# OpenShift Container Platform 3.5

## Release Notes

Last Updated: 2019-02-23

## Legal Notice



## Abstract

# Table of Contents

# CHAPTER 1. OVERVIEW

The following release notes for OpenShift Container Platform 3.5 summarize all new features, major corrections from the previous version, and any known bugs upon general availability.

## 1.1. 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).

The OpenShift Container Platform version must match between master and node hosts, excluding temporary mismatches during cluster upgrades. For example, in a 3.5 cluster, all masters must be 3.5 and all nodes must be 3.5. However, OpenShift Container Platform will continue to support older **oc** clients against newer servers. For example, a 3.4 **oc** will work against 3.3, 3.4, and 3.5 servers.

Changes of APIs for non-security related reasons will involve, at minimum, two minor releases (3.1 to 3.2 to 3.3, for example) to allow older **oc** to update. Using new capabilities may require newer **oc**. A 3.2 server may have additional capabilities that a 3.1 **oc** cannot use and a 3.2 **oc** may have additional capabilities that are not supported by a 3.1 server.

**Table 1.1. Compatibility Matrix**

| | **X.Y** (**oc** Client) | **X.Y+N** [a] (**oc** Client) |
|---|---|---|
| **X.Y** (Server) | 1 | 3 |
| **X.Y+N** [a] (Server) | 2 | 1 |

[a] Where **N** is a number greater than 1.

1 Fully compatible.

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

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

# CHAPTER 2. OPENSHIFT CONTAINER PLATFORM 3.5 RELEASE NOTES

## 2.1. OVERVIEW

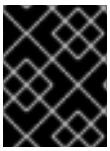
Red Hat OpenShift Container Platform provides developers and IT organizations with a cloud application platform for deploying new applications on secure, scalable resources with minimal configuration and management overhead. OpenShift Container Platform supports a wide selection of programming languages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

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

## 2.2. ABOUT THIS RELEASE

Red Hat OpenShift Container Platform version 3.5 (RHBA-2017:0884) is now available. This release is based on OpenShift Origin 1.5. New features, changes, bug fixes, and known issues that pertain to OpenShift Container Platform 3.5 are included in this topic.

OpenShift Container Platform 3.5 is supported on RHEL 7.2 and 7.3 with the latest packages from Extras, including Docker 1.12.



**IMPORTANT**

OpenShift Container Platform 3.5 is not certified with older versions of Docker and RHEL 7.1 or earlier.

TLSV1.2 is the only supported security version in OpenShift Container Platform version 3.4 and later. You must update if you are using TLSV1.0 or TLSV1.1.

For initial installations, see the Installing a Cluster topics in the Installation and Configuration documentation.

To upgrade to this release from a previous version, see the Upgrading a Cluster topics in the Installation and Configuration documentation.

## 2.3. NEW FEATURES AND ENHANCEMENTS

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

### 2.3.1. Container Orchestration

#### 2.3.1.1. Shared Memory for Containers within the Same Pod

When running more than one container in the same pod, those containers can now share memory segments because ***/dev/shm*** is shared across containers in a pod. ***/dev/shm*** is limited to 64 MB and charged under the **`memcg cgroup`**.

#### 2.3.1.2. Ability to Refer to External Endpoints by Name

It is very common to add an external service to your application. This is a service that does not live on the OpenShift Container Platform. Before this feature, you had to add that endpoint to your application via IP address. These services often had pools of IPs or required access via fully qualified domain name (FQDN), which is similar to SNI in OpenShift Container Platform's load balancer. Now, OpenShift Container Platform has a new **`ExternalName`** type that will accept an FQDN.

#### 2.3.1.3. Setting a NodeSelector and Whitelisting Labels

OpenShift Container Platform 3.5 now supports setting a **`nodeSelector`** on a namespace, with the ability to whitelist labels during the selection. In **`cpodNodeSelectorPluginConfig`**, there are four added levels of logic to allow for various control of the selection process:

1. **`clusterDefaultNodeSelector: region=west`** - The default global (cluster level) list of labels (L1).
2. **`ns1: os=centos,region=west`** and **`ns2: os=rhel`** - The whitelist of labels per namespace (L2).
3. **`scheduler.alpha.kubernetes.io/node-selector`** - The default list of labels per namespace (L3).
4. **`podSpec.nodeSelector`** - The list of labels per pod (L4).

#### 2.3.1.4. Kubelet Directory Should Not Share SELinux Labels with its Containers

Previously, the kubelet directory had to be labeled **`svirt_sandbox_file_t`**. This widens the attack surface available to containers escaping namespace isolation. Now, the kubelet directory has an SELinux context and the kubelet cannot share SELinux labels that are known, preventing the intruder from doing more damage. With OpenShift Container Platform 3.5, the use of **`svirt_sandbox_file_t`** is removed on the kublet directory and ***openshift.local.volumes*** directory. The need to carry a matching SElinux MAC is removed through the use of a **`:Z`** bind mount flag for container directories.

#### 2.3.1.5. Horizontal Pod Autoscaler Ignores Pod Startup Spikes

The Horizontal Pod Autoscaler (HPA) now ignores the CPU usage for pods that are not marked as **`ready`**, so that any extra CPU usage caused by initialization is not the cause for a scale-up. Pods in the **`unready`** state will have zero CPU usage when scaling up, and the HPA ignores them when scaling down. Pods without known metrics will have a 0% CPU usage when scaling up, and 100% CPU when scaling down. This allows for more stability during the HPA decision.

#### 2.3.1.6. Control Over Multiple Secret File Permissions in a Volume

Many applications require that permissions on the mounted file (that hold their configuration) be only owner readable. Users can now specify the mode for the mount point in the downward API annotation or the **`configMap`**.

#### 2.3.1.7. Kubelet Removes Memory-backed Volumes Upon Pod Termination

Previously, when a pod terminated, volumes used by **`emptyDir`**, **`configMaps`**, and secrets on the node were kept. Now, the kubelet deletes memory-backed volumes when the associated pods enter into terminated mode. This makes the vectors less likely to be used as an attack vector.

#### 2.3.1.8. Rate Limiting Controller Retries to Improve Cluster Stability

Controllers on the master can become unstable or cause resource issues when they continuously retry on unresponsive endpoints in the cluster. OpenShift Container Platform 3.5 introduces logic to the controllers that controls how many times they retry calls, and intelligently backs them off so that they do not hang.

These controllers now contain the logic:

- replication controller
- replica set
- daemonset
- certificates
- deployments
- endpoints
- pod disruption budget
- jobs

#### 2.3.1.9. Kubelet Collection of Node Attributes for Scheduling Considerations (Technology Preview)

The kubelet is now able to collect any attribute on the node for scheduling considerations. This feature is currently in Technology Preview.

The cluster operator must advertise a per-node opaque resource on one or more nodes. Users must request the opaque resource in pods. To advertise a new opaque integer resource, the cluster operator should submit a PATCH HTTP request to the API server to specify the available quantity in the **`status.capacity`** for a node in the cluster. After this operation, the node's **`status.capacity`** will include a new resource. The **`status.allocatable`** field is updated automatically with the new resource asynchronously by the kubelet.

See Opaque Integer Resources for more information.

#### 2.3.1.10. StatefulSets (Technology Preview)

**`StatefulSets`** (currently in Technology Preview and formerly known as **`PetSets`**) offer more control over scale, network naming, handling of PVs, and deployment sequencing.

This new controller allows for the deployment of application types that require changes to their configuration or deployment count (instances) to be done in a specific and ordered manner.

Supported:

- Declaration of the Ordinal Index.
- Stable network ID nomenclature.
- Controlled or manual handling of PVs.
- Sequence control at deployment time.
- Ordered control during scale up or scale down, based on instance status.

Not Supported:

- Slow to iterate through the Ordinal Index and, therefore, slow on scale up and scale down.
- No deployment or pod specification post deployment verification of what is deployed versus what is configured in the JSON file.
- Locality awareness of zones or regions when dealing with scale up or scale down ordinality changes or mounted PVs.

**IMPORTANT**

If you have any existing **`PetSets`** in your cluster, you must remove them before upgrading to OpenShift Container Platform 3.5. Automatically migrating **`PetSets`** to **`StatefulSets`** in OpenShift Container Platform 3.5 is not supported. Follow the instructions for manually migrating **`PetSets`** to **`StatefulSets`**.

See more information about web console enhancements related to this feature for OpenShift Container Platform 3.5.

### 2.3.2. Registry

OpenShift Container Platform now allows control of whether or not an image is cached locally in the internal OpenShift Container Registry via the **`oc tag`** command with the **`--reference-policy=local`** and **`--scheduled=true`** options.

The storage of the manifest is moved to the OpenShift Container Registry, instead of storing it in etcd. There are two processes that will clean up existing images' metadata from etcd:

- **`push`** and **`prune`** will gradually migrate all etcd images to not have the manifest attached.
- Use a provided script manually to do them all at once.

Create an image stream from a Docker image and tell it to store locally in the internal OpenShift Container Platform registry:

```
$ oc tag --reference-policy=local --source=docker docker.io/image:tag
myimagestream:tag
```

Schedule the image stream to track new image changes in the external registry:

```
$ oc tag --scheduled=true --source=docker docker.io/image:tag
myimagestream:tag
```

See Extended Registry Configuration for more information.

### 2.3.3. Platform Management

#### 2.3.3.1. Application Service Certificate Regeneration (Technology Preview)

Application service certificate regeneration is currently in Technology Preview.

The controller will now look over the expiry of application certificates that have used the **service.alpha.openshift.io/serving-cert-secret-name** API and regenerate them.

Set the **service.alpha.openshift.io/serving-cert-secret-name** to the name you want to use for your secret. Then, your **PodSpec** can mount that secret. When it is available, your pod will run. The certificate will be good for the internal service DNS name, **<service.name>.<service.namespace>.svc**. The certificate and key are in PEM format, stored in ***tls.crt*** and ***tls.key***, respectively.

```
$ oc get secret ssl-key -o yaml

kind: Secret
metadata:
  annotations:
    service.alpha.openshift.io/expiry: 2017-03-19T08:07:07Z
```

When the regenerator finds a certificate that does not have the expiry annotation, it will regenerate as well. However, the existing secret is not invalidated. Therefore, no manual intervention is required to get the regeneration behavior.

See Service Serving Certificate Secrets for more information.

#### 2.3.3.2. Configurable Expiry Range for Framework Certificates

By default, the certificates used to govern the etcd, master, and kubelet expire after two to five years. There is now an **oc** command to change this expiry to be end-user configurable. This has not been implemented in the Ansible installer yet.

Use the **oc adm ca** command, specifying a validity period greater than two years:

```
# oc adm ca create-master-certs --hostnames=example.org --signer-expire-
days=$[365*2+1]`
```

See Creating New Configuration Files for more information.

#### 2.3.3.3. can-i Command and scc-review Command Options

The **can-i** and **scc-review** command options allow users to better understand their permissions and security context constraints (SCC) setting in their projects. Users see a list of the commands they are allowed to execute.

The **can-i** command option tests scopes in terms of the user and role. The **scc-review** command option checks which **ServiceAccount** can create a pod.

**scc-subject-review** can check whether a user or a **ServiceAccount** can create a pod.

List which permissions a particular user or group has in the project by project administrator:

```
$ oc policy can-i --list --user=**
$ oc policy can-i --list --groups=**
```

List which permissions a particular user or group has in the project by system administrator role:

```
$ oc policy can-i --list --user=** -n <project>
$ oc policy can-i --list --groups=**  -n <project>
```

Determine if users can have all the combination of verbs and resources from **`oc policy can-i --list [--user|--groups]`**

```
$ oc policy can-i <verb> <resource> --[--user|--groups]
```

Test the SCCs with scopes: **`oc policy can-i [--user|--groups]`**

```
$ oc policy can-i <verb> <resource> [--user|--groups] --scopes=user:info
$ oc policy can-i <verb> <resource> [--user|--groups] --
scopes=user:info,role:admin:<namespace>
$ oc policy can-i <verb> <resource> [--user|--groups] --scopes=role:view:*
$ oc policy can-i <verb> <resource> [--user|--groups] --scopes=role:edit:*
$ oc policy can-i <verb> <resource> [--user|--groups] --
scopes=role:admin:*
$ oc policy can-i <verb> <resource> [--user|--groups] --
scopes=role:admin:*:!
```

Test with the **`ignore-scopes`** flag in the **`oc policy can-i [--user|--groups]`** command:

```
$ oc policy can-i <verb> <resource> [--user|--groups] --ignore-scopes=true
```

The lower-level user cannot list project administrator or system administrator roles:

```
$ oc policy can-i --list --user project admin
$ oc policy can-i --list --user system:admin
```

Check whether a user or a **`ServiceAccount`** can create a pod:

```
$ oc policy scc-subject-review -f examples/hello-openshift/hello-pod.json
RESOURCE ALLOWED BY
Pod/hello-openshift restricted
```

See Authorization for more information.

#### 2.3.3.4. GitHub Identity Provider Can Optionally Require a Team

Users can now test for GitHub team membership at log in.

There is now a list of one or more GitHub teams to which a user must have membership in order to authenticate. If specified, only GitHub users that are members of at least one of the listed teams will be allowed to log in. If this is not specified, then any person with a valid GitHub account can log in.

See Authentication for more information.

### 2.3.4. Storage

#### 2.3.4.1. Qualification of External Dynamic Provisioner Interface and Third-party PV

In OpenShift Container Platform 3.5, there is now the qualification of the Kubernetes interface for an external dynamic provisioner so that Red Hat can support a customer using a third-party storage solution such as NetApp Trident.

There is a concept of *in-tree* and *out-of-tree* with Kubernetes storage. Out-of-Tree means that it is not in the Kubernetes source tree and does not ship in Kubernetes or OpenShift Container Platform. The ability is provided post-installation. Many of the third-party storage vendors gravitate towards out-of-tree because it allows them to ship on their own schedule and own the distribution of their code.

See Available Dynamically Provisioned Plug-ins for more information.

#### 2.3.4.2. Dynamic Provisioner for Azure Block Storage

Dynamic provisioning is now available for Azure block storage. Just like AWS and GCE, you declare the Azure cloud provider in the ***cloud-config*** file, and then create **`StorageClasses`** with the Azure block storage options and connection information.

**Configure the Cloud Provider for Azure**

```
kubernetesMasterConfig:
  ...
  apiServerArguments:
    cloud-provider:
      - "azure"
    cloud-config:
      - "/etc/azure/azure.conf"
  controllerArguments:
    cloud-provider:
      - "azure"
    cloud-config:
      - "/etc/azure/azure.conf"
```

**Example StorageClass**

```
kind: StorageClass
apiVersion: storage.k8s.io/v1beta1
metadata:
  name: slow
provisioner: kubernetes.io/azure-disk
parameters:
  skuName: Standard_LRS
  location: eastus
  storageAccount: azure_storage_account_name
```

See Dynamic Provisioning and Creating Storage Classes for more information.

### 2.3.5. Scale

#### 2.3.5.1. Scalability Enhancements for Metrics

With OpenShift Container Platform 3.5, the default value of the **`METRICS_RESOLUTION`** parameter is now **`30`** (seconds). This change was introduced to better match the cAdvisor housekeeping interval of 30 seconds (BZ#1421834).

Increasing the **METRICS_RESOLUTION** interval helped achieve better results in relation to how many pods can be monitored by one set of metrics pods. In OpenShift Container Platform 3.5, tests showed that OpenShift metrics collection was stable for test cases up to 25,000 monitored pods in a OpenShift Container Platform cluster.

See Scaling Cluster Metrics for more information.

Currently, up to 100 container native storage (CNS) volumes on one trusted storage pool (TSP) is supported. For more information, see the Persistent Storage Using GlusterFS.

## 2.3.6. Networking

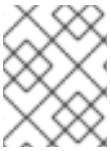
### 2.3.6.1. Multicast Support

OpenShift Container Platform 3.5 introduces multicast support. Pods can now send or receive traffic with other pods subscribed to the same multicast group.

This requires the **ovs-multitenant** plug-in and only works with annotated namespaces:

```
netnamespace.network.openshift.io/multicast-enabled: "true"
```

Pods in different tenants can subscribe to same multicast group, but cannot see each other's traffic. Administrator tenant (default project) multicast traffic does not appear in other projects. Overlay (OVS and tenants) and underlay (virtual machine and a physical server) multicast traffic never mix.



**NOTE**

Multicast is best used for low bandwidth coordination or service discovery and not a high-bandwidth solution.

See Managing Networking for more information.

### 2.3.6.2. CLI Understand Wildcard Routes

In OpenShift Container Platform 3.5, there is the added ability to see the subdomain wildcard routes added in OpenShift Container Platform 3.4, create them, and edit them using the CLI.

Add the wildcard support. Enable this on the router. The default is **off**:

```
$ oc env dc/router ROUTER_ALLOW_WILDCARD_ROUTES=true
```

Create an application or service, then create the wildcard route:

```
$ oc expose svc service-unsecure --wildcard-policy=Subdomain --name=app --
hostname=app.example.com
```

Create an edge, passthrough, or reencrypt route, for example:

```
$ oc create route edge edgeroute --service=service-secure --wildcard-
policy=Subdomain --hostname=edge.edgeroute.com
```

Test the route:

```
$ curl --resolve edge2.edgeroute.com:443:$router_ip
https://edge2.edgeroute.com -k
```

Support was also added to the web console.

#### 2.3.6.3. Allow Host Claims to be Disabled in the Router

This new feature provides the ability to create claims from different namespaces on the first directory of the path. The goal is to be able to split an application into different pods running in different namespaces.

This works by providing a way to disable the host claims is sufficient (initially). The administrator handles the routes and forbids projects from manipulating them.

For example:

Create a route in namespace 1 with:

- host name **`foo.com`**
- path= **`/bar`**

Create a route in namespace 2 with:

- host name **`foo.com`**
- path= **`/foo`**

```
namespace 2 →/bar       (1)
namespace 2 →/          (2)
namespace 2 →/bar/test  (3)
```

1. Should be rejected.
2. Should be admitted.
3. Should be admitted.

**WARNING**

This is for controlled environments only. If users can create routes, and they are untrusted, then there is a security concern.

#### 2.3.6.4. Network Policy Plug-in (Technology Preview)

Network Policy (currently in Technology Preview) is an optional plug-in specification of how selections of pods are allowed to communicate with each other and other network endpoints.

Network Policy works by way of namespace isolation at the network layer using defined labels. You can also limit connections to specific ports (e.g., only TCP ports 80 and 443).

```
kind: NetworkPolicy
apiVersion: extensions/v1beta1
metadata:
  name: allow-http-and-https
spec:
podSelector:
ingress:
- ports:
  - protocol: TCP
    - port: 80
    - port: 443
```

After installing the Network Policy plug-in, an annotation must first be set on the namespace, which flips the namespace from **`allow all traffic`** to **`deny all traffic`**. At that point, you can create **`NetworkPolicies`** that define what traffic to allow. The annotation is as follows:

```
$ oc annotate namespace ${ns}
'net.beta.kubernetes.io/network-policy={"ingress":
{"isolation":"DefaultDeny"}}'
```

With Network Policy in Technology Preview, not all features are available. Multi-tenant isolation is not available by default. Currently, it must be configured by creating default isolation policies for each namespace, and there is currently no clean path to upgrade or migrate from the multi-tenant plug-in.

See Managing Networking for more information.

#### 2.3.6.5. Ingress Object Support (Technology Preview)

In OpenShift Container Platform 3.5, there is added support for the K8s Ingress object, a set of rules that allow inbound connections to reach cluster services.

Ingress is disabled in the router, by default. When enabled, Ingress objects are handled equivalently to routes. The precedence rules apply to both if they claim the same host name.

**NOTE**

To use Ingress, the router must be given permission to read all cluster secrets.

**Example Testing Ingress Object with TLS**

**test-secret.yaml**

```
apiVersion: v1
kind: Secret
metadata:
  name: test-secret
data:
  tls.crt: `base64 -w 0 /some/path/tls.crt`
  tls.key: `base64 -w 0 /some/path/tls.key`
```

**test-ingress.yaml**

```
$ cat ingress.yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: test-ingress
spec:
  tls:
  - secretName: test-secret
  backend:
    serviceName: test-service
    servicePort: 8080
```

See Ingress Resources for more information.

### 2.3.7. Installation

OpenShift Container Platform and OpenShift Online operations are now using the same Ansible upgrade playbooks.

Lots of work around idempotency resulted in an increase in installer and upgrade stability.

Main features include:

- pre- and post- hooks for master upgrades. Integration points are now added so that users can perform custom tasks, such as cycling hosts in and out of load balancers during the upgrade process.
- Open vSwitch (OVS) and etcd version increases.
- Rolling updates of certificates.
- More customization possible during upgrade steps to meet local needs.
- Code refactoring for idempotency.
- Deployment of router shards during installation is now possible. This allows administrators to establish swim lanes to specific route shards for labeled routes.

### 2.3.8. Metrics and Logging

OpenShift Container Platform 3.5 includes enhanced Ansible playbooks to better handle deployments and upgrades. This deprecates the deployer deployment procedure and replaces it with Ansible in a manner that is more consistent with the installation of the rest of the product.

Administrators can declare variables in the inventory file to cause playbooks ***openshift_metrics.yml*** and ***openshift_logging.yml*** to behave differently. The metrics and EFK stacks can be deployed without requiring Java to be installed on the master node.

Ansible handles:

- Metrics stack for OpenShift Container Platform 3.5.
- Fresh deployment of metrics and logging.
- Upgrading of metrics from OpenShift Container Platform version 3.3 to 3.5 and OpenShift Container Platform version 3.4 to 3.5.

- Upgrading of logging from OpenShift Container Platform version 3.3 to 3.4.
- Re-installation of metrics and logging (**`cleanup`** and **`install`**).
- Scaling metrics and logging.

See Enabling Cluster Metrics and Aggregating Container Logs for more information.

### 2.3.9. Developer Experience

#### 2.3.9.1. Pulling Artifacts from Remote Resources

Previously, **`oc start-build`** only allowed a local file to be specified, but did not allow a URL to a remote resource. Now, users can pull in artifacts via **`oc start-build --from-file=<some URL>`**.

This feature only works against GET-based endpoints that do not require authentication and use either no transport layer security (TLS), or TLS with a certificate trusted by the client. This feature does not reinvent **`curl`**. The file is downloaded by the CLI, then uploaded to the binary build endpoint.

#### 2.3.9.2. Setting Environment Variables When Creating an Application from a Template

Users now also have the ability to set environment variable when creating an object (for example, an application) from a template. Previously, this was a separate step following template creation.

#### 2.3.9.3. Support for -p Parameter Values

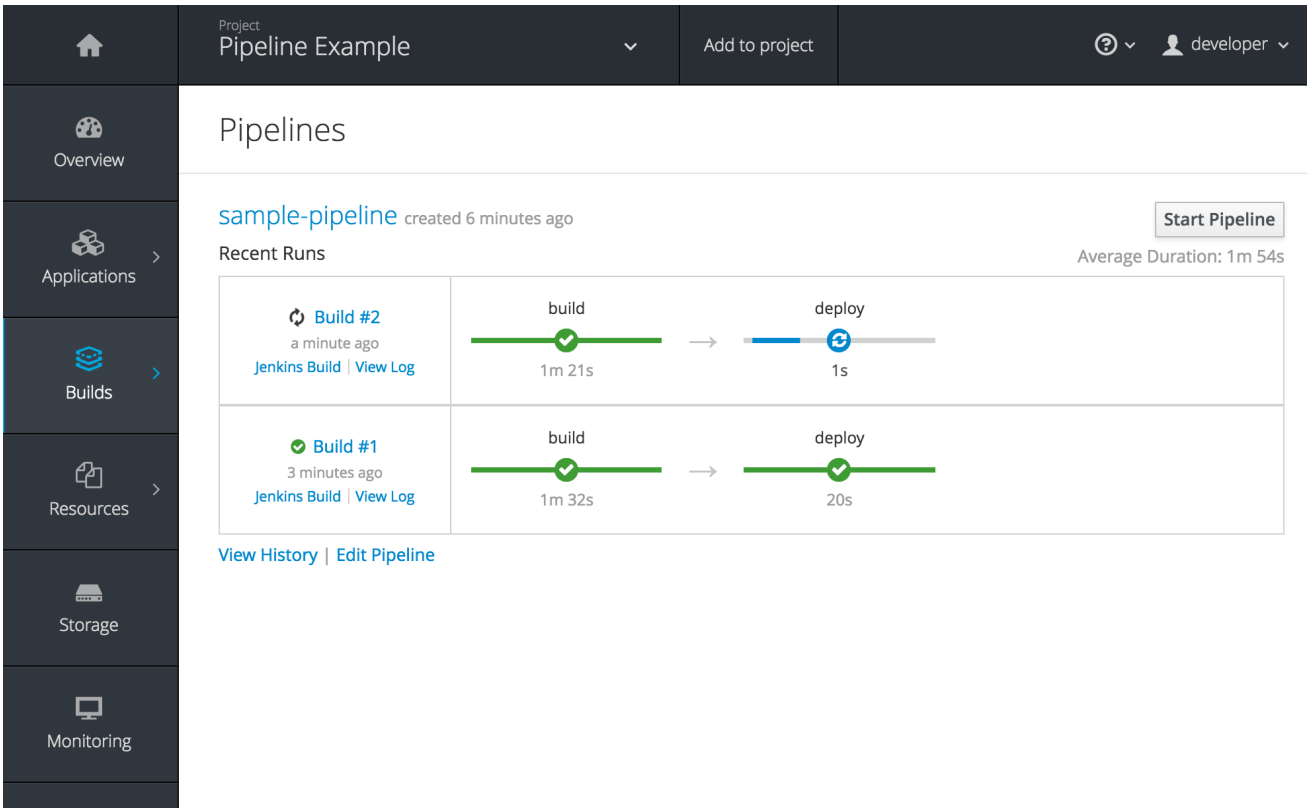
Both **`oc new-app`** and **`oc process`** now support **`-p`** for parameter values. The **`-v`** flag is deprecated.

#### 2.3.9.4. CI/CD Pipeline

In OpenShift Container Platform 3.5, enablement materials regarding use of CI/CD pipelines with OpenShift Container Platform are improved. The complexity and number of pipeline samples provided is increased.

Support is added to **`oc new-app`** and **`oc new-build`** so that the commands are pipeline aware.

**Figure 2.1. Pipelines Page**




See Promoting Applications Across Environments and Creating New Applications for more information.

#### 2.3.9.5. Default Hard Eviction Thresholds

OpenShift Container Platform uses the following default configuration for **`eviction-hard`**.

```
...
kubeletArguments:
  eviction-hard:
  - memory.available<100Mi
  - nodefs.available<10%
  - nodefs.inodesFree<5%
  - imagefs.available<15%
...
```

See Handling Out of Resource Errors for more information.

### 2.3.10. Web Console

#### 2.3.10.1. Run and Deploy on OpenShift Container Platform

In OpenShift Container Platform 3.5, there is now a "Run on OpenShift" experience that allows you to provide external links in the web console to deploy templates.


Use the URL pattern to select a template or image. You can customize it to have it come from separate project. The end-user is prompted for the project.

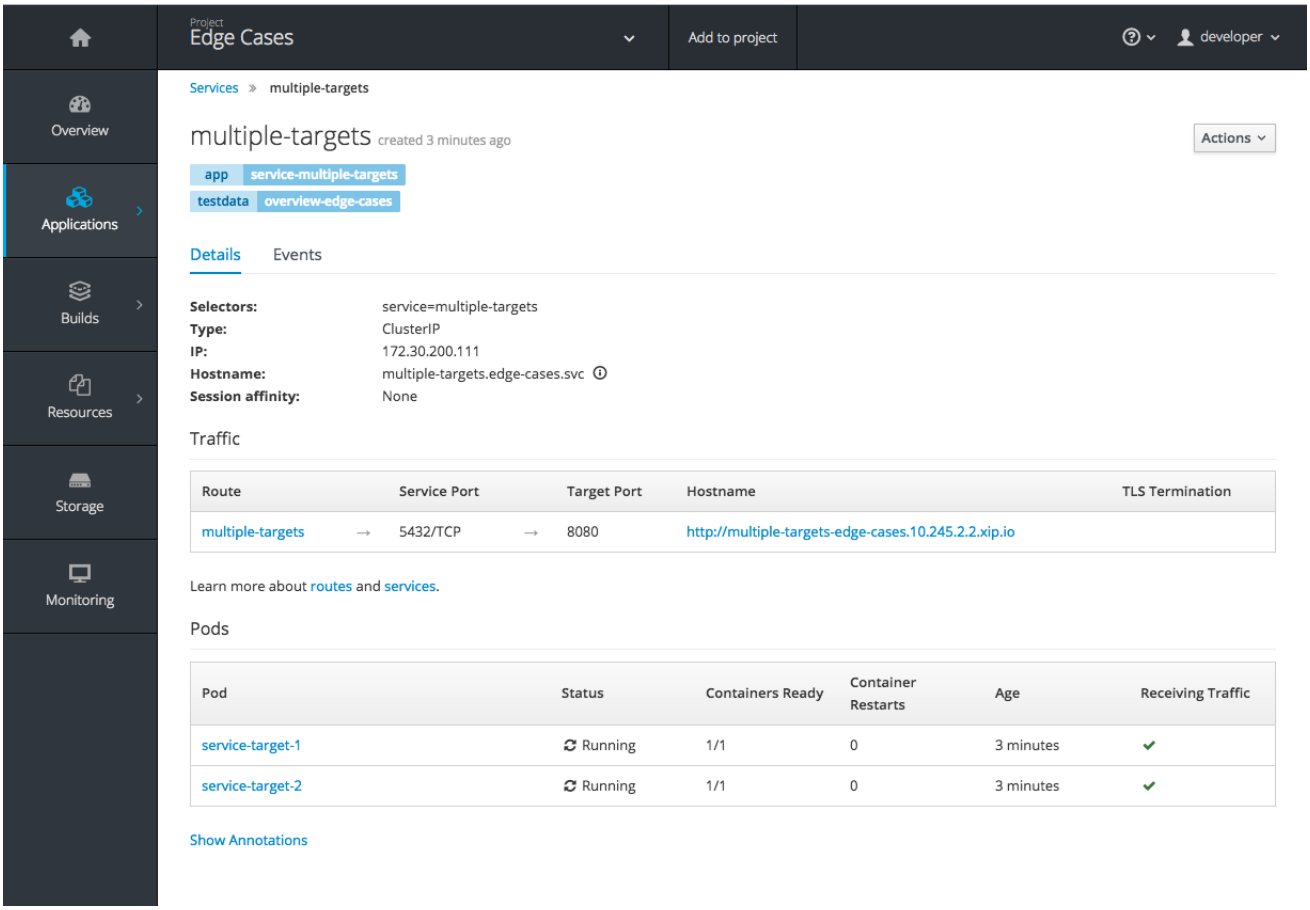
See Create From URL for more information.

### 2.3.10.2. Added Service Details

There are now added service details on configuration, traffic, routes, and pods.

There is a new section highlighting routes, service and target ports, host name, and TLS. There is also a section iterating pods and their status.

**Figure 2.2. Service Details View**




### 2.3.10.3. ConfigMap: Create, List, Detail

In OpenShift Container Platform 3.5, there is now the ability to easily work with configuration data decoupled from the container image. You can:

- Create new **`ConfigMap`**
- List out existing **`ConfigMaps`**
- Work with the configuration details.
- Easily consume them from various other pages.

**Figure 2.3. Create a ConfigMap**

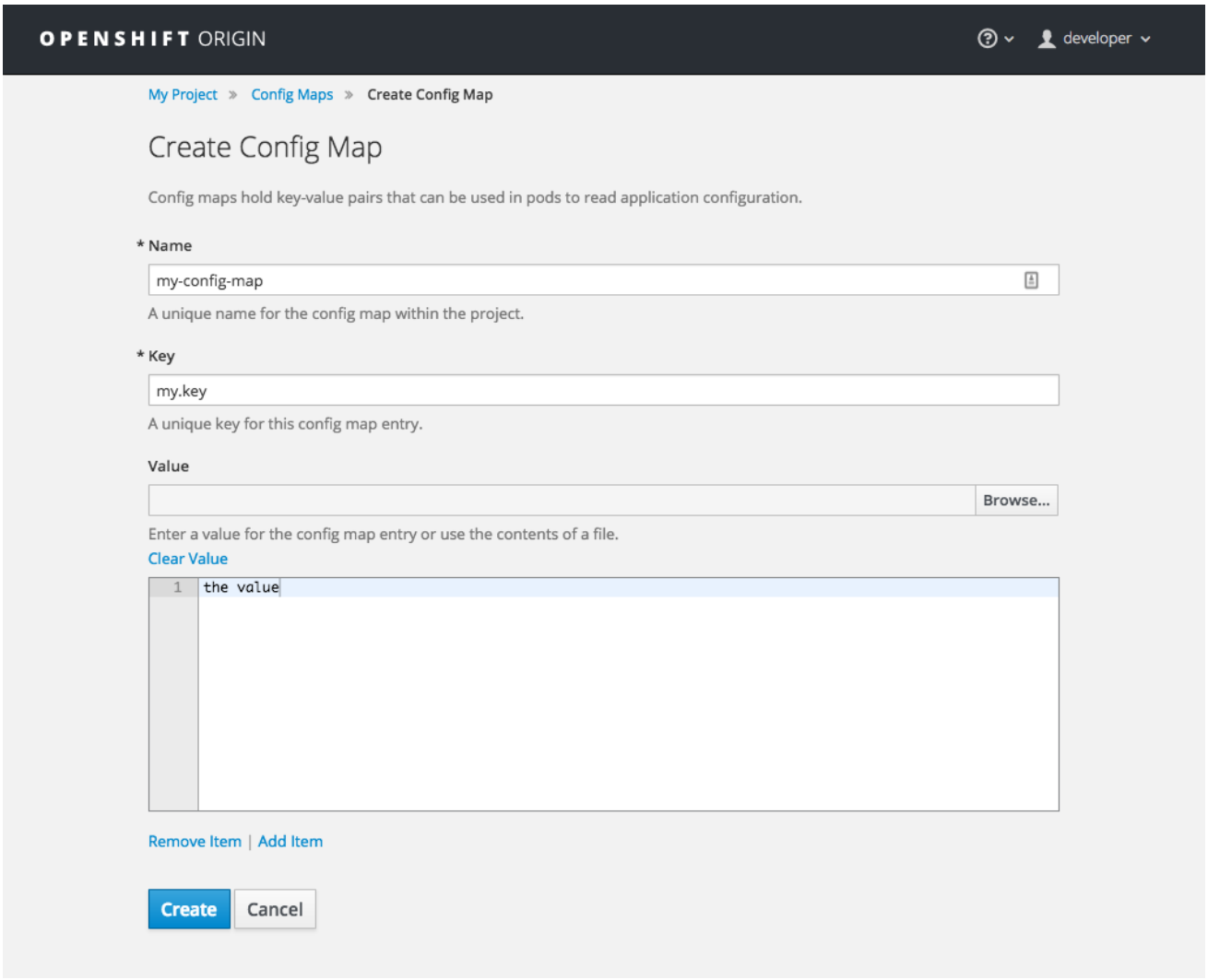
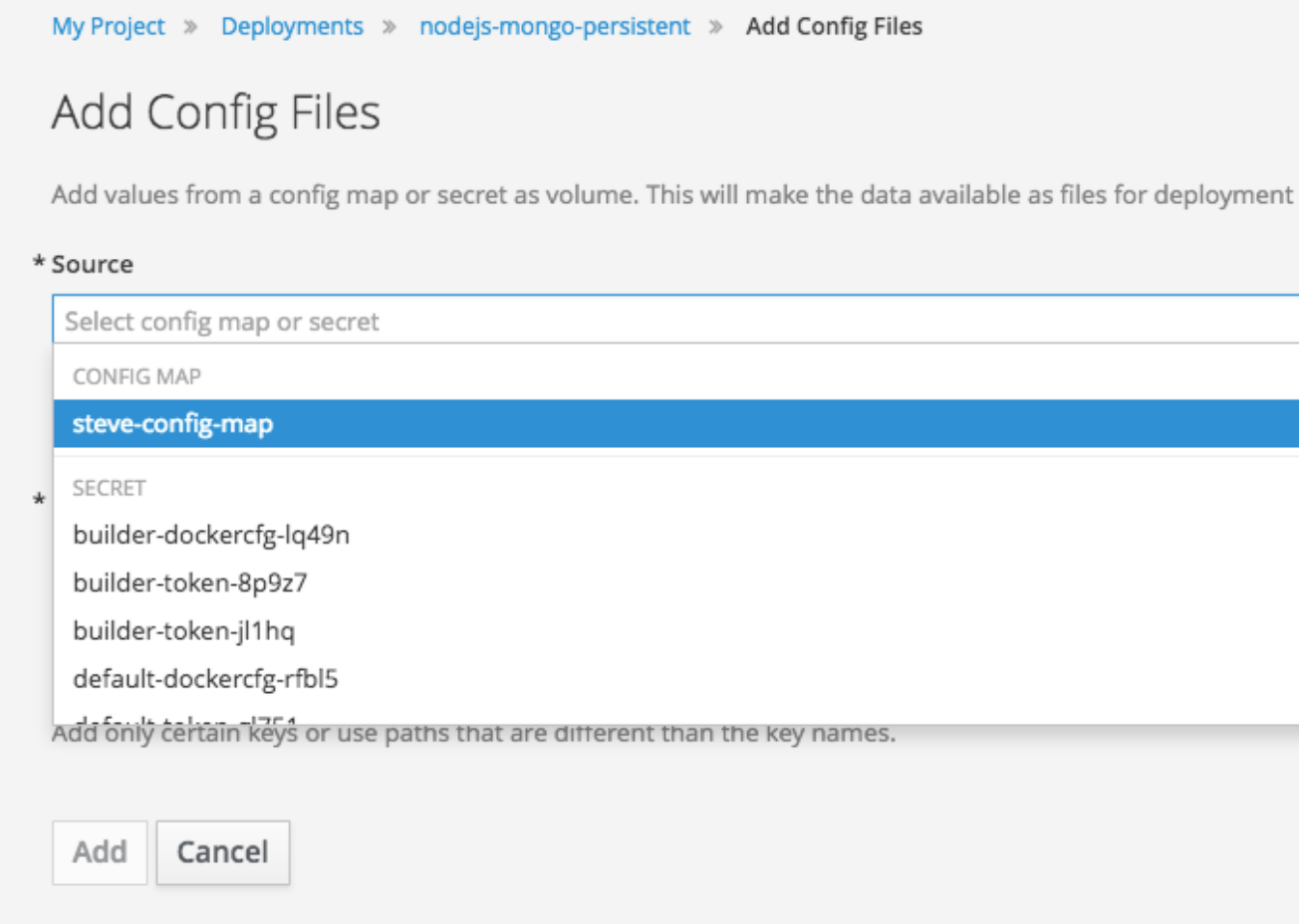


**Figure 2.4. Add Config Files**




### 2.3.10.4. Show Build Failures

Users no longer have to search logs to gain a better understanding of why build failed. Individual build status messages are now updated with details that are available via the web console and the CLI.

**Figure 2.5. Build Failures as Seen in the Web Console**

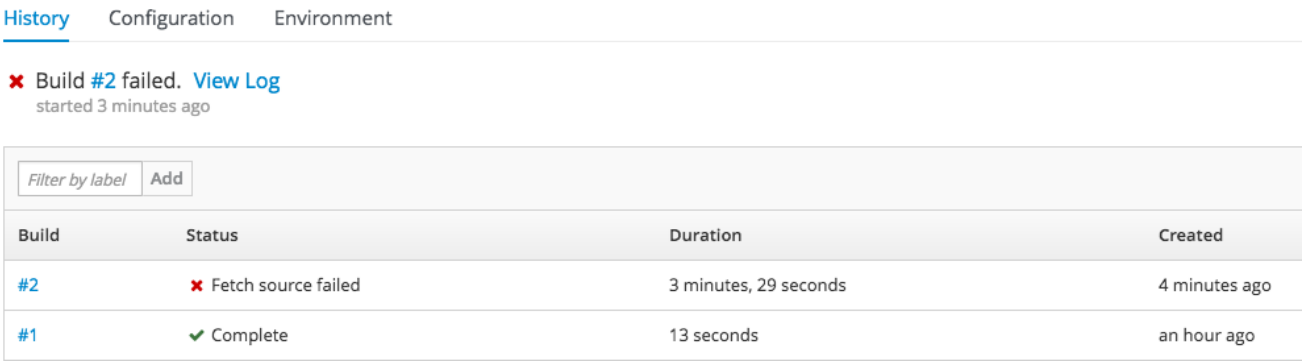
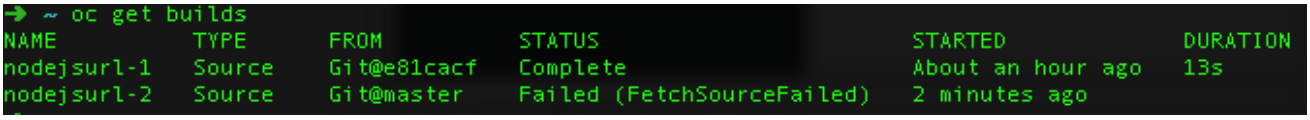



**Figure 2.6. Build Failures as Seen in the CLI**




### 2.3.10.5. StatefulSets (Technology Preview)

Custom resource listing and details pages for **`StatefulSets`** (formerly known as **`PetSets`**) is now available. Users can get details of all **`StatefulSets`**, including deployments and replica sets.

See the Web Console documentation for more information.

## 2.4. NOTABLE TECHNICAL CHANGES

OpenShift Container Platform 3.5 introduces the following notable technical changes.

### Updated Infrastructure Components

- OpenShift Container Platform 3.5 is supported on RHEL 7.2 and 7.3 with the latest packages from Extras, including Docker 1.12.
- OpenShift Container Platform 3.5 is *not* certified with older versions of Docker and RHEL 7.1 or earlier.
- Kubernetes has been updated to v1.5.
- etcd has been updated to 3.1.
- Open vSwitch (OVS) was upgraded to 2.6 and the package is now provided via the Red Hat Enterprise Linux Fast Datapath channel.

### Miscellaneous Changes

- **`activeDeadlineSeconds`** is now configurable for deployer pods via the deployment configuration API.
- In OpenShift Container Platform 3.5, **`ScheduledJob`** is renamed **`CronJob`**. If you want to keep your scheduled jobs, you need to export them from the 3.4 cluster (using **`oc export`** or **`oc get -o yaml`**) and create them again, after the upgrade, on the 3.5 cluster. The storage prefix has changed, along with the name, and newly created clusters do not know where to look for **`ScheduledJob`**. Cluster version 3.5 operates on **`CronJob`**, but it also understands **`ScheduledJob`** submitted to it. It performs rapid conversion, saving your newly created object as a **`CronJob`**, resulting in all subsequent read operations returning **`CronJob`** instead. See Cron Jobs for more information.
- The default value for **`ingressIPNetworkCIDR`** was previously a non-private range (**`172.46.0.0/16`**) and has been changed to a private range (**`172.29.0.0/16`**). Clusters configured with the non-private range run the risk of routing issues, and updating to a private range is advised.

**WARNING**

When **`ingressIPNetworkCIDR`** changes, any external IPs allocated from the previous range will be reallocated from the new range.

- The **`groups`** field in the user object is now deprecated. Instead, create Group API objects containing the names of the users that are members of the group.

- **`oc whoami --token`** was deprecated in OpenShift Container Platform 3.4 in favor of **`oc whoami -t`**. Also, **`oc whoami --context`** is deprecated in favor of **`oc whoami -c`**. The **`--token`** and **`--context`** options now behave consistently with all other **`oc`** commands, indicating the specified token or context should be used.

- **`extensions/v1beta1.Job`** is deprecated in favor of using **`batch/v1.Job`**. The storage should be updated to keep the Jobs readable in future versions of the cluster. See Manual Upgrades for more information.

- OpenShift Container Platform 3.5 requires that the **`rhel-7-fast-datapath`** repository be enabled.

- Template instantiation now respects namespaces defined in the template objects (meaning it will create the object in specified namespace) if and only if the namespace definition uses a parameter reference. Previously, it never respected the namespace defined in the object.

## 2.5. BUG FIXES

This release fixes bugs for the following components:

### Authentication

- There was a bug in how policies were listed internally when used to build role bindings. Filtering of role bindings based on selectors did not work correctly. With this bug fix, the internal listing of policies was updated to the correct behavior. As a result, the filtering of role bindings based on selectors now works as expected. (**BZ#1423215**)

### Builds

- Source-to-Image builds expect image commits to take no longer than two minutes. Commits that take longer than two minutes result in a timeout and a failed build. With this bug fix, the timeout is removed so that image commits can take indeterminate lengths of time. As a result, commits that take an excessive amount of time will not result in a failed build. (**BZ#1391665**)

- The build failure reason was not getting set or saved correctly. Therefore, the build failure reason was not shown in command output. The code is now updated to correctly save the build failure reason and the build failure reason now shows correctly in command output. (**BZ#1415946**, **BZ#1419810**)

- Previously, running a custom build with an image containing a Docker binary that was a different version than the Docker container running on the OpenShift Container Platform node would result in an error. The build would fail with a message about mismatched Docker API version. Now, you can set the **`DOCKER_API_VERSION`** environment variable in the **`BuildConfig`** to match the API version running on the node. For example:

  ```
  $ oc set env bc/buildcfg DOCKER_API_VERSION=1.22
  ```

  Note that this will only work if the version of the Docker binary on the custom builder image is newer than the version running on the OpenShift Container Platform node. (**BZ#1422798**)

- The build duration was not being consistently calculated. Therefore, the build duration displayed in the web console and on the command line was inaccurate. With this bug fix, the duration of completed builds is now consistently calculated and a consistent build duration value is reported for builds under all circumstances. (**BZ#1318403**)

- Previously, the **`oc new-app`** command would try to interpret its argument as a path and would

exit with an error when a component of this path existed, but was not a directory. Running **oc new-app X/Y' with a file named `X** in the current directory would cause an exit with an error, even though **X/Y** denotes a valid Docker image. When **oc new-app** tries to interpret the input component as a directory and object with that name exists on a file system but is not a directory, try another possible interpretation instead of exiting with an error. As a result, running **oc new-app X/Y** creates a new application based on Docker image X/Y, even in the case when file X exists in the current directory. (BZ#1347512)

- There were different code paths for retrieving and setting the commit information. Therefore, the **OPENSHIFT_BUILD_COMMIT** environment variable was only set in the output image when the build was triggered by a webhook. To fix this issue, use a common code path for retrieving and setting the commit information so it is always available to be added to the image. As a result of this bug fix, the **OPENSHIFT_BUILD_COMMIT** environment variable is always present in the output image. (BZ#1408879)

- Previously, a race condition could cause builds with short-running post-commit hooks to hang. This bug fix resolves the issue and builds no longer hang. (BZ#1425824)

- Master returned an internal server error HTTP code when the Docker image lookup failed due to unreachable registry. This happened for every image lookup in disconnected OpenShift Container Platform environments. Therefore, **oc new-app** reported the internal server error as a warning to the user, which can make the user think there is something wrong with their OpenShift Container Platform deployment. Change the wording of the error **oc new-app** prints to not include the string "internal server error". As a result, the warning that is printed does not sound more severe than it is. (BZ#1398330)

### Command Line Interface

- The latest version of Docker for Mac/Windows uses the Community Edition versioning scheme. This causes **oc cluster up** to halt with an error because the new version cannot be parsed by the **semver** library. This bug fix changes the behavior to display a warning instead of exiting with an Error. (BZ#1428978)

- The race condition is seen when updating a batch of nodes in the cluster using **oc adm manage-node** to be schedulable or unschedulable. Therefore, several nodes could not be updated with the "object has been modified" error. Use a patch on the **unschedulable** field of the node object instead of a full update. With this bug fix, all nodes can be properly updated as schedulable or unschedulable. (BZ#1279303)

- Previously, the **--overwrite** option for **oc volume** was confusing. This bug fix improves the **oc set volume --override** flag description so that users understand that they are not replacing the current volume that is being used. (BZ#1319964)

- Previously, a confusing error message was generated when **oc set probe** was run without providing a port with a get-url. With this bug fix, the error is now formatted to be much more readable to the user. (BZ#1332871)

- The **oc get** command would return the message "No resources found", even in cases where resources did exist, but could not be retrieved due to a connection error. The command **oc get** was updated to only show the message "No resources found" in cases when resources truly did not exist in the server. As a result of this bug fix, **oc get** no longer displays "No resources found" in cases when there is an error retrieving resources from the server. (BZ#1393289)

- The new responsive terminal would wrap long lines in the output of CLI commands. The **oc adm diagnostics** indentation did not work well, and no longer had color in its output. This bug fix

bypasses the responsive terminal in **`oc adm diagnostics`** (currently only being used in CLI help output). As a result, **`oc adm diagnostics`** now has proper indentation and colorized output. (BZ#1397995)

- Output from the **`oc idle`** command was confusing to end users. A user could not easily tell what was being done by the **`oc idle`** command. With this bug fix, the output of the **`oc idle`** command was updated to clarify what the command had done and is now easier to understand. (BZ#1402356)

- Previously, **`oc status`** tried to generate a status for the "default" cluster namespace if a user had not yet created a project after logging in. The user would see a forbidden status error "cannot get projects in project" when their context was still in the cluster's "default" namespace after logging in, and did not have permissions to "LIST" in this namespace. With this bug fix, **`oc status`** now checks to see if a user cannot list projects in the default namespace. As a result, the user no longer sees the error message "cannot get projects in project <default cluster namespace>" when they execute **`oc status`** and have no projects in their current namespace. They instead see a message prompting them to create a new project, or to contact their administrator to have one created for them. (BZ#1405636)

- After running ` oc adm drain -h`, the user would try to open the provided link **`http://kubernetes.io/images/docs/kubectl _drain.svg`**, but would receive a "404 page not found" error. This bug fix corrects an extra space in the link path and the link now works as expected. (BZ#1415985)

- Although a **`MasterConfig`** load error is stored globally, it is only printed the first time that it is encountered during a diagnostics check. This bug fix ensures that, even if the error has already been encountered once, its message gets printed in subsequent diagnostic checks. (BZ#1419472)

- Deleting an access token using the **OAuthAccessTokens** client would fail for users that had logged in using a **serviceaccount** token. A failure from the access token client would prevent the token from being deleted from the local configuration, causing a user to be unable to log out. With this bug fix, the failure is now logged, ensuring that an attempt to remove the token from the user's local configuration always takes place. A user is now able to log out after logging in with a **serviceaccount** token. (BZ#1422252)

- Tags were not sorted according to **`http://semver.org/`** and, therefore, the "highest" tags were not imported when the image import limit was cutting down the amount of imported images. With this bug fix, tags are now sorted according to semantic versioning rules. The "highest" tags are now properly imported, even when only a limited number of tags is allowed to be imported. (BZ#1339754)

- Previously, the ***.kubeconfig*** file was being generated with a server URL that did not include a port number. Although the port number was safely assumed to be **443** with an HTTPS protocol, it prevented the certificate from being successfully verified during the login sequence (an exact match including the port was required). Therefore, the user was prompted with the warning "The server uses a certificate signed by an unknown authority" every time they attempted to log in using an OpenShift Container Platform installation completed through **`openshift-ansible`**. With this bug fix, the command **`oc adm create-kubeconfig`** (used by the **`openshift-ansible`** playbook) was patched to normalize the server URL so that it included the port with the server URL in the generated ***.kubeconfig*** file every time. As a result, the user no longer sees the message "The server uses a certificate signed by an unknown authority" when logging in using a ***.kubeconfig*** file generated by an **openshift-ansible** installation. (BZ#1393943)

- There was a duplicated resource "quota" in the **`oc describe`** list of valid resources. Therefore, "quota" was printed twice. This bug fix removes one entry on "quota" in the **`oc describe`** list of valid resources. Now, each resource type is only printed once. (**BZ#1396397**)

- Multi-line output for a template description did not display all lines with correct indentation under **`oc new-app`**. Therefore, the output for template descriptions was hard to read. This bug fix added a new helper function **`formatString`**, which indents all lines for a multi-line template description. Template descriptions for **`oc new-app <my_template>`** are now easier to read. (**BZ#1370104**)

- The **`.spec.dockerImageMetadata`** field was unnecessarily used when patching an image stream tag. As a consequence, the **`oc edit`** command could not succeed. This bug fix modifies the patch mechanism used in **`oc edit`** to always replace the contents of the **`.spec.dockerImageMetadata`** field. As a result, users should be able to invoke **`oc edit`** on any image stream tag. (**BZ#1403134**)

- There was previously no information about the **`--generator`** parameter explaining its use in the help output of the **`oc expose`** command. This bug fix adds an explanation that gives example usage (**BZ#1420165**)

### Containers

- This enhancement updates the Jenkins examples to remove the need for a slave, which makes configuration simpler. (**BZ#1374249**)

### Deployments

- The rolling updater was not ignoring pods marked for deletion and was counting them as ready. This bug fix updates the rolling updater to ignore such pods. (**BZ#1307004**)

### Image

- This enhancement allows Maven and Node.js slave image paths to be specified explicitly. Disconnected environments were unable to pull the images from the hardcoded paths, so **`MAVEN_SLAVE_IMAGE`** and **`NODEJS_SLAVE_IMAGE`** environment variables can now be used to control where to pull the images from, overriding the hardcoded defaults. (**BZ#1397260**)

### Image Registry

- The OpenShift Container Registry (OCR) was not able to handle forwarded headers provided by an HAProxy in front of it, making it unusable when exposed on insecure port 80. Pushes failed because the registry generated incorrect URLs. An upstream fix has been backported to the OCR. As a result, the OCR now handles forwarded headers and it is usable again when exposed on an insecure port. (**BZ#1383439**)

- The master API previously investigated the incorrect object when determining the docker image reference of a new image stream mapping when the referenced image already existed. This created image stream tags containing misleading information about an image's location, pointing to the original image stream. This bug fix updates the master API to now properly determine docker image references for new image stream mappings. As a result, image stream tags now show proper docker image references pointing to managed images. (**BZ#1408993**)

- The OpenShift Container Registry (OCR) did not consider insecure import policies of image stream tags when deciding whether to fall back to insecure transport when serving blobs from external registries. This meant images imported from external insecure (no HTTPS or a bad certificate) with an **`--insecure`** flag applied could not be pulled through the OCR. With this bug

fix, the OCR now considers the insecure import policy of image stream tags where the requested image is tagged. As a result, the OCR allows serving images from insecure external registries if they are tagged with an insecure import policy. (BZ#1421954)

**Kubernetes**

- Using **`hostPath`** for storage could lead to running out of disk space, and the root disk could become full and unusable. This bug fix adds support for pod eviction based on disk space. If a pod using **`hostPath`** uses too much space, it may be evicted from the node. (BZ#1349311)

- Horizontal pod autoscalers (HPAs) would fail to scale when it could not retrieve metrics for pods matching its target selector. Therefore, dead pods and newly created pods would cause HPAs to skip scaling. This bug fix adds logic to the HPA controller which assumes conservative metric values, depending on the state of the pod and the direction of the scale, when metrics are missing or pods are marked as unready or not active. As a result, newly created or dead pods will no longer block scaling. (BZ#1382855)

- Previously, pod evictions due to disk pressure did not resolve until the pod was deleted from the API server. This bug fix causes local storage to be freed on pod termination (i.e., eviction) rather than pod deletion. (BZ#1390963)

- Previously, I/O could be saturated on a node due to the collection of per-container disk stats from a thin pool with a large amount of metadata. This bug fix disables the collection of these statistics until such time that an efficiently way to collect them can be found. (BZ#1405347)

- Previously, docker could refuse to start new containers due to reaching **`dm.min_free_space`** (default 10%), but the devicemapper thin pool usage did not exceed **`image-gc-high-threshold`** (default 90%), so the image reclaim occurred and the node was stuck. This bug fix changes the default **`image-gc-high-threshold`** to 85%, which causes image reclaim to occur before the default **`dm.min_free_space`** is reached. (BZ#1408309)

- The kubelet previously had a fixed constant for how long it would tolerate the docker daemon being down before reporting the node as **`NotReady`**. That was previously set to 5 minutes, which meant that it could take up to 5 minutes for the kubelet to report it was no longer ready. This bug fix introduces new behavior so that the kubelet will wait 30 seconds for the container runtime to be down before reporting the node as **`NotReady`**. As a result, the node now reports **`NotReady`** faster when the docker daemon is down. (BZ#1418461)

- The **`oc adm drain --force`** command was ignoring any pods that indicated they were managed by a daemonset even if the managing daemonset was missing. This bug fix updates the command to detect when a daemonset pod is orphaned and warn about the missing daemonset rather than generating an error. As a result, the command removes orphaned daemonset pods. (BZ#1424678)

- When attempting to connect to etcd to acquire a leader lease, the master controllers process only tried to reach a single etcd cluster member even if multiple were specified. If the selected etcd cluster member was unavailable, the master controllers process was unable to acquire the leader lease and would not start up and run properly. This bug fix updates this process to attempt to connect to all of the specified etcd cluster members until a successful connection is made. As a result, the master controllers process can acquire the leader lease and start up properly. (BZ#1426733)

- Excessive logging to the journal caused masters to take longer to restart. This bug fix reduces the amount of logging that occurs when initial list/watch actions happen against etcd. As a result, the journal is no longer pegged with a lot of messages that cause logging messages to be rate

limited and dropped. Server restart time should be improved on clusters with larger data sets. (BZ#1427532)

- OpenShift Container Platform nodes configured with OpenStack as the cloud provider could previously move into **`NotReady`** state if contact with the OpenStack API was lost. With this bug fix, nodes now remain in **`Ready`** state even if the OpenStack API is not responding. Note that a new node process configured to use OpenStack cloud integration cannot start without the OpenStack API being responsive. (BZ#1400574)

- The admission plug-in **`LimitPodHardAntiAffinityTopology`** has been disabled by default. Enabling it by default previously caused conflict with one of the end to end tests. (BZ#1413748)

### Logging

- The Diagnostic Tool (**`oc adm diagnostics`**) now correctly reports the presence of the **`logging-curator-ops`** pod. The **`logging-curator-ops`** was not in the list of pods to investigate, resulting in an error that indicated the pod was missing. (BZ#1394716)

- Switching between indices in the Kibana UI now displays the appropriate log entries. Because default field mappings were being applied in Elasticsearch, the user might receive the **`Apply these filters?`** error message. (BZ#1426061)

### Web Console

- The **Browse** tab now shows the local host name of a service. (BZ#1395821)

- On a project's **Settings** tab, the Quota terminating scope descriptions are not clear. The spinning icon on the **Browse** tab that indicates a pod is running no longer appears jittery. In some browser/operating system combinations, font and line-height issues could make a spinning icon wobble. Those issues have been corrected. (BZ#1365301)

- A link to documentation on using persistent volumes was added to the **Create Storage** page. (BZ#1367718)

- If the web console encounters an error updating Hawkular Metrics charts, the console will automatically attempt to update again. If the error(s) persist, the web console will show an alert at the top of the page with a **Retry** link. Previously, the user would need to reload the browser if an update error occurred. (BZ#1388493)

- On the web console **About** page, the user can copy the CLI code to log into OpenShift Container Platform using the current session token. The token is now permanently hidden and the web console now appends the user token if the user copies the CLI example using the **Copy to Clipboard** button. (BZ#1388770)

- The web console now displays any Kubernetes **`StatefulSet`** objects (formerly called **`PetSets`**) in a project with the same level of detail as other resources. (BZ#1393202)

- On the **Create Secret** page, if the user uploads a file that is not a properly formed file, the **Create** button will now be disabled. Previously, the **Create** button was enabled if an improper file was uploaded. (BZ#1400775)

- The screen to edit a JSON-formatted template in YAML format now displays the entire template file in YAML. Previously, because of space restrictions, some of the JSON formatting would not be converted to YAML. (BZ#1402260)

- When a build is in the **Pending** state, the **Duration** time will not be calculated. The duration time starts when the build changes to **Running**. This change was made to prevent negative duration

times that could arise from differences in the browser clock time and the server clock time. (BZ#1404417)

- Previously, under specific circumstances, a single build could appear twice in the **Overview** page of the web console. The web console now correctly lists each specific build one time on the **Overview** page. (BZ#1410662)
- In the JVM console, for Apache Camel diagrams, the **Breakout suspended at** slideout window can be closed and appears only when a breakout is suspended. Previously, the window could not be closed, which could prevent the user from selecting Camel route elements. (BZ#1411296)
- The web console now validates deployment controller and replication controller memory limits that are specified in kB. Previously, validation of memory units in kB would incorrectly fail as being too small for the limit range. This happened only for kB, and not other memory units. (BZ#1413516)
- The links to the documentation in the web console now point to the correct product. Previously, the links led to the OpenShift Origin documentation. (BZ#1426061)
- When editing a deployment configuration (DC) through the web console, the memory unit is properly retained. Previously, the requested memory was not retained. (BZ#1413842)
- Project display names that contain less than (<) and greater than (>) characters always display in the **Choose Existing Project** list. Previously, if a display names contained these characters in a way that mimicked HTML (such as: **`<displayname>`**) would result in the display name not appearing or not appearing correctly in the list. (BZ#1414195)
- Client-side validation for persistent volume claim limit ranges has been added to the "Create Storage" page in the web console allowing the user to specify minimum and maximum values for capacity. (BZ#1414229)
- When using self-defined stage names for a pipeline, the **`stage`** parameter must include a block argument, for example: **`stage('build is the greatest stage') {}`** in the Jenkinsfile. (BZ#1414661)
- Name validation in the web console is now consistent with the CLI. Periods are now allowed in the names, and the maximum length has been increased to 253 characters. Previously, the validation in the web console was more strict than in the CLI. Validation has been relaxed for the following forms in the web console to match the command line:
  - Add Autoscaler
  - Add Storage
  - Create Config Map
  - Create Route
  - Create Secret (BZ#1414691)
- In the JVM Console, the **Preference** button in the User page of the JVM Console has been added back to the interface. Previously, the **Preferences** button was missing. (BZ#1415463)
- In the web console, when deploying an application based on an image, the **Next Step** page correctly appears. Previously, the web console would incorrectly redirect to the **Overview** page. (BZ#1415602)

- The web console now displays an error message when a user with an unauthorized role tries to grant the `serviceaccount:builder` role to a user. Previously, the web console did not display an error message. (BZ#1420247)

- If you accessed the **Build Configuration** edit page using the page URL, the **Create New Secret** button correctly appears. Previously, if you accessed the edit page using the URL, the **Create New Secret** button would not appear. (BZ#1421097)

- Logs in the web console for a pod with multiple containers have been fixed to address a situation where it was possible for log output from more than one container to appear. (BZ#1427289)

- The pod metrics graph for CPU in the web console would not render if there is zero CPU activity. Previously, the graph line would not connect to the zero baseline. (BZ#1427360)

- On the **Application Deployment** page, the annotations associated with the deployment might appear truncated in the **Show Annotations** list, if the annotation is too long. Click the **See All** button to display the full annotation or **Collapse** to hide the truncated section of the annotation. (BZ#1233511)

- In the web console, environment variables in the build file are no longer truncated after the **=** character. Previously, the environment variable values that contained an **=** character were being truncated. (BZ#1357107)

- In the pod metrics page, the donut chart for current usage now appears to the right of the metrics sparkline. The new position allows you to see more metric data on the screen. Previously the donut chart was above the sparkline. (BZ#1387286)

- Previously, some changes to a health check command or deployment hook command in the web console would not be saved. This happened when editing an existing command and adding or removing a single argument. The web console has been fixed to correctly save all edits to health check and deployment hook commands. (BZ#1411258)

- Previously, you had to enter weights between `0` and `256` as integer values. When creating or editing routes that send traffic to two services in the web console, you can now specify the service weights as percentages using a slider control. You can still enter integer weights if desired. (BZ#1416882)

- On the **Add to Project** page of the web console, if you entered an invalid setting for some advanced options, then hide the advanced options, the form would be submitted with invalid values, causing errors when creating some resources like horizontal pod autoscalers. The web console has been changed to correctly validate these fields so that you cannot submit the form with invalid values. (BZ#1419887)

- Previously, the link to download the OpenShift CLI linked to the incorrect version (of OpenShift Origin). The link has been updated, and the link downloads the correct version. (BZ#1421949)

- Previously, the **Create a Secret** and **Add Config Files** buttons when creating using the web console mistakenly linked to the other page. The buttons have been corrected. (BZ#1425728)

- Dates now use the word form over the number form to avoid ambiguity (For example, May 4, 2016 instead of 05/04/2017). (BZ#1333101)

- Previously, dropdown menus on the web console overlay the navigation menu dropdowns, blocking the view and usability of the navigation menu dropdowns. The navigation menu dropdown's z-index has been set to a value greater than that of page content dropdowns, resulting in navigation menu dropdowns to always appear on top of page content dropdowns. (BZ#1366090)

- A DOM element under the label filter component was being removed during certain navigation situations, preventing the **Clear Filters** link from appearing until the browser was refreshed. The correct element is now removed under these navigation situations meaning the **Clear Filters** link will always appear when any label filters are active. (**BZ#1375862**)

- When using the **Deploy Image** tab from the **Add to Project** page, changing the name input value no longer causes the displayed image name to change. The correct image name is now displayed. (**BZ#1403097**)

**Metrics**

- Previously, the Heapster image and pod did not specify the user it should be run under and defaulted to using the root user. If the user is running with the `MustRunAsNonRoot` SCC, then it would fail since it its not allowed to be run as a root user. This bug fix ensured it would specify a default user for the Heapster image meaning users can run with the `MustRunAsNonRoot` SCC without issues. (**BZ#1393103**)

- The Hawkular Metrics log data was missing the date in its timestamps. This bug fix enables the timestamps in the logs. (**BZ#1423014**, **BZ#1427666**)

- Previously, JDK and Cassandra could not determine the filesize for extremely large filesystems, such as EFS, because Cassandra tries to and read the filesystem size when it configures itself, but notices the invalid size and fail to start properly. Cassandra has been patched to work around the failure encountered and will be able to start on systems that are using extremely large filesystems. (**BZ#1418748**)

**Networking**

- Previously, wildcard route support was not exposed in the CLI. This fix enables support, meaning you can now create wildcard routes in the CLI. (**BZ#1391786**)

- Previously, unidling connections could time out if the pod took longer than 30s to start, because clients had connections closed with no data. The timeout has been increased to 120 seconds so that slow pods do not break clients. (**BZ#1416037**)

- To be consistent with edge routes, this bug fix makes is possible to configure insecure termination for all types of routes from the CLI. (**BZ#1403155**)

- This bug feature adds an environment variable to configure haproxy router logging facility, so that the syslog facility can be set. Now, users can separate log traffic as desired. (**BZ#1419127**)

- Previously, the CIDR for multicast addresses was incorrect. Leading to addresses that were in the mis-claimed portion being treated incorrectly, as multicast would not work. This fix allows the range to be the IETF assigned one (per RFC 5771), meaning that addresses that were in the wrong portion of the range now work. (**BZ#1420032**)

**REST API**

- Previously, there was a code difference with the code used to build the root etcd prefix between etcdv2 and etcdv3. This resulted in, when migrating from etcdv2 to etcdv3, the cluster not being able to find any data if a root etcd prefix was used that did not start with a "/" (which is the default case for OpenShift). Now, the same code is used to build the root etcd prefix for both etcdv2 and etcdv3, meaning that after a migration, the cluster is able to find migrated data as expected. (**BZ#1393744**)

**Routing**

- The max connection was too low, causing the pod to restart. With this fix, the default value of the connection was increased. As a result, the pod does not restart. (**BZ#1405440**)

- Previously, if you created two ipfailover instances and had them run on the same node, it would fail because both would to use hostPort 1985. This was corrected by using the ServicePort as a mechanism to prevent multiple pods for same configuration from starting on the same node. (**BZ#1411501**)

- Previously, as routers were removed, the route status was not regularly cleared. This fix added a script to clean out the defunct route status, and documented expectations of operators. As a result, route statuses are clear and correct. (**BZ#1356819**)

- Previously, permissions would reset to preset values on a periodic basis causing the scripts to lose execute permissions. This fix set the correct preset value in the RC. (**BZ#1408172**)

- Previously, default host name generation did not take into account that routes could have the "." character. Therefore when a generated host name was used for a route that included a "." in the name, and had allowed **`wildcardpolicy`**, there would be an extra subdomain. This fix changed the host name generator to replace "." in a route's name to "-" in the generated host name. As a result, generated host names cannot create additional subdomains. (**BZ#1414956**)

- To match user expectations, this feature makes the default for routes with multiple active services be round-robin. Without this feature, users needed to set an annotation on a route as well as weights to make it behave correctly. (**BZ#1416869**)

- Previously, re-encryption routes were not correctly supporting redirect access from HTTP to HTTPS. As a result, it was not possible to set a re-encrypt routes insecure termination policy to redirect. The HAproxy template file was edited to correctly implement redirect as a valid insecure termination policy for redirect routes. Now re-encrypt routes can be configured to redirect HTTP to HTTPS traffic. (**BZ#1317159**)

**Storage**

- Previously, the Azure provisioner was not enabled, causing a failure to provision Azure disks. This fix enabled the Azure provisioner. As a result, it is now able to provision Azure disks. (**BZ#1415466**)

- Previously, OpenShift Container Platform used the wrong **`InstanceID`** for checking that volumes were attached to nodes, causing it to think that a volume was detached while it is still attached. This resulted in volumes remaining attached when they were not needed, and unable to be deleted according to their reclaim policy. With this fix, OpenShift Container Platform now uses the right **`InstanceID`** for all attach, detach, and check operations. And as a result, volumes are detached and deleted when they are not needed. (**BZ#1420645**)

- Previously, **ceph-common** packages were not installed in the infra container, causing failure to provision Ceph RBD volumes. With this fix, **ceph-common** packages are installed in the infra container. As a result, Ceph RBD volumes now provision correctly. (**BZ#1420698**)

- Previously, the AWS device IDs were incorrect. This caused failure to attach EBS volume due to **`InvalidParameterValue`** for the parameter device. This fix updated the AWD device IDs, and as a result, the EBS volume is successfully attached. (**BZ#1422457**)

- Previously, OpenShift Container Platform contained a race condition in NFS recycler handling. This caused some pods to fail to start, and failed to recycle the corresponding NFS share when recycler pods for multiple NFS shares were started at the same time. With this fix, the race condition was corrected. As a result, all scheduled NFS recycler pods are started and NFS shares are recycled. (**BZ#1392338**)

- Previously, the device name provided by Cinder was being used for volume mounting into a pod, however, the device name provided by Cinder is unreliable for the actual mounting. This caused some Cinder volumes to fail to be mounted into a pod, and resulted in an inconclusive message to appear in the logs. This fix enables a detection to be performed using the Cinder ID. As a result, Cinder volumes are reliably being mounted into appropriate pods. (**BZ#1408867**)

- Previously, the same iSCSI device could not successfully be used by multiple pods on same node. When one pod would shut down, the iSCSI device for the other pod would be unavailable. The code was changed with this fix. As a result, the iscsi device are successfully run. (**BZ#1426778**, **BZ#1426775**)

- Previously, if a mount was in progress and pod was deleted, the pod would fail to be cleaned up properly. This meant the pod was left with volumes attached to the node. This fix makes sure that the pending operation is completed before volume is unmounted from node. As a result, the pod gets cleaned up properly even if mount was in flight when deletion request is received. (**BZ#1432949**)

## 2.6. TECHNOLOGY PREVIEW FEATURES

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

Technology Preview Features Support Scope

The following new features are now available in Technology Preview:

- Kubelet Collection of Node Attributes for Scheduling Considerations
- `StatefulSets`
- Application Service Certificate Regeneration
- Network Policy Plug-in
- Kubelet Collection of Node Attributes for Scheduling Considerations
- Ingress Object Support
- Init containers

The following features that were formerly in Technology Preview from a previous OpenShift Container Platform release remain in Technology Preview:

- Kubernetes Deployments Support -Pod Distribution Budgets
- Cron Jobs (formerly called Scheduled Jobs)

See more details on technical changes related to Cron Jobs in OpenShift Container Platform 3.5.

## 2.7. KNOWN ISSUES

- In OpenShift Container Platform 3.4, the master connected to the etcd cluster using the host name of the etcd endpoints. In OpenShift Container Platform 3.5, the master now connects to etcd via IP address. When configuring a cluster to use proxy settings, this change causes the

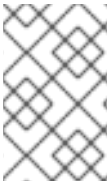
master-to-etcd connection to be proxied as well, rather than being excluded by host name in each host's **NO_PROXY** setting.

Workarounds for setting the IP addresses manually in each host's **NO_PROXY** setting are documented in the installation and upgrade steps. The installer will be updated in a future release to handle this scenario automatically during installation and upgrades. (**BZ#1466783**)

## 2.8. ASYNCHRONOUS ERRATA UPDATES

Security, bug fix, and enhancement updates for OpenShift Container Platform 3.5 are released as asynchronous errata through the Red Hat Network. All OpenShift Container Platform 3.5 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 3.5. Versioned asynchronous releases, for example with the form OpenShift Container Platform 3.5.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 upgrading your cluster properly.

### 2.8.1. RHBA-2017:0903 - atomic-openshift-utils Bug Fix and Enhancement Update

Issued: 2017-04-12

OpenShift Container Platform bug fix and enhancement advisory RHBA-2017:0903, providing updated **atomic-openshift-utils**, **ansible**, and **openshift-ansible** packages that fix several bugs and add enhancements, is now available.

Space precluded documenting all of the bug fixes and enhancements for this release in the advisory. See the following sections for notes on upgrading and details on the bug fixes and enhancements included in this release.

#### 2.8.1.1. Upgrading

To apply this update, run the following on all hosts where you intend to initiate Ansible-based installation or upgrade procedures:

```
# yum update atomic-openshift-utils
```

#### 2.8.1.2. Bug Fixes

- When CloudFront was enabled, the installer did not use the private key for the registry, and the registry failed to deploy successfully. This bug fix adds new steps to ensure the private key creates a secret and attaches to the CloudFront registry. (**BZ#1395168**)

- Previously, the facts generation procedures may have incorrectly determined major release versions prior to package installation. Because the playbooks are now version specific, this defaulting has been eliminated, ensuring that OpenShift Container Platform 3.5 playbooks receive 3.5 content in all scenarios. (**BZ#1395637**)

- OpenShift Container Platform 3.4 and 3.3 introduced a requirement on the **`conntrack`** executable, but this dependency was not enforced at install time. This made it possible for service proxy management to fail post installation. This bug fix updates the installer to ensure that **`conntrack`** is installed. (**BZ#1420182**)

- An Ansible release introduced a regression that caused datastructures to fail to serialize when writing them out to a YAML document. Users would trigger the regression during the pre-run fact fetching, causing their installation to crash. Ansible introduced a new YAML serializing system in an update. The old serializing system was replaced with the new one, **`AnsibleDumper`**. As a result, the quick installer can run the "Gathering information from hosts" actions now without triggering the error during serializing. (**BZ#1420970**)

- Previously, if **`ansible_user`** was a Windows domain user in the format of **`dom\user`**, the installation playbooks would fail. This bug fix escapes this user name properly, ensuring playbooks run successfully. (**BZ#1426703**)

- In containerized environments, the CNI data directory located at ***/var/lib/cni*** was not properly configured to persist on the node host. This bug fix updates the installer to ensure that pod IP allocation data is persisted when restarting containerized nodes. (**BZ#1427789**)

- The command line option that flags unattended mode was not being checked when the scaleup routine was ran, and users would be prompted to enter host information. This bug fix ensures the unattended mode flag is checked during the scaleup routine. As a result, users are kicked out and given instructions on how to continue if the unattended mode flag is set during a scaleup run. (**BZ#1390135**)

- A **when** clause was present on the **firewalld** service installation task, causing the installation to be skipped for **firewalld** when running a containerized install. This bug fix removes the **when** clause from the **firewalld** installation task, and as a result **firewalld** is installed properly when running a containerized install. (**BZ#1413447**)

- A custom systemd unit file was used for the **docker** service specifying **`Requires=iptables`**. This resulted in the **docker** service being restarted when **iptables** was restarted. This bug fix updates the custom systemd unit file to specify **`Wants=iptables`**. This will still ensure that **iptables** is started before **docker**, but will not cause **docker** to restart when **iptables** is restarted. (**BZ#1416156**)

- Re-running the installation playbook with **`openshift_hosted_logging_deploy=true`** would redeploy the logging deployer pod using the install mode and would fail because logging was already installed. The Ansible playbook fails due to waiting on the deployer pod to complete successfully. In OpenShift Container Platform 3.5, the logging deployer pod is no longer used to install logging, but rather the **`openshift_logging`** role. As a result, it is able to handle previously installed logging, and the playbook now completes successfully. (**BZ#1417525**)

- The fact **`etcd_is_atomic`** was detected incorrectly due to the role ordering of some fact setting

operations. Atomic Host systems do not support yum, repoquery, and rpm commands; Atomic Host systems would attempt to run commands specific to managing and inspecting repositories and packages when they should not. This bug fix changes the ordering of role calls and fact updates and wrapped in a meta-role to ensure they stay in the correct order. As a result, Atomic Host systems will no longer attempt to run the problematic commands, because the **`etcd_is_atomic`** fact is now correctly detected. (BZ#1427067)

- Previously, **atomic-openshift-docker-excluder** was disabled before the **docker** installation, and **docker** could be installed with newer version that is not compatible. This bug fix enables **atomic-openshift-docker-excluder** during the **docker** installation so that **docker** is installed with a version that is compatible. (BZ#1430612)

- Previously, **atomic-openshift-excluder** was not enabled after installation, meaning OpenShift Container Platform components were not protected from accidental package updates. This bug fix enables **atomic-openshift-excluder** correctly. (BZ#1430613)

- The example inventories were incorrect for the logging public master URL, and **`loggingPublicURL`** was not being set as expected. This bug fix updates the example inventories to be accurate for the new logging role. As a result, **`loggingPublicURL`** is correctly set as expected. (BZ#1430628)

- Previously, the **atomic-openshift-excluder** and **atomic-openshift-docker-excluder** packages were not acknowledged during node or master scale-up. This meant that these excluder packages were not installed on new nodes or masters. This bug fix ensures that the excluder packages are installed on new nodes and masters as well. As a result, the excluder packages are installed on new nodes and masters when scaling up. (BZ#1432868)

- The quick installer used a system for counting the number of plays in a run that was not 100% accurate due to conditional play includes. The reported number of plays could be bigger or smaller than the original estimate. With this bug fix, at the end of the playbook run, the installer now explains why the actual play count may be different than the estimate. Users still have an idea of about how far along their install is and if the number of tasks do not match the original estimate they understand why. (BZ#1388739)

- A bug was fixed in the **openvswitch** upgrade to v2.6 (BZ#1420636)

- Automatic migration is not possible from **`PetSets`** to **`StatefulSets`**. An additional validation step was added to the pre-upgrade validation playbook. **`PetSets`** are searched for in the cluster and if any existing **`PetSets`** are detected, the installation errors and quits. The user is given an information message (including documentation references) describing what went wrong, why, and what the users choices are for continuing the upgrade without migrating **`PetSets`**. (BZ#1428229)

- In some situations, node upgrade could terminate a running pod that was upgrading the router or registry, causing the router or registry to fail to upgrade. The router and registry upgrade is now re-ordered to follow node upgrade when performing a full cluster in-place upgrade. As a result, nodes are no longer taken offline for upgrade while the router or registry is still running. (BZ#1395081)

- Previously, an error in the upgrade playbooks prevented ansible from detecting when a host had successfully been rebooted. This error has been corrected and upgrades that use **`openshift_rolling_restart_mode=system`** now work properly. (BZ#1421002)

- The **atomic-openshift-excluder** and **atomic-openshift-docker-excluder** packages are now updated to the latest available packages when upgrading OpenShift Container Platform. (BZ#1426070)

- The **atomic-openshift-docker-excluder** package was not updated during containerized cluster upgrades. If this package was not up to date, the cluster was not protected from accidentally upgrading to an unsupported **docker** version. This bug fix ensures the package is now updated during containerized upgrade. (**BZ#1430700**)

- Previously, **tar** was not listed as a dependency for the installer. On systems where **tar** was not part of the base system, the installer could fail. This bug fix adds **tar** as a dependency, and as a result the installer is now able to use **tar** during installations and upgrades. (**BZ#1388445**)

- The upgrade plays were updating the excluder packages on Atomic Host systems. This caused the plays to fail as the excluders are not supported on Atomic Host. This bug fix skips excluders on Atomic Host systems, and as a result these plays no longer fail. (**BZ#1431077**)

- The default **docker** log driver has been changed to **journald** in order to provider higher performance and lower logging latency. (**BZ#1388191**)

- Previously, the file specified in **`openshift_master_ca_certificate`** was not deployed when performing a master scaleup. The scaleup playbooks have been updated to ensure that this certificate is deployed. (**BZ#1427003**, **BZ#1426677**)

- When using **`grep`** to find **`pluginOrderOverride`** within the ***/etc/origin/master/master-config.yaml*** file, if the string was not found the task failed, causing the playbook to fail. This bug fix updates the task to no longer fail if the RC != 0 (the string was not found). As a result, if the string is missing, the playbook no longer fails and instead continues to run as expected. (**BZ#1425400**)

- Previously, the registry certificate was not properly updated when running the certificate re-deploy playbooks, which may have prevented pushing or pulling images. This bug fix updates the playbooks to ensure that the registry certificate is correctly updated. (**BZ#1418191**)

- Previously, the installer may have failed to add **iptables** rules if other **iptables** rules were being updated at the same time. This bug fix updates the installer to wait to obtain a lock when updating **iptables** rules, ensuring that rules are properly created. (**BZ#1415800**)

- The certificate re-deploy playbooks have been updated to ensure all internal certificates have been updated and, when possible, the update is done in a rolling manner, preventing downtime. See Redeploying Certificates for the latest information on updating certificates. (**BZ#1397958**)

- The registry console deployment now allows both the prefix and version to be specified if the user needs to reference an alternate registry or specific version of the registry console. These values may be configured by setting, for example, **`openshift_cockpit_deployer_prefix='registry.example.com/openshift'`** and **`openshift_cockpit_deployer_version='3.5.0'`**, which would result in use of **`registry.example.com/openshift/registry-console:3.5.0`**. (**BZ#1383275**)

#### 2.8.1.3. Enhancements

- This enhancement enables the **`gitNoProxy`** default value for builds to be set explicitly via the installer. Previously, the value was inherited from the **`no_proxy`** settings for the installer, which was insufficiently configurable. The build default **`gitNoProxy`** value can now be directly controlled by the **`openshift_builddefaults_git_no_proxy`** setting. (**BZ#1384753**)

- This enhancement adds the new option **`openshift_docker_selinux_enabled`**. This allows users to override the default installation **docker** options setting of **`--selinux-enabled`**. Placing **`openshift_docker_selinux_enabled=false`** in user inventory file will remove the **`--selinux-enabled`** docker option. (**BZ#1392742**)

### 2.8.2. RHBA-2017:1129 - OpenShift Container Platform 3.5.5.8 Bug Fix and Enhancement Update

Issued: 2017-04-26

OpenShift Container Platform release 3.5.5.8 is now available. The list of packages, bug fixes, and enhancements included in the update are documented in the RHBA-2017:1129 advisory. The list of container images included in the update are documented in the RHBA-2017:1130 advisory.

#### 2.8.2.1. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.3. RHBA-2017:1235 - OpenShift Container Platform 3.5.5.15 Bug Fix Update

Issued: 2017-05-18

OpenShift Container Platform release 3.5.5.15 is now available. The list of packages and bug fixes included in the update are documented in the RHBA-2017:1235 advisory. The list of container images included in the update are documented in the RHBA-2017:1236 advisory.

#### 2.8.3.1. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.4. RHBA-2017:1425 - OpenShift Container Platform 3.5.5.24 Bug Fix Update

Issued: 2017-06-15

OpenShift Container Platform release 3.5.5.24 is now available. The packages and bug fixes included in the update are documented in the RHBA-2017:1425 advisory. The container images included in the update are provided by the RHBA-2017:1426 advisory and listed in Images.

Space precluded documenting all of the bug fixes and images for this release in the advisory. See the following sections for notes on upgrading and details on the bug fixes and images included in this release.

#### 2.8.4.1. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

#### 2.8.4.2. Bug Fixes

- Containers were being force killed by the build process. Some container resources were not freed when they were force killed, resulting in resource exhaustion and the inability to run new builds. Containers are now gracefully killed, allowing them to clean up their resources. Resource exhaustion no longer occurs and builds continue to run normally. (**BZ#1437121**)

- The OpenShift Container Registry (OCR) takes the **`dockerImageReference`** from an image object. The **`dockerImageReference`** is shared across image streams and the same image is fetched using the same **`dockerImageReference`** for all image streams. Now, the

**dockerImageReference** is taken from an image stream. OCR fetches the image from different places for different image streams. (**BZ#1433232**)

- This bug fix adjusted conditionals to allow audit configuration for non-HA environments. (**BZ#1447019**)
- Missing rules in the Network Policy SDN plug-in did not allow proper off-cluster access. Off-cluster resources were not reachable. The rules are corrected and resources accessible. (**BZ#1445500**)
- Multiple node IP addresses were reported in random order by node status. Consequently, the SDN controller picked up a random one each time. This bug fix maintains the stickiness of the IP once it is chosen until valid, and IP addresses are no longer switched unexpectedly. (**BZ#1451830**)
- The ARP cache size tuning parameters were not set when performing an installation on bare metal hosts. The bare metal profiles are now updated to ensure that the ARP cache is set correctly on bare metal hosts. (**BZ#1452235**)
- If the pod was using a persistent volume and had been deleted while the controller was down, the volume was never detached from the node. The restarted controller was not able to find the attached volume and never tried to detach it. Now, the fixed controller examines the node volumes on startup, determines which ones need to be detached, then detaches them properly. (**BZ#1377486**)
- Volumes attached to non-running AWS instances were being incorrectly marked as detached by a periodic routine that verified if volumes were attached because non-running AWS instances were not considered nodes by the routine. Volumes that were incorrectly marked detached were never detached if or when they needed to be later. By considering non-running AWS instances to be nodes in the routine, the issue is fixed. Volumes attached to non-running AWS instances are correctly tracked as attached and will be detached when they need to be later. (**BZ#1455675**)

#### 2.8.4.3. Images

This release updates the Red Hat Container Registry (**registry.access.redhat.com**) with the following images:

```
openshift3/ose-pod:v3.5.5.24-2
rhel7/pod-infrastructure:v3.5.5.24-2
openshift3/ose:v3.5.5.24-2
openshift3/ose-docker-registry:v3.5.5.24-2
openshift3/ose-egress-router:v3.5.5.24-2
openshift3/ose-keepalived-ipfailover:v3.5.5.24-2
openshift3/ose-f5-router:v3.5.5.24-2
openshift3/ose-deployer:v3.5.5.24-2
openshift3/ose-haproxy-router:v3.5.5.24-2
openshift3/node:v3.5.5.24-2
openshift3/ose-recycler:v3.5.5.24-2
openshift3/ose-sti-builder:v3.5.5.24-2
openshift3/ose-docker-builder:v3.5.5.24-2
openshift3/logging-deployer:v3.5.5.24-2
openshift3/metrics-deployer:v3.5.5.24-2
openshift3/openvswitch:v3.5.5.24-2
openshift3/logging-auth-proxy:3.5.0-15
openshift3/logging-curator:3.5.0-13
```

```
openshift3/logging-elasticsearch:3.5.0-23
openshift3/logging-fluentd:3.5.0-15
openshift3/logging-kibana:3.5.0-14
openshift3/metrics-cassandra:3.5.0-18
openshift3/metrics-hawkular-metrics:3.5.0-21
openshift3/metrics-hawkular-openshift-agent:3.5.0-15
openshift3/metrics-heapster:3.5.0-15
openshift3/registry-console:3.5-13
```

### 2.8.5. RHBA-2017:1492 - OpenShift Container Platform 3.5.5.26 Bug Fix Update

Issued: 2017-06-20

OpenShift Container Platform release 3.5.5.26 is now available. The list of packages and bug fixes included in the update are documented in the RHBA-2017:1492 advisory. The list of container images included in the update are documented in the RHBA-2017:1493 advisory.

#### 2.8.5.1. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.6. RHBA-2017:1666 - atomic-openshift-utils Bug Fix and Enhancement Update

Issued: 2017-06-29

OpenShift Container Platform bug fix and enhancement advisory RHBA-2017:1666, providing updated **atomic-openshift-utils** and **openshift-ansible** packages that fix several bugs and add an enhancement, is now available.

Space precluded documenting all of the bug fixes and enhancements for this release in the advisory. See the following sections for notes on upgrading and details on the bug fixes and enhancements included in this release.

#### 2.8.6.1. Upgrading

To apply this update, run the following on all hosts where you intend to initiate Ansible-based installation or upgrade procedures:

```
# yum update atomic-openshift-utils
```

#### 2.8.6.2. Bug Fixes

- During control plan upgrades, a subset of pre-check and verification tasks for upgrades is run. However, the tasks were run over non-control plane nodes as well. Some of the tasks need ***-excluder** RPMs to be disabled in order to work properly. Given the excluders are disabled on control plane hosts only, the tasks run over the remaining nodes caused a failure. With this bug fix, all pre-check and verification tasks are run over control plane nodes only. (**BZ#1440167**)
- Starting with OpenShift Container Platform 3.4, OpenShift's SDN plug-ins no longer reconfigure the **`docker`** bridge MTU; instead, pods are configured properly on creation. Because of this change, non-OpenShift containers may have an MTU configured that is too large to allow access to hosts on the SDN. This bug fix updates the installer to align the MTU setting for the **`docker`** bridge with the MTU used inside the cluster, thus avoiding the problem. (**BZ#1460235**)

- The OpenShift CA redeployment playbook (***playbooks/byo/openshift-cluster/redeploy-openshift-ca.yml***) would fail to restart services if certificates were previously expired. This bug fix ensures that service restarts are now skipped within the OpenShift CA redeployment playbook when expired certificates are detected. Expired cluster certificates may be replaced with the certificate redeployment playbook (***playbooks/byo/openshift-cluster/redeploy-certificates.yml***) after the OpenShift CA certificate has been replaced via the OpenShift CA redeployment playbook. (**BZ#1460969**)

- Previously, installation would fail in multi-master environments in which the load balanced API was listening on a different port than that of the OpenShift Container Platform API and web console. This bug fix accounts for this difference and ensures the master loopback client configuration is configured to interact with the local master. (**BZ#1462276**)

- The use of **`oc patch`** to update router images was setting additional configuration items to defaults, even if they were configured differently in the environment. This bug fix converts those tasks to use Ansible modules, which are much more precise and change only the provided parameter. (**BZ#1462721**)

- When using the **`openshift_upgrade_nodes_label`** variable during upgrades, if the label did not match any hosts, the upgrade would silently proceed with upgrading all nodes given. This bug fix verifies the provided label matches a set of hosts prior to upgrading, and the upgrade fails if no nodes match. (**BZ#1462995**)

- During certificate expiration checking or redeployment, certificiates with large serial numbers could not be parsed using the existing manual parser workaround on hosts that were missing the OpenSSL python library. This bug fix updates the manual parser to account for the format of certificates with large serial numbers. As a result, these certificates can now be parsed. (**BZ#1464543**)

#### 2.8.6.3. Enhancements

- Previously, it was only possible to redeploy the etcd CA certificate by also redeploying the OpenShift CA certificate, which was unnecessary maintenance. With this enhancement, the etcd CA certificate may now be replaced independent of the OpenShift CA certificate using the etcd CA certificate redeployment playbook (***playbooks/byo/openshift-cluster/redeploy-etcd-ca.yml***). Note that the OpenShift CA redeployment playbook (***playbooks/byo/openshift-cluster/redeploy-openshift-ca.yml***) now only replaces the OpenShift CA certificate. Similarly, the etcd CA redeployment playbook only redeploys the etcd CA certificate. (**BZ#1463772**)

### 2.8.7. RHBA-2017:1640 - OpenShift Container Platform 3.5.5.31 Bug Fix Update

Issued: 2017-07-11

OpenShift Container Platform release 3.5.5.31 is now available. The packages and bug fixes included in the update are documented in the RHBA-2017:1640 advisory. The container images included in the update are documented in the RHBA-2017:1646 advisory.

Space precluded documenting all of the bug fixes for this release in the advisory. See the following sections for notes on upgrading and details on the bug fixes included in this release.

#### 2.8.7.1. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

#### 2.8.7.2. Bug Fixes

- When doing an incremental build, the S2I builder pulls its builder image before calling the ***save-artifacts*** script and does not ensure that the builder image is still there when it calls ***assemble***. This leaves a gap of time between the start of the build and the calling of the ***assemble*** script in which the image can be removed. If the image is removed, the build fails. This bug fix adds a call to ensure that the builder image exists right before calling the ***assemble*** script. As a result, the chance of the ***assemble*** script running and not finding an available builder image is greatly reduced. (**BZ#1464537**)

- The Elasticsearch (ES) default value for sharing storage between ES instances was wrong. The incorrect default value allowed an ES pod starting up (when another ES pod was shutting down, e.g., during deployment configuration redeployments) to create a new location on the persistent volume (PV) for managing the storage volume. This duplicated data, and in some instances, potentially caused data loss. With this bug fix, all ES pods now run with `node.max_local_storage_nodes` set to `1`. As a result, the ES pods starting up or shutting down will no longer share the same storage, preventing data duplication and data loss. (**BZ#1463046**)

- The version of Netty that is part of Cassandra 3.0.9 had a memory leak. This bug fix updates Cassandra to 3.0.13, which has a version of Netty that has a fix for the memory leak. (**BZ#1457501**)

- VNID allow rules were being incorrectly removed before they were actually no longer in use. When containers had startup errors, it could cause the tracking to get out of sync. The rules that allowed communication for a namespace were removed too early, so that if there were still pods in that namespace on the node, they could not communicate with one another. This bug fix changes the way that the tracking is done to avoid edge cases around pod creation and deletion failures. As a result, the VNID tracking no longer fails, allowing traffic to flow. (**BZ#1462338**)

- When an IP address was re-used, it would be generated with a random MAC address that would be different from the previous one. Any node with an ARP cache that still held the old entry for the IP would not be able to communicate with the node. This bug fix generates the MAC address deterministically from the IP address. As a result, a re-used IP address will always have the same MAC address, so the ARP cache no longer gets out of sync, allowing traffic to flow. (**BZ#1462955**)

- Due to a coding error, `Pop()` operations could panic and cause the router to stop. This bug fix updates this logic and as a result panics no longer occur. (**BZ#1464563**)

- When master controller routines watch for persistent volume Recycle success events, they may never receive one, but still keep trying indefinitely. This caused the potential for high CPU usage by the master controller as it leaks routines. This bug fix updates the routines to stop watching for these Recycle success events when they will never be received. As a result, the chance of high CPU usage by the master controller is reduced. (**BZ#1460289**)

### 2.8.8. RHBA-2017:1828 - OpenShift Container Platform 3.5.5.31 Bug Fix Update

Issued: 2017-08-31

OpenShift Container Platform release 3.5.5.31 is now available. The packages and bug fixes included in the update are documented in the RHBA-2017:1828 advisory. The container images included in the update are provided by the RHBA-2017:1829 advisory and listed in Images.

Space precluded documenting all of the images for this release in the advisory. See the following sections for notes on upgrading and details on the images included in this release.

#### 2.8.8.1. Images

This release updates the Red Hat Container Registry (**`registry.access.redhat.com`**) with the following images:

```
openshift3/ose-pod:v3.5.5.31.19-2
rhel7/pod-infrastructure:v3.5.5.31.19-2
openshift3/ose-ansible:v3.5.5.31.19-2
openshift3/ose:v3.5.5.31.19-2
openshift3/ose-docker-registry:v3.5.5.31.19-2
openshift3/ose-egress-router:v3.5.5.31.19-2
openshift3/ose-keepalived-ipfailover:v3.5.5.31.19-2
openshift3/ose-f5-router:v3.5.5.31.19-2
openshift3/ose-deployer:v3.5.5.31.19-2
openshift3/ose-haproxy-router:v3.5.5.31.19-2
openshift3/node:v3.5.5.31.19-2
openshift3/ose-recycler:v3.5.5.31.19-2
openshift3/ose-sti-builder:v3.5.5.31.19-2
openshift3/ose-docker-builder:v3.5.5.31.19-2
openshift3/logging-deployer:v3.5.5.31.19-2
openshift3/logging-curator:v3.5.5.31.23-2
openshift3/metrics-deployer:v3.5.5.31.19-2
openshift3/openvswitch:v3.5.5.31.19-2
openshift3/logging-auth-proxy:3.5.0-28
openshift3/logging-elasticsearch:3.5.0-37
openshift3/logging-fluentd:3.5.0-26
openshift3/logging-kibana:3.5.0-30
openshift3/metrics-cassandra:3.5.0-33
openshift3/metrics-hawkular-metrics:3.5.0-36
openshift3/metrics-hawkular-openshift-agent:3.5.0-26
openshift3/metrics-heapster:3.5.0-26
openshift3/registry-console:3.5-26
```

#### 2.8.8.2. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.9. RHBA-2017:2670 - OpenShift Container Platform 3.5.5.31.24 Bug Fix Update

Issued: 2017-09-07

OpenShift Container Platform release 3.5.5.31.24 is now available. The packages and bug fixes included in the update are documented in the RHBA-2017:2670 advisory. The container images included in the update are provided by the RHBA-2017:2643 advisory.

#### 2.8.9.1. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.10. RHBA-2017:3049 - OpenShift Container Platform 3.5.5.31.36 Bug Fix Update

Issued: 2017-10-25

OpenShift Container Platform release 3.5.5.31.36 is now available. The list of packages included in the update are documented in the RHBA-2017:3049 advisory. The container images included in the update are provided by the RHBA-2017:3050 advisory.

Space precluded documenting all of the bug fixes and images for this release in the advisories. See the following sections for notes on upgrading and details on the bug fixes and images included in this release.

### 2.8.10.1. Bug Fixes

#### Image Registry

- The size of cached layers was previously uncounted, causing an image's layer size for cached layers to be zero. This bug fix ensures cached layers are now properly counted, and as a result images now have the proper layer sizes. (**BZ#1457043**)

- The OpenShift Container Registry used to append the forwarded target port to redirected location URLs. The registry client would get confused by the received location containing a superfluous port, and could not match it against the original host. This happened when exposed with TLS-termination other than passthrough. The client's new request to the target location lacked credentials, and as a consequence, the image push failed due to authorization error. This bug fix rebases the registry to a newer version, which fixes forwarding processing logic. As a result, the registry no longer confuses its clients; clients can push images successfully to the exposed registry using arbitrary TLS-termination. (**BZ#1489039**)

- Images younger than the threshold were not added to the dependency graph. Blobs used by a young image and by a prunable image were deleted because they had no references in the graph. This bug fix adds young images to the graph and marks them as non-prunable. As a result, blobs now have references and are not deleted. (**BZ#1498123**)

- Neither documentation nor CLI help talked about insecure connections to the secured registry. Errors used to be hard to decipher when users attempted to prune the secured registry with a bad CA certificate. This bug fix ensures that errors are now printed with hints, CLI help has been updated, and new flags have been provided to allow for insecure fallback. As a result, users can now easily enforce both secure and insecure connections and understand any HTTPS errors and how to resolve them. (**BZ#1474446**)

#### Logging

- Messages were previously read into Fluentd's memory buffer and were lost if the pod was restarted. Because Fluentd considers them read even though they have not been pushed to storage, any message not stored but already read by Fluentd was lost. This bug fix replaces the memory buffer with a file-based buffer. As a result, file-buffered messages are pushed to storage once Fluentd restarts. (**BZ#1477515**)

- The pattern for container logs in the journal field **`CONTAINER_NAME`** changed. The pattern was not matching for logs from pods in the **`default`**, **`openshift`**, or **`openshift-infra`** namespaces. Logs from these namespaces were being stored in indices matching **`project.default.*`**, for example rather than **`.operations.*`**. This bug fix updates the pattern matcher to match the correct pattern. As a result, logs from pods in the affected namespaces are correctly written to the **`.operations.*`** indices. (**BZ#1494310**)

- Memory was not being set properly for the Kibana container. This bug fix uses underscores instead of dashes, and memory settings are now honored by the Node.js runtime. (**BZ#1469711**)

- When moving to use the journald log driver instead of the json-file log driver, the code that

parses the journald Kubernetes records was not preserving the **message** field created by the Kubernetes meta filter plug-in. This caused the raw JSON message to be put back into the **message** field. This bug fix preserves the **message** field created when the Kubernetes meta filter parses the JSON blob. As a result, the parsed **message** field is preserved. (BZ#1439504)

**Web Console**

- Previously, the web console Overview page would not finish loading if a resource name contained only digits, for example a deployment configuration "54321". This bug fix updates the web console, and the Overview page now works as expected for any valid resource name. (BZ#1485892)

**Master**

- In some failure cases, the etcd client used by OpenShift will not rotate through all the available etcd cluster members. The client will end up repeatedly trying the same server. If that server is down, then requests will fail for an extended time until the client finds the server invalid. If the etcd leader goes away when it is attempted to be contacted for something like authentication, then the authentication fails and the etcd client is stuck trying to communicate with the etcd member that does not exist. User authentication would fail for an extended period of time. With this bug fix, the etcd client now rotates to other cluster members even on failure. If the etcd leader goes away, the worst that should happen is a failure of that one authentication attempt. The next attempt will succeed because a different etcd member will be used. (BZ#1490428)

- The upstream discovery was not set to prefer version v1. It preferred the API group version instead. The old **oc** client was able to get discovery information when talking to the newer server and prefer the grouped API version of the resource. However, the version was not recognized. With this bug fix, the upstream directory is now set to prefer version v1. (BZ#1463576)

**Networking**

- Conntrack entries for UDP traffic were not cleared when an endpoint was added for a service that previously had no endpoints. The system could end up incorrectly caching a rule that would cause traffic to that service to be dropped rather than being sent to the new endpoint. With this bug fix, the relevant conntrack entries are now deleted at the right time and UDP services work correctly when endpoints are added and removed. (BZ#1497768)

**Pod**

- When describing multiple instances on AWS, each node is supplied as a filter. This fails to work if the cluster is large enough because AWS only allows up to 200 filters to a request. As a result, **DescribeInstances** calls fail, resulting in broken load balancer and storage functionality in AWS. This bug fix implements batching of describeinstance calls to get over the filtering limit. **DescribeInstances** calls now also work for larger clusters. (BZ#1460388)

- Disabling the use of the proxy via **--disable-proxy** triggers a panic because the service stores have nil values. When disabling the proxy, the node will never start leaving the system in an indeterminate state. With this bug fix, the logic was reworked to ensure that the service stores are populated with non-nil values when the proxy is disabled. Using **--disable=proxy** no longer causes a panic and overall node start failure. (BZ#1484272)

**Storage**

- When the **atomic-openshift-node** service was restarted, all processes in its control group were terminated, including the glusterfs mounted points. Each glusterfs volume in OpenShift corresponds to one mounted point. If all mounting points are lost, so are all of the volumes. This

bug fix sets the control group mode to terminate only the main process and leave the remaining glusterfs mounting points untouched. When the **`atomic-openshift-node`** service is restarted, no glusterfs mounting point is terminated. (**BZ#1472370**)

#### 2.8.10.2. Images

This release updates the Red Hat Container Registry (**`registry.access.redhat.com`**) with the following images:

```
openshift3/logging-curator:v3.5.5.31.36-4
openshift3/logging-elasticsearch:3.5.0-46
openshift3/logging-kibana:3.5.0-43
openshift3/metrics-deployer:v3.5.5.31.36-5
openshift3/metrics-heapster:3.5.0-33
openshift3/ose-deployer:v3.5.5.31.36-4
openshift3/ose:v3.5.5.31.36-4
openshift3/ose-egress-router:v3.5.5.31.36-4
openshift3/ose-haproxy-router:v3.5.5.31.36-4
openshift3/openvswitch:v3.5.5.31.36-5
openshift3/ose-recycler:v3.5.5.31.36-4
openshift3/ose-sti-builder:v3.5.5.31.36-4
openshift3/registry-console:3.5.0-33
openshift3/ose-f5-router:v3.5.5.31.36-4
openshift3/logging-auth-proxy:3.5.0-37
openshift3/logging-deployer:v3.5.5.31.36-5
openshift3/logging-fluentd:3.5.0-38
openshift3/metrics-cassandra:3.5.0-41
openshift3/metrics-hawkular-metrics:3.5.0-49
openshift3/metrics-hawkular-openshift-agent:3.5.0-33
openshift3/ose-docker-builder:v3.5.5.31.36-4
openshift3/ose-docker-registry:v3.5.5.31.36-4
openshift3/ose-keepalived-ipfailover:v3.5.5.31.36-4
openshift3/node:v3.5.5.31.36-5
openshift3/ose-pod:v3.5.5.31.36-4
```

#### 2.8.10.3. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.11. RHSA-2017:3389 - Moderate: OpenShift Container Platform 3.5.5.31.47 Security, Bug Fix, and Enhancement Update

Issued: 2017-12-06

OpenShift Container Platform release 3.5.5.31.47 is now available. The list of packages included in the update are documented in the RHSA-2017:3389 advisory. The container images included in the update are provided by the RHBA-2017:3390 advisory.

Space precluded documenting all of the bug fixes, enhancements, and images for this release in the advisories. See the following sections for notes on upgrading and details on the bug fixes and images included in this release.

#### 2.8.11.1. Images

This release updates the Red Hat Container Registry (**`registry.access.redhat.com`**) with the following images:

```
openshift3/logging-curator:v3.5.5.31.47-10
openshift3/logging-deployer:v3.5.5.31.47-10
openshift3/metrics-deployer:v3.5.5.31.47-10
openshift3/node:v3.5.5.31.47-10
openshift3/openvswitch:v3.5.5.31.47-10
openshift3/ose-ansible:v3.5.5.31.47-10
openshift3/ose-base:v3.5.5.31.47-10
openshift3/ose-deployer:v3.5.5.31.47-10
openshift3/ose-docker-builder:v3.5.5.31.47-10
openshift3/ose-docker-registry:v3.5.5.31.47-10
openshift3/ose-egress-router:v3.5.5.31.47-10
openshift3/ose-f5-router:v3.5.5.31.47-10
openshift3/ose-haproxy-router:v3.5.5.31.47-10
openshift3/ose-keepalived-ipfailover:v3.5.5.31.47-10
openshift3/ose-pod:v3.5.5.31.47-10
openshift3/ose-recycler:v3.5.5.31.47-10
openshift3/ose-sti-builder:v3.5.5.31.47-10
openshift3/ose:v3.5.5.31.47-10
```

### 2.8.11.2. Bug Fixes

#### Authentication

- During upgrades, reconciliation happens only for cluster roles automatically, but this role needs to be adjusted in 3.6 due to enablement of API groups in this release. The Ansible upgrade code has been changed to address this role upgrade. (**BZ#1493213**)

#### Image Registry

- The size of a cached layer did not get counted. Therefore, the layer size for cached layers was zero. Counting the size for cached layers now allows images to have proper layer sizes. (**BZ#1457042**)

#### Logging

- **`openshift-elasticsearch-plugin`** was creating ACL roles based on the provided name, which could include slashes and commas. This caused the dependent library to not properly evaluate roles. With this bug fix, hash the name when creating ACL roles so they no longer contain the invalid characters. (**BZ#1494239**)

- If the logging system is under a heavy load, it may take longer than the five-second timeout for Elasticsearch to respond, or it may respond with an error indicating that Fluentd needs to back off. In the former case, Fluentd will retry to send the records again, which can lead to having duplicate records. In the latter case, if Fluentd is unable to retry, it will drop records, leading to data loss. For the former case, the fix is to set the **`request_timeout`** to 10 minutes, so that Fluentd will wait up to 10 minutes for the reply from Elasticsearch before retrying the request. In the latter case, Fluentd will block attempting to read more input, until the output queues and buffers have enough room to write more data. This bug fix greatly reduces chances of duplicate data (though it is not entirely eliminated). Also, there is no data loss due to back pressure. (**BZ#1497836**, **BZ#1501948**, **BZ#1506854**)

#### Management Console

- The management console was defaulting to the legacy API group **`extensions`** for jobs. As a result, the legacy API group appeared in the UI in places such as **Edit YAML**. With this bug fix, the console snow uses the new **`batch`** API group as the default for job resources. The API group and version on a job resource now appear as **`batch/v1`** wherever it is visible in the console. (BZ#1506233)

**Metrics**

- Extra, unnecessary queries were being performed on each request. The GET **`/hawkular/metrics/metrics`** endpoint could fail with timeouts. With this bug fix, the extra queries are only performed when explicitly requested. By default, do not execute the extra queries that provide optional data. The endpoint is now more stable and not as susceptible to timeouts. (BZ#1458186)

- When either a certificate within the chain at **`serviceaccount/ca.crt`** or any of the certificates within the provided truststore file contained a white space after the **`BEGIN CERTIFICATE`** declaration, the Java keytool rejected the certificate with an error, causing Origin Metrics to fail to start. As a workaround, Origin Metrics will now attempt to remove the spaces before feeding the certificate to the Keytool, but administrators should ensure their certificates do not contain such spaces. (BZ#1471251, BZ#1500464, BZ#1500471)

**Networking**

- A slow image pull made the network diagnostics fail. With this bug fix, the timeout for the image pull was increased. The diagnostics now run in slow environments. (BZ#1481550)

- The OpenShift node proxy previously did not support using a specified IP address. This could prevent correct operation on hosts with multiple network interface cards. The OpenShift node process already accepts a **`--bind-address=<ip address>:<port>`** command-line flag and **`bindAddress:`** configuration file option for the multiple network interface card case. The proxy functionality has been fixed to respect these options. When **`--bind-address`** or **`bindAddress`** are used, the OpenShift node proxy should work correctly when the OpenShift node host has multiple network interface cards. (BZ#1489023, BZ#1489024)

- Iptables called too often and unnecessarily. Therefore, time-outs would wait for iptables operations to finish. This bug fix changes the code so that it skips reloads when the iptables rules are unchanged. There are now fewer calls to iptables and, therefore, less time-outs. (BZ#1501517)

**Pod**

- There was a symbolic link error for the log file of every pod started when the docker log driver was journald. Log symlink creation that fails when using journald logging driver was skipped. This bug fix resolves the issue. (BZ#1434942)

- Currently, pod anti-affinity is respected across projects. Pod A from Project 1 will not land on node where Pod B from Project 2 is running, if pod anti-affinity is enabled when scheduling Pod A. While scheduling Pod A, check for pod anti-affinity only within the project of Pod A. Pod anti-affinity will not be respected across projects. (BZ#1492194)

**Storage**

- The volumePath that included the datastore name was parsed incorrectly. The same applies to volumePath that included datacluster and datastore names. It is not possible to attach persistent volumes that have the above described volumePath values. volumePath is now parsed correctly. Persistent volumes that have the above described volumePath values are attached correctly. (BZ#1497042)

**Security**

- An attacker with knowledge of the given name used to authenticate and access Elasticsearch can later access it without the token, bypassing authentication. This attack also requires that the Elasticsearch be configured with an external route, and the data accessed is limited to the indices. (**BZ#1501986**)

#### 2.8.11.3. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.12. RHBA-2017:3438 - OpenShift Container Platform 3.5.5.31.48 Bug Fix and Enhancement Update

Issued: 2017-12-14

OpenShift Container Platform release 3.5.5.31.48 is now available. The list of packages included in the update are documented in the RHBA-2017:3438 advisory. The container images included in the update are provided by the RHBA-2017:3439 advisory.

Space precluded documenting all of the bug fixes, enhancements, and images for this release in the advisories. See the following sections for notes on upgrading and details on the bug fixes and images included in this release.

#### 2.8.12.1. Images

This release updates the Red Hat Container Registry (**`registry.access.redhat.com`**) with the following images:

```
openshift3/logging-curator:v3.5.5.31.48-2
openshift3/metrics-deployer:v3.5.5.31.48-3
openshift3/node:v3.5.5.31.48-2
openshift3/openvswitch:v3.5.5.31.48-2
openshift3/ose-base:v3.5.5.31.48-2
openshift3/ose-deployer:v3.5.5.31.48-2
openshift3/ose-docker-builder:v3.5.5.31.48-2
openshift3/ose-docker-registry:v3.5.5.31.48-2
openshift3/ose-egress-router:v3.5.5.31.48-2
openshift3/ose-f5-router:v3.5.5.31.48-2
openshift3/ose-haproxy-router:v3.5.5.31.48-3
openshift3/ose-keepalived-ipfailover:v3.5.5.31.48-2
openshift3/ose-pod:v3.5.5.31.48-2
openshift3/ose-sti-builder:v3.5.5.31.48-2
openshift3/ose:v3.5.5.31.48-2
```

#### 2.8.12.2. Bug Fixes

- Custom **openshift-ansible** modules were not being loaded due to a missing dependency configuration. The issue was first identified when using Ansible 2.4.1, which included a change in how tasks and roles were dynamically included at runtime. (**BZ#1516469**)

- Previously, the registry CA, certificate, and key were not updated when running the certificate redeploy playbooks. This bug fix updates the playbooks to ensure that these are all updated whenever new certificates are deployed, ensuring that pushes to the registry work as expected.

(BZ#1383965)

- When using **`openshift_metrics_heapster_standalone=true`** in an inventory file, the CA certificate was not generated, causing playbooks to fail. This bug fix allows the CA certificate to be generated when **`openshift_metrics_heapster_standalone=true`** is set as well. (BZ#1443741)

- Previously, replacement of router certificates through use of the certificate redeployment playbook (***redeploy-certificates.yml***) or the router certificate redeployment playbook (***redeploy-router-certificates.yml***) would fail when a custom router certificate was provided. With this bug fix, custom router certificates set by **`openshift_hosted_router_certificate`** in inventory files now be redeployed. (BZ#1446737)

- In some mixed-node environments, it was possible that host facts were not collected for containerized hosts, causing a conditional to fail. This bug fix adds a conditional to allow the check to complete correctly. (BZ#1462517)

- Invalid PEM data could be left in the route configuration file during extended validation, causing the router to crash. This bug fix sanitizes PEM data from route configuration, and as a result extended validations now properly catch malformed certificates. (BZ#1511732)

- The configuration management for **`BuildDefaults`** attempted to remove environment variables that were previously defined, but have since been removed from the configuration. In situations where no environment variables have been configured, this was failing because the **env** key did not exist. This bug fix updates the process to skip the cleanup when the **env** key does not exist. (BZ#1515459)

#### 2.8.12.3. Enhancements

- When installing logging, the master configuration is now updated to add the **`loggingPublicURL`** parameter. Without **`loggingPublicURL`** being set, a user would not see a link to view historical logs from the web console. With this enhancement, the **`loggingPublicURL`** and its value is now added to the master configuration when installing logging and restarted so that when logging into the web console users can see the link to view historical logs. (BZ#1414706)

#### 2.8.12.4. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.13. RHBA-2018:0076 - OpenShift Container Platform 3.5.5.31.48-10 Images Update

Issued: 2018-01-10

OpenShift Container Platform release 3.5.5.31.48-10 is now available. The list of container images included in the update are documented in the RHBA-2018:0076 advisory.

The container images in this release have been updated using the latest base images.

#### 2.8.13.1. Images

This release updates the Red Hat Container Registry (**registry.access.redhat.com**) with the following images:

```
openshift3/logging-kibana:3.5.0-54
openshift3/node:v3.5.5.31.48-10
openshift3/openvswitch:v3.5.5.31.48-11
```

#### 2.8.13.2. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.14. RHBA-2018:1106 - OpenShift Container Platform 3.5.5.31.66 Bug Fix Update

Issued: 2018-04-12

OpenShift Container Platform release 3.5.5.31.66 is now available. The list of packages included in the update are documented in the RHBA-2018:1106 advisory. The list of container images included in the update are documented in the RHBA-2018:1107 advisory.

Space precluded documenting all of the bug fixes for this release in the advisories. See the following sections for notes on upgrading and details on the bug fixes included in this release.

#### 2.8.14.1. Bug Fixes

**Command Line Interface**

- When the master configuration specifies a default **nodeSelector** for the cluster, test projects created by **oadm diagnostics NetworkCheck** got this **nodeSelector**, and therefore the test pods were also confined to this **nodeSelector**. This caused **NetworkCheck** test pods to only be scheduled on a subset of nodes, preventing the diagnostic covering the entire cluster; in some clusters, this might even result in too few pods running for the diagnostic to succeed even if the cluster health is fine. With this bug fix, **NetworkCheck** now creates the test projects with an empty **nodeSelector** so they can land on any schedulable node. As a result, the diagnostic should be more robust and meaningful. (**BZ#1534775**)

**Installer**

- OpenShift Container Platform requires that host names conform to standards which preclude the use of uppercase letters. With this bug fix, the installer now ensures that the host names for node objects are created with lowercase letters. (**BZ#1543748**)

**Logging**

- Fluentd was adding the level field with a value of **3** or **6**, overwriting any existing level field. The level field set by the application was being removed, and the **3** or **6** value was not useful. With this bug fix, if there is already a **level** field in the record, then see if it is a "close" match to one of the canonical **level** field values. For example, if **level** is **CRITICAL**, convert to **crit**; if level is **WARN**, convert to **warning**. Otherwise, if it cannot be used directly or normalized, convert it to its string representation (ruby **to_s** method) and store the string value in the **level** field. As a result, if the record already has a level field, the value is normalized or preserved, otherwise, a value like **info** or **err** is used. (**BZ#1514110**)

**Web Console**

- A common problem with Angular (the application framework) and some template logic caused certain kinds of template logic to cause instability while rendering, resulting in **`$digest loop`** errors. The template render loop reached its maximum number of cycles and stopped processing, resulting in broken templates on the pages using the **pods-table** component. This bug fix backports a fix from OpenShift Container Platform 3.6 to the **pods-table** component that eliminates the issue causing the **`$digest`** cycle to hit its maximum. As a result, the pages using the **pods-table** component should now function as normal (tested with 25+ pod replicas). (**BZ#1538431**)

- Use of JavaScript Array methods not supported by Internet Explorer 11 (**`Array.find`**, **`Array.findIndex`**) prevented some A-MQ pages from showing content when the user clicks on a tree node. This bug fix replace the unsupported Array methods with the Lodash library, and as a result the A-MQ pages show their content as expected. (**BZ#1543402**)

#### 2.8.14.2. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

### 2.8.15. RHSA-2018:3624 - Critical: OpenShift Container Platform 3.5 Security Update

Issued: 2018-12-05

A security update to OpenShift Container Platform 3.5 is now available. The list of packages included in the update are documented in the RHSA-2018:3624 advisory.

#### 2.8.15.1. Upgrading

To upgrade an existing OpenShift Container Platform 3.4 or 3.5 cluster to this latest release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

# CHAPTER 3. XPAAS RELEASE NOTES

The release notes for xPaaS docs have migrated to their own book on the Red Hat customer portal.

# CHAPTER 4. COMPARING WITH OPENSHIFT ENTERPRISE 2

## 4.1. OVERVIEW

OpenShift Container Platform 3 is based on the OpenShift version 3 (v3) architecture, which is very different product than OpenShift version 2 (v2). Many of the same terms from OpenShift v2 are used in v3, and the same functions are performed, but the terminology can be different, and behind the scenes things may be happening very differently. Still, OpenShift remains an application platform.

This topic discusses these differences in detail, in an effort to help OpenShift users in the transition from OpenShift v2 to OpenShift v3.

## 4.2. ARCHITECTURE CHANGES

**Gears vs Containers**

Gears were a core component of OpenShift v2. Technologies such as kernel namespaces, cGroups, and SELinux helped deliver a highly-scalable, secure, containerized application platform to OpenShift users. Gears themselves were a form of container technology.

OpenShift v3 takes the gears idea to the next level. It uses Docker as the next evolution of the v2 container technology. This container architecture is at the core of OpenShift v3.

**Kubernetes**

As applications in OpenShift v2 typically used multiple gears, applications on OpenShift v3 will expectedly use multiple containers. In OpenShift v2, gear orchestration, scheduling, and placement was handled by the OpenShift broker host. OpenShift v3 integrates Kubernetes into the master host to drive container orchestration.

## 4.3. APPLICATIONS

Applications are still the focal point of OpenShift. In OpenShift v2, an application was a single unit, consisting of one web framework of no more than one cartridge type. For example, an application could have one PHP and one MySQL, but it could not have one Ruby, one PHP, and two MySQLs. It also could not be a database cartridge, such as MySQL, by itself.

This limited scoping for applications meant that OpenShift performed seamless linking for all components within an application using environment variables. For example, every web framework knew how to connect to MySQL using the **`OPENSHIFT_MYSQL_DB_HOST`** and **`OPENSHIFT_MYSQL_DB_PORT`** variables. However, this linking was limited to within an application, and only worked within cartridges designed to work together. There was nothing to help link across application components, such as sharing a MySQL instance across two applications.

While most other PaaSes limit themselves to web frameworks and rely on external services for other types of components, OpenShift v3 makes even more application topologies possible and manageable.

OpenShift v3 uses the term "application" as a concept that links services together. You can have as many components as you desire, contained and flexibly linked within a project, and, optionally, labeled to provide grouping or structure. This updated model allows for a standalone MySQL instance, or one shared between JBoss components.

Flexible linking means you can link any two arbitrary components together. As long as one component can export environment variables and the second component can consume values from those

environment variables, and with potential variable name transformation, you can link together any two components without having to change the images they are based on. So, the best containerized implementation of your desired database and web framework can be consumed directly rather than you having to fork them both and rework them to be compatible.

This means you can build anything on OpenShift. And that is OpenShift's primary aim: to be a container-based platform that lets you build entire applications in a repeatable lifecycle.

## 4.4. CARTRIDGES VS IMAGES

In OpenShift v3, an image has replaced OpenShift v2's concept of a cartridge.

Cartridges in OpenShift v2 were the focal point for building applications. Each cartridge provided the required libraries, source code, build mechanisms, connection logic, and routing logic along with a preconfigured environment to run the components of your applications.

However, cartridges came with disadvantages. With cartridges, there was no clear distinction between the developer content and the cartridge content, and you did not have ownership of the home directory on each gear of your application. Also, cartridges were not the best distribution mechanism for large binaries. While you could use external dependencies from within cartridges, doing so would lose the benefits of encapsulation.

From a packaging perspective, an image performs more tasks than a cartridge, and provides better encapsulation and flexibility. However, cartridges also included logic for building, deploying, and routing, which do not exist in images. In OpenShift v3, these additional needs are met by Source-to-Image (S2I) and configuring the template.

**Dependencies**

In OpenShift v2, cartridge dependencies were defined with **`Configure-Order`** or **`Requires`** in a cartridge manifest. OpenShift v3 uses a declarative model where pods bring themselves in line with a predefined state. Explicit dependencies that are applied are done at runtime rather than just install time ordering.

For example, you might require another service to be available before you start. Such a dependency check is always applicable and not just when you create the two components. Thus, pushing dependency checks into runtime enables the system to stay healthy over time.

**Collection**

Whereas cartridges in OpenShift v2 were colocated within gears, images in OpenShift v3 are mapped 1:1 with containers, which use pods as their colocation mechanism.

**Source Code**

In OpenShift v2, applications were required to have at least one web framework with one Git repository. In OpenShift v3, you can choose which images are built from source and that source can be located outside of OpenShift itself. Because the source is disconnected from the images, the choice of image and source are distinct operations with source being optional.

**Build**

In OpenShift v2, builds occurred in application gears. This meant downtime for non-scaled applications due to resource constraints. In v3, builds happen in separate containers. Also, OpenShift v2 build results used rsync to synchronize gears. In v3, build results are first committed as an immutable image and

published to an internal registry. That image is then available to launch on any of the nodes in the cluster, or available to rollback to at a future date.

**Routing**

In OpenShift v2, you had to choose up front as to whether your application was scalable, and whether the routing layer for your application was enabled for high availability (HA). In OpenShift v3, routes are first-class objects that are HA-capable simply by scaling up your application component to two or more replicas. There is never a need to recreate your application or change its DNS entry.

The routes themselves are disconnected from images. Previously, cartridges defined a default set of routes and you could add additional aliases to your applications. With OpenShift v3, you can use templates to set up any number of routes for an image. These routes let you modify the scheme, host, and paths exposed as desired, with no distinction between system routes and user aliases.

## 4.5. BROKER VS MASTER

A master in OpenShift v3 is similar to a broker host in OpenShift v2. However, the MongoDB and ActiveMQ layers used by the broker in OpenShift v2 are no longer necessary, because **etcd** is typically installed with each master host.

## 4.6. DOMAIN VS PROJECT

A project is essentially a v2 domain.

# CHAPTER 5. REVISION HISTORY: RELEASE NOTES

## 5.1. WED JAN 10 2018

| Affected Topic | Description of Change |
|---|---|
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2018:0076 - OpenShift Container Platform 3.5.5.31.48-10 Images Update. |

## 5.2. THU DEC 14 2017

| Affected Topic | Description of Change |
|---|---|
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:3438 - OpenShift Container Platform 3.5.5.31.48 Bug Fix and Enhancement Update. |

## 5.3. WED DEC 06 2017

| Affected Topic | Description of Change |
|---|---|
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHSA-2017:3389 - Moderate: OpenShift Container Platform 3.5.5.31.47 Security, Bug Fix, and Enhancement Update. |

## 5.4. WED OCT 25 2017

| Affected Topic | Description of Change |
|---|---|
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:3049 - OpenShift Container Platform 3.5.5.31.36 Bug Fix Update. |

## 5.5. THU SEP 07 2017

| Affected Topic | Description of Change |
|---|---|
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:2670 - OpenShift Container Platform 3.5.5.31.24 Bug Fix Update. |

## 5.6. THU AUG 31 2017

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:1828 - OpenShift Container Platform 3.5.5.31 Bug Fix Update. |

## 5.7. TUE JUL 11 2017

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:1640 - OpenShift Container Platform 3.5.5.31 Bug Fix Update. |

## 5.8. FRI JUL 07 2017

| Affected Topic | Description of Change |
| --- | --- |
| Overview | Clarified that "client" referred to the **oc** client. |
| OpenShift Container Platform 3.5 Release Notes | Added **BZ#1466783** to Known Issues. |

## 5.9. THU JUN 29 2017

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:1666 - atomic-openshift-utils Bug Fix and Enhancement Update. |

## 5.10. THU JUN 22 2017

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:1492 - OpenShift Container Platform 3.5.5.26 Bug Fix Update. |
| | Added issued dates for all Asynchronous Errata Updates. (**BZ#1463721**) |

## 5.11. WED JUN 14 2017

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:1425 - OpenShift Container Platform 3.5.5.24 Bug Fix Update. |

## 5.12. THU MAY 18 2017

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:1235 - OpenShift Container Platform 3.5.5.15 Bug Fix Update. |

## 5.13. TUE APR 25 2017

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for RHBA-2017:1129 - OpenShift Container Platform 3.5.5.8 Bug Fix and Enhancement Update. |

## 5.14. WED APR 12 2017

OpenShift Container Platform 3.5 Initial Release

| Affected Topic | Description of Change |
| --- | --- |
| OpenShift Container Platform 3.5 Release Notes | Added release notes for initial release. |
| | Added release notes for RHBA-2017:0903 - atomic-openshift-utils Bug Fix and Enhancement Update. |