OpenShift Container Platform 4.5

Release notes

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

Last Updated: 2020-09-24

OpenShift Container Platform 4.5 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 4.5 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

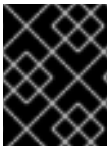
Red Hat OpenShift Container Platform (RHBA-2020:2409) is now available. This release uses Kubernetes 1.18 with CRI-O runtime. New features, changes, and known issues that pertain to OpenShift Container Platform 4.5 are included in this topic.

Red Hat did not publicly release OpenShift Container Platform 4.5.0 as the GA version and, instead, is releasing OpenShift Container Platform 4.5.1 as the GA version.

OpenShift Container Platform 4.5 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 4.5 is supported on RHEL 7, version 7.7 or later, as well as Red Hat Enterprise Linux CoreOS (RHCOS) 4.5.

You must use RHCOS for the control plane, which are also known as master machines, and can use either RHCOS or RHEL 7, version 7.7 or later, for compute machines, which are also known as worker machines.



IMPORTANT

Because only RHEL 7, version 7.7 or later, is supported for compute machines, you must not upgrade the RHEL compute machines to version 8.

With the release of OpenShift Container Platform 4.5, version 4.2 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 vSphere using installer-provisioned infrastructure

OpenShift Container Platform 4.5 introduces support for installing a cluster on vSphere using installer-provisioned infrastructure.

1.2.1.2. Installing a cluster on GCP using user-provisioned infrastructure and a shared VPC

OpenShift Container Platform 4.5 introduces support for installing a cluster on Google Cloud Platform (GCP) using user-provisioned infrastructure and a shared VPC.

1.2.1.3. Three-node bare metal deployments

You can install and run three-node clusters in OpenShift Container Platform with no workers. This provides smaller, more resource efficient clusters for deployment, development, and testing.

For more information, see Running a three-node cluster.

1.2.1.4. Restricted network cluster upgrade improvements

The Cluster Version Operator (CVO) can now verify the release images if the image signature is available as a ConfigMap in the cluster during the upgrade process for a restricted network cluster. This removes the need for using the **--force** flag during upgrades in a restricted network environment.

This improved upgrade workflow is completed by running the enhanced **oc adm release mirror** command. The following actions are performed:

- Pulls the image signature from the release during the mirroring process.
- Applies the signature ConfigMap directly to the connected cluster.

1.2.1.5. Migrating Azure private DNS zones

There is now a new **openshift-install migrate** command available for migrating Azure private DNS zones. If you installed an OpenShift Container Platform version 4.2 or 4.3 cluster on Azure that uses installer-provisioned infrastructure, your cluster might use a legacy private DNS zone. If it does, you must migrate it to the new type of private DNS zone.

1.2.1.6. Built-in help for install-config.yaml supported fields

There is a new **openshift-install explain** command available that lists all the fields for supported **install-config.yaml** file versions including a short description explaining each resource. It also provides details on which fields are mandatory and specifies their default value. Using the **explain** command reduces the need to continually look up configuration options when creating or customizing the **install-config.yaml** file.

1.2.1.7. Encrypt EBS instance volumes with a KMS key

You can now define a KMS key to encrypt EBS instance volumes. This is useful if you have explicit compliance and security guidelines when deploying to AWS. The KMS key can be configured in the **install-config.yaml** file by setting the optional **kmsKeyARN** field. For example:

```
apiVersion: v1
baseDomain: example.com
compute:
- architecture: amd64
  hyperthreading: Enabled
  name: worker
  platform:
    aws:
      rootVolume:
```

```
        kmsKeyARN: arn:aws:kms:us-east-2:563456982459:key/4f5265b4-16f7-xxxx-xxxx-
xxxxxxxxxxxx
...
```

If no key is specified, the account's default KMS key for that particular region is used.

1.2.1.8. Install to pre-existing VPC with multiple CIDRs on AWS

You can now install OpenShift Container Platform to a VPC with more than one CIDR on AWS. This lets you select secondary CIDRs for the machine network. When the VPC is provisioned by the installer, it does not create multiple CIDRs or configure the routing between subnets. Installing to a pre-existing VPC with multiple CIDRs is supported for both user-provisioned and installer-provisioned infrastructure installation workflows.

1.2.1.9. Adding custom domain names to AWS Virtual Private Cloud (VPC) DHCP option sets

Custom domain names can now be added to AWS Virtual Private Cloud (VPC) DHCP option sets. This enables Certificate Signing Request (CSR) approval of new nodes when custom DHCP options are used.

1.2.1.10. Provisioning bare metal hosts using IPv6 with Ironic

Binaries required for IPv6 provisioning using the UEFI networking stack have now been introduced in Ironic. You can now provision bare metal hosts using IPv6 with Ironic. The **snpnoly.efi** bootloader executable and compatible iPXE binaries are now included in the **tftpboot** directory.

1.2.1.11. Custom networks and subnets for clusters on RHOSP

OpenShift Container Platform 4.5 introduces support for installing clusters on Red Hat OpenStack Platform (RHOSP) that rely on preexisting networks and subnets.

1.2.1.12. Additional networks for clusters on RHOSP

OpenShift Container Platform 4.5 introduces support for multiple networks in clusters that run on RHOSP. You can specify these networks for both control plane and compute machines during installation.

1.2.1.13. Improved RHOSP load balancer upgrade experience for clusters that use Kuryr

Clusters that use Kuryr now have improved support for Octavia load-balancing services on RHOSP clusters that were upgraded from 13 to 16. For example, these clusters now support the Octavia OVN provider driver.

For more information, see The Octavia OVN driver.

1.2.1.14. Multiple version schemes accepted when installing RPM packages

When installing RPM packages, OpenShift Container Platform now accepts both three-part and two-part version schemes. A three-part version scheme follows the **x.y.z** format whereas a two-part version scheme follows the **x.y** format. Packages that use either scheme can be installed. See BZ#1826213 for more information.

1.2.1.15. SSH configuration no longer required for debug information

Gathering debug information from the bootstrap host no longer requires SSH configuration. See BZ#1811453 for more information.

1.2.1.16. Master nodes can be named any valid hostname

Master nodes can now be named as any valid hostname. See BZ#1804944 for more information.

1.2.1.17. Octavia OVN provider driver supported on previous RHOSP versions

OpenShift Container Platform clusters that were deployed before RHOSP supported the Octavia OVN provider driver can now use the driver. See BZ#1847181 for more information.

1.2.1.18. Octavia OVN provider driver supports listeners on same port

The **ovn-octavia** driver now supports listeners on the same port for different protocols. This was not supported previously on the **ovn-octavia** driver, but now it is supported and there is no need to block it. This means that it is possible to have, for instance, the DNS service expose port 53 in both TCP and UDP protocols when using **ovn-octavia**. See BZ#1846452 for more information.

1.2.2. Security

1.2.2.1. Using the oauth-proxy imagestream in restricted network installations

The **oauth-proxy** image can now be consumed by external components in restricted network installations by using the **oauth-proxy** imagestream.

1.2.3. Images

1.2.3.1. Mirroring release images to and from files

You can now mirror release images from a registry to a file and from a file to a registry.

1.2.3.2. Mirroring release image signatures

The **oc adm release mirror** command was extended to also create and apply ConfigMap manifests that contain the release image signature, which the Cluster Version Operator can use to verify the mirrored release.

1.2.4. Machine API

1.2.4.1. AWS MachineSets support spot instances

AWS MachineSets now support spot instances. This lets you create a MachineSet that deploys machines as spot instances so you can save costs compared to on-demand instance prices. You can configure spot instances by adding the following line under the **providerSpec** field in the MachineSet YAML file:

```
providerSpec:
  value:
    spotMarketOptions: {}
```

1.2.4.2. Autoscaling the minimum number of machines to 0

You can now set the minimum number of replicas for a MachineAutoscaler to **0**. This allows the autoscaler to be more cost-effective by scaling between zero machines and the machine count necessary based on the resources your workloads require.

For more information, see the MachineAutoscaler resource definition.

1.2.4.3. MachineHealthCheck with empty selector monitors all machines

A MachineHealthCheck resource that contains an empty **selector** field now monitors all machines.

For more information on the **selector** field in the MachineHealthCheck resource, see the Sample MachineHealthCheck resource.

1.2.4.4. Describing machine and MachineSet fields by using oc explain

A full OpenAPI schema is now provided for machine and MachineSet Custom Resources. **oc explain** now provides descriptions for fields included in machine and MachineSet API resources.

1.2.5. Nodes

1.2.5.1. New descheduler strategy is available (Technology Preview)

The descheduler now allows you to configure the **RemovePodsHavingTooManyRestarts** strategy. This strategy ensures that Pods that have been restarted too many times are removed from nodes. Likewise, the Descheduler Operator now supports full upstream Descheduler strategy names, allowing for more one-to-one configuration.

See Descheduler strategies for more information.

1.2.5.2. Vertical Pod Autoscaler Operator (Technology Preview)

OpenShift Container Platform 4.5 introduces the Vertical Pod Autoscaler Operator (VPA). The VPA 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 create individual custom resources (CR) to instruct the VPA to update all of the Pods associated with a workload object, such as a Deployment, Deployment Config, StatefulSet, Job, DaemonSet, ReplicaSet, or ReplicationController. The VPA helps you to understand the optimal CPU and memory usage for your Pods and can automatically maintain Pod resources through the Pod lifecycle.

1.2.5.3. Anti-affinity control plane node scheduling on RHOSP

If separate physical hosts are available on an RHOSP deployment, control plane nodes will be scheduled across all of them.

1.2.6. Cluster monitoring

1.2.6.1. Monitor your own services (Technology Preview)

The following improvements are now available to further enhance monitoring your own services:

- Allow cross-correlation of the metrics of your own service with cluster metrics.
- Allow using metrics of services in user namespaces in recording and alerting rules.

- Add multi-tenancy support for the Alertmanager API.
- Add the ability to deploy user recording and alerting rules with higher availability.
- Add the ability to introspect Thanos Stores using the Thanos Querier.
- Access metrics of all services together in the web console from a single view.

For more information see Monitoring your own services .

1.2.7. Cluster logging

1.2.7.1. Elasticsearch version upgrade

Cluster logging in OpenShift Container Platform 4.5 now uses Elasticsearch 6.8.1 as the default log store.

The new Elasticsearch version introduces a new Elasticsearch data model. With the new data model, data is no longer indexed by type (infrastructure and application) and project. Data is only indexed by type:

- The application logs that were previously in the **project-** indices in OpenShift Container Platform 4.4 are in a set of indices prefixed with **app-**.
- The infrastructure logs that were previously in the **.operations-** indices are now in the **infra-** indices.
- The audit logs are stored in the **audit-** indices.

Because of the new data model, the update does not migrate existing custom Kibana index patterns and visualizations into the new version. You must re-create your Kibana index patterns and visualizations to match the new indices after updating.

Elasticsearch 6.x also includes a new security plug-in, Open Distro for Elasticsearch. Open Distro for Elasticsearch provides a comprehensive set of advanced security features designed to keep your data secure.

1.2.7.2. New Elasticsearch log retention feature

The new index management feature relies on the Elasticsearch rollover feature to maintain indices. You can configure how long to retain data before it is removed from the cluster. The index management feature replaces Curator. In OpenShift Container Platform 4.5, Curator removes data that is in the Elasticsearch index formats prior to OpenShift Container Platform 4.5, and will be removed in a later release.

1.2.7.3. Kibana link in web console moved

The link to launch Kibana has been moved from the **Monitoring** menu to the Application Launcher  at the top of the OpenShift Container Platform console.

1.2.8. Web console

1.2.8.1. New Infrastructure Features filters for Operators in OperatorHub

You can now filter Operators by **Infrastructure Features** in OperatorHub. For example, select **Disconnected** to see Operators that work in disconnected environments.

1.2.8.2. Developer Perspective

You can now use the **Developer** perspective to:

- Make informed decisions on installing Helm Charts in the **Developer Catalog** using the description and docs for them.
- Uninstall, upgrade, and rollback Helm Releases.
- Create and delete dynamic Knative event sources.
- Deploy virtual machines, launch applications in them, or delete the virtual machines.
- Provide Git webhooks, Triggers, and Workspaces, manage credentials of private git repositories, and troubleshoot using better logs for OpenShift Pipelines.
- Add health checks during or after application deployment.
- Navigate efficiently and pin frequently searched items.

1.2.8.3. Streamlined steps for configuring alerts from cluster dashboard

For **AlertManagerReceiversNotConfigured** alerts that display on the cluster dashboard of the web console, a new **Configure** link is available. This link goes to the Alertmanager configuration page. This streamlines the steps it takes to configure your alerts. For more information, see **BZ#1826489**.

1.2.9. Scale

1.2.9.1. Cluster maximums

Updated guidance around Cluster maximums for OpenShift Container Platform 4.5 is now available.

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

1.2.10. Networking

1.2.10.1. Migrating from the OpenShift SDN default CNI network provider (Technology Preview)

You can now migrate to the OVN-Kubernetes default Container Network Interface (CNI) network provider from the OpenShift SDN default CNI network provider.

For more information, see Migrate from the OpenShift SDN default CNI network provider .

1.2.10.2. Ingress enhancements

There are two noteworthy Ingress enhancements introduced in OpenShift Container Platform 4.5:

- You can enable access logs for the Ingress Controller .
- You can specify a wildcard route policy through the Ingress Controller .

1.2.10.3. HAProxy upgraded to version 2.0.14

The HAProxy used for Ingress has been upgraded from version 2.0.13 to 2.0.14. This upgrade provides a router reload performance improvement. The router reload optimization is most beneficial for clusters with thousands of routes.

1.2.10.4. HTTP/2 Ingress support

You can now enable transparent end-to-end HTTP/2 connectivity in HAProxy. This feature allows application owners to make use of HTTP/2 protocol capabilities, including single connection, header compression, binary streams, and more.

You can enable HTTP/2 connectivity in HAProxy for an individual Ingress Controller or for the entire cluster. For more information, see HTTP/2 Ingress connectivity.

To enable the use of HTTP/2 for the connection from the client to HAProxy, a route must specify a custom certificate. A route that uses the default certificate cannot use HTTP/2. This restriction is necessary to avoid problems from connection coalescing, where the client re-uses a connection for different routes that use the same certificate.

The connection from HAProxy to the application Pod can use HTTP/2 only for re-encrypt routes and not for edge-terminated or insecure routes. This restriction comes from the fact that HAProxy uses Application-Level Protocol Negotiation (ALPN), which is a TLS extension, to negotiate the use of HTTP/2 with the back-end. The implication is that end-to-end HTTP/2 is possible with passthrough and re-encrypt and not with insecure or edge-terminated routes.

IMPORTANT

A connection that uses the HTTP/2 protocol cannot be upgraded to the WebSocket protocol. If you have a back-end application that is designed to allow WebSocket connections, it must not allow a connection to negotiate use of the HTTP/2 protocol or else WebSocket connections will fail.

1.2.11. Developer experience

1.2.11.1. oc new-app now produces Deployment resources

The **oc new-app** command now produces Deployment resources instead of DeploymentConfig resources by default. If you prefer to create DeploymentConfig resources, you can pass the **--as-deployment-config** flag when invoking **oc new-app**. For more information, see Understanding Deployments and DeploymentConfigs.

1.2.11.2. Support node affinity scheduler in image registry CRD

The node affinity scheduler is now supported to ensure image registry deployments complete even when an infrastructure node does not exist. The node affinity scheduler must be manually configured.

See Controlling Pod placement on nodes using node affinity rules for more information.

1.2.11.3. Virtual hosted buckets for custom S3 endpoints

Virtual hosted buckets are now supported to deploy clusters in new or hidden AWS regions.

1.2.11.4. Node pull credentials during build and imagestream import

Builds and imagestream imports will automatically use the pull secret used to install the cluster if a pull secret is not explicitly set. Developers do not need to copy this pull secret into their namespace.

1.2.12. Backup and restore

1.2.12.1. Gracefully shutting down and restarting a cluster

You can now gracefully shut down and restart your OpenShift Container Platform 4.5 cluster. You might need to temporarily shut down your cluster for maintenance reasons, or to save on resource costs.

See Shutting down the cluster gracefully for more information.

1.2.13. Disaster recovery

1.2.13.1. Automatic control plane certificate recovery

First introduced in OpenShift Container Platform 4.4.8, OpenShift Container Platform can now automatically recover from expired control plane certificates. The exception is that you must manually approve pending **node-bootstrapper** certificate signing requests (CSRs) to recover kubelet certificates.

See Recovering from expired control plane certificates for more information.

1.2.14. Storage

1.2.14.1. Persistent storage using the AWS EBS CSI Driver Operator (Technology Preview)

You can now use the Container Storage Interface (CSI) to deploy the CSI driver you need for provisioning AWS Elastic Block Store (EBS) persistent storage. This Operator is in Technology Preview. For more information, see AWS Elastic Block Store CSI Driver Operator.

1.2.14.2. Persistent storage using the OpenStack Manila CSI Driver Operator

You can now use CSI to provision a PersistentVolume using the CSI driver for the OpenStack Manila shared file system service. For more information, see OpenStack Manila CSI Driver Operator.

1.2.14.3. Persistent storage using CSI inline ephemeral volumes (Technology Preview)

You can now use CSI to specify volumes directly in the Pod specification, rather than in a PersistentVolume. This feature is in Technology Preview and is available by default when using CSI drivers. For more information, see CSI inline ephemeral volumes.

1.2.14.4. Persistent storage using CSI volume cloning

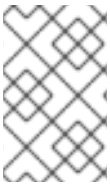
Volume cloning using CSI, previously in Technology Preview, is now fully supported in OpenShift Container Platform 4.5. For more information, see CSI volume cloning.

1.2.15. Operators

1.2.15.1. Bundle Format for packaging Operators and opm CLI tool

The Bundle Format for Operators is a new packaging format introduced by the Operator Framework that is supported starting with OpenShift Container Platform 4.5. To improve scalability and better

enable upstream users hosting their own catalogs, the Bundle Format specification simplifies the distribution of Operator metadata.



NOTE

While the legacy Package Manifest Format is deprecated in OpenShift Container Platform 4.5, it is still supported and Operators provided by Red Hat are currently shipped using the Package Manifest Format.

An Operator bundle represents a single version of an Operator and can be scaffolded with the Operator SDK. On-disk *bundle manifests* are containerized and shipped as a *bundle image*, a non-runnable container image that stores the Kubernetes manifests and Operator metadata. Storage and distribution of the bundle image is then managed using existing container tools like **podman** and **docker** and container registries like Quay.

See Packaging formats for more details on the Bundle Format.

The new **opm** CLI tool is also introduced alongside the Bundle Format. The **opm** CLI allows you to create and maintain catalogs of Operators from a list of bundles, called an index, that are equivalent to a "repository". The result is a container image, called an *index image*, which can be stored in a container registry and then installed on a cluster.

An index contains a database of pointers to Operator manifest content that can be queried via an included API that is served when the container image is run. On OpenShift Container Platform, OLM can use the index image as a catalog by referencing it in a CatalogSource, which polls the image at regular intervals to enable frequent updates to installed Operators on the cluster.

See Managing custom catalogs for more details on **opm** usage.

1.2.15.2. v1 CRD support in Operator Lifecycle Manager

Operator Lifecycle Manager (OLM) now supports Operators using v1 CustomResourceDefinitions (CRDs) when loading Operators into catalogs and deploying them on cluster. Previously, OLM only supported v1beta1 CRDs; OLM now manages both v1 and v1beta1 CRDs in the same way.

To support this feature, OLM now enforces CRD upgrades are safer by ensuring existing CRD storage versions are not missing in the upgraded CRD, avoiding potential data loss.

1.2.15.3. Report etcd member status conditions

The etcd cluster Operator now reports etcd member status conditions.

1.2.15.4. Admission webhook support in OLM

Validating and mutating admission webhooks allow Operator authors to intercept, modify, and accept or reject resources before they are saved to the object store and handled by the Operator controller. Operator Lifecycle Manager (OLM) can manage the lifecycle of these webhooks when they are shipped alongside your Operator.

See Managing admission webhooks in Operator Lifecycle Manager for more details.

1.2.15.5. ConfigMap configurations added from **openshift-config** namespace

ConfigMap configurations are now added from the **openshift-config** namespace using the Insights Operator. This allows you to see if certificates are used for cluster certificate authority and to gather other cluster-related settings from the **openshift-config** namespace.

1.2.15.6. Read-only Operator API (Technology Preview)

The new Operator API is now available as a Technology Preview feature in read-only mode. Previously, installing Operators using Operator Lifecycle Manager (OLM) required cluster administrators to be aware of multiple APIs, including CatalogSources, Subscriptions, ClusterServiceVersions, and InstallPlans. This single Operator API resource is a first step towards a more simplified experience discovering and managing the lifecycle of Operators in a OpenShift Container Platform cluster.

Currently only available using the CLI and requiring a few manual steps to enable, this feature previews interacting with Operators as a first-class API object. Cluster administrators can discover previously installed Operators using this API in read-only mode, for example using the **oc get operators** command.

To enable this Technology Preview feature:

Procedure

1. Disable Cluster Version Operator (CVO) management of the OLM:

```
$ oc patch clusterversion version \
    --type=merge -p \
    '{
      "spec":{
        "overrides":[
          {
            "kind":"Deployment",
            "name":"olm-operator",
            "namespace":"openshift-operator-lifecycle-manager",
            "unmanaged":true,
            "group":"apps/v1"
          }
        ]
      }
    }'
```

2. Add the **OperatorLifecycleManagerV2=true** feature gate to the OLM Operator.

   a. Edit the OLM Operator's Deployment:

```
$ oc -n openshift-operator-lifecycle-manager \
    edit deployment olm-operator
```

   b. Add the following flag to the Deployment's **args** section:

```
...
    spec:
      containers:
      - args:
...
        - --feature-gates
        - OperatorLifecycleManagerV2=true
```

c. Save your changes.

3. Install an Operator using the normal OperatorHub method if you have not already; this example uses an etcd Operator installed in the project **test-project**.

4. Create a new Operator resource for the installed etcd Operator.

   a. Save the following to a file:

   etcd-test-op.yaml file

   ```
   apiVersion: operators.coreos.com/v2alpha1
   kind: Operator
   metadata:
     name: etcd-test
   ```

   b. Create the resource:

   ```
   $ oc create -f etcd-test-op.yaml
   ```

5. To have the installed Operator opt in to the new API, apply the **operators.coreos.com/etcd-test** label to the following objects related to your Operator:

   - Subscription
   - InstallPlan
   - ClusterServiceVersion
   - Any CRDs owned by the Operator

   NOTE

   In a future release, these objects will be automatically labeled for any Operators where the CSV was installed using a Subscription.

   For example:

   ```
   $ oc label sub etcd operators.coreos.com/etcd-test="" -n test-project
   $ oc label ip install-6c5mr operators.coreos.com/etcd-test="" -n test-project
   $ oc label csv etcdoperator.v0.9.4 operators.coreos.com/etcd-test="" -n test-project
   $ oc label crd etcdclusters.etcd.database.coreos.com operators.coreos.com/etcd-test=""
   $ oc label crd etcdbackups.etcd.database.coreos.com operators.coreos.com/etcd-test=""
   $ oc label crd etcdrestores.etcd.database.coreos.com operators.coreos.com/etcd-test=""
   ```

6. Verify your Operator has opted in to the new API.

   a. List all **operators** resources:

   ```
   $ oc get operators

   NAME        AGE
   etcd-test   17m
   ```

b. Inspect your Operator's details and note that the objects you labeled are represented:

```
$ oc describe operators etcd-test

Name:         etcd-test
Namespace:
Labels:       <none>
Annotations:  <none>
API Version:  operators.coreos.com/v2alpha1
Kind:         Operator
Metadata:
  Creation Timestamp:  2020-07-02T05:51:17Z
  Generation:          1
  Resource Version:    37727
  Self Link:           /apis/operators.coreos.com/v2alpha1/operators/etcd-test
  UID:                 6a441a4d-75fe-4224-a611-7b6c83716909
Status:
  Components:
    Label Selector:
      Match Expressions:
        Key:       operators.coreos.com/etcd-test
        Operator:  Exists
    Refs:
      API Version:  apiextensions.k8s.io/v1
      Conditions:
        Last Transition Time:  2020-07-02T05:50:40Z
        Message:               no conflicts found
        Reason:                NoConflicts
        Status:                True
        Type:                  NamesAccepted
        Last Transition Time:  2020-07-02T05:50:41Z
        Message:               the initial names have been accepted
        Reason:                InitialNamesAccepted
        Status:                True
        Type:                  Established
      Kind:                    CustomResourceDefinition
      Name:                    etcdclusters.etcd.database.coreos.com 1
...
      API Version:             operators.coreos.com/v1alpha1
      Conditions:
        Last Transition Time:  2020-07-02T05:50:39Z
        Message:               all available catalogsources are healthy
        Reason:                AllCatalogSourcesHealthy
        Status:                False
        Type:                  CatalogSourcesUnhealthy
      Kind:                    Subscription
      Name:                    etcd 2
      Namespace:               test-project
...
      API Version:             operators.coreos.com/v1alpha1
      Conditions:
        Last Transition Time:  2020-07-02T05:50:43Z
        Last Update Time:      2020-07-02T05:50:43Z
        Status:                True
        Type:                  Installed
      Kind:                    InstallPlan
```

```
    Name:              install-mhzm8 3
    Namespace:         test-project
...
    Kind:              ClusterServiceVersion
    Name:              etcdoperator.v0.9.4 4
    Namespace:         test-project
Events:                <none>
```

1. One of the CRDs.
2. The Subscription.
3. The InstallPlan.
4. The CSV.

1.2.15.7. Upgrading metering and support for respecting a cluster-wide proxy configuration

You can now upgrade a Metering Operator to 4.5 from 4.2 through 4.4. Previously, you had to uninstall your current metering installation and then reinstall the new version of the Metering Operator. For more information, see Upgrading metering.

With this update, support for respecting a cluster-wide proxy configuration is available. Additionally, the upstream repository moved from the operator-framework organization to kube-reporting.

1.2.16. OpenShift Virtualization

1.2.16.1. OpenShift Virtualization support on OpenShift Container Platform 4.5

Red Hat OpenShift Virtualization is supported to run on OpenShift Container Platform 4.5. Previously known as container-native virtualization, OpenShift Virtualization enables you to bring traditional virtual machines (VMs) into OpenShift Container Platform where they run alongside containers, and are managed as native Kubernetes objects.

1.3. NOTABLE TECHNICAL CHANGES

OpenShift Container Platform 4.5 introduces the following notable technical changes.

Operator SDK v0.17.2
OpenShift Container Platform 4.5 supports Operator SDK v0.17.2, which introduces the following notable technical changes:

- The **--crd-version** flag was added to the **new**, **add api**, **add crd**, and **generate crds** commands so that users can opt-in to **v1** CRDs. The default setting is **v1beta1**.

Ansible-based Operator enhancements include:

- Support for relative Ansible roles and playbooks paths in the Ansible-based Operator Watches files.
- Event statistics output to the Operator logs.

Helm-based Operator enhancements include:

- Support for Prometheus metrics.

terminationGracePeriod parameter support
OpenShift Container Platform now properly supports the **terminationGracePeriodSeconds** parameter with the CRI-O container runtime.

/readyz configuration for API server health probe
All OpenShift Container Platform 4.5 clusters using user-provisioned infrastructure must be configured to use the **/readyz** endpoint for API server health checking to remain supported. Any clusters using user-provisioned infrastructure installed on versions prior to OpenShift Container Platform 4.5 must be reconfigured to use **/readyz**.

Clusters using user-provisioned infrastructure without **/readyz** configured can suffer from API outages when the API server restarts. The API server can restart after events such as configuration changes, certificate updates, or control plane machine reboots. The load balancer must be configured to take a maximum of 30 seconds from the time the API server turns off the **/readyz** endpoint to the removal of the API server instance from the pool. Within the time frame, the **readyz** endpoint must be removed or added, depending on whether it returned an error or became healthy. The readiness check is recommended to probe every 5 or 10 seconds, with two consecutive successful requests to become healthy and three consecutive failed requests to become unhealthy.

For more information, see the network topology requirements in the user-provisioned infrastructure installation documentation for your cloud provider.

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 4.5, 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*
- **REM**: *Removed*

Table 1.1. Deprecated and removed features tracker

Feature	OCP 4.3	OCP 4.4	OCP 4.5
Service Catalog	DEP	DEP	REM
Template Service Broker	DEP	DEP	REM
OpenShift Ansible Service Broker	DEP	REM	REM
OperatorSources	DEP	DEP	DEP

Feature	OCP 4.3	OCP 4.4	OCP 4.5
CatalogSourceConfigs	DEP	DEP	REM
Operator Framework's Package Manifest Format	GA	DEP	DEP
v1beta1 CRDs	GA	GA	DEP

1.4.1. Deprecated features

1.4.1.1. Jenkins Pipeline build strategy

The Jenkins Pipeline build strategy is now deprecated. You should use Jenkinsfiles directly on Jenkins or OpenShift Pipelines instead.

1.4.1.2. v1beta1 CRDs

The **apiextensions.k8s.io/v1beta1** API version for CustomResourceDefinitions (CRDs) is now deprecated. It will be removed in a future release of OpenShift Container Platform.

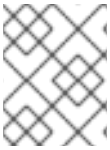
See v1 CRD support in Operator Lifecycle Manager for related details.

1.4.1.3. Custom label no longer in use

The **flavor.template.kubevirt.io/Custom** label is no longer used to identify Custom flavors.

1.4.1.4. OperatorSources and CatalogSourceConfigs block cluster upgrades

OperatorSources and CatalogSourceConfigs have been deprecated for several OpenShift Container Platform releases. Starting in OpenShift Container Platform 4.4, if there are any custom OperatorSources or CatalogSourceConfigs objects present on the cluster, the **marketplace** cluster Operator sets an **Upgradeable=false** condition and issues a **Warning** alert. This means that upgrades to OpenShift Container Platform 4.5 are blocked if the objects are still installed.



NOTE

Upgrades to OpenShift Container Platform 4.4 z-stream releases are still permitted in this state.

In OpenShift Container Platform 4.5, OperatorSources are still deprecated and only exist for the use of the default OperatorSources. CatalogSourceConfigs, however, are now removed.

See the OpenShift Container Platform 4.4 release notes for how to convert OperatorSources and CatalogSourceConfigs to using CatalogSources directly, which clears the alert and enables cluster upgrades to OpenShift Container Platform 4.5.

1.4.1.5. Ignition config spec v2

The v2 Ignition config spec is now deprecated for use when deploying new nodes as part of a fresh OpenShift Container Platform 4.6 installation. The v2 Ignition config spec is still supported for machine configurations.

If you have created custom Ignition v2 spec configurations to deploy new clusters, you must convert these to spec v3 when installing new OpenShift Container Platform 4.6 clusters. You should use the Ignition Config Converter tool to complete the conversion process. In general, v2 can be directly translated to v3. In certain edge cases, you might need to modify the output to make explicit configuration details that were assumed by spec v2.

1.4.2. Removed features

1.4.2.1. OpenShift CLI commands and flags removed

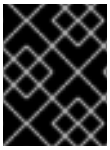
The following **oc** commands and flags are affected:

- The **oc policy can-i** command was deprecated in OpenShift Container Platform 3.9 and has been removed. You must use **oc auth can-i** instead.
- The **--image** flag previously used for the **oc new-app** and **oc new-build** commands was deprecated in OpenShift Container Platform 3.2 and has been removed. You must use the **--image-stream** flag with these commands instead.
- The **--list** flag previously used in the **oc set volumes** command was deprecated in OpenShift Container Platform 3.3 and has been removed. The **oc set volumes** lists volumes without a flag.
- The **-t** flag previously used in the **oc process** command was deprecated in OpenShift Container Platform 3.11 and has been removed. You must use the **--template** flag with this command instead.
- The **--output-version** flag previously used in the **oc process** command was deprecated in OpenShift Container Platform 3.11 and has been removed. This flag was already ignored.
- The **-v** flag previously used in the **oc set deployment-hook** command was deprecated in OpenShift Container Platform 3.11 and has been removed. You must use the **--volumes** flag with this command instead.
- The **-v** and **--verbose** flags previously used in the **oc status** command were deprecated in OpenShift Container Platform 3.11 and have been removed. You must use the **--suggest** flag with this command instead.

1.4.2.2. The oc run OpenShift CLI command now only creates Pods

The **oc run** command can now only be used to create Pods. Use the **oc create** command instead to create other resources.

1.4.2.3. Service Catalog, Template Service Broker, and their Operators

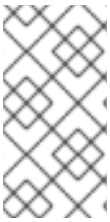


IMPORTANT

Service Catalog is not installed by default in OpenShift Container Platform 4; however, it now blocks upgrades to OpenShift Container Platform 4.5 if installed.

Service Catalog, Template Service Broker, Ansible Service Broker, and their associated Operators were deprecated starting in OpenShift Container Platform 4.2. Ansible Service Broker, including Ansible Service Broker Operator and related APIs and APBs, were removed in OpenShift Container Platform 4.4.

Service Catalog, Template Service Broker, and their associated Operators are now removed in OpenShift Container Platform 4.5, including the related **.servicecatalog.k8s.io/v1beta1** API.



NOTE

Templates are still available in OpenShift Container Platform 4.5, but they are no longer handled by Template Service Broker. By default, the Samples Operator handles Red Hat Enterprise Linux (RHEL)-based OpenShift Container Platform ImageStreams and Templates. See Configuring the Samples Operator for details.

The **service-catalog-controller-manager** and **service-catalog-apiserver** cluster Operators were set to **Upgradeable=false** in 4.4. This means that they block cluster upgrades to the next minor version, 4.5 in this case, 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.

If Service Catalog and Template Service Broker are enabled in 4.4, specifically if their management state is set to **Managed**, the web console warns cluster administrators that these features are still enabled. The following alerts can be viewed from the **Monitoring → Alerting** page on a 4.4 cluster and have a **Warning** severity:

- **ServiceCatalogAPIServerEnabled**
- **ServiceCatalogControllerManagerEnabled**
- **TemplateServiceBrokerEnabled**

If they are still enabled on a 4.4 cluster, cluster administrators can see Uninstalling Service Catalog and Uninstalling Template Service Broker in the OpenShift Container Platform 4.4 documentation to uninstall it, which permits cluster upgrades to 4.5.

In 4.5, a pair of Jobs are created in a new **openshift-service-catalog-removed** namespace to run during the cluster upgrade process. Their behavior depends on the management state of Service Catalog:

- **Removed**: The Jobs remove the following Service Catalog items:
  - Operators
  - namespaces
  - Custom Resources (CRs)
  - ClusterRoles
  - ClusterRoleBindings
- **Unmanaged**: The Jobs skip removal and do nothing.
- **Managed**: The Jobs report an error in logs. This state is unlikely to occur because upgrades would have been blocked. The Jobs take no other actions.

The Jobs and **openshift-service-catalog-removed** namespace will be removed in a future OpenShift Container Platform release.

NOTE

As of OpenShift Container Platform 4.5, all Red Hat-provided service brokers have been removed. Any other broker installed by users is not removed by the upgrade process. This is to avoid removing any services that might have been deployed using the brokers. Users must remove these brokers manually.

1.4.2.4. CatalogSourceConfigs removed

CatalogSourceConfigs are now removed. See OperatorSources and CatalogSourceConfigs block cluster upgrades for more details.

1.5. BUG FIXES

apiserver-auth

- Previously, **oc login** was performing an HTTP request to decide which CA bundle to use to connect to the remote login server. This generated a **remote error: tls: bad certificate** error in the OAuth server logs upon every login attempt, even though the login would succeed. The server certificate chain is now retrieved from an insecure TLS handshake, so the correct CA bundle is chosen and the OAuth server no longer logs bad certificate errors on login attempts. (BZ#1819688)

- Previously, the incomplete security context of the OAuth server Pods might cause the Pods to crashloop when they pick up a custom security context constraint (SCC) that reverts the default behavior. The security context of the OAuth server Pods was modified and a custom SCC no longer prevents the OAuth server Pods from running. (BZ#1824800)

- Previously, the Cluster Authentication Operator always disabled challenge authentication flows for any OIDC identity provider, which meant that logging in with **oc login** was not successful. Now, when an OIDC identity provider is configured, the Cluster Authentication Operator checks whether it allows for the Resource Owner Password Credentials grant and allows challenge-based login if it does. You can now log in using **oc login** for OIDC identity providers that allow the Resource Owner Password Credentials authorization grant. (BZ#1727983)

- Previously, the Cluster Authentication Operator did not properly close connections to the OAuth server, causing the rate of traffic to the OAuth server to grow as connections were being opened faster than they were being dropped. The connections are now properly closed and the Cluster Authentication Operator does not degrade the service of its own payload. (BZ#1826341)

- Previously, the **oauth-proxy** container exited with an error if there was an error reaching the **kube-apiserver** during configuration. This caused multiple container restarts if the **kube-apiserver** and controllers were not stable or fast enough. Now, multiple attempts to perform checks against the **kube-apiserver** are allowed when the **oauth-proxy** container starts, so that it only fails when the underlying infrastructure is truly broken. (BZ#1779388)

Bare Metal Hardware Provisioning

- Because the UEFI boot process was using the **ipxe.efi** binary when using IPv4 networks, the boot process reported that there were no network devices found. As a result, the Preboot eXecution Environment (PXE) boots the machines with **No network devices**. The **dnsmasq.conf** file has been updated to use the **snponly.efi** binary for IPv4 networks. The machines booting with PXE utilize the UEFI network drivers and are able to deploy as they have network connectivity. (BZ#1830161)

- If a cluster has networking issues during install (for example a slow image download) the install could fail. To address this problem, the PXE boot has been changed to include retries and the networking maximum number of retries has been increased for communication between the bare metal provisioner and the nodes being provisioned. The installer will now handle slow network conditions. (BZ#1822763)

Build

- Before starting a build, the OpenShift Container Platform builder would parse the supplied Dockerfile and reconstruct a modified version of it to use for the build. This process included adding labels and handling substitutions of the images named in **FROM** instructions. The generated Dockerfile did not always correctly reconstruct **ENV** and **LABEL** instructions; sometimes the generated Dockerfile would include **=** characters, although the original Dockerfile did not include them. This caused the build to fail with a syntax error. When generating the modified Dockerfile, the original text for **ENV** and **LABEL** instructions are now used verbatim, fixing this issue. (BZ#1821858)

- Previously, the last few lines of error logs were not being attached to a build if a failure occurred in a build Pod init container. Subsequently, build errors in init containers, such as malformed Git URLs, were hard to diagnose. The build controller has been updated so that error logs are attached to a build when failures occur in init containers. Build failures are now easier to diagnose. (BZ#1809862)

- Previously, build failures caused by failed image imports or invalid Dockerfiles were only categorized as generic build errors. Non-default build logging levels were required to diagnose such issues. New failure reasons have now been introduced for failed image imports and invalid Dockerfiles. Build failures relating to failed image imports or invalid Dockerfiles can now be identified within the build object status. (BZ#1809861)

- Previously, build label generation and validation did not include complete Kubernetes validation routines. Builds with certain valid build configuration names would fail due to an invalid build label value being created. The build controller and build API server now use complete Kubernetes validation routines to ensure added build labels meet label criteria. Builds with any valid build configuration name will now result in a valid build label value being created. (BZ#1777337)

- Previously Buildah interpretted variables in Dockerfiles literally, rather than parsing the value contained within a variable. Subsequently, builds would fail when Dockerfiles contained variables. Buildah has been updated to expand Dockerfile variables. Buildah will now parse Dockerfile environment variable values when building container images. (BZ#1810174)

- With the **RunOnceDuration** admission plug-in being disabled in OpenShift 4, an **activeDeadlineSeconds** value was not automatically applied to build Pods. Pods with **activeDeadlineSeconds** set to nil are matched to resource quotas that include **NotTerminating** scope. Subsequently, build Pods failed to start due to quota limitations, in namespaces that had resource quotas with **NotTerminating** scope defined. The build controller now applies a suitable default **activeDeadlineSeconds** value to build Pods. Build Pods are now handled properly in namespaces that have resource quotas that include **NotTerminating** scope. (BZ#1829447)

Cloud compute

- The cluster autoscaler expects provider IDs across node and machine objects to be an exact match. Previously, if a machine configuration included a resource group name that had a mix of upper and lower case characters, the cluster autoscaler would terminate the machine after fifteen minutes, given that a match was not found. Resource group names are now sanitized so

that all characters are set to lowercase. Now, matching provider IDs are correctly identified even when resource group names are entered using a mix of upper and lower case characters. (BZ#1837341)

- Previously, the **metadata** field within machine and MachineSet specifications was not validated when MachineSets were created or updated. Invalid metadata caused unmarshalling errors leading to controllers not being able to process objects. The **metadata** field is now validated when MachineSets are created or updated and invalid entries return an error. Invalid metadata is now identified before MachineSets are created so that subsequent object processing errors are prevented. (BZ#1702089)

- Occasionally during scale down operations, the last machine in a MachineSet will contain deletion annotations. That machine will not be removed by the autoscaler if the minimum MachineSet size is reached before its deletion. Previously, the last machine's deletion annotations would not have been removed after a scale down. A fix has been introduced that changes the way machine annotations are unmarked after a scale down. Now, the annotations no longer persist on the last machine in the MachineSet. (BZ#1820410)

- Previously, the AWS Identity and Access Management (IAM) role assigned to worker nodes did not have sufficient permissions to access the AWS Key Management Service (KMS) key to decrypt the Amazon Elastic Block Store (EBS) volume on mount. Subsequently, Amazon Elastic Compute Cloud (EC2) instances would be accepted, but they would fail to start because they could not read from their root drive. The required permissions have now been granted for EC2 instances to be able to decrypt KMS encrypted EBS volumes with Customer Managed Keys. When using a Customer Managed Key for encrypting EBS volumes, instances now have the required permissions to start successfully. (BZ#1815219)

- The **replicas** field in a MachineSet specification can be set to nil. Previously, if the autoscaler could not determine the number of replicas within a MachineSet, autoscaling operations were prevented. Now, if the **replicas** field is not set, the autoscaler makes a scaling decision based on the last number of observed replicas according to the MachineSet. Autoscaling operations can now proceed even if the **replicas** field in a MachineSet specification is set to nil, assuming that the MachineSet controller has recently synchronized the number of replicas to **MachineSet.Status.Replicas**. (BZ#1820654)

- Previously, the autoscaler would reduce the size of a node group by one on every call to **DeleteNodes**, even if an existing node deletion had not yet completed. This resulted in a cluster having less than the minimum required node count. Now, if a node's machine already has a deletion timestamp, the size of the node group is not reduced further. This prevents the autoscaler from reducing the node count to less than the required capacity when it calls **DeleteNodes**. (BZ#1804738)

Cloud Credential Operator

- Cloud Credential Operator (CCO) could crash loop when the original cluster was installed with OpenShift Container Platform 4.1. CCO would be unable to reconcile the permissions requests found in the CredentialsRequest objects. This bug fix updates CCO to no longer assume that parts of the Infrastructure fields are available. As a result, CCO can work with clusters that were originally installed with OCP 4.1. (BZ#1813343)

- Cloud Credential Operator (CCO) no longer bypasses Security Context Constraints (SCCs). Previously, CCO would run with excess permissions that are not required for CCO to perform its tasks. With this enhancement, there is no unnecessary bypassing of SCCs for CCO. (BZ#1806892)

Cluster Version Operator

- The Cluster Version Operator (CVO) had a race condition where it would consider a timed-out update reconciliation cycle a successful update. This only happened for restricted network clusters where the Operator timed out attempting to fetch release image signatures. This bug caused the CVO to enter its shuffled-manifest reconciliation mode, which could break the cluster if the manifests were applied in an order that the components could not handle. The CVO now treats timed-out updates as failures, so it no longer enters reconciling mode before the update succeeds. (BZ#1843526)

- Failures to roll-out deployments during updates was logged only in CVO logs and only a general error message was reported to ClusterVersion. This general error message made it difficult for users and teams to debug the failure unless looking at the CVO logs. This bug fix updates CVO to expose underlying errors to roll-out to ClusterVersion. As a result, debugging is now easier for deployment roll-outs during upgrades. (BZ#1768260)

Console Kubevirt plugin

- With this release, if a VM is configured to use a disk with an invalid or unrecommended bus type, the **Disks** tab on the created VM view displays a disk interface warning. (BZ#1803780)

- Previously, all DataVolumes were categorized as VM Disk imports. This incorrect categorization caused the the Activity card to disappear for DataVolumes that did not have an owner reference to a VM. With this release, only DataVolumes with an owner reference to a VM are categorized as VM Disk imports and the Activity card does not disappear for DataVolumes that do not have an owner reference to a VM. (BZ#1815138)

- Previously, DataVolumes and their associated PersistentVolumeClaims (PVCs) were not deleted when the VM disk was removed. These objects were only deleted when the VM was deleted, and there was not an option to preserve a DataVolume during VM deletion. With this release, the user can choose to preserve or delete DataVolumes and PVCs when deleting a VM disk or VM. This does not apply for disks that are deleted using the CD-ROM modal. (BZ#1820192)

- Previously, the number of disks in the inventory did not match the number of disks in the disk list. The inventory view is now updated to show CD-ROMs and disks separately. (BZ#1803677)

- Previously, it was not possible to create a VM with the default YAML used by the VM wizard because the default YAML VM template did not contain values required by the VM wizard. With this release, the default YAML VM template contains all required values. (BZ#1793962)

- The web console previously reported that failed VM migrations had succeeded. When migrating a VM, the web console now correctly reports when a VM migration fails. (BZ#1806974)

- Previously, the VM wizard did not generate the cloud-init configuration in correct format, and as a result it was not applied on the VM. With this release, the format generated by the wizard has been corrected and the cloud-init configuration that is provided in the VM wizard is applied on the VM. (BZ#1821024)

- Previously, VM template sockets were not reflected in the final VM created by the VM wizard, which caused the number of vCPUs to be doubled after the VM was created. With this release, the VM template sockets, cores, and threads are reflected when creating a VM and the resulting number of vCPUs is correct. (BZ#1810372)

- A change in the URL for the VM templates list caused the user to be redirected to the wrong page after deleting a VM template. The URL has been fixed in this release. (BZ#1810379)

- Previously, when a running VM was removed, the associated VMI appeared in the Virtual Machine list with the satus **VM error**. With this release, stale VMIs with deleted associated VMs are no longer listed. (BZ#1803666)

- Previously, the disk import process only expected VM import resources. As a result, the VM resource link for import activity from a VM template or VMI pointed to a nonexistent VM. With this release, the import process recognizes VM templates and VMIs that are import resources and links to the correct resource. (BZ#1840661)

- With this release, the VM disk import process no longer reports a process value of **NaN%**. (BZ#1836801)

- Previously, the virtual machine wizard used **virtIO** as the default interface for the VM root disk instead of using interface specified in the common template. However, the **virtIO** interface is not compatible with all operating systems. With this release, the correct default interface for the operating system is selected based on common template used. (BZ#1803132)

Console Metal3 plugin

- Previously, there was no space between the **Powering on/off** message and the bare metal host link in the web console. A space has been added so that the message now reads properly. (BZ#1819614)

- Previously, for bare metal installations, the Bare Metal Host Details page would not load when some nodes were not available. Now the Bare Metal Host Details page shows 0 Pods instead. (BZ#1827490)

Web console (Developer perspective)

- Previously, it was difficult to see the list of Pods or resources associated with a Knative service in the **Topology** view. With this bug fix, when you select the Knative service, the sidebar displays a list of Pods along with a link to see the logs. (BZ#1801752)

- When you edited an existing query using the PromQL editor in the **metrics** tab of the **Monitoring** view, the cursor moved to the end of the line. With this bug fix, the PromQL editor works as expected. (BZ#1806114)

- For Knative images, in the **Add → From Git** option, the **Advanced Options** for **Routing** would not provide a prefetched container port option. Also, if you created the service without updating the default port value of **8080**, the revisions would not show. With this bug fix, the user can select from the available port options using the drop-down list or provide input if they want to use another port and the revisions are shown as expected. (BZ#1806552)

- Previously, a Knative service created using the CLI could not be edited using the console because the images could not be fetched. Now, if the associated ImageStreams are not found while editing, the value provided by the user for the container image in the YAML file is used. This allows the user to edit the service using the console, even if the service was created using the CLI. (BZ#1806994)

- In the **Topology** view, editing the image name in the external image registry for a Knative service did not create a new revision. With this bug fix, a new revision of the service is created when the name of the service is changed. (BZ#1807868)

- When you used the **Add → Container Image** option, and then selected the **Image stream tag from internal registry** option, the **ImageStreams** drop-down list did not list the option to deploy images from the **OpenShift** namespace. However, you were able to access them

through the CLI. With this bug fix, all users have access to images in the **OpenShift** namespace through the console and the CLI. (BZ#1822112)

- Previously, in the **Pipeline Builder**, when you edited a Pipeline that referenced a Task that did not exist, the entire screen would go white. This fix now displays an icon to indicate that an action is required and a drop-down list is displayed to easily update the Task reference. (BZ#1839883)

- In the **Pipelines Details** page, when you changed existing fields in the **Parameters** and the **Resources** tabs, the **Save** button was disabled even though the new changes were detected. The validation criteria has now been modified and the **Save** button is enabled to submit changes. (BZ#1804852)

- In the **Add → From Git** option, the Pipeline templates provided by the OpenShift Pipelines Operator would fail when the **Deployment** or **Knative Services** resource options were selected. This bug fix adds support to use the resource type as well as the runtime to determine the Pipeline template, thus providing resource-specific Pipeline templates. (BZ#1796185)

- When a Pipeline was created using the **Pipeline Builder** and a Task parameter of the type array was used, the Pipeline did not start. With this bug fix, both array and string type parameters are supported. (BZ#1813707)

- In the **Topology** view, filtering nodes by application returned an error when the namespace had Operator-backed services. This bug fix adds the logic to filter out the Operator-backed service nodes based on the selected application group. (BZ#1810532)

- The **Developer Catalog** showed no catalog results until you selected the **Clear All Filters** option. With this bug fix, all catalog items are seen by default and you do not need to clear all filters. (BZ#1835548)

- Previously, users were unable to add environment variables for **knative** services. As a result, apps where **envVariables** would be needed might not have worked as expected. Now, support has been added for environment variables. (BZ#1839114)

- The Developer Console Navigation menu is now available and is aligned with the latest UX designs. (BZ#1801278)

- Time Range and Refresh Interval drop menus have been added in the Monitoring dashboard tab in Developer Perspective. (BZ#1807210)

- No Pipeline Resources were created in the namespace although the Start Pipeline modal required one. The user would see a disabled and empty dropdown above fields, losing some context of what the fields were for. With this bug fix, **Create Pipeline Resource** gives the user context of what they were doing inline in the Start Pipeline modal. The user now has a better experience starting a Pipeline from the start modal when there are no Pipeline Resources created in the namespace. (BZ#1826526)

- Layout padding was missing, which allowed the title to flow over the **Close** button. If text was over the **Close** button, it made it difficult to click. The layout is now fixed to prevent the title from overlapping the **Close** button and the button is now always accessible via mouse click. (BZ#1796516)

- Pipeline Builder incorrectly interpreted a default value of an empty string (**''**) as having no default. Some Operator-provided tasks needed this to be the default and, therefore, had issues working without it. Check for a default property and do not assume the validity of the value. Now, any values that the OpenShift Pipeline Operator deems as a valid default value are respected. (BZ#1829567)

- The Pipeline Builder reads Task/ClusterTask definitions and incorrectly assumed that all Parameters were of type **string**. When a Task Param of type **array** was encountered, it would cast the array to a string and represent it, losing the type; it would produce a value to the Task param as **string**, thus breaking the contract with the Task. The **array** type is now supported in the UI and the type is properly retrained. Managing both types allows the Pipeline Builder to work the way it was intended. (BZ#1813707)

- The Pipeline page was inconsistent with other pages. The **Create Pipeline** button was always enabled and did not take into consideration when no projects were available. The **Create Pipeline** button is now removed when the Getting Started guide is enabled. (BZ#1792693)

- Metrics queries for the **Dashboard & Metrics** tab got updated in the design document. The code need to be synced with w.r.t queries. The queries are now updated and the order of the metrics queries and their labels are synced with the design. (BZ#1806518)

- The tile description variable was incorrectly set to be the CRD description appended with the CSV description. This caused the tile descriptions to be wrong. The tile descriptions are now back to the original value and the appended value is now moved to its own variable. (BZ#1814639)

- The **eventSources** API Group is updated to the latest supported API Group, **sources.knative.dev**. This update allows sources generated by the new API Group to be recognized in the **Topology** view of the web console. (BZ#1836805)

- With the release of Red Hat OpenShift Serverless 1 Serverless Operator version 1.7.1, the Operator is generally available. The Tech Preview badge in the Developer perspective of the web console has been removed. (BZ#1827042)

DNS

- Previously, CoreDNS metrics were being exposed over an insecure channel within a cluster. Now the proper TLS components and a kube-rbac-proxy sidecar have been added to secure the CoreDNS metrics endpoint and expose CoreDNS metrics over a secure channel. (BZ#1809197)

- Previously,adding arbitrary taints to nodes could cause problems related to the DNS operator's operand. Now the DNS operator's operand tolerates any taint added to a node. The operand runs on and updates /etc/hosts on all Linux node hosts. **Missing CNI default network** events may be observed when the operand starts on a node that is still initializing, but such errors are transient and can be ignored. (BZ#1813479)

- Previously, there was a dependency on having specific DNS names for master nodes. Now any legal hostname can be used for master nodes. (BZ#1807234)

- Previously, when the **dnses.operator.openshift.io/default** object existed but its corresponding DaemonSet was not available, clusteroperators/dns reported the **Available** condition with an incorrect **NoDNS** reason and **No DNS resource exists** message. Now under these same conditions the correct reason and message will appear. (BZ#1835725)

etcd

- Previously, the etcd peer certificate did not include the IPv6 localhost address and failed to connect on **https://[::1]:2379** messages. This bug fix includes the **::1** as one of the hosts in the peer certificate. Now repeated failed attempts to connect using **https://[::1]:2379** are no longer shown. (BZ#1810997)

- Previously, the CVO was overwriting certificates in a ConfigMap every 10 minutes. This caused a lot of overhead and negatively impacted cluster performance and stability. Now, certificates are created only once in a ConfigMap for improved performance and stability. (BZ#1819472)

- Previously, the Cluster etcd Operator health status reporting was hard to understand. The was caused by improper log messaging construction, which often resulted in uncertainty of the cluster's status. This has been fixed by properly analyzing the Operator statuses in a separate function to construct a proper log message and event about the etcd status. Now the status of the etcd Pods on all master nodes are more meaningful. (BZ#1821286)

- Previously, the TLS certificates were mistakenly signed for 10 years, even though the documentation said that they were signed for three years. Now, the certificates are signed for only three years. (BZ#1837594)

- gRPC-go 1.23.0 had client-side load-balancer bug. This bug could could cause deadlock. gRPC-go has been upgraded to version 1.23.1, in which the bug was fixed. (BZ#1823993)

- After stopping all Pods, the restore process only restarted etcd, api-server, api-scheduler, and controller-manager. It did not restart network Pods. As a result, kubelets could not communicate, and bare metal clusters could not stand up. Now, the restore service no longer stops Pods that it cannot restart. Clusters stand up after the restore process runs. (BZ#1835146)

Etcd Operator

- Previously, there were missing properties in the etcd spec, causing the **oc explain etcd** command to incorrectly list properties referenced from the spec. The applicable CRD has been updated to describe the missing properties. Now the **oc explain etcd** command fully describes the properties of etcd. (BZ#1809282)

- The Etcd Operator was performing improper health checks, leading to incorrect event reports and misleading log messages. Health statuses are now detected correctly with improved messaging, providing accurate health statuses. (BZ#1832986)

Image

- Previously, the Nodeca daemon was created only when the registry was set to **managed**. When the registry was removed, the Nodeca daemon is not created. With this bug fix, Nodeca daemons are always created and Nodeca daemons are created even if the registry is removed. (BZ#1807471)

Image registry

- Previously if you deleted registry configuration without a proper storage configuration, the resource was never finalized due to the lack of storage configuration and the operator could not remove the storage because it did not know about it. This bug fix made storage configuration optional, which allows the resource to be completely finalized. (BZ#1798618)

- Previously, the Image Registry Operator was not setting the **nodeSelector** label on Image Registry created resources. This could have caused future issue because of not specifying in what nodes resources can run, and could end up running the registry over unsupported platforms. This bug fix added the missing label to created resources. Now, it is possible to see the label on the created resources. (BZ#1809005)

- Previously pushing an image to a namespace that does not exist caused the Image Registry to return a 500 error code. This bug fix changed the return code to indicate the lack of permissions. Now when pushing images to a namespace that does not exist a permission denied

error is returned. (BZ#1804160)

- The Azure infrastructure name is used for generated Azure containers and storage accounts. Therefore, if the Azure infrastructure name contained uppercase letters, the container would successfully be created, but the storage account creation would fail. This bug fix adjusts the container name creation logic to discard invalid characters, allowing the image registry to deploy on an infrastructure that contains invalid characters in its name. (BZ#1827807)

- When deleting a non-empty image registry with GCP storage, the image registry hostname was not being removed from the image configuration file. This prevented you from creating a new image registry. The code has been changed to remove the image registry hostname from the image configuration file when you delete an image registry. As a result, you can delete and create image registries as expected. (BZ#1827075)

- Because the image registry was not removing objects from a bucket before it removes the bucket, you could not delete a bucket with images. The code has been changed to remove images before removing a bucket. You can delete non-empty buckets as expected. (BZ#1827075)

- Because images in the image registry were not clearing their yum cache, the image sizes can get large. The image registry Dockerfile was changed to include a **yum clean all** command. The size of the images are smaller. (BZ#1804493)

- The **keepYoungerThan** parameter in an image pruning custom resource, uses nanoseconds and cannot be configured to use a larger period of time. Nanoseconds are not an appropriate period to use in an image pruner. A new parameter has been added to the image pruning custom resource, **keepYoungerThanDuration** that replaces and overrides the **keepYoungerThan** parameter. (BZ#1835004)

- The Image Registry Operator did not properly clean up the storage status when the user changes the operator to **Removed** state. As a result, when the user changed the operator back to **Managed**, the operator could not create a new storage Pod. The operator was changed to properly clean up the storage status and the operator can create a new storage Pod. (BZ#1785534)

- Because the Image Registry Operator was not cleaning logs, you could see improper messages in the logs. The code has been changed to clean the logs to remove these improper messages. The logs now display proper information. (BZ#1797840)

- Because the default Image Registry Operator was configured with 0 replicas, problems could result unless the value is manually changed. The operator was updated to install with 1. (BZ#1811846)

- Registry credentials used during the cluster install were not available to specific namespaces, the user needed to create new credentials. The code was changed so that if the credentials for a registry were provided during the install, users can import images using those credentials. (BZ#1816534)

- Because the Image Registry Operator was being installed with only one Pod, it did not meet requirements. The operator is now installed with two Pods for high availability. (BZ#1810317)

Installer

- On the Azure platform, the **cifs-utils** package is required to create volume mounts for Pods. With this release, **cifs-utils** is included in the packages installed for RHEL 7 hosts when installing OpenShift Container Platform. (BZ#1827982)

- When recovering from an expired control plane certificate, the cluster is unable to connect to the recovery API server on port 7443. This is caused by the recovery API server's port conflicting with the HAProxy port used for OpenStack, oVirt, bare metal, and vSphere. This results in an **Unable to connect to the server: x509: certificate signed by unknown authority** error. HAProxy now listens on port 9443, allowing the recovery API server to use port 7443 to facilitate the recovery process for an expired control plane certificate. (BZ#1821720)

- Previously, the RHOSP installer created security groups using **remote_group_id** to allow traffic origins. Using the **remote_group_id** in the security rules was very inefficient, triggering a lot of computation by the OVS agent to generate the flows. This process sometimes exceeded the time allocated for flow generation. In such cases, especially in environments already under stress, master nodes would be unable to communicate with worker nodes, leading the deployment to fail. Now IP prefixes for whitelisting traffic origins are used instead of the **remote_group_id**. This lessens the load on Neutron resources, reducing the occurrence of timeouts. (BZ#1825286)

- Previously, the installation program required the user to manually create a virtual machine template before it could create an OpenShift Container Platform cluster on Red Hat Virtualization (RHV). This is because the installation program did not meet the following requirements in RHV version 4.3.9: 1) The installation program must pass the ignition to the virtual machine, and 2) The template must specify its OS type as Red Hat CoreOS (RHCOS). The installation program now creates a template that specifies RHCOS as the OS type, and it passes the ignition to the VM. The user no longer needs to create a virtual machine template. (BZ#1821151)

- Previously, the Keepalive process that provides failover for both API-VIP and INGRESS-VIP addresses in bare metal installer-provisioned infrastructure clusters used an IPV4 local address in a script that monitors local component status to decide which node should own the VIP even if the deployment used IPV6 addresses. Because of this, in IPv6 deployments, Keepalived sometimes received incorrect component status. Now, the Keepalived script uses localhost, which resolves to 127.0.0.1 in V4 deployments and ::1 in V6 deployments, so it always uses the correct local IP address. (BZ#1800969)

- Previously, in bare metal clusters that use installer-provisioned infrastructure, the VIP did not always fail over to a control plane machine with a healthy load balancer. Because of this, the control plane machine continues to own the API-VIP IP address even though the local load balancer is unhealthy and the OpenShift Container Platform API is unreachable for ~10 seconds. Now, the Keepalived check for API-VIP script also monitors the self-hosted load balancer health, and the API-VIP will failover to a control plane node with a working load balancer and prevent service downtime for the OpenShift Container Platform API. (BZ#1835974)

- Previously, the installation program did not explicitly check for an overlap between the **machineCIDR** and **provisioningNetworkCIDR** ranges. As a result, the error message when the network ranges overlapped was unclear. Now, the installation program explicitly checks for overlap between these network ranges and presents a clear error message if they overlap. (BZ#1813422)

- Because Operators in the control plane can start before bootstrap process completes, the bare metal provisioning infrastructure might be active on both the bootstrap and control plane at the same time. Previously, both sets of provisioning infrastructure could provision compute machines, and the machines did not all use the same infrastructure. Now, the bootstrap provisioning infrastructure provisions only control plane machines, so both provisioning infrastructures can be online at the same time. (BZ#1800746)

- Previously, the wrong port number was used when blocking DHCP traffic to the bootstrap node on IPv6. Because of this, a race condition was introduced where a control plane machine sometimes incorrectly obtained a DHCP lease from the bootstrap node. Now, the correct port is

blocked for DHCPv6, and control plane machines are provisioned from only the bare metal infrastructure that runs in the cluster (BZ#1809691)

- Previously, with a bare metal cluster that uses installer-provisioned infrastructure, using VRRP to manage the virtual IP addresses for OpenShift Container Platform clusters meant that if you ran several clusters, virtual router IDs might already be in use in the broadcast domain. Because of this, nodes might be assigned virtual IP addresses that are already in use. Now, you can use a tool to check which virtual router IDs will be used for the chosen cluster name before you deploy a cluster. (BZ#1821667)

- OpenShift Container Platform version 4.1 clusters did not use the **infrastructure.status.infraPlatform** parameter. Because of this, Operators must check and use old fields for clusters that originally installed version 4.1, which causes errors during upgrades. Now, the migration controller sets the new fields for all clusters during upgrade by using information that is available in the cluster, so the Operators can use all of the new parameters and reduce upgrade errors. (BZ#1814332)

- Because the AWS API that is used to fetch resources for clusters is extremely slow in reacting to previously deleted resources, trying to delete already deleted hosted zones caused failures if you tried to destroy a cluster multiple times. Because of this, the destroy command looped until the AWS APIs removed the HostedZone from their response. Now, the installation program skips the **notfound** error for hosted zone, and the destroy command completes more quickly. (BZ#1817201)

- Previously, the bootstrap server endpoint used the 'api' endpoint that goes through the external load-balancer. Because of this, you needed to open another port to add RHEL nodes to the cluster. Now the bootstrap server endpoint uses the internal 'api-int' endpoint, and you no longer need to open another port on the external load balancer. (BZ#1792822)

- Previously, for bare metal clusters, in order to support nodes DNS resolution, the node's **/etc/resolv.conf** file pointed to the local instance of the infrastructure coredns by prepending the node's control plane IP address to the node's **/etc/resolv.conf** file. Because of this, when a host already had three nameservers listed in its **/etc/resolv.conf** file, Pods generated a "nameserver limits were exceeded" alert. Now, only the first three nameservers are included in the generated **/etc/resolv.conf** file, so the alert is no longer generated by Pods. (BZ#1825909)

- Previously, the **ipxe.efi** file was not present in the running ironic container, so the booting UEFI failed in cases where **ipxe.efi** was needed. Now, the **ipxe.efi** file is copied to the **/shared** directory at runtime, so UEFI boot is no longer impacted. (BZ#1810071)

- Previously, rate limiting from AWS sometimes caused a failure to create records for the cluster. Now, the installation program uses an exponential back-off to allow for a longer wait timeout, which creates fewer failures due to rate limiting. (BZ#1766691)

- Previously, rate limiting from AWS sometimes causes a failure to fetch zones for the cluster, which would prevent the cluster from installing. Now, the installation program uses an exponential back-off to allow for a longer wait timeout, which creates fewer failures due to rate limiting. (BZ#1779312)

- Previously, the installation program did not check for symlinks when determining relative path to the configuration file, so the installation fails if the installation program runs from a symlink. Now, the installation program checks for symlinks, and you can run the installation program from a symlinked directory. (BZ#1767066)

- Previously, the AWS Terraform provider that the installation program used occasionally caused a race condition with the S3 bucket, and the cluster installation failed with the following error: **When applying changes to module.bootstrap.aws_s3_bucket.ignition, provider"**

level=error msg="\"aws\" produced an unexpected new value for was present, but now absent. Now, the installation program uses different AWS Terraform provider code, which now robustly handles S3 eventual consistency, and the installer-provisioned AWS cluster installation does not fail with that error. (BZ#1745196)

- Previously, the CoreDNS forward plugin used a random server selection policy by default. As a result, clusters failed to resolve the OpenStack API hostname if given multiple external DNS resolvers. The plugin now uses DNS servers in the order they are provided. (BZ#1809611)

- Due to performance variability among RHOSP clouds where OpenShift Container Platform can be installed, installation times vary. As a result, the installer can time out before the installation succeeds. As a workaround, check your cluster's status after the installer indicates failure. The cluster might be healthy. (BZ#1819746)

- On RHOSP, control plane and compute nodes inject their IP addresses into their **/etc/resolv.conf** files as their preferred nameservers. As a result, hosts that already had three nameservers in the file generated nameserver limit warnings. Now, only the first three nameservers in /etc/resolve.conf are preserved. In this situation, Pods no longer generate nameserver warnings. (BZ#1791008)

- Previously, RHOSP clouds without trunk ports could return an error that the installer misinterpreted as a failure. As a result, cluster destruction would loop before timing out. With this update, the installer now correctly interprets the error, allowing for successful cluster destruction on clouds that do not support trunk ports. (BZ#1814593)

- RHOSP resources that share names cannot be removed. Previously, if security groups that shared a name existed, cluster destruction using Ansible playbooks failed on RHOSP clouds. Now, the **down-security-groups.yaml** playbook uses group IDs instead of names when destroying clusters. All security groups are deleted if the playbook finishes successfully. (BZ#1841072)

- Some RHOSP environments might enforce a policy that disallows VMs from booting with ephemeral disks. As a result, cluster installations failed when bootstrap machines attempted to boot from ephemeral disks. Now, bootstrap machines follow rootVolume settings from the control plane machine pool, allowing cluster installations to succeed in environments that disallow VMs from booting with ephemeral disks. (BZ#1820434)

- Previously, a prerequisite Terraform step did not always happen before floating IP address (FIP) association on clusters that ran on RHOSP. As a result, a race condition could occur that would cause installations to fail. The Terraform step now always occurs before FIP association. (BZ#1846297)

- Because a RHOSP user-provisioned installation script was not compatible with some Ansible versions, installations could fail. The script was updated to assure broad compatibility. Now, installations succeed regardless of the your Ansible version. (BZ#1810916)

- Currently, the RHOSP user-provisioned infrastructure playbooks do not delete Cinder volumes that were created in the cluster's lifetime. Resultantly, destroyed clusters leak Cinder volumes. As a workaround, delete Cinder volumes manually after cluster destruction. (BZ#1814651)

- Previously, clusters on RHOSP did not process all certificates that were passed to it in certificate authority (CA) file bundles. As a result, clusters could not be installed with intermediate certificates that were signed by a non-default trusted authority. CA files are now split and processed separately, allowing installations that use intermediate certificates signed by non-default trusted authorities. (BZ#1809780)

kube-apiserver

- Previously, some users could not upgrade from 4.2 to 4.3 due to an upstream bug that prevented the running of clusters that used a mix of Kubernetes 1.14 and 1.16 components. This fix includes a merge from upstream so that OpenShift Container Platform 4.3 is now compatible with OpenShift Container Platform 4.2 when upgrading. (BZ#1816302)

- Previously when creating a new version of an Operator, it could take several minutes before the lock was released and the new version of the Operator could continue because the leader election setup was not releasing the lock when the Operator received a UNIX signal to shut down. With this fix, the Operator rollout time has improved significantly because control plane Operators now respect the graceful termination period and do not have to wait for the lock to be released on startup. (BZ#1775224)

- Previously during upgrades, the OpenShift Container Platform API server would sometimes be added back to the GCP load balancer, despite not yet being able to serve traffic because routes on the node were misconfigured. This was caused by a race condition between the node and GCP load balancer. This has been fixed by moving route configurations to iptables and differentiating between local and non-local traffic; non-local traffic is now always accepted. Now during API server upgrades, connections are gracefully terminated, and new connections are load-balanced only to running API servers. (BZ#1802534)

kube-scheduler

- Previously, Pods that were evictable because they would fit on a certain node might not be evicted because the Descheduler would return early in the node-checking loop to determine if Pods were evictable in a NodeAffinity strategy. Now, the break condition of the node-checking loop has been fixed so that all nodes are considered when checking evictability. (BZ#1820253)

Logging

- Previously, the Fluentd buffer queue was not limited and a high volume of incoming logs could flood the filesystem of a node and cause it to crash. As a result, applications would be rescheduled. To prevent this type of crash, the Fluentd buffer queue is now limited to a fixed amount of chunks per output (default: 32). (BZ#1780698)

- In an IPv6 bare metal deployment, Elasticsearch was binding on the IPv4 loopback address instead of the cluster IPv6 address. As a result, the Elasticsearch cluster failed to start. The downward API was changed to set the binding and publish host for Elasticsearch. Elasticsearch is able to bind to the network interface and starts as expected. (BZ#1811867)

- Because the cluster logging cluster service version (CSV) was using incorrect paths to obtain the status of some cluster logging components, the status was not being reported. As a result, cluster logging was not functioning properly. The paths have been corrected and cluster logging is working as expected. (BZ#1840888)

- Because the Elasticsearch Operator create a second deployment when more than 3 Elasticsearch nodes are configured, the Cluster Logging Operator was not reading the correct number of Elasticsearch nodes. As a result, the Cluster Logging Custom Resource always reported he number of nodes associated with one deployment. The Cluster Logging Operator was changed to corretly compute the number of Elasticsearch nodes. (BZ#1732698)

Machine Config Operator

- Multiple available networks on worker nodes make it difficult to pick an address on the control plane for CRI-O. This causes CRI-O to often bind to a non-control plane interface. This bug fix updates the CRI-O systemd service to depend on a service that chooses the correct interface and configures the CRI-O service. As a result, CRI-O binds to an address in the control plane as expected. (BZ#1808018)

- Previously, when configuring OperatorHub for restricted networks in an IPv6 bare metal deployment, multiple interfaces could come up on OpenShift Container Platform (OCP) nodes without DHCP-provided names nor reverse resolution. This caused the multicast DNS publishing service to start with the default **localhost** name. This bug fix ensures that the Machine Config Operator only configures non-default names and waits until those are available. As a result, the correct host names are published to multicast DNS. (BZ#1810333)

- The Ingress Virtual IP management configuration was using a fixed string for its password. If two VRRP keepalived instances in separate clusters had the same Virtual Router ID, they would have the same password and potentially belong to a single virtual router. This bug fix makes the password change depending on cluster configuration. As a result, different cluster Ingress Virtual IPs now have a different password. (BZ#1803232)

- Previously, the systemd service doing control plane IP detection and configuring for Kubelet and CRI-O could run before any control plane IP was configured, resulting in a Kubelet and CRI-O failure message that the nodeip-configuration 'Failed to find suitable node ip'. Now, the system retries until the interface has a control plane IP configured. (BZ#1819484)

- Previously, when Coredns would forward DNS requests to the list of servers in the **/etc/resolv.conf** file, if the file was changed, the change would not be reflected in the Coredns Corefile. With this fix, the Coredns-monitor Pod now verifies that the Coredns forward list is synced with **/etc/resolv.conf** so that the list of servers appear in the file. (BZ#1790819)

- Previously, when the interface that keepalived uses was bridged, it was possible for users to dynamically put interfaces in bonds or bridges, and doing so could prevent keepalived from resuming operation, disrupting Virtual IP management. With this fix, the monitor interface now changes and reloads keepalived so that it reads the new configuration and virtual IP management can operate with minimal disruption. (BZ#1751978)

- Because some routes contained the **expires** field, IPv6 (**non_virtual_ip** script) could not process the route. As a result, services that need to be configured with a non_virtual_ip fail. The **non_virtual_ip** script has been updated. Routes are parsed and services are configured correctly. (BZ#1817236

Web console (Administrator perspective)

- Invalid monitoring flags were set when the console was started due to a missing Prometheus link on the monitoring metrics query page. Now, the proper flags have been set and Prometheus monitoring is available on the metrics query page. (BZ#1811481)

- When a user tried to install into an unsupported namespace, the form would not be submitted because it was not clear to the user which installation mode is supported by the Operator group. Now, an alert has been added for the supported Operator's install mode. The alert will clarify why the picked namespace can be used by the install Operator. (BZ#1821407)

- Machine Health Checks and Machine Config were not visually separated, causing confusion to the user. Now, a divider has been added between the Machine Health Checks and the Machine Config for clarity. (BZ#1817879)

- An error message would appear in the browser's console due to a missing property for the react component. Now, the property has been added for the react component and the error message does not occur. (BZ#1800769)

- Multiple alert receivers could be created with the same name. If one of the same named alerts were deleted, all would be deleted. Now, In the Create Receiver form, users are prompted with an error message if the name already exists, and the Create button is disabled. Users cannot create two receivers with the same name. (BZ#1805133)

- PVCs were sorted alphabetically, and now they are sorted numerically. (BZ#1806875)
- Services were listed in alphabetical order, so that **oc** was not the first option. Now, the **oc** option is appended to the front of the list. (BZ#1802429)
- After Alerts were changed to a Silenced state, the Status card and Notification Drawer would continue to show the silenced Alert. Now, the Dashboard and the Notification Drawer do not show silenced Alerts. (BZ#1802034)
- After Alerts were changed to a Silenced state, the Status card and Notification Drawer would continue to show the silenced Alert. Now, the Dashboard and the Notification Drawer do not show silenced Alerts. (BZ#1808059)
- Sorting was not based on data in the column, causing erroneous sorting. Now, data is sorted by the correct operand status values. (BZ#1812076)
- Status descriptor paths can be longer than the space allotted for them inside the donut chart. Status descriptor paths that are very long can be clipped on the right and left sides, obscuring the value. Now, the status descriptor is path below the donut chart so it can wrap as needed and allow more than one status descriptor per row. Status descriptor paths with long values are full visible, and less scrolling is required to view all status descriptors. (BZ#1823870)
- The console would display an inaccurate update status of **Error Retrieving** when the version did not appear in the update channel. This suggested the version should be available, but it was not. Now, the console display has been updated to **Verion not found** when the version does not appear in the update channel. (BZ#1819892)
- The installed Operators list was only sortable by the **Name** column, limiting sorting options for users. Now, users can sort the list by more than just the name column. (BZ#1797931)
- The Pods details page did not include conditions. Without the conditions, it was difficult to know the status of the Pod. There is now a conditions section on the Pod details page and it is easier to discern the status of the Pod. (BZ#1804869)
- The query browser results were rendered with a hard-coded sort. The hard-coded sort could override the sort specified in the query, thus rendering a different result than requested. The hard-coded sort is now removed so the sort specified in the query is preserved. (BZ#1808394)
- Previously, the web console was experiencing runtime errors on certain pages due to the ts-loader using the incorrect **tsconfig.json** in some cases. The ts-loader issue is resolved, allowing all web console pages to load properly. (BZ#1811886)
- When navigating to the **Advanced → Project Details → Inventory** section from the Developer perspective of the web console, DeploymentConfigs were not listed. The DeploymentConfigs are now tracked and are included in the Inventory section of the dashboard. (BZ#1825228)
- Previously, the web console did not display user details when the user name contained special characters such as **#**. The web console now displays user details regardless of special characters in the user name. (BZ#1835460)
- Previously, when an object was edited in the YAML editor, it did not verify the presence of the required **metadata** field. If the field was missing when the object was saved, an error was logged in the browser's JavaScript console, but no visible feedback was provided. Now if the required **metadata** field is missing, the UI presents an actionable error message. (BZ#1787503)

- Previously, when editing an object by using the form view, switching to the YAML editor for the object did not synchronize all existing data. Now, all data is correctly synced between the form view and the YAML editor. (BZ#1796539)

- Previously, when navigating with the tab key, the notification drawer might be triggered and expand. With this bug fix, the notification drawer is not triggered when tabbing through UI elements. (BZ#1810568)

- Previously, when listing existing instances of a custom resource definition (CRD), the wrong API was used to populate the list. Now the correct API is used to populate the list. (BZ#1819028)

- On the **Operators → Installed Operators** page, when viewing the available custom resource (CR) list for a selected Operator, the **Version** column displays the value **Unknown**. Because no version information is available for a CR, this field is now removed from the UI. (BZ#1829052)

- Previously, when completing the Create Operator Subscription form, if the Update Channel field was changed, the target namespace for the subscription was erroneously reset and the form could not be submitted. Now when adjusting the Update Channel, the target namespace value is preserved and the form can be submitted successfully. (BZ#1798851)

- Previously, the metric **openshift_console_operator_build_info** was not properly exposed. With this bug fix, the metric is available in Prometheus. (BZ#1806787)

- Previously, in the administration perspective, when viewing the Workloads tab with a side panel visible, the notification drawer when expanded is hidden beneath the side panel. This bug fix adjusts the CSS **z-index** so that the notification drawer is visible. (BZ#1813052)

- Previously, the OperatorHub was visible in the web console to only cluster administrators. With this update, the web console now shows the OperatorHub to users who are assigned the **aggregate-olm-view** and **aggregate-olm-edit** cluster role bindings. (BZ#1819938)

- Previously on the **Home → Events** page from the Administrator perspective of the web console, the node name did not show for several node events. With this update, all events now correctly link to the corresponding node. (BZ#1809813)

- Previously, the **Home → Overview** menu item from the Administrator perspective of the web console was hidden from users who could not list namespaces, but otherwise have permissions to see cluster metrics. With this update, the **Overview** navigation item is now visible for all users who have authority to view cluster metrics. (BZ#1811757)

- Previously in the OperatorHub on the Installed Operators page, the link to view more APIs for an Installed Operator did not open the correct tab. With this update, the **View x more** link under Provided APIs goes to the Details tab for the Installed Operator. (BZ#1824254)

- Previously in the OperatorHub, overflow of a container background was not hidden in mobile view. This update fixes the gray background and hides the overflow. (BZ#1809812)

- Previously, the **fieldDependency** specDescriptor did not work as expected. As a result, the Control Field did not control the visibility of the Dependent Field. The visibility of the Dependent Field is now correctly enabled or disabled by the Control Field. (BZ#1826074)

- Previously, the default CA certificate was being used inside the console Pod. This bug fix configures the console to use the **default-ingress-cert** ConfigMap if that ConfigMap exists; if it does not exist, the console configures the default CA certificate instead. This allows the default Ingress certificate to be used, if available, to verify access to the routes the Ingress controller creates. (BZ#1824934)

- Previously, when creating a new Alert Receiver, the web console did not indicate that routing labels were required. A red asterisk has been added as a visual indicator that the routing labels are required. (BZ#1803614)

- Previously, the Role Bindings tab in the web console ClusterRole details page could show bindings for a namespaced Role with the same name. The tab now correctly shows only bindings for the ClusterRole. (BZ#1624328)

- Previously, markdown tables for OLM Operators could render poorly when they had a lot of content. The web console has improved the display of these tables and added a horizontal scrollbar, when necessary. (BZ#1831315)

- Previously, when checking all PVCs in the web console, it was hard to distinguish which storage class the PVC belonged to. A PVC Storage Classes column has been added to the web console so it is easier to find storage class info for PVCs. (BZ#1800459)

- Previously, creating a new MachineConfigPool using the console's **Compute → Machine Config Pools → Create Machine Config Pool**button resulted in a MachineConfigPool that did not match the node. This was 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 web console to display a Machine Selector which now matches the appropriate node. (BZ#1813369)

- Previously, **oc** was not listed first on the CLI downloads page because the CLI downloads were listed alphabetically. Because **oc** is the primary CLI for OpenShift Container Platform, it is now listed at the top of the CLI downloads page. (BZ#1824934)

- Previously, the **Explorer** view presented **Access Review** tabs to users who lacked the required permissions to view these tabs. Users without this authorization saw an error message and instructions to try reloading the tab, but retrying would not change the result. With this release, the **Access Review** tabs are hidden from users who do not have permission to view the contents of the tabs. (BZ#1786251)

- Previously, memory consumption data in the **Cluster Utilization** card view and the top consumers popover view was inconsistent because these two views used different methods to calculate memory usage. With this release, the two views use the same method to calculate memory usage so that the data they provide is consistent. (BZ#1812096)

- Previously, users were able to create two routing labels for a single alert receiver. When two routing labels had the same key, the list page only showed the latest created one. However, exactly if one of the routing labels used regular expressions, the details page separated them as two distinct routing labels. With this release, users can no longer create two routing labels for a single alert receiver. (BZ#1804049)

- With this release, an update to a library that is used by the web console resolved performance and display issues on some views. (BZ#1796658)

- Select links in the mast head had an href value of **#** with an OnClick handler containing the target destination. As a result, those links have the option to open in a new tab, however the **#** resolved to the dashboard instead of the intended target destination. Now, any links with an href of **#** are updated to a button element so the **Open Link In New Tab**option is not available. Links that have the **Open Link In New Tab**option show the correct URL. (BZ#1703757)

Monitoring

Previously, mishandling of metadata related to the Prometheus PVC name could cause upgrade failures to or from versions 4.4.0-4.4.8. Now data is copied from old physical volumes to the new ones in order to retain the metric data and allow the upgrades to complete. (BZ#1832124)

Previously, Thanos Querier could be scheduled on both on master and worker nodes, but it is only meant to be scheduled on worker nodes. Now the toleration allowing Thanos Querier to be scheduled on master nodes has been removed, so Thanos Querier is only deployed on worker nodes. (BZ#1812834)

Previously, the evaluation of some Prometheus recording rules occasionally failed and caused metrics to generated from the rule to go missing. Now the recording rules have been fixed. (BZ#1802941)

Previously, the CPU usage rate was showing incorrect results dues to statistical smoothing of the data. Now the method for calculating CPU usage has been updated and the results **oc adm top** are similar to the Linux **top** utility. (BZ#1812004)

Previously, custom configurations to cluster monitoring were being lost because the 'cluster-monitoring-config` map was invalid and the cluster monitoring operator defaulted to use the default configuration. Now when the cluster monitoring operator can not decode the cluster-monitoring-config config map, it does not use the default configuration and fires an alert warning instead. (BZ#1807430)

Networking

- Changes on kube-proxy metrics implementation made some metrics disappear during the Kubernetes 1.17 rebase. This bug fix change how metrics are published in SDN, keeping them from disappearing. (BZ#1811739)
- Previously when the installer introduced **machineNetwork**, the Cluster Network Operator was not modified to add it to **proxy.status.noProxy**. This bug fix set **proxy.status.noProxy** to contain the expected fields, including **machineNetwork**. (BZ#1797894)
- Previously, the node detected its self IP incorrectly preventing it from owning the egressIP it was assigned. This bug fix assigns the node IP from the Kubernetes API instead. (BZ#1802557)
- A code change inadvertently stopped setting the status for third-party plug-ins, which meant the Cluster Network Operator status never indicated that it was working. This bug fix added code to set the status when a third-party plug-in is in use. Now Cluster Network Operator correctly reports status when a third-party plug-in is in use. (BZ#1807611)
- Previously, the Cluster Network Operator on Kuryr bootstrapping had no logic to remove deprecated security group rules when they were replaced by new ones. On OpenShift Container Platform upgrades, the old security group rules were left on the security groups meaning that tightening them to increase security was not done on environments upgraded from 4.3 to 4.4. This bug fix ensures the Cluster Network Operator is removing old security group rules, and as a result the security group rules get removed on 4.3 to 4.4 upgrade and Pods are correctly getting the restricted access to host VMs. (BZ#1832305)
- Previously, in order to enforce a Network Policy that blocks any traffic, the service matched by that policy should have the corresponding load balancer blocking the traffic, and the way Octavia provided this was by using ACLs and setting off the admin state on the load balancer listeners. As a consequence, the mismatch of the security groups on the Kuryr annotation for the OpenShift Container Platform endpoints and the actual security group set for the Pods made some load balancers to be considered for a network policy update, and so having the traffic blocked with the admin state disabled. With this bug fix, the security groups field on the Kuryr annotation for the endpoints match the existent security groups of the selected Pods. Now all load balancer listeners have the admin state enabled if no network policy blocks it. (BZ#1824258)

- Previously, iptables experienced locking problems. In rare circumstances, a Pod could fail to start, and the command **oc describe pod** would show an event including the text, "Failed create pod sandbox ... could not set up pod iptables rules: Another app is currently holding the xtables lock." This bug fix passes **-w** to iptables in the relevant piece of code, and as a result iptables wait for the lock and does not fail spuriously. (BZ#1810505)

- Previously on node deletion, the chassis record for the node would not get removed from the south-bound database. Stale chassis records resulted in stale logical flows for that chassis which were never removed. This bug fix added a node sync mechanism in **ovnkube-master** to purge chassis records of deleted nodes. Now there are no more stale chassis records or stale logical flows corresponding to deleted nodes in the south-bound database. (BZ#1809747)

- When etcd was running slowly, **openshift-sdn** could miss namespace creation events due to a race condition. This could lead to Pods in that namespace having no connectivity. With this bug fix, the race condition was removed. As a result, Pods eventually have connectivity. (BZ#1825355)

Node

- Previously, the **kubepods.slice** memory cgroup was not set to the maximum limit, minus the reservations. This caused the nodes to become overloaded with out of memory errors, and not evict workloads. The **kubepods.slice** memory reservation is now set correctly. (BZ#1800319)

- Previously, the device mapper for devices was missing metrics, so none were available if the system was using a device mapper for the root device. The cadvisor was fixed and metrics are now available whether or not the device mapper is used for the root device. (BZ#1849269)

Node Tuning Operator

- The Node Tuning Operator did not ship with fixes to address tuned daemon behavior related to (BZ#1702724) and (BZ#1774645). As a result, when an invalid profile was specified by the user, a Denial of Service (DoS) of the operand's functionality occurred. Also, correcting the profile did not restore the operand's functionality. This was fixed by applying the aforementioned bug fixes, allowing the tuned daemon to process and set a new, corrected profile. (BZ#1823941)

- Previously, tuned Pods did not mount **/etc/sysctl.{conf,d/}** from the host. This gave the ability for settings provided by the host to be overridden by tuned profiles. Now **/etc/sysctl.{conf,d/}** is mounted from the host in tuned Pods, which prevents tuned profiles from overriding the host sysctl settings in **/etc/sysctl.{conf,d/}**. (BZ#1825322)

oc

- Previously, printer flags were not wired properly and the **oc adm group sync** command was missing output options. The flags are now wired properly and all of the output options are working correctly. (BZ#1828194)

- Previously, the format result function had a hard-coded size, so panic occurred when the array was filled with less than the hard-coded limit. The number of LDAP entries is now limited based on the actual array capacity and the function correctly formats results. (BZ#1806876)

- Previously, the **oc image mirror** command would give an error if only the **--from-dir** option is specified, even though it should override the **--dir** option. Now, **--from-dir** properly overrides **--dir**, and the command succeeds. (BZ#1807807)

- Previously, the help examples for the **oc adm release** command were not displayed correctly. They have been updated so that they now display properly. (BZ#1810310)

OLM

- Custom resources installed by Operator Lifecycle Manager (OLM) are given OwnerReferences to the InstallPlan they were applied from. Deleting an InstallPlan deletes the custom resources that were applied from it. This bug fix updates OLM to point OwnerReferences for custom resources to the CSV that they were installed for. As a result, deleting an InstallPlan no longer deletes the custom resources that were applied from it. (BZ#1808113)

- Previously, the garbage collection resource event queue was not configured correctly. This caused cluster-scoped resources generated for Operators managed by Operator Lifecycle Manager (OLM) to never get cleaned up when the Operator was uninstalled. This bug fix updates OLM to reconfigure garbage collection queues to be hit for owner labels referencing any namespace. As a result, cluster-scoped resources generated for Operators managed by OLM are now properly cleaned up when the Operator is uninstalled. (BZ#1834136)

- If an Operator is being upgraded that provides a required API whose group, version, and kind (GVK) has not changed since the previous version of the Operator, and the Operator that depends on the API uses a **skipRange** instead of the **spec.Replaces** field, Operator Lifecycle Manager (OLM) fails to generate the "upgraded CSV" with the correct **replaces** field. Specifically, OLM would:

  1. Add the new Operator to the generation and mark the APIs it provides as **present**.

  2. Remove the old Operator from the generation and mark the APIs it provides as **absent**, despite being provided by the new version of the Operator.

  3. Attempt to resolve the **missing APIs**, overwriting the new version of the Operator with a copy that does not have its **spec.Replaces** field set.
     This caused certain Operators to fail to upgrade to new versions. This bug fix updates OLM to remove the old Operator from the current generation before adding the new Operator to the generation. As a result, the upgrade succeeds as expected. (BZ#1818788)

- Invalid CatalogSource configurations were causing a nil-pointer exception and a panic. The **catalog-operator** Pod would crash every time an invalid CatalogSource was reconciled. This bug fix adds runtime nil checks and CatalogSource validation. As a result, invalid CatalogSources are given a representative condition, and the **catalog-operator** Pod no longer crashes. (BZ#1817833)

- Operator Lifecycle Manager (OLM) allows users to specify volumes and volumeMounts using the **subscriptionConfig** field of a Subscription. Using this feature updates the Deployment defined in the ClusterServiceVersion (CSV). Occasionally, OLM would not have the Subscription created for a CSV in its cache, and the CSV would be placed in the "installing phase" without creating the Deployment with the volumes or volumeMounts defined in the Subscription. OLM would then be unable to move the CSV into the "Succeeded phase" because the calculated Deployment hash would not equal the actual Deployment hash on the Deployment. This error would not be resolved because OLM does not update or recreate the Deployment in the "installing phase", and the issue would persist until five minutes passed, when OLM would resync CSVs. As a result, OLM would occasionally be delayed while installing CSVs. This bug fix ensures that, if OLM encounters a Deployment hash error when installing a CSV, OLM now recreates the Deployment. As a result, OLM is no longer delayed by an incorrect Deployment hash. (BZ#1826443)

- Previously, Operator Lifecycle Manager (OLM) did not anticipate running multiple APIServices on a single Deployment and only mounted the CA associated with the last APIService created by OLM. This caused OLM to be unable to run multiple APIServices on a single Deployment. This bug fix updates OLM to use the same CA for all APIServices on a single Deployment. As a result, OLM can now run multiple APIServices on a single Deployment. (BZ#1805412)

- Previously, Operator Lifecycle Manager (OLM) did not deprecate the v1alpha2 version OperatorGroup CustomResourceDefinition (CRD) correctly when introducing a structural schema. This caused v1alpha2 OperatorGroups to no longer be supported and they could not be created. This bug fix reintroduces the v1alpha2 OperatorGroup CRD, and as a result, OLM again supports the v1alpha2 OperatorGroup CRDs. (BZ#1798051)

- The application of a newly, non-deterministically resolved set of dependencies was triggered when previously-resolved InstallPlans no longer contained an equivalent set of manifests. When more than one valid set of dependencies for an Operator existed, this caused an equivalent but distinct resolution to sometimes be applied over an existing one. This bug fix adds a **generation** field to the status of the InstallPlan API and increments it upon every resolution, only applying the InstallPlan with the newest status generation. As a result, only one set of dependencies for an Operator exists on the cluster at a given time. (BZ#1784024)

- The **OperatorHub** type definition was missing an additional **+genclient** marker comment required for Kubernetes client generation. This caused the generated client not to be available in the **openshift/client-go** config client. This bug fix adds the missing **+genclient** marker comment to the **OperatorHub** config type, and as a result, the generated client is now available as expected. (BZ#1816483)

openshift-apiserver

- Previously, the OpenShift API server was not available to clients during upgrades, causing failures. Now the OpenShift API server remains available to clients during upgrades. (BZ#1791162)

openshift-controller-manager

- Previously, the client used to create pull secrets for the OpenShift Container Platform internal registry had a low rate limit. If a large number of namespaces were created in a short time window, it would take a long time for image registry pull secrets to be created. The client's rate limit has been increased, so internal registry pull secrets are now created quickly, even with high traffic. (BZ#1785023)

- Previously, metrics such as workqueue_depth did where unavailable in Prometheus metrics. With this bug fix, the missing metrics are now available. (BZ#1825324)

- If an **openshift-controller-manager** Pod failed, no termination message was provided. Now if the Pod terminates, a termination message is provided. (BZ#1804432)

- Previously, metrics were not properly registered for the OpenShift Container Platform control plane. With this bug fix, metrics for the control plane are now available. (BZ#1809699)

- Previously, a pull secret for the internal registry could be orphaned when the associated token was deleted. In this bug fix, a reference is created between a pull secret and token so that a pull secret is no longer orphaned when the associated token is deleted. (BZ#1765294)

- Previously, if OpenShift Container Platform was configured with a global proxy, the proxy was not used when connecting to external image registries. Now when pull images from an external registry, OpenShift Container Platform uses the cluster-wide proxy configuration. (BZ#1805168)

- Previously, during a deployment rolling update the controller might be unavailable for an excessive amount of time. This bug fix minimizes any delay by allowing the controller to proactively release its lease as Pods in the deployment terminate. (BZ#1809719)

- Previously, the **openshift-controller-manager-operator** might potentially run with access to elevated SELinux privileges. With this bug fix, the correct security context constraints are now applied. (BZ#1806913)

- Previously, during an upgrade the **openshift-controller-manager** erroneously reported that the Operator had been upgraded and was available. Now, the Operator correctly reports when it is successfully updated. (BZ#1804434)

- During installation or upgrade, the **openshift-controller-manager** did not correctly report its progress condition. As a result, an installation or upgrade might fail. Now the Operator correctly reports its progress upon a successful installation or upgrade. (BZ#1814446)

- Previously, the **image-resolve-plugin** did not resolve images if the **alpha.image.policy.openshift.io/resolve-names** annotation was added after resource creation. The **image-resolve-plugin** was fixed to resolve images even if the **alpha.image.policy.openshift.io/resolve-names** annotation is added after resource creation. (BZ#1805155)

- Previously, the Controller Manager Operator did not expose its metrics when over an IPv6 cluster. Subsequently, metrics were not being properly scraped, which left users with no way to graph or query performance data. The Controller Manager Operator now properly binds to IPv6 interfaces, so metrics are properly scraped and presented to users. (BZ#1813233)

Routing

- Previously, service load balancers could not include Azure master nodes, which broke ingress on compact clusters where worker nodes are also master nodes. Azure only allows a node's network interface card (NIC) to be associated with a single load balancer at any point in time. With this update, the installer was changed to create a unified load balancer and network security group that are used for both the API and services of type **LoadBalancer**. Now, service load balancers can include master nodes on Azure and ingress works on compact clusters. (BZ#1794839)

- Previously, the openshift-router did not establish a watch of default certificate secret contents if the secret was invalid. Upon starting, the openshift-router failed to read the invalid secret, which must exist for the router Pod to start. As a result, the user had to update the invalid secret and delete the current router Pods. With this update, the router now watches for any changes in the default certificate secret without deleting the router Pods. If the secret is invalid, the router uses and serves the default router certificate. If the secret is valid, the router serves the default certificate from that secret. (BZ#1820400)

- Previously, the ingress operator failed to configure DNS when running in AWS China regions. With this update, the ingress operator now detects when it is running in AWS China regions and can configure DNS in Route 53 API endpoint. (BZ#1805224)

- Previously, the ingress operator continuously upserted DNS records that it managed on Azure and Google Cloud Provider (GCP). With this update, the ingress operator avoids upserting a DNS record if the record is already published and neither the record nor the DNS zone configuration has changed since the controller last upserted the record. The ingress operator now makes fewer calls to the cloud provider API, which might prevent **cloud provider rate limited** events in the **openshift-ingress** namespace. Additionally, the ingress operator logs now show fewer **upserted DNS record** log messages. (BZ#1809354)

- Previously, the ingress-to-route controller used the ingresses resource from the extensions/v1beta1 API group, which was deprecated in Kubernetes 1.18. With this update, the ingress-to-route controller now uses the ingresses resource from the networking.k8s.io/v1beta1 API group. (BZ#1801415)

- Previously, the router was not promoting inactive Routes when a conflicting Route was deleted. Now when a Route is deleted, the router reprocesses all inactive Routes and activates Routes that no longer conflict with the deleted Route. (BZ#1821095)

- Previously, when a Service with type LoadBalancer or an IngressController with the LoadBalancerService endpoint publishing strategy type was deleted, the Service remained present and in a **pending** state. The service controller was changed in OpenShift Container Platform 3.10 to prevent unnecessary GetLoadBalancer cloud-provider API calls when non-LoadBalancer Services were created or deleted. A subsequent change in Kubernetes 1.15 prevented these unnecessary API calls in a different way. As a result, interaction between these two changes broke the service controller's clean-up logic for Services with type LoadBalancer. With this update, the change added in OpenShift Container Platform 3.10 was removed. Deletion of Services with a type LoadBalancer and IngressControllers with a LoadBalancerService type can now complete. (BZ#1798282)

- Previously, a confusing LoadBalancerManager status condition reason was set by the Ingress Operator when the endpoint publishing strategy did not include managing a load balancer. When an IngressController is configured to use an endpoint publishing strategy type other than LoadBalancerService, the ingress operator does not manage a load balancer for that IngressController. With this update, the LoadBalancerManager status condition more clearly states why the operator is not managing a load balancer for the IngressController. The message now does not use phrases such as **unsupported** or **does not support**. (BZ#1826113)

- Previously, a Forwarded HTTP header with a non-standard **proto-version** parameter was added when the ingress controller forwarded an HTTP request to an application. As a result, the Forwarded header was not standards-compliant and might have caused problems when applications tried to parse the header value. With this update, the Forwarded header is now standards-compliant and the ingress controller does not specify a **proto-version** parameter in the Forwarded header. (BZ#1803001)

- Previously, Prometheus counters that show the number of active sessions were preserved across router restarts and increased indefinitely. With this update, **haproxy_frontend_current_session** and **haproxy_server_current_session** now accurately depict the number of active sessions. The value of these counters are now reset upon router restart. (BZ#1832539)

- If the backing Pods of a service exposed via a route are unavailable (e.g., crashlooping, deleted), the router responds with a 503 error. Previously, the **haproxy_server_http_responses_total** metric for that route was no longer available, thus monitoring on the route was no longer possible. With this update, all backend metrics are now reported and users can track when no Pods are up. (BZ#1835845)

Samples

- Previous versions of the Samples Operator did not bootstrap as removed on s390x and ppc64le architectures, although samples content had not been made available on those architectures yet. This caused cluster upgrades on s390x and ppc64le architectures to fail because samples content was expected, although it was not available. Now the Samples Operator is forced to upgrade, even if it does not contain the necessary samples content. This fixes the cluster upgrade failures caused by unavailable sample content for the s390x and ppc64le architectures. (BZ#1835112)

- If a sample imagestream available in a prior OpenShift Container Platform release was removed in a subsequent release, then during upgrade to that subsequent release, the removed imagestream could be incorrectly tracked as needing imagestreamimports to complete. Since no imagestreamimmports were occurring, samples were not reporting their upgrade as complete. This caused the cluster upgrade to fail. The Samples Operator has been updated to

ignore the tracking of imagestreams that existed in a prior release but not in the release for which the upgrade is intended. Now imagestreams removed between releases no longer cause the Samples Operator to fail during upgrade. (BZ#1811143)

- Previously, the Samples Operator would send alerts about an invalid configuration or missing image pull secrets when it was bootstrapped as removed. This caused misleading alerts to users because invalid configurations and missing image pull secrets should not be sent by the Samples Operator if it is removed. The Samples Operator has been updated to not send alerts related to importing samples when it is bootstrapped as removed. (BZ#1813175)

- Previously, sample templates that were available in a prior release and then removed in a subsequent release might be marked as needing updates. Attempting to update the templates resulted in various errors and failure statuses. These templates were updated to not receive updates after their removal. As a result, removed sample templates do not generate errors or failures. (BZ#1828065)

Storage

- Previously, volumes might have failed to provision in certain Azure regions that were created without proper availability zone support. With this fix, availability zone support is now detected during provisioning to enable volume provisioning in all Azure regions. (BZ#1828174)

- Previously, namespaces would get stuck in **Terminating** and VolumeSnapshots would linger on the cluster when a VolumeSnapshotClass was removed before the associated VolumeSnapshots because it was no longer possible to delete the associated resources. With this fix, VolumeSnapshot functionality now examines whether the associated VolumeSnapshotClass has already been deleted so that VolumeSnapshots can be successfully deleted as long as no corresponding VolumeSnapshotClass exists. (BZ#1808123)

- Previously, the CSI Snapshot Controller might crash when VolumeSnapshotContents were nil. The system now checks to see if the VolumeSnapshotContent is nil before it gets used. (BZ#1814280)

- Previously when upgrading the Local Storage Operator, the associated diskmaker and provisioner Pods might both be outdated unless the LocalVolume resource was also modified. With this fix, the DaemonSet's hash is included in an annotation. If the hash does not match, the Pods are deployed so that the diskmaker and provisioner Pods are now successfully updated when the Local Storage Operator is updated. (BZ#1822213)

- Previously, **oc get volumesnapshot** would only display the name and creation time of the resource, and not volumesnapshot status. With this fix, **oc get volumesnapshot** now includes additional details, such as the associated VolumeSnapshotContent, VolumeSnapshot source, and other relevant information. (BZ#1800437)

- Previously, **oc get volumesnapshotclass** would only display the name and creation time of the resource, and not deletion policy or driver information. With this fix, **oc get volumesnapshotclass** now includes additional details, such as the associated CSI Driver and deletion policy. (BZ#1800470)

- Previously, **oc get volumesnapshotcontent** would only display the name and creation time of the resource, and not additional relevant information. With this fix, **oc get volumesnapshotcontent** now includes additional details, such as the associated VolumeSnapshot, VolumeSnapshotClass, and other relevant information. (BZ#1800477)

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

Feature	OCP 4.3	OCP 4.4	OCP 4.5
Precision Time Protocol (PTP)	TP	TP	TP
oc CLI Plug-ins	TP	TP	TP
experimental-qos-reserved	TP	TP	TP
Pod Unidler	TP	TP	GA
Ephemeral Storage Limit/Requests	TP	TP	TP
Descheduler	-	TP	TP
Podman	TP	TP	TP
Sharing Control of the PID Namespace	TP	TP	GA
OVN-Kubernetes Pod network provider	TP	TP	TP
HPA custom metrics adapter based on Prometheus	TP	TP	TP
HPA for memory utilization	TP	TP	TP
Machine health checks	TP	GA	GA
Three-node bare metal deployments	TP	TP	GA
Helm CLI	TP	GA	GA
Service Binding	TP	TP	TP
Log forwarding	TP	TP	TP

Feature	OCP 4.3	OCP 4.4	OCP 4.5
User workload monitoring	TP	TP	TP
OpenShift Serverless	TP	GA	GA
Compute Node Topology Manager	TP	TP	GA
Raw Block with Cinder	TP	TP	TP
External provisioner for AWS EFS	TP	TP	TP
CSI volume snapshots	-	TP	TP
CSI volume cloning	-	TP	GA
CSI AWS EBS Driver Operator	-	-	TP
OpenStack Manila CSI Driver Operator	-	-	GA
CSI inline ephemeral volumes	-	-	TP
OpenShift Pipelines	-	TP	TP
Vertical Pod Autoscaler	-	-	TP
Operator API	-	-	TP
OpenShift Virtualization	TP	TP	GA

1.7. KNOWN ISSUES

- When upgrading to a new OpenShift Container Platform z-stream release, connectivity to the API server might be interrupted as nodes are upgraded, causing API requests to fail. (BZ#1845411)
- When upgrading to a new OpenShift Container Platform z-stream release, connectivity to routers might be interrupted as router Pods are updated. For the duration of the upgrade, some applications might not be consistently reachable. (BZ#1809665)
- When upgrading to a new OpenShift Container Platform release with the default CNI network provider set to OVN-Kubernetes, the upgrade fails and the cluster is left in an unusable state. (BZ#1854175)
- Because the **ImageContentSourcePolicy** for image registry pull-through is not yet supported, the deployment Pod cannot mirror images by using a digest ID if the imagestream has the pull-through policy enabled. In this case, an **ImagePullBackOff** error displays. (BZ#1787112)
- If you scale up with a RHEL worker while running a cluster on RHOSP that uses user-provisioned

infrastructure, all routes are inaccessible if the Ingress port VIP is on the RHEL worker. As a workaround, you must reschedule the router Pod to an RHCOS node and make the Ingress VIP migrate to the RHCOS node. To do this, add the **node.openshift.io/os_id: rhcos** label to the Ingress Controller before upgrade:

```
$ oc -n openshift-ingress-operator edit ingresscontroller/default -o yaml
spec:
  nodePlacement:
    nodeSelector:
      matchLabels:
        kubernetes.io/os: linux
        node-role.kubernetes.io/worker: ""
        node.openshift.io/os_id: rhcos
```

(BZ#1848945)

- The Che Workspace Operator was updated to use the DevWorkspace custom resource instead of the Workspace custom resource. However, the OpenShift web terminal continues to use the Workspace custom resource. Because of this, the OpenShift web terminal fails to work with the latest version of the Che Workspace Operator. (BZ#1846969)

- A **basic-user** is unable to view the **Dashboard** and **Metrics** tabs in the **Monitoring** view of the **Developer** perspective. (BZ#1846409)

- In the **Topology** view, when you right-click a Knative service, the **Edit Application Grouping** option is displayed twice in the context menu. (BZ#1849107)

- The Special Resources Operator (SRO) cannot be deployed successfully on OpenShift Container Platform 4.5. This prevents the deployment of NVIDIA drivers, which are required by the cluster to run workloads requiring GPU resources. Also, the Topology Manager feature could not be tested with GPU resources as a result of this known issue. (BZ#1847805)

- The web console includes the option to create VM vNICS with a SLIRP binding, but this is not supported. Attempting to use this option will cause the VM to fail to boot. Do not select this option. (BZ#1828744)

- There is an issue where Pods that use the OpenShift SDN default CNI network provider in a node can lose network communication, causing the Pods to crash. This can sometimes happen when upgrading a cluster. As a workaround, you can delete and re-create the Pods. (BZ#1855118)

- There is a known issue where the custom pool is not supported on the master node. The command **oc label node** applies the new custom role to the target master node, but the Machine Config Operator does not apply changes specific to the custom pool. This results in an error, which can be seen in the Machine Config Controller Pod logs. As a suggested workaround to ensure that control plane nodes remain stable, it is recommended to not apply multiple roles on master. (BZ#1797687)

- The logging performance for clusters is degraded compared to past versions of OpenShift Container Platform. This is being actively investigated and will be updated in a future release of OpenShift Container Platform. (BZ#1833486)

- You might receive a message that the system is unable to mount a volume when the volume contains a large number of files. This can happen when a Pod mounts a volume that is set with **FSGroup SecurityContext** because the GID ownership of the files must be recursively updated

for all files on the volume. Users should expect that Pods using volumes with a large number of files and the **FSGroup SecurityContext** setting can take considerable time to start. (BZ#1515907)

- Running Pods with frequent probes can cause the number of conmon processes to grow quickly. A conmon process is a program that detaches from its parent, CRI-O, and is used to exec the container runtime. If the probes happen frequently enough, systemd has trouble reaping all of its new children, and some conmon processes can become zombies. (BZ#1852064)

- On Microsoft Azure when upgrading from 4.4 to 4.5, the Ingress Operator can fail to ensure a DNSRecord due to errors refreshing the token. Restarting the Ingress Operator fixes the issue. (BZ#1854383)

- When running OpenShift Container Platform on Azure with installer-provisioned infrastructure, there is a known issue where **oc** commands fail intermittently with TLS handshake timeout errors. (BZ#1851549)

- For clusters on VMware vSphere instances using installer-provisioned infrastructure, bootstrap workers fail. The default resource pool resolves to multiple instances. (BZ#1852545)

- There is an issue where the Machine Config Operator (MCO) becomes degraded during the installation of an OpenShift Container Platform cluster. This is caused by a MachineConfig ordering problem during the bootstrap process. As a workaround, you must prefix any custom MachineConfig file with a differing priority of **98-** instead of **99-**. (BZ#1826150)

- Git clone operations that go through an HTTPS proxy fail. Non-TLS (HTTP) proxies can be used successfully. (BZ#1750650)

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

- An issue has been found for the s390x and ppc64le architectures that renders nodes unavailable for workloads after a forced reboot or power down. Do not force reboot or power down nodes.
  If a forced reboot or power down is unavoidable and a node that comes back up is unavailable for workloads:

  1. SSH into the node.
  2. Stop the CRI-O and kubelet services.
  3. Run the command **rm -rf /var/lib/containers**.
  4. Restart the CRI-O and kubelet services.
     (BZ#1858411)

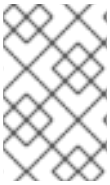
- If an AWS account is configured to use AWS Organizations service control policies (SCPs) that use a global condition to deny all actions or require a specific permission, OpenShift Container Platform AWS installations fail, even if the provided credentials have the required permissions for installation.
  A workaround for this issue is introduced in OpenShift Container Platform 4.5.8. (BZ#1829101)

1.8. ASYNCHRONOUS ERRATA UPDATES

Security, bug fix, and enhancement updates for OpenShift Container Platform 4.5 are released as asynchronous errata through the Red Hat Network. All OpenShift Container Platform 4.5 errata is

available on the Red Hat Customer Portal . See the OpenShift Container Platform Life Cycle for more information about asynchronous errata.

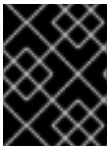
Red Hat Customer Portal users can enable errata notifications in the account settings for Red Hat Subscription Management (RHSM). When errata notifications are enabled, users are notified via email whenever new errata relevant to their registered systems are released.



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 4.5. Versioned asynchronous releases, for example with the form OpenShift Container Platform 4.5.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.

1.8.1. RHBA-2020:2409 - OpenShift Container Platform 4.5 image release and bug fix advisory

Issued: 2020-07-13

OpenShift Container Platform release 4.5 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:2409 advisory. The RPM packages included in the update are provided by the RHBA-2020:2408 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 4.5.1 container image list

1.8.2. RHSA-2020:2412 - Moderate: OpenShift Container Platform 4.5 security update

Issued: 2020-07-13

Container image updates are now available for OpenShift Container Platform 4.5. Details of the updates are documented in the RHSA-2020:2412 advisory.

1.8.3. RHSA-2020:2413 - Moderate: OpenShift Container Platform 4.5 security update

Issued: 2020-07-13

A package update is now available for OpenShift Container Platform 4.5. Details of the update are documented in the RHSA-2020:2413 advisory.

1.8.4. RHBA-2020:2909 - OpenShift Container Platform 4.5.2 bug fix update

Issued: 2020-07-16

OpenShift Container Platform release 4.5.2 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:2909 advisory. The RPM packages included in the update are provided by the RHBA-2020:2908 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 4.5.2 container image list

1.8.4.1. Bug Fixes

- Upgrades to OpenShift Container Platform 4.5.1 failed on nodes with Secure Boot configured. For clusters configured with Secure Boot, one node from both the control plane and compute Machine Config Pools failed to reboot, which caused the Machine Config Operator (MCO) to be degraded. The cluster subsequently failed to upgrade. The issue is not present in this release. (**BZ#1856501**)

1.8.5. RHBA-2020:2956 - OpenShift Container Platform 4.5.3 bug fix update

Issued: 2020-07-22

OpenShift Container Platform release 4.5.3 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:2956 advisory. The RPM packages included in the update are provided by the RHBA-2020:2955 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 4.5.3 container image list

1.8.5.1. Bug Fixes

- Previously, an issue caused nodes to become unavailable for workloads after a forced reboot or power down. This has been fixed. (**BZ#1857224**)

- Previously, the web console would choose Operator icons to display in OperatorHub by returning the icon from the first channel declared in the package. This sometimes caused the displayed icon to be different than the latest icon published to the package. This has been fixed by choosing the icon from the default channel, which ensures the latest icon is displayed. (**BZ#1844588**)

- Previously, the container image signature policy used in OpenShift Container Platform builds did not contain any configuration for local images. When only allowing images from specific registries, postCommit scripts in builds failed because it was not allowed to use local images. The container image signature policy has been updated to always allow images that reference local storage layers directly. Now builds can successfully complete if they contain a postCommit hook. (**BZ#1849173**)

1.8.6. RHBA-2020:3028 - OpenShift Container Platform 4.5.4 bug fix update

Issued: 2020-07-30

OpenShift Container Platform release 4.5.4 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:3028 advisory. The RPM packages included in the update are provided by the RHBA-2020:3027 and RHEA-2020:3208 advisories.

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 4.5.4 container image list

1.8.6.1. Features

1.8.6.1.1. IBM Z and LinuxONE

With this release, IBM Z and LinuxONE is now compatible with OpenShift Container Platform 4.5. See Installing a cluster on IBM Z and LinuxONE for installation instructions.

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:
  - OpenShift virtualization
  - Log forwarding
  - Precision Time Protocol (PTP) hardware
  - CSI volume snapshots
  - OpenShift Pipelines
- The following OpenShift Container Platform features are unsupported:
  - Red Hat OpenShift Service Mesh
  - OpenShift Do (odo)
  - CodeReady Containers (CRC)
  - OpenShift Container Platform Metering
  - Multus CNI plug-in
  - OpenShift Container Platform upgrades phased rollout
  - FIPS cryptography
  - Encrypting data stored in etcd
  - Automatic repair of damaged machines with machine health checking
  - Tang mode disk encryption during OpenShift Container Platform deployment
  - OpenShift Serverless
  - Helm command-line interface (CLI) tool

    - Controlling overcommit and managing container density on nodes
    - etcd cluster operator
    - CSI volume cloning
- Worker nodes must run Red Hat Enterprise Linux CoreOS (RHCOS).
- Persistent shared storage must be of type Filesystem: NFS.
- These features are available for OpenShift Container Platform on IBM Z for 4.5, but not for OpenShift Container Platform 4.5 on x86:
    - HyperPAV enabled on IBM System Z for the virtual machine for FICON attached ECKD storage.

1.8.6.1.2. IBM Power Systems

With this release, IBM Power Systems are now compatible with OpenShift Container Platform 4.5. See Installing a cluster on IBM Power or Installing a cluster on IBM Power in a restricted network .

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

- OpenShift Container Platform for IBM Power Systems does not include the following Technology Preview features:
    - Container-native virtualization (CNV)
    - OpenShift Serverless
- The following OpenShift Container Platform features are unsupported:
    - Red Hat OpenShift Service Mesh
    - OpenShift Do (odo)
    - CodeReady Containers (CRC)
    - OpenShift Pipelines based on Tekton
    - OpenShift Container Platform Metering
    - SR-IOV CNI plug-in
- Worker nodes must run Red Hat Enterprise Linux CoreOS (RHCOS).
- Persistent storage must be of the **Filesystem** mode using local volumes, Network File System (NFS), OpenStack Cinder, or Container Storage Interface (CSI).
- Networking must use either DHCP or static addressing with Red Hat Openshift SDN.

Supported Features

- Currently, 3 operators are supported:
    - Cluster-Logging-Operator

- Cluster-NDF-Operator
- Elastic Search-Operator

1.8.6.2. Bug Fixes

- Previously, the action menu for an operand on the operand list could close immediately after opening. This behavior was observed when clicking the tab for an Operator-provided API on the **Installed Operators → Operator Details** page. The menu now functions correctly and does not close without user interaction. (BZ#1842717)
- Previously, when filtering the OperatorHub catalog in the web console, some Operator icons did not appear until the user scrolled down on the page. With this release, the icons appear immediately when filtered. (BZ#1844503)
- Previously, the quota gauge charts on the **Resource Quota Details** page of the web console rendered with a width of zero and were not visible. The issue has been resolved in this release. (BZ#1845125)
- Previously, clicking **Create EtcdRestore** on the **EtcdRestores** page caused the web console to stop responding. With this release, the **Create EtcdRestore** form view workflow loads correctly. (BZ#1847277)
- Previously, in the **Create Knative Serving** form view workflow for the OpenShift Serverless Operator, some fields that should only accept numeric characters accepted non-numeric characters. The issue has been resolved in this release. (BZ#1847283)
- Previously, clicking **Create** on the **Create ManilaDriver** form view workflow for the Manila CSI Driver Operator did not create a ManilaDriver instance or any response in the web console. The issue has been resolved in this release. (BZ#1853274)

1.8.6.3. Upgrading

To upgrade an existing OpenShift Container Platform 4.5 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.8.7. RHSA-2020:3207 - Moderate: OpenShift Container Platform 4.5 security update

Issued: 2020-07-30

An update for **jenkins-2-plugins** is now available for OpenShift Container Platform 4.5. Details of the update are documented in the RHSA-2020:3207 advisory.

1.8.8. RHBA-2020:3188 - OpenShift Container Platform 4.5.5 bug fix update

Issued: 2020-08-10

OpenShift Container Platform release 4.5.5 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:3188 advisory. The RPM packages included in the update are provided by the RHBA-2020:3189 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 4.5.5 container image list

1.8.8.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.5 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.8.9. RHBA-2020:3330 - OpenShift Container Platform 4.5.6 bug fix update

Issued: 2020-08-17

OpenShift Container Platform release 4.5.6 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:3330 advisory. The RPM packages included in the update are provided by the RHBA-2020:3331 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 4.5.6 container image list

1.8.9.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.5 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.8.10. RHSA-2020:3453 - Important: OpenShift Container Platform 4.5 security update

Issued: 2020-08-17

An update for **jenkins-2-plugins** and **python-rsa** is now available for OpenShift Container Platform 4.5. Details of the update are documented in the RHSA-2020:3453 advisory.

1.8.11. RHBA-2020:3436 - OpenShift Container Platform 4.5.7 bug fix update

Issued: 2020-08-24

OpenShift Container Platform release 4.5.7 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:3436 advisory. The RPM packages included in the update are provided by the RHBA-2020:3437 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 4.5.7 container image list

1.8.11.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.5 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.8.12. RHSA-2020:3519 - Important: OpenShift Container Platform 4.5 security update

Issued: 2020-08-24

An update for **jenkins** and **openshift** is now available for OpenShift Container Platform 4.5. Details of the update are documented in the RHSA-2020:3519 advisory.

1.8.13. RHSA-2020:3520 - Moderate: OpenShift Container Platform 4.5 security update

Issued: 2020-08-24

An update for **openshift-enterprise-hyperkube-container** is now available for OpenShift Container Platform 4.5. Details of the update are documented in the RHSA-2020:3520 advisory.

1.8.14. RHBA-2020:3510 - OpenShift Container Platform 4.5.8 bug fix update

Issued: 2020-09-08

OpenShift Container Platform release 4.5.8 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:3510 advisory. The RPM packages included in the update are provided by the RHBA-2020:3511 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 4.5.8 container image list

1.8.14.1. Features

1.8.14.1.1. Added **projectID** field for network interfaces

The new **projectID** field is now available to configure in a MachineSet custom resource under **.spec.template.spec.providerSpec.networkInterfaces**. This field allows machines to be booted in shared VPCs.

```
...
providerSpec:
  ...
  networkInterfaces:
    - network: <infrastructureID>-network
      subnetwork: <infrastructureID>-<role>-subnet
      projectID: <projectID>
...
```

For more information, see BZ#1868751.

1.8.14.1.2. Added **credentialsMode** parameter to bypass inaccurate AWS permissions validation

For AWS installations, OpenShift Container Platform depends on an AWS policy simulator API to validate permissions. If an AWS account is configured to use AWS Organizations service control policies (SCPs), permissions are checked against the policies that are set in the SCPs. When SCPs include policies that use a global condition to deny all actions or require a specific permission, the policy simulator API does not correctly validate permissions. For example, policies with conditions such as for all regions except **us-east-1** and **us-west-2**, or for all roles except **role-xyz**, cause the AWS API to return

false negatives. When the permissions cannot be validated, OpenShift Container Platform AWS installations fail, even if the provided credentials have the required permissions to install OpenShift Container Platform.

With this release, you can bypass the policy simulator permissions check by setting a value for the **credentialsMode** parameter in the **install-config.yaml** configuration file.

Example install-config.yaml configuration file

```
apiVersion: v1
baseDomain: cluster1.example.com
credentialsMode: Mint (1)
compute:
- architecture: amd64
  hyperthreading: Enabled
...
```

(1) This line is added to set the **credentialsMode** parameter to **Mint**.

Setting a value for **credentialsMode** bypasses the permissions check for AWS accounts configured to use SCPs and allows the installation to proceed. When bypassing this check, ensure that the credentials you provide have the permissions that are required for the specified mode.

The value of **credentialsMode** changes the behavior of the Cloud Credential Operator (CCO) as follows:

- **Mint** - The CCO uses the provided admin-level cloud credential to run the installer. If the credential is not removed after installation, it is stored and used by the CCO to process **CredentialRequests** in the cluster and create new users for each with specific required permissions.
- **Passthrough** - The CCO uses the provided non-admin cloud credential that has enough permissions to perform the installation to run the installer. For more information about locating the permissions specified in the **CredentialRequests** for the version of OpenShift Container Platform being installed, see Manually creating IAM for AWS.

1.8.14.2. Bug fixes

- Previously, intermittent API server errors were reported for the **ImageChangesInProgress** condition instead of the **SamplesExists** condition of the Samples Operator config object. When the API server reported that all samples were installed, the Samples Operator failed to switch the **Progressing** condition to **false** because there was unexpected data in its **ImageChangesInProgress** condition. This incorrectly caused upgrades to be marked as incomplete. This bug fix updates the **SamplesExists** condition to report errors on the API server, so upgrades are no longer blocked if intermittent API server errors occur while the Samples Operator is upgrading. (BZ#1857201)
- Previously, the **ironic-image** container configuration was missing the setting to enable the **idrac-redfish-virtual-media** boot driver. Because of this, users were unable to select the **idrac-virtual-media** boot URL for Metal3. The missing **ironic-image** container configuration is now included, so users are able to select the **idrac-virtual-media** URL for Metal3. (BZ#1859488)
- Previously, the Operand form array and object fields did not have logic to retrieve and show field descriptions on the form. As a result, descriptions were not rendered for array or object type fields. This bug fix adds logic to now display array and object field descriptions on the

Operand creation form. (BZ#1861433)

- Previously, Buildah erased image architecture and OS fields on images. This caused common container tools to fail because the resulting images could not identify their architecture and OS. This bug fix prevents Buildah from overwriting the image and architecture unless there are explicit overrides. This ensures that images always have architecture and OS fields, and the image mismatch warning does not appear. (BZ#1868401)

- Previously, intermittent invalid memory address or nil pointer dereference errors occurred and were followed by timeouts for Kube API access when running CoreDNS 1.6.6. This is now fixed by correctly handling errors with Endpoint Tombstones. Now CoreDNS behaves as intended without intermittent panics. (BZ#1869309)

- Previously, the controller for BareMetalHost objects mirrored status data to an annotation, including a timestamp of the latest status update. This was not needed by the cluster. This could result in the BareMetalHost entering a state of continuous flux where affected BareMetalHosts would be subject to longer back-offs between reconciliation to prevent the controller from overwhelming the Kubernetes API. The annotation causing the problem is no longer written, which fixes the issue. (BZ#1851531)

- Previously, the Cluster Version Operator (CVO) was not syncing the **shareProcessNamespace** parameter in the Pod spec, which caused the Registry Operator to not update the **shareProcessNamespace** setting. The CVO now syncs **shareProcessNamespace**, **DNSPolicy**, and **TerminationGracePeriodSeconds**, fixing the Registry Operator update issues. (BZ#1868478)

1.8.14.3. Upgrading

To upgrade an existing OpenShift Container Platform 4.5 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.8.15. RHSA-2020:3578 - Moderate: OpenShift Container Platform 4.5 security update

Issued: 2020-09-08

An update for **cluster-network-operator-container**, **cluster-version-operator-container**, **elasticsearch-operator-container**, **logging-kibana6-container**, and **ose-cluster-svcat-controller-manager-operator-container** is now available for OpenShift Container Platform 4.5. Details of the update are documented in the RHSA-2020:3578 advisory.

1.8.16. RHBA-2020:3618 - OpenShift Container Platform 4.5.9 bug fix update

Issued: 2020-09-14

OpenShift Container Platform release 4.5.9 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:3618 advisory. The RPM packages included in the update are provided by the RHBA-2020:3619 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 4.5.9 container image list

1.8.16.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.5 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.8.17. RHBA-2020:3719 - OpenShift Container Platform 4.5.11 bug fix update

Issued: 2020-09-21

OpenShift Container Platform release 4.5.11 is now available. The list of container images and bug fixes included in the update are documented in the RHBA-2020:3719 advisory. The RPM packages included in the update are provided by the RHBA-2020:3720 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 4.5.11 container image list

1.8.17.1. Upgrading

To upgrade an existing OpenShift Container Platform 4.5 cluster to this latest release, see Updating a cluster by using the CLI for instructions.

1.8.18. RHSA-2020:3780 - Moderate: OpenShift Container Platform 4.5 security update

Issued: 2020-09-21

An update for **ose-cluster-svcat-apiserver-operator-container** is now available for OpenShift Container Platform 4.5. Details of the update are documented in the RHSA-2020:3780 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 4.5 cluster, all masters must be 4.5 and all nodes must be 4.5. If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.5. 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.