImageStreams with Kubernetes Resources</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Device Manager</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Persistent Volume Resize</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Huge Pages</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>CPU Pinning</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Admission Webhooks</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>External provisioner for AWS EFS</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Pod Unidler</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Node Problem Detector</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Ephemeral Storage Limit/Requests</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Descheduler</td>
<td>-</td>
<td>-</td>
<td>TP</td>
</tr>
<tr>
<td>Podman</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Kuryr CNI Plug-in</td>
<td>TP</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Sharing Control of the PID Namespace</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Cluster Administrator console</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Cluster Autoscaling</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Feature</td>
<td>OCP 4.2</td>
<td>OCP 4.3</td>
<td>OCP 4.4</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>Container Storage Interface (CSI)</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Operator Lifecycle Manager</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Red Hat OpenShift Service Mesh</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>&quot;Fully Automatic&quot; Egress IPs</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Pod Priority and Preemption</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Multi-stage builds in Dockerfiles</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>OVN-Kubernetes Pod network provider</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>HPA custom metrics adapter based on Prometheus</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Machine health checks</td>
<td>TP</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Persistent Storage with iSCSI</td>
<td>TP</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Raw Block with iSCSI</td>
<td>TP</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Raw Block with Cinder</td>
<td>-</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>OperatorHub</td>
<td>GA</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Three-node bare metal deployments</td>
<td>TP</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>SR-IOV Network Operator</td>
<td>TP</td>
<td>GA</td>
<td>GA</td>
</tr>
<tr>
<td>Helm CLI</td>
<td>-</td>
<td>TP</td>
<td>GA</td>
</tr>
<tr>
<td>Service Binding</td>
<td>-</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>Log forwarding</td>
<td>-</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>User workload monitoring</td>
<td>-</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>OpenShift Serverless</td>
<td>TP</td>
<td>TP</td>
<td>GA</td>
</tr>
<tr>
<td>Compute Node Topology Manager</td>
<td>-</td>
<td>TP</td>
<td>TP</td>
</tr>
<tr>
<td>CSI volume snapshots</td>
<td>-</td>
<td>-</td>
<td>TP</td>
</tr>
<tr>
<td>CSI volume cloning</td>
<td>-</td>
<td>-</td>
<td>TP</td>
</tr>
</tbody>
</table>
1.7. KNOWN ISSUES

- There is an issue with the Machine Config Operator (MCO) supporting Day 2 proxy support, which describes when an existing non-proxied cluster is reconfigured to use a proxy. The MCO should apply newly configured proxy CA certificates in a ConfigMap to the RHCOS trust bundle; this is not working. As a workaround, you must manually add the proxy CA certificate to your trust bundle and then update the trust bundle:

$$\text{cp /opt/registry/certs/<my_root_ca>.crt /etc/pki/ca-trust/source/anchors/}$$
$$\text{update-ca-trust extract}$$
$$\text{oc adm drain <node>}$$
$$\text{systemctl reboot}$$

(BZ#1784201)

- When using a self-signed Red Hat OpenStack Platform (RHOSP) 16 cluster, you cannot pull from or push to an internal image registry. As a workaround, you must set `spec.disableRedirects = true` in the `configs.imageregistry/cluster` resource. This allows the client to pull the image layers from the image registry rather than from links directly from Swift. (BZ#1810461)

- The cluster proxy configuration `HTTP_PROXY` is only available for OpenShift Container Platform components, not user applications. As a workaround, you must run the following command to enable cluster proxy configuration for user applications:

$$\text{oc set env dc/jenkins}$$
$$\text{  \$http_proxy=$(oc get proxy cluster -o jsonpath={.status.httpProxy})}$$
$$\text{  \$https_proxy=$(oc get proxy cluster -o jsonpath={.status.httpsProxy})}$$
$$\text{  \$no_proxy=$(oc get proxy cluster -o jsonpath={.status.noProxy})}$$

(BZ#1780125)

- All `git clone` operations that go through an HTTPS proxy fail. HTTP proxies can be used successfully. (BZ#1750650)

- All `git clone` operations fail in builds running behind a proxy if the source URIs use the `git://` or `ssh://` scheme. (BZ#1751738)

- When using a mirror to build images, the build fails when the pull secret for the mirror registry only links to the builder service account. The pull secret must also link to the build config object. (BZ#1810904)

- In Red Hat OpenStack Platform (RHOSP) 13 with Kuryr, if FIPS is disabled, you cannot enable Service Catalog. Pods for Service Catalog’s controller manager and API server components show a status of `CrashLoopBackOff`. This is due to the `https://etcd.openshift-etcd.svc.cluster.local:2379` URL not always resolving. There is a new technique for getting the etcd cluster URL in OpenShift Container Platform 4. (BZ#1821589)

- Installing RHOSP 16 with Kuryr does not work due to the `ovn_controller` crashing after initial setup. (BZ#1812009, BZ#1818844)

- The Red Hat Virtualization (RHV) machine `instance-state` annotation and the `providerStatus.instanceState` status do not always match. This mismatch causes the client to fail or incorrectly patch the RHV machine status. (BZ#1815394)
When scaling up a MachineSet on RHV, the new machine cannot exit the **Provisioned** phase. This causes the machine to never run. (BZ#1815435, BZ#1817853)

- OpenShift Container Platform cluster autoscaling on RHV fails due to cluster resource computation errors. (BZ#1822118)

- When using the Firefox browser to select a node or a group of nodes in the **Topology** view, the backgrounds of all the associated labels and nodes become transparent. (BZ#1822337)

- In the **Topology view**, when a user selects a node or workload, and then clicks **Monitoring → View monitoring dashboard** on the side panel, the user sees the monitoring dashboard for that specific workload. This filtered workload dashboard view is not clearly named, which causes confusion with the generic dashboard that displays metrics for all the workloads. (BZ#1822331)

- When invalid characters such as a period (.) are entered in the serverless traffic distribution tag from the **Developer** perspective, the traffic distribution feature stops working. However, it displays no error messages to prevent invalid characters from being entered in the tag. (BZ#1822344)

- If an identity provider (IDP) takes longer than 60 seconds to authenticate a user, the authentication might fail before trying other IDPs. As a workaround, you can remove the faulty IDP from the list of IDPs so that users can use a successful IDP to authenticate. (BZ#1826484)

- When updating cluster logging from version 4.3 to 4.4, the Elasticsearch Pods might get stuck in the **CrashLoopBackOff** status. You can work around this issue by deleting the Elasticsearch deployments in sequence. (BZ#1824006)

- OpenShift Container Platform 4.4 is not shipping with a v.4.4 Metering Operator. Customers can install or continue to run the v.4.3 Metering Operator on OpenShift Container Platform 4.4 clusters. (BZ#1829035)

- When updating an OpenShift Container Platform cluster from version 4.3 to 4.4, the etcd Operator sometimes fails to upgrade because it is in a degraded state. This is caused by an InstallerPod failure. As a workaround, you must force a new revision on etcd to overcome the InstallerPod failure, which allows the etcd Operator to recover:

  1. Force a new revision on etcd:

     ```bash
     $ oc patch etcd cluster -p="{"spec":{"forceRedeploymentReason":"recovery-"$( date --rfc-3339=ns )"}"} --type=merge
     ```

  2. Verify the nodes are at the latest revision:

     ```bash
     $ oc get etcd -o=jsonpath={range .items[0].status.conditions[?(@.type=="NodeInstallerProgressing")].reason="\n"}.message="\n"}"
     ```

     (BZ#1830789)

- The web console downloads deployment will fail on nodes configured with `ipv6.disable=1`. (BZ1795325)

### 1.8. ASYNCHRONOUS ERRATA UPDATES

Security, bug fix, and enhancement updates for OpenShift Container Platform 4.4 are released as asynchronous errata through the Red Hat Network. All OpenShift Container Platform 4.4 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 4.4. Versioned asynchronous releases, for example with the form OpenShift Container Platform 4.4.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.


Issued: 2020-05-04

OpenShift Container Platform release 4.4 is now available. The list of container images and bug fixes includes in the update are documented in the RHBA-2020:0581 advisory. The RPM packages included in the update are provided by the RHBA-2020:0582 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 4.4.3 container image list

1.8.2. RHSA-2020:1936 - Moderate: OpenShift Container Platform 4.4 Security Update

Issued: 2020-05-04

An update for haproxy is now available for OpenShift Container Platform 4.4. Details of the update are documented in the RHSA-2020:1936 advisory.

1.8.3. RHSA-2020:1937 - Moderate: OpenShift Container Platform 4.4 Security Update

Issued: 2020-05-04

An update for cri-o is now available for OpenShift Container Platform 4.4. Details of the update are documented in the RHSA-2020:1937 advisory.
1.8.4. RHSA-2020:1938 - Moderate: OpenShift Container Platform 4.4 Security Update
Issued: 2020-05-04

An update for hadoop-container is now available for OpenShift Container Platform 4.4. Details of the update are documented in the RHSA-2020:1938 advisory.

1.8.5. RHSA-2020:1939 - Moderate: OpenShift Container Platform 4.4 Security Update
Issued: 2020-05-04

An update for ose-machine-config-operator-container is now available for OpenShift Container Platform 4.4. Details of the update are documented in the RHSA-2020:1939 advisory.

1.8.6. RHSA-2020:1940 - Moderate: OpenShift Container Platform 4.4 Security Update
Issued: 2020-05-04

An update for ose-cluster-policy-controller-container is now available for OpenShift Container Platform 4.4. Details of the update are documented in the RHSA-2020:1940 advisory.

1.8.7. RHSA-2020:1942 - Moderate: OpenShift Container Platform 4.4 Security Update
Issued: 2020-05-04

An update for presto-container is now available for OpenShift Container Platform 4.4. Details of the update are documented in the RHSA-2020:1942 advisory.

1.8.8. RHBA-2020:2133 - OpenShift Container Platform 4.4.4 Bug Fix Update
Issued: 2020-05-18

OpenShift Container Platform release 4.4.4 is now available. The list of container images and bug fixes includes in the update are documented in the RHBA-2020:2133 advisory. The RPM packages included in the update are provided by the RHBA-2020:2132 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 4.4.4 container image list

1.8.8.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.4 cluster to this latest release, see Updating a cluster by using the web console for instructions.

1.8.9. RHSA-2020:2136 - Important: OpenShift Container Platform 4.4 Security Update
Issued: 2020-05-18
An update for cluster-image-registry-operator is now available for OpenShift Container Platform 4.4. Details of the update are documented in the RHSA-2020:2136 advisory.
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 4.4 cluster, all masters must be 4.4 and all nodes must be 4.4. If you installed an earlier version of oc, you cannot use it to complete all of the commands in OpenShift Container Platform 4.4. 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

<table>
<thead>
<tr>
<th></th>
<th>X.Y (oc Client)</th>
<th>X.Y+N[a] (oc Client)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X.Y (Server)</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>X.Y+N[a] (Server)</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

[a] Where N is a number greater than 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.