OpenShift Enterprise 3.2

Release Notes

Last Updated: 2019-01-21

Legal Notice

Abstract

Table of Contents

CHAPTER 1. OVERVIEW

The following release notes for OpenShift Enterprise 3.2 and xPaaS summarize all new features, major corrections from the previous version, and any known bugs upon general availability.

CHAPTER 2. OPENSHIFT ENTERPRISE 3.2 RELEASE NOTES

2.1. OVERVIEW

OpenShift Enterprise by Red Hat is a Platform as a Service (PaaS) that 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 Enterprise supports a wide selection of programming languages and frameworks, such as Java, Ruby, and PHP.

Built on Red Hat Enterprise Linux and Kubernetes, OpenShift Enterprise provides a secure and scalable multi-tenant operating system for today's enterprise-class applications, while providing integrated application runtimes and libraries. OpenShift Enterprise brings the OpenShift PaaS platform to customer data centers, enabling organizations to implement a private PaaS that meets security, privacy, compliance, and governance requirements.

2.2. NEW FEATURES AND ENHANCEMENTS

OpenShift Enterprise version 3.2 is now available. Ensure that you follow the instructions on upgrading your OpenShift cluster properly, including steps specific to this release.

IMPORTANT

For any release, always review the Installation and Configuration guide for instructions on upgrading your OpenShift cluster properly, including any additional steps that may be required for a specific release.

2.2.1. For Administrators

2.2.1.1. Updated Infrastructure Components

- Kubernetes has been updated to v1.2.0-36.
- etcd has been updated to v2.2.5.
- OpenShift Enterprise 3.2 requires Docker 1.9.1.

2.2.1.2. Configuration and Administration

- Admission controllers are now included, which intercept requests to the master API prior to persistence of a resource, but after the request is authenticated and authorized. The following admission control plug-ins can be configured:
  - The number of projects an individual user can create can be limited via the **`ProjectRequestLimit`** admission controller. See Limiting Number of Self-Provisioned Projects Per User for details.
  - A build defaults admission controller can be used to set default environment variables on all builds created, including global proxy settings. See Configuring Global Build Defaults and Overrides for details.
  - The **`PodNodeConstraints`** admission control plug-in has been added, which constrains the use of the **`NodeName`** field in a pod definition to roles which have the **`pods/binding`** permission. This allows administrators, via **`NodeSelectorLabelBlacklist`**, to specify

node labels by setting them in the **`NodeSelector`** field of the pod definition. See Controlling Pod Placement for details.

- Multiple web login providers can now be configured at the same time.
- The **`oc adm diagnostics`** command can now launch a diagnostic pod that reports on more potential issues with pod networking, DNS configuration, and registry authentication.
- Support for security context constraints (SCCs) has been added to the **`oc describe`** command.
- The **`NO_PROXY`** environment variable will now accept a CIDR in a number of places in the code for controlling which IP ranges bypass the default HTTP proxy settings. See Configuring Hosts for Proxies for details.
- Masters are now taken down one at a time during upgrades through rolling restarts. The **`openshift_rolling_restart_mode`** parameter can now be used in Ansible inventories to control this behavior: **`services`** for service restarts or **`system`** for full system restarts. See Configuring Cluster Variables for details.

2.2.1.3. Security

- The new **`Volumes`** field in SCCs allows an administrator full control over which volume plug-ins may be specified.
  - In order to maintain backwards compatibility, the **`AllowHostDirVolumePlugin`** field takes precedence over the **`Volumes`** field for the host mounts. You may use **`*`** to allow all volumes.
  - By default, regular users are now forbidden from directly mounting any of the remote volume type; they must use a persistent volume claim (PVC).
- The new **`ReadOnlyRootFilesystem`** field in SCCs allows an administrator to force containers to run with a read-only root file system.
  - If set to true, containers are required to run with a read-only root file system by their **`SecurityContext`**. Containers that do not set this value to true will be defaulted. Containers that explicitly set this value to false will be rejected.
  - If set to false, containers may use a read-only root file system, but they are not forced to run with one.
- By default, the **restricted** and **anyuid** SCCs drop Linux capabilities that could be used to escalate container privileges. Administrators can change the list of default or enforced capabilities.
- A constant-time string comparison is now used on webhooks.
- Only users authenticated via OAuth can request projects.
- A GitLab server can now be used as an identity provider. See Configuring Authentication for details.
- The **`SETUID`** and **`SETGID`** capabilities have been added back to the **anyuid** SCC, which ensures that programs that start as root and then drop to a lower permission level will work by default.
- Quota support has been added for **`emptydir`**. When the quota is enabled on an XFS system,

nodes will limit the amount of space any given project can use on a node to a fixed upper bound. The quota is tied to the **FSGroup** of the project. Administrators can control this value by editing the project directly or allowing users to set **FSGroup** via SCCs.

- The **DaemonSet** object is now limited to cluster administrators because pods running under a **DaemonSet** are considered to have higher priority than regular pods, and for regular users on the cluster this could be a security issue.
- Administrators can prevent clients from accessing the API by their **User-Agent** header the new **userAgentMatching** configuration setting.

2.2.1.4. Integrated Docker Registry

- A readiness probe and health check have been added to the integrated registry to ensure new instances do not serve traffic until they are fully initialized.

2.2.1.5. Routes

- You can limit the frequency of router reloads using the **--interval=DURATION** flag or **RELOAD_INTERVAL** environment variable to the router. This can minimize the memory and CPU used by the router while reloading, at the cost of delaying when the route is exposed via the router.
- Routers now report back status to the master about whether routes are accepted, rejected, or conflict with other users. The CLI will now display that error information, allowing users to know that the route is not being served.
- Using *router sharding*, you can specify a selection criteria for either namespaces (projects) or labels on routes. This enables you to select the routes a router would expose, and you can use this functionality to distribute routes across a set of routers, or shards.

2.2.1.6. Storage

- The **NoDiskConflicts** scheduling predicate can be added to the scheduler configuration to ensure that pods using the same Ceph RBD device are not placed on the same node. See Scheduler for details.

2.2.1.7. Administrator CLI

- The administrative commands are now exposed via **oc adm** so you have access to them in a client context. The **oadm** commands will still work, but will be a symlink to the **openshift** binary.
- The help output of the **oadm policy** command has been improved.
- Service accounts are now supported for the router and registry:
  - The router can now be created without specifying **--credentials** and it will use the router service account in the current project.
  - The registry will also use a service account if **--credentials** is not provided. Otherwise, it will set the values from the **--credentials** file as environment on the generated deployment configuration.
- Administrators can pass the **--all-namespaces** flag to **oc status** to see status information across all namespaces and projects.

2.2.1.8. Web Console

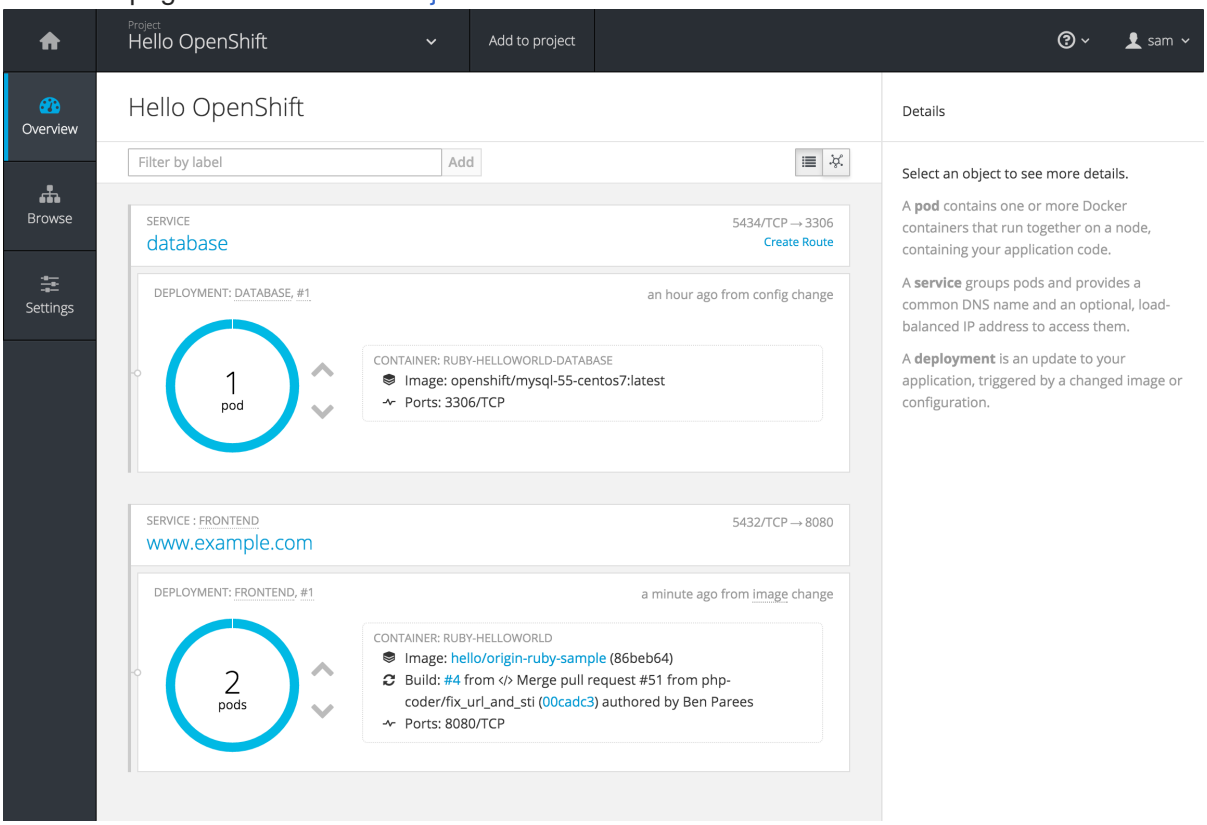
- Users can now be presented with a customized, branded page before continuing on to a login identity provider. This allows users to see your branding up front instead of immediately redirecting to identity providers like GitHub and Google. See Customizing the Login Page for details.

- CLI download URLs and documentation URLs are now customizable through web console extensions. See Adding or Changing Links to Download the CLI for details.

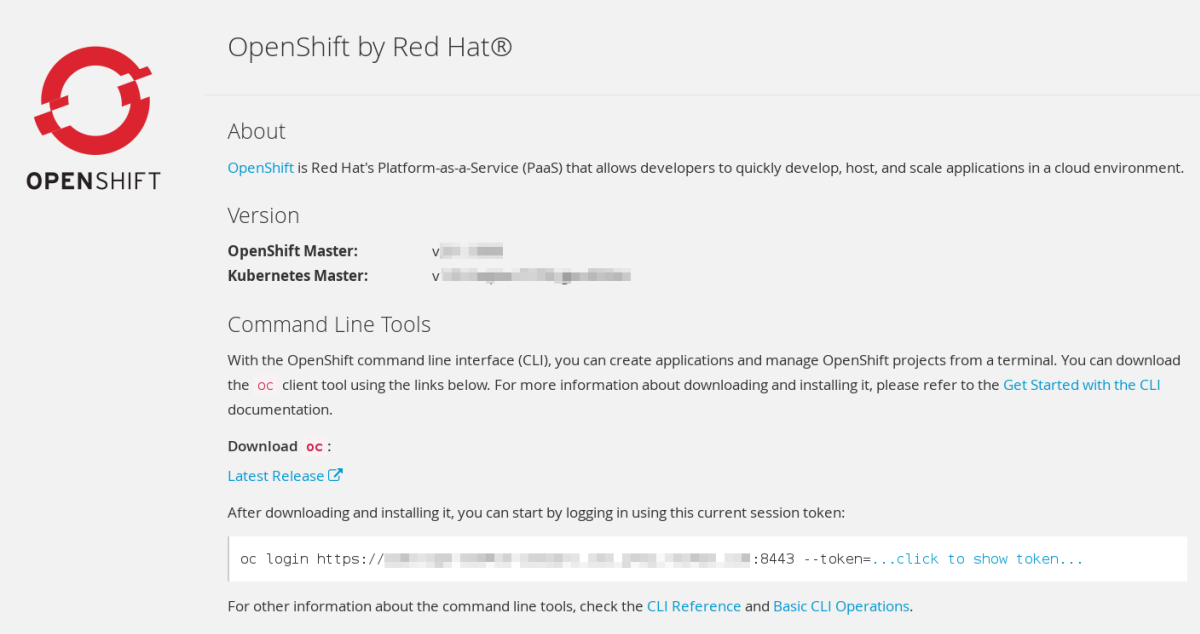
2.2.2. For Developers

2.2.2.1. Web Console

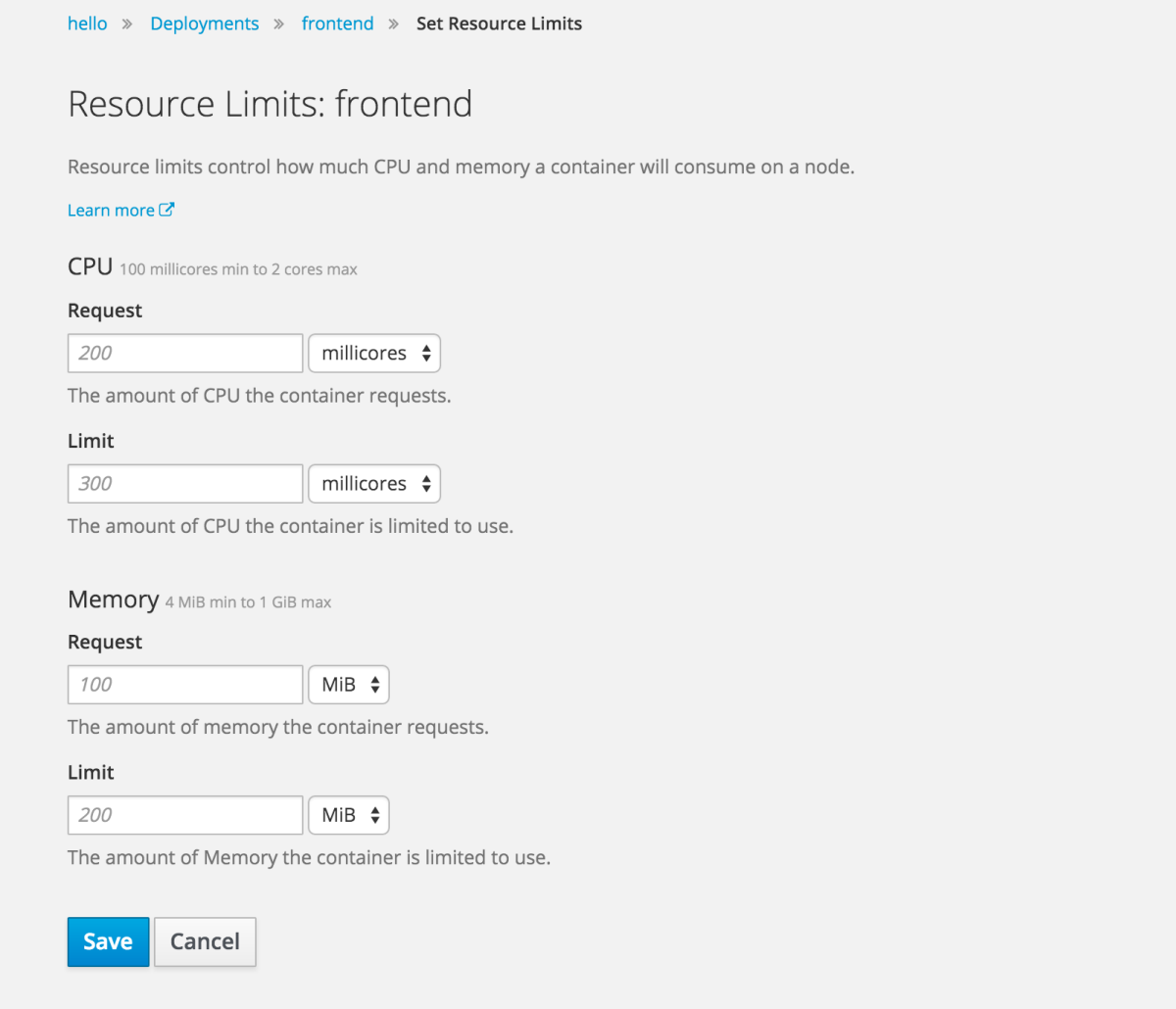
- The web console uses a brand new theme that changes the look and feel of the navigation, tabs, and other page elements. See Project Overviews for details.



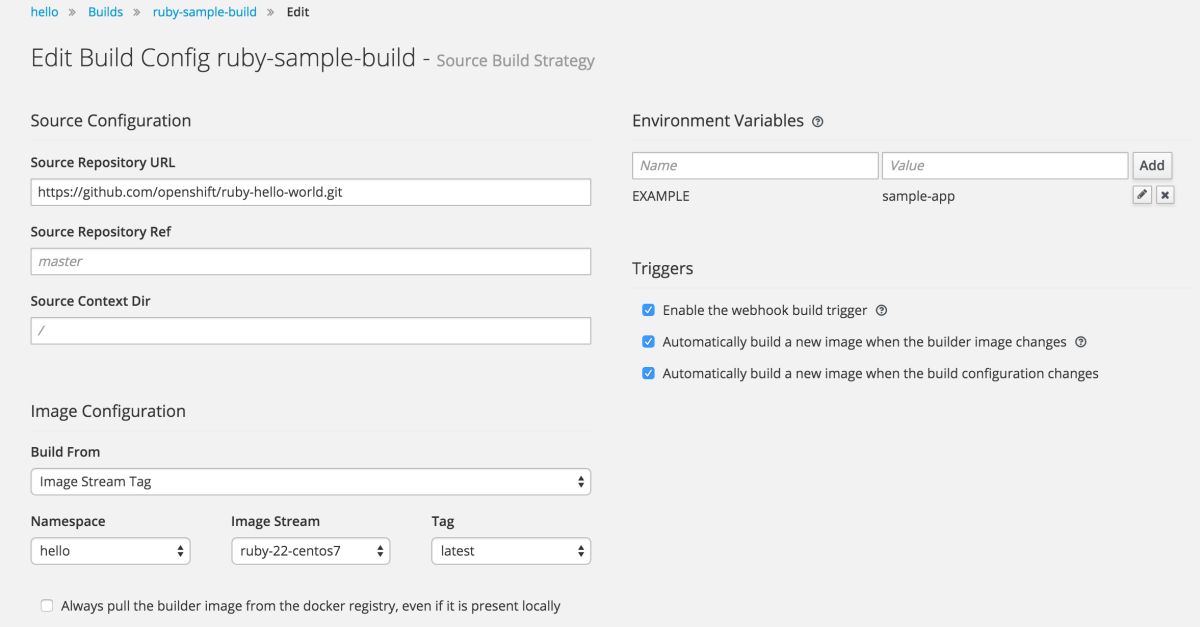
- A new **About** page provides developers with information about the product version, **oc** CLI download locations, and a quick access to their current token to login using **`oc login`**. See CLI Downloads for details.



- You can now add or edit resource constraints for your containers during **Add to Project** or later from the deployment configuration.



- A form-based editor for build configurations has been added for modifying commonly edited fields directly from the web console.



- All **Browse** resource pages (e.g, viewing a particular pod) now have a tab for **Events** related to that pod.
- Limits, quotas, and quota scopes are now displayed.
- More error and warning information is now displayed about routes, their configuration, and their use in the system.
- Support has been added for filtering and sorting on all **Events** pages.
- You can now edit a project's display name and description from the **Settings** page.
- Existing persistent volume claims (PVCs) can now be listed and attached to deployments and deployment configurations.
- More detailed pod status is now provided on all pages.
- Better status and alert messages are now provided.
- Improved ***Dockerfile*** build keyword highlighting has been added when editing builds.
- More accurate information is now displayed about routes based on which addresses the router exposed them under.
- The layout and display of logs have been improved.

2.2.2.2. Developer CLI

- The following commands have been added to **`oc create`**, allowing more objects to be created directly using the CLI (instead of passing it a file or JSON/YAML):

Command	Description
`namespace`	Create a namespace with the specified name.
`secret`	Create a secret using a specific subcommand: **`docker-registry`** or **`generic`**.

Command	Description
configmap	Create a **ConfigMap** from a local file, directory, or literal value.
serviceaccount	Create a service account with the specified name.
route	Expose containers externally via secured routes. Use the **edge**, **passthrough**, or **reencrypt** subcommands and specify the secret values to be used for the route.

- Display more information about the application being created by the **oc new-app** command, including any display name or description set on the image as a label, or whether the image may require running as root.
- If you have set up the **latest** tag in an image stream to point to another tag in the same image stream, the **oc new-app** command will follow that reference and create the application using the referenced tag, not **latest**. This allows administrators to ensure applications are created on stable tags (like **php:5.6**). The default image streams created in the **openshift** project follow this pattern.
- You can view the logs of the oldest pod in a deployment or build configuration with:

```
$ oc logs dc/<name>
```

- The **oc env** and **oc volume** commands have been moved to **oc set env** and **oc set volume**, and future commands that modify aspects of existing resources will be located under this command.
- When a pod is crash-looping, meaning it is starting and exiting repeatedly, an error is now displayed in **oc status** output and provides more information about possible causes.
- The new **oc debug** command makes it easy to obtain shell access in a misbehaving pod. It clones the exact environment of the running deployment configuration, replication controller, or pod, but replaces the run command with a shell.
- The new **oc set trigger** command can be used to update deployment and build configuration triggers.
- More information is displayed about liveness and readiness probes in the **oc status** and **oc describe** commands.

2.2.2.3. Builds and Image Sources

- Builds can now be supplied with input files from unrelated images. Previously, all input to a build had to come from the builder image itself, or a Git repository. It is now possible to specify additional images and paths within those images to use as an input to a build for things like external dependencies.
  Use the **--source-image=<image>** and **--source-image-path=<source>:<destination>** flags with the **oc new-build** command to specify images.

  The example shown below injects the ***/usr/lib/jenkins/jenkins.war*** file out of the image currently tagged with **jenkins:latest** into the ***installed-apps*** directory of the build input:

```
apiVersion: v1
kind: BuildConfig
metadata:
  name: imagedockerbuild
spec:
  source:
    images:
    - from:
        kind: ImageStreamTag
        name: jenkins:latest
      paths:
      - destinationDir: installed-apps/
        sourcePath: /usr/lib/jenkins/jenkins.war
```

Ensure that you set an image change trigger for **jenkins:latest** if you want to rebuild every time that image is updated.

- Builds can now be supplied with secrets for use during the build process. Previously, secrets could be used for Git cloning but now secrets can also be made available to the build process itself so that build operations such as Maven packaging can use a secret for credentials. See Using Secrets During a Build for details.

- Builds now properly use Git submodules when checking out the source repository. When a build configuration is deleted (via **`oc delete`**), all associated builds are now deleted as well. To prevent this behavior, specify **`--cascade=false`**.

- Custom build configurations can now specify the API version to use. This API version will determine the schema version used for the serialized build configuration supplied to the custom build pod in the **`BUILD`** environment variable.

- Resource limits are now enforced on the container launched by S2I builds, and also on the operations performed within containers as part of a **`docker build`** of a ***Dockerfile***. Previously, the resource limit only applied to the build pod itself and not the containers spawned by the build process.

- You can now provide a command to be triggered after a build succeeds but before the push. You can set **`shell`** (to run a shell script), **`command`**, or **`args`** to run a command in the working directory of the built image. All S2I builders set the user's source repository as the working directory, so commands like **`bundle exec rake test`** should work. See Build Hooks for details.

2.2.2.4. Image Imports

- You can now import images from Docker v2 registries that are authenticated via Basic or Token credentials. To import, create a secret in your project based on a ***.docker/config.json*** or ***.dockercfg*** file:

```
$ oc secrets new hub .dockerconfigjson=$HOME/.docker/config.json
Created secret/hub

$ oc import-image auth-protected/image-from-dockerhub
The import completed successfully.

Name:       image-from-dockerhub
Created:    Less than a second ago
```

```
Tag     Spec                                    Created
latest  default/image-from-dockerhub:latest     Less than a second ago
...
```

When importing, all secrets in your project of those types will be checked. To exclude a secret from being a candidate for importing, use the **`openshift.io/image.excludeSecret`** annotation set to **true**:

```
$ oc annotate secret/hub openshift.io/image.excludeSecret=true
```

- Image stream tags can be set to be automatically imported from remote repositories when they change (public or private). OpenShift Enterprise will periodically query the remote registry and check for updates depending on the configuration the administrator sets. By default, images will be checked every 15 minutes.
  To set an image to be imported automatically, use the **`--scheduled`** flag with the **`oc tag`** command:

  ```
  $ oc tag --source=docker redis:latest myredis:latest --scheduled
  Tag myredis:latest set to import redis:latest periodically.
  ```

  You can see which images are being scheduled using:

  ```
  $ oc describe is myredis
  ```

  Administrators can control whether scheduling is enabled, the polling interval, and the rate at which images can be imported via the **`imagePolicyConfig`** section in the ***/etc/origin/master/master-config.yaml*** file.

- The integrated Docker registry now supports *image pullthrough*, allowing you to tag a remote image into OpenShift Enterprise and directly pull it from the integrated registry as if it were already pushed to the OpenShift Enterprise registry. If the remote registry is configured to use content-offload (sending back a temporary redirect URL to the actual binary contents), that value will be passed through the OpenShift Enterprise registry and down to the Docker daemon, avoiding the need to proxy the binary contents.
  To try pullthrough, tag an image from the DockerHub:

  ```
  $ oc tag --source=docker redis:latest redis:local
  $ oc get is redis
  NAME      DOCKER REPO                     TAGS     UPDATED
  mysql     172.30.1.5:5000/default/redis   local    Less than a
  second ago
  ```

  Log into your local Docker registry, then pull the image from the integrated registry:

  ```
  $ docker pull 127.30.1.5:5000/default/redis:local
  Using default tag: local
  Trying to pull repository 127.30.1.5:5000/default/redis ... latest:
  Pulling from 127.30.1.5:5000/default/redis
  47d44cb6f252: Pull complete
  838c1c5c4f83: Pull complete
  5764f0a31317: Pull complete
  60e65a8e4030: Pull complete
  449f8db3c25a: Pull complete
  ```

```
a6b6487c42f6: Pull complete
Digest:
sha256:c541c66a86b0715bfbb89c5515929268196b642551beccf8fbd452bb00170
cde
Status: Downloaded newer image for
127.30.1.5:5000/default/redis:local
```

You can use pullthrough with private images; the integrated registry will use the same secret you imported the image with to fetch content from the remote registry.

- The **`oc describe`** command now reports overall image size for imported images as well as the individual layers and size of each layer.

- When importing an entire remote repository, only the first five tags are imported by default. OpenShift Enterprise preferentially imports the **latest** tag and the highest semantically versioned tag (i.e., tags in the form **v5**, **5.0**, or **5.0.1**). You can import the remaining tags directly. Lists of tags will be sorted with the latest tag on top, followed by the highest major semantic tags, in descending order.

2.2.2.5. Test Deployments

It is now possible to create a "test" deployment that will scale itself down to zero when a deployment is complete. This deployment can be used to verify that an image will be correctly rolled out without requiring the pods to be running all the time. To create a test deployment, use the **`--as-test`** flag on **`oc new-app`** or set the **`spec.test`** field of a deployment configuration to **`true`** via **`oc edit`**.

The deployment triggers like any other deployment configuration, scaling up to the current **`spec.replicas`** value when triggered. After the deployment has completed with a success or failure, it is then scaled down to zero. You can use deployment hooks to test or verify the deployment; because hooks run as part of the deployment process, a test suite running in your hook can ensure your application is correct and pass or fail the deployment.

You can add a local database or other test container to the deployment pod template, and have your application code verify itself before passing to the next step.

Scaling a test deployment will only affect the next deployment.

2.2.2.6. Recreate Strategy

- The Recreate deployment strategy now supports **`mid`** hooks, which run while all old pods have been scaled down and before any new pods are scaled up; use it to run migrations or configuration changes that can only happen while the application is completely shut down.

- The Recreate deployment strategy now has the same behavior as the Rolling strategy, requiring the pod to be "Ready" before continuing with the deployment. A new field **`timeoutSeconds`** was added to the strategy that is the maximum allowed interval between pods becoming ready; it defaults to **`120s`**.

2.2.2.7. Other Enhancements

- The new Kubernetes 1.2 ConfigMap resource is now usable.

- Pods being pulled or terminating are now distinguished in the pod status output, and the size of images is now shown with other pod information.

- The Jenkins image can now be used as an S2I-compatible build image. See Using Jenkins as a Source-to-Image Builder for details.

2.3. NOTABLE TECHNICAL CHANGES

OpenShift Enterprise 3.2 introduces the following notable technical changes:

2.3.1. For Administrators

2.3.1.1. Services with External IPs Rejected by Default

By default, services with external IPs are now rejected because, in some cases, they can be used to allow services to pretend to act as nodes. The new **`networkConfig.externalIPNetworkCIDR`** parameter has been added to the ***master-config.yaml*** file to control the allowable values for external IPs. By default, it is empty, which rejects all values. Cluster administrators can set it to **`0.0.0.0/0`** to emulate the behavior from OpenShift Enterprise 3.1.

2.3.1.2. Build Strategy Permissions Separated into Distinct Roles

Build strategy permissions have been separated into distinct roles. Administrators who have denied access to Docker, Source, or Custom builds must now assign users or groups to those roles by default. See Securing Builds by Strategy for details.

2.3.1.3. FSGroup Enabled by Default for restricted and hostaccess SCCs

`FSGroup` is now enabled by default in the **restricted** and **hostaccess** SCCs. This means that pods matched against those SCCs will now:

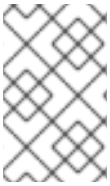
- Have the **`pod.spec.securityContext.fsGroup`** field populated to a namespace-wide allocated value automatically.
- Have their **emptyDir**-derived (**emptyDir**, **gitRepo**, **secret**, **configMap**, and **downwardAPI**) and block device volumes (basically every network volume except **ceph** and **nfs**) owned by the **`FSGroup`**.
- Run with the **`FSGroup`** in each container's list of supplemental groups.

2.3.1.4. Tightened Directory Permissions on Hosts

Permissions on the ***/etc/origin*** directory have been tightened to prevent unprivileged users from reading the contents of this directory tree. Administrators should ensure that, if necessary, they have provided other means to access the generated CA certificate.

2.3.1.5. DNS Changes

- By default, new nodes installed with OpenShift Enterprise 3.2 will have Dnsmasq installed and configured as the default nameserver for both the host and pods.
- By default, new masters installed with OpenShift Enterprise 3.2 will run SkyDNS on port 8053 rather than 53. Network access controls must allow nodes to connect to masters on port 8053. This is necessary so that Dnsmasq may be configured on all nodes.



NOTE

The above DNS changes only apply to new installations of OpenShift Enterprise 3.2. Clusters upgraded from OpenShift Enterprise 3.1 to 3.2 do not currently have these changes applied during the upgrade process.

2.3.1.6. New Default Values for Pod Networking

The default values for pod networking have changed:

master-config.yaml Field	Ansible Variable	Old Value	New Value
`clusterNetworkCIDR`	`osm_cluster_network_cidr`	10.1.0.0/16	10.128.0.0/14 (i.e., 10.128.0.0 - 10.131.255.255)
`hostSubnetLength`	`osm_host_subnet_length`	8 (i.e., /24 subnet)	9 (i.e., /23 subnet)

2.3.1.7. API Changes

- Due to a change in the upstream JSON serialization path used in Kubernetes, some fields that were previously accepted case-insensitively are no longer accepted. Please validate that your API objects have the correct case for all attributes.
- When creating a deployment configuration, omitting the **`spec.selector`** field will default that value to the pod template labels.
- **`ImageStreamTag`** objects now return the spec tag **`tag`**, the current status conditions, and latest status generation **`generation`**, so clients can get an accurate view of the current tag.
- **`ImageStreamTag`** objects can be updated via **`PUT`** to set their spec tag in a single call.
- Deployment configuration hooks now default the container name if there is only a single container in the deployment configuration.

2.3.1.8. Increased Default MaxPodsPerNode

The default value for **`MaxPodsPerNode`** has been increased to **`110`** to reflect updated capacity.

2.3.1.9. High Availability Masters Support

Administrators are recommended to take advantage of the native HA method for multiple masters built in to OpenShift instead of previous solutions such as Pacemaker. Starting with OpenShift Enterprise 3.2, the Pacemaker HA method is no longer supported by the installer and upgrade playbooks, and administrators should upgrade to the native HA method before upgrading the cluster to 3.2. See the Upgrading from Pacemaker to Native HA in the OpenShift Enterprise 3.1 documentation for instructions.

2.3.2. For Developers

2.3.2.1. Developer CLI

The **oc rsh** command now launches **/bin/sh**, not **/bin/bash**. To have the old behavior, run:

```
$ oc rsh <name> -- /bin/bash
```

2.4. BUG FIXES

The following bugs have been fixed:

- Passthrough routes may not be specified with paths. Because passthrough does not decode the route, there is no way for the router to check the path without decoding the request. The **oc status** command will now warn you if you have any such routes.
- The **oc new-app** command now returns more information about errors encountered while searching for matches to user input.
- When using images from registries that are not the DockerHub, do not insert the **library** prefix.
- The image ID returned from the **ImageStreamImage** API was not the correct value.
- The router health check was not correct on all systems when using host networking. It now defaults to using **localhost**.
- OAuth client secrets are now correctly reset in HA master configurations.
- Improved the web console's performance when displaying many deployments or builds.
- The router unique host check should not reprocess routes that did not change.
- Added the **AlwaysPull** admission controller to prevent users from being able to run images that others have already pulled to the node.
- Fixed **oc edit** when editing multiple items in a list form.
- The recycler for persistent volumes now uses a service account and has proper access to restricted content.
- The block profiler in **pprof** is now supported.
- Additional **cGroup** locations are now handled when constraining builds.
- Scratch images from **oc new-app** are now handled.
- Added support for paged LDAP queries.
- Fixed a performance regression in **cAdvisor** that resulted in long pauses on Kubelet startup.
- The **oc edit** command was not properly displaying all errors when saving an edited resource failed.
- More information is now shown about persistent volume claims and persistent volumes in a number of places in the CLI and web console.
- Some commands that used the API PATCH command could fail intermittently when they were executed on the server and another user edited at the same time.

- Users are now warned when trying to import a non-existent tag with the **`oc import-image`** command.
- Singular pods are now shown in **`oc status`** output.
- Router fixes:
  - More information is now shown from the router reload command in the router logs.
  - Routes that changed at the same time could compete for being exposed if they were in different namespaces. The check for which route gets exposed has been made predictable.
  - The health check is now used when restarting the router to ensure the new process is correctly running before continuing.
- Better errors are displayed in the web console when JavaScript is disabled.
- Failed deployments now update the status of the deployment configuration more rapidly, reducing the time before the old deployment is scaled back up.
- Persistent volume claims (PVCs) are no longer blocked by the default SCC policy for users.
- Continue to support host ports on the **`oadm router`** command. Administrators can disable them with **`--host-ports=false`** when **`--host-network=false`** is also set.
- Events are now emitted when the cancellation of a deployment fails.
- When invoking a binary build, retry if the input image stream tag does not exist yet (because it may be in the process of being imported).
- Fixed a race condition in Kubernetes where endpoints might be partially updated (only have some pods) when the controller is restarted.
- Docker containers do not allow CPU quota less than **`10m`**, so set the minimum value.
- Do not sync **`DaemonSet`** objects that match all pods.
- The **`oc new-build`** command no longer fails when creating a binary build on a Git repository that does not have an upstream remote set.
- Fixed a race condition between scaled up routers where some changes might be ignored.
- Enable the etcd watch cache for Kubernetes resources, reducing memory use and duplicate watches.
- Change the **`RunOnce`** pod duration restrictor to act as a limit instead of override.
- Guarantee partially completed builds are cleaned up when cancelled.
- Check **`claimRef`** UID when processing a recycled persistent volume (PV) to prevent races.
- The **`ProjectRequestLimit`** plug-in now ignores projects in terminating state.
- The **`ConfigMap`** volume is now readable as non-root.
- The **system:image-auditor** role has been added for managing the image registry.

- Dynamic volume provisioning can now be disabled.
- Deployment pods should now be cancelled when deployments are cancelled in all cases.
- The deployer controller should now ensure deployments that are cancelled cannot become completed.
- Concurrent deployer pod creation is now prevented.
- Fixed an issue where a pod would never terminate if the registry it pulls images from was unavailable.
- Fixed precision of CPU to millicore and memory to Mi in the UI.
- The HAProxy router should now obfuscate the pod IP in when using cookies for session affinity.

2.5. 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 features are in Technology Preview:

- Introduced in OpenShift Enterprise 3.1.1, dynamic provisioning of persistent storage volumes from Amazon EBS, Google Compute Disk, OpenStack Cinder storage providers remains in Technology Preview for OpenShift Enterprise 3.2.

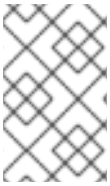
2.6. KNOWN ISSUES

- At the general availability release of OpenShift Enterprise 3.2, there was a known issue with upgrades for containerized installation environments from OpenShift Enterprise 3.1 to 3.2. Upgrades were only supported for clusters using the RPM-based installation method. As of the release of the RHBA-2016:1208 advisory, this issue has been resolved, and containerized upgrades are now supported after updating the **atomic-openshift-utils** package. (**BZ#1331097**, **BZ#1331380**, **BZ#1326642**, **BZ#1328950**)
- Internally-managed images cannot be pulled from an image reference referencing another image stream. See Deploying a Docker Registry for more information.
- See also the Known Issues for OpenShift Enterprise 3.2.1.1 .

2.7. ASYNCHRONOUS ERRATA UPDATES

Security, bug fix, and enhancement updates for OpenShift Enterprise 3.2 are released as asynchronous errata through the Red Hat Network. All OpenShift Enterprise 3.2 errata is available on the Red Hat Customer Portal. See the OpenShift Enterprise 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 Enterprise entitlements for OpenShift Enterprise 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 Enterprise 3.2. Versioned asynchronous releases, for example with the form OpenShift Enterprise 3.2.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.

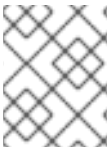
IMPORTANT

For any release, always review the instructions on upgrading your OpenShift Enterprise cluster properly.

2.7.1. RHBA-2016:1208 - atomic-openshift-utils Bug Fix Update

Issued: 2016-06-07

OpenShift Enterprise bug fix advisory RHBA-2016:1208, providing updated **atomic-openshift-utils** and **openshift-ansible** packages that fix several bugs, is now available.



NOTE

The instructions for applying this update are provided in the Solution section of the advisory.

Space precluded documenting all of the bug fixes in the advisory. This release includes the following bug fixes:

BZ#1331346

The installer's global proxy configuration support did not correctly configure the **`BuildDefaults`** admission controller. The installer has been updated to properly configure the **`BuildDefaults`** admission controller.

BZ#1337438

The installer was incorrectly adding extra single quotes to the ***/etc/sysconfig/docker*** file on each run due to an errant newline in the Ansible role. This bug fix updates the installer to remove the newline, and as a result the extra quotes no longer appear.

BZ#1334187

Due to **docker-1.9.1-40** packaging changes, it is no longer possible to use **`yum downgrade`** to downgrade from **docker-1.9.1** to **docker-1.8.2** as required for OpenShift Enterprise 3.1 and 3.0 installations. The installer has been updated to use **`yum swap`** to perform this downgrade when necessary.

BZ#1336780

Due to packaging changes in **docker-1.9.1-40**, containerized nodes did not have the correct Docker components mounted from the host into the node container. This prevented pods from being correctly configured to use the SDN. The missing components have been added to the containerized node configuration.

BZ#1330934

The installer did not properly convert the **`openshift_generate_no_proxy_hosts`** Ansible variable to a boolean so it may have been ignored. This bug fix updates the installer and the **`openshift_generate_no_proxy_hosts`** variable is now properly converted into a boolean ensuring that this variable produces the desired effect.

BZ#1330935

Containerized installations of OpenShift Enterprise (OSE) 3.1 were incorrectly receiving configuration defaults intended only to be used with OSE 3.2 installations. This meant that **dnsmasq** was configured for OSE 3.1 installs when it should not have been. This bug fix updates the fixed containerized version detection so that the correct default configurations are applied to OSE 3.1 installations. This means **dnsmasq** will no longer be included by default on OSE 3.1 containerized installations. This bug only affected containerized installations.

BZ#1331097

Previously under certain configurations, running the ***config.yml*** playbook could fully upgrade a containerized OpenShift Enterprise environment to the latest available image versions in configured registries. This bug fix updates the ***config.yml*** playbook to ensure images are not updated in these scenarios, and as a result the playbook can be run safely without inadvertently upgrading images to a newer version.

BZ#1331365

The quick installer has been updated to help make proxy-related questions more clear as to what information is being requested.

BZ#1331239

The quick installer incorrectly prompted for global proxy configuration settings when installing OpenShift Enterprise (OSE) 3.1. The installer has been updated to no longer prompt for global proxy settings in OSE 3.0 and 3.1 installations because this feature requires OSE 3.2.

BZ#1331236

Proxy variables previously were not written correctly to Ansible inventories by the quick installer. This bug fix updates the quick installer to ensure the **`openshift_http_proxy`**, **`openshift_https_proxy`**, **`openshift_no_proxy`** variables are written to inventories.

BZ#1334895

The NetworkManager dispatcher script which configures **dnsmasq** in OpenShift Enterprise 3.2 did not account for static network configurations. The dispatcher script has been updated to work for static network configurations.

BZ#1330920

The example Ansible inventories used the incorrect syntax for the **`openshift_generate_no_proxy_hosts`** variable. If administrators had copied and pasted the example syntax, it would not have taken effect. This bug fix updates the example inventories with the correct syntax for setting this variable.

BZ#1335063

The installer's global proxy configuration incorrectly quoted values in the master's ***sysconfig*** files. This meant that containerized installs using proxy configurations created by the installer would have failed. The installer has been updated to use proper quoting syntax.

BZ#1337425

The installer uses the **`repoquery`** command, which is provided by the **yum-utils** package and is not in Minimal installations of Red Hat Enterprise Linux 7.x. Ansible 1.9 installed this package before calling the command, but it is no longer installed starting with Ansible 2.0. This bug fix updates the installer to check that the **yum-utils** package is installed, and attempts to install it if it is not.

BZ#1334639

When configuring Ansible variables in inventories using raw booleans, installations could fail due to broken master configurations. This bug fix updates the installer to ensure that these values are properly converted to the master configuration files.

BZ#1334148

The default for the **`openshift_docker_hosted_registry_insecure`** Ansible variable is **`true`** but if it was set explicitly to **`true`** in an inventory, the installation would product an error. Setting the variable to **`false`** caused it to be ignored. This bug fix updates the installer to respect explicitly setting this value.

BZ#1329496

Previously, the **`osm_default_subdomain`** Ansible variable did not take effect when set. This was due to a backwards compatibility issue in the installer. This bug fix updates the installer to once again respect setting this variable.

BZ#1326045

The legacy **`cli_docker_options`** and **`cli_docker_log_options`** Ansible variables were not working due to use of an outdated host group that was since refactored. The variables were supposed to be migrated to the new format, using the **`openshift_docker_options`** and **`openshift_docker_log_options`** variables, respectively. This bug fix updates the installer so that the legacy variables can be used again.

BZ#1326642

During an upgrade, if the **`openshift_image_tag`** Ansible variable was set in an inventory to an image version that was older than the latest available, the latest available version was still set in the **systemd** unit files. This bug fix updates the installer to ensure the version set by **`openshift_image_tag`** is what actually gets set in the **systemd** unit files.

BZ#1336202

Upgrades from OpenShift Enterprise (OSE) 3.1 to 3.2 on RPM-based installations incorrectly attempted to pull the **openshift3/ose:latest** image. This step is only required for containerized installations and has been removed from RPM-based installations, eliminating the need to pull an unexpected image.

BZ#1331389

Previously, the **`cli_docker_additional_registries`** Ansible variable did not take effect during an upgrade. This was due to legacy options (**`cli_*`**) not being migrated during upgrades. This bug fix updates the installer to migrate these options correctly.

2.7.2. RHBA-2016:1343 - OpenShift Enterprise 3.2.1.1 bug fix and enhancement update

Issued: 2016-06-27

OpenShift Enterprise release 3.2.1.1 (RHBA-2016:1343) is now available.

See the following sections for notes on upgrading and details on the enhancements, bug fixes, and known issues included in this release.

2.7.2.1. Upgrading

Currently, you must use the manual cluster upgrade steps to apply this asynchronous errata update from OpenShift Enterprise 3.2.0 to 3.2.1.1. An automated playbook for this minor upgrade path is in development, and the upgrade documentation will be updated with instructions when it is available.

However, if you are upgrading from OpenShift Enterprise 3.1, you can use the ***v3_1_to_v3_2*** upgrade playbook as described in the Upgrading to OpenShift Enterprise 3.2 automated cluster upgrade steps to upgrade all the way to the latest asynchronous release at once.

2.7.2.2. Enhancements

Docker 1.10 Now Supported

Red Hat Enterprise Linux (RHEL) 7 Server and RHEL Atomic 7.2.5 ship Docker 1.10. OpenShift Enterprise 3.2 supported Docker 1.9.1 at its general availability release, and starting with OpenShift Enterprise 3.2.1 now supports Docker 1.10 as well. OpenShift Enterprise 3.2.1 also still supports Docker 1.9.1. If any images exist on a host when Docker is started after upgrading to 1.10, a lengthy upgrade process is triggered automatically for the remaining images. As such, Red Hat recommends removing all images before upgrading to Docker 1.10; this step is detailed in the upgrade documentation.

IMPORTANT

See Known Issues for more details on using OpenShift Enterprise and Docker 1.10.

2.7.2.3. Bug Fixes

BZ#1324179

Creation of the **`builder`** and **`deployer`** service accounts could be delayed for newly-created projects, during which time users could not build or deploy applications. This was caused by an issue when project templates defined a quota for secrets. This bug fix ensures that service accounts and their tokens are created quickly in this scenario (within seconds), and as a result users do not have to wait after project creation to build or deploy applications.

BZ#1327500

Pod and build names allow for up to 256 characters, however label values cannot be more than 64 characters. This caused builds to fail for build configurations with names longer than 64 characters, due to the invalid length set for the build pod's label. This bug fix truncates the value of build pod labels to 64 chars and relies on the build annotation to get the full name. As a result, builds no longer fail in this scenario.

BZ#1334249

When attempting to run a PostgreSQL slave pod from the upstream replica template, the pod could get stuck in CrashLoopBackOff status, citing a "MEMORY_LIMIT_IN_BYTES: unbound variable" error. This bug fix ensures that cgroup limits are properly handled for such pods, and as a result this issue no longer occurs.

BZ#1333122

Events related to quota failures for compute resources produced multiple identical events. This was due to errors describing why a request was rejected having a variable ordering of responses for the same root cause. This bug fix sorts resources in quota errors, and as a result duplicate events are avoided.

BZ#1334501

Previously when etcd watch cache was enabled, the API server would deliver a 410 HTTP response when a watch was attempted with a resourceVersion that was too old. The expected result was a 200 HTTP status, with a single watch event of type ERROR. This bug fix updates the API server to produce the same results in this case, regardless of whether watch cache is enabled. The "410 Gone" error is now returned as a watch error event, rather than as a HTTP 410 response.

BZ#1333172

Previously in the web console, it was difficult to tell the difference between links to route host names linking to actual running applications versus navigation links within the console. This was particularly difficult on the Browse page for a route. This bug fix updates the web console so that route host names are displayed with its entire web URL (protocol included), making it more obvious that it is a link to the host name.

BZ#1333898

If a project had a large number of builds, and then many were deleted, the graph in the web console showing the builds could become truncated and display poorly. This bug fix updates the web console to avoid this issue.

BZ#1334485

When a project had no services but had a deployment configuration with no deployments, the empty Overview page in the web console displayed a "No services to show" message. This bug fix updates the message to more specifically read "There are no services and no running deployments or pods."

BZ#1333003

Previously, information on downloading and logging in to the CLI tool was shown on the **About** page in the web console, linked from the **?** drop-down menu in the top navigation. This bug fix updates the web console to include a separate **Command Line Tools** page in this drop-down menu so that it is more obvious at a glance. A link has also been added to the new page from the **About** page.

BZ#1333118

When adding environment variable or label name-value pairs in the web console via **Add to Project**, it was previously unclear whether it required clicking the **Add** button to actually the commit the changes before hitting **Create** at the bottom of the page. This bug fix updates the web console to disable the **Create** button while uncommitted name-value pairs are entered. A "Please add or clear this name-value pair" message is also displayed until the pair has been added or cleared.

BZ#1331816

The web console has been updated to more accurately reflect memory limit values.

BZ#1333158

When scaling deployments in the web console, if multiple scaling requests were made in a short amount of time, it was possible for the operation to result with an incorrect number of replicas. This bug fix addresses a timing issue, and as a result the correct number of replicas are now set in this scenario.

BZ#1333590

Previously, template descriptions in the web console were collapsed into a single line and truncated with no way to expand. Because the description could contain important information or warnings, this bug fix updates the web console to now display the full text, split into multiple lines if needed.

BZ#1333163

When editing a YAML object in the web console, pressing CTRL+F or CMD+F to attempt to search the text did not appear to do anything. This bug fix updates the web console so that doing so causes a search box to appear in the UI, as expected.

BZ#1336526

The Documentation link in the **?** drop-down menu was hard-coded instead of using the method described in Customizing the Web Console. This bug fix updates the web console, and now this link can be customized as expected.

BZ#1322271

In previous releases, network metrics were not included when cluster metrics were enabled. This bug fix allows for network metrics to now be shown via REST API.

BZ#1340324

Due to newer releases of docker changing the path of the docker executable, containerized nodes

could fail to initialize the SDN because they cannot execute docker properly. This bug fix updates the containerized node image to accommodate this change, and as a result containerized nodes work properly with current and future versions of docker.

BZ#1334866

Previously, it was possible to set the **`metadata.deletionTimestamp`** parameter during the update of an object. However, **`deletionTimestamp`** and **`deletionGracePeriodSeconds`** fields should only be able to be set as a result of a delete API operation. This bug fix ensures that the parameter cannot be set during update, and any attempts now produce a "field is immutable; may only be changed via deletion" error.

BZ#1333932

The etcd watch cache was enabled in a previous release for Kubernetes resource types. This bug fix enables the etcd watch cache for all OpenShift resource types, as well.

BZ#1326523

This bug fix adds the **`MYSQL_MAX_ALLOWED_PACKET`** environment variable to the MySQL image, for setting the maximum size of one packet or any generated or intermediate string (default: 200M).

BZ#1320233

When the default HAProxy router reloaded its configuration during a resync (default interval: 10 minutes), it was possible to experience dropped connections to routes. This bug fix updates the **openshift3/ose-haproxy-router** image to suppress reloads during sync events, and as a result the HAProxy router no longer reloads periodically and connections to routes are no longer interrupted for this reason.

2.7.2.4. Known Issues

- **Registry pushes using AWS S3 storage considerably slower with Docker 1.10**:
  When pushing a local image to clusters using the registry with Amazon Simple Storage Service (S3) storage back end from the Amazon Web Services platform, the push takes considerably more time when using Docker 1.10 than Docker 1.9 when the Docker registry is version 2.2.x or earlier. OpenShift Enterprise 3.2 currently ships Docker registry 2.2.1. If you are using S3 storage with your registry, it is recommended that you do not upgrade to Docker 1.10 and OpenShift Enterprise 3.2.1 at this time, until a subsequent OpenShift Enterprise update is released that addresses the issue. (**BZ#1347022**)

- **Images from Docker Hub fail due to v2 image schema:**
  Docker Hub recently switched to only supporting v2 image schema, and Docker 1.10 defaults to converting to and using v2 schema when pushing and pulling images. OpenShift Enterprise 3.2 currently ships Docker registry 2.2.1, which does not support the v2 schema (none of the images provided in the Red Hat Registry at registry.access.redhat.com are currently v2 schema). If any image with v2 schema is introduced to the cluster, for example during an interaction with images from Docker Hub, Docker operations will fail. The issue exists for Docker 1.9 as well specifically as it relates to Docker Hub interactions. If you expect Docker Hub images to be used in your environment, it is recommended that you do not upgrade to Docker 1.10 at this time, until a subsequent OpenShift Enterprise update is released that addresses the issue. (**openshift/origin#8596**, **openshift/origin#9491**)

2.7.3. RHBA-2016:1383 - OpenShift Enterprise 3.2.1.4 bug fix and enhancement update

Issued: 2016-07-05

OpenShift Enterprise release 3.2.1.4 (RHBA-2016:1383) is now available. The list of bug fixes included in the update are documented in the Description section of the advisory.

2.7.3.1. Upgrading

At the initial release of OpenShift Enterprise 3.2.1.4, only the manual cluster upgrade steps were available for applying this asynchronous errata update from OpenShift Enterprise 3.2.x to 3.2.1.4. An automated playbook for this minor upgrade path was still in development at the time, which has now been released as of OpenShift Enterprise 3.2.1.9.

If you are upgrading from OpenShift Enterprise 3.1, you can use the ***v3_2*** upgrade playbook (previously located in a ***v3_1_to_v3_2*** directory) as described in the Upgrading to OpenShift Enterprise 3.2 automated cluster upgrade steps to upgrade all the way to the latest asynchronous release at once.

2.7.4. RHBA-2016:1466 - OpenShift Enterprise 3.2.1.9 security and bug fix update

Issued: 2016-07-20

OpenShift Enterprise release 3.2.1.9 (RHBA-2016:1466) is now available. The list of security and bug fixes included in the update are documented in the Description section of the advisory.

2.7.4.1. Upgrading

With the release of OpenShift Enterprise 3.2.1.9, an automated upgrade playbook is now available and supported for applying asynchronous errata updates within the OpenShift Enterprise 3.2 minor version (e.g., 3.2.1.4 to 3.2.1.9). See Upgrading to OpenShift Enterprise 3.2 Asynchronous Releases for instructions.

If you are upgrading from OpenShift Enterprise 3.1, you can use the ***v3_2*** upgrade playbook (previously located in a ***v3_1_to_v3_2*** directory) as described in the Upgrading to OpenShift Enterprise 3.2 automated cluster upgrade steps to upgrade all the way to the latest asynchronous release at once.

2.7.5. RHBA-2016:1608 - OpenShift Enterprise 3.2.1.13 bug fix and enhancement update

Issued: 2016-08-11

OpenShift Enterprise release 3.2.1.13 is now available. The list of packages, container images, and bug fixes included in the update are documented in the RHBA-2016:1608 advisory.

2.7.5.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated Cluster Upgrades for instructions.

2.7.6. RHBA-2016:1639 - atomic-openshift-utils Bug Fix and Enhancement Update

Issued: 2016-08-18

OpenShift Enterprise bug fix advisory RHBA-2016:1639, providing updated **atomic-openshift-utils** and **openshift-ansible** packages that fix several bugs and add enhancements, is now available.

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

2.7.6.1. 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
```

To update the default image streams to include the .NET Core S2I image, see Updating the Default Image Streams and Templates.

2.7.6.2. Enhancements

Image Streams for .NET Core S2I Image

Image stream definitions for the .NET Core on RHEL S2I image are now added during OpenShift Enterprise 3.2 installations. (**BZ#1365285**)

Ansible 2.x Now Required

The OpenShift Enterprise 3.2 playbooks now require Ansible 2.x. (**BZ#1359236**)

Named CA Certificates

Administrators can now add the CA for named certificates to the generated CA using the **`cafile`** option with the **`openshift_master_named_certificates`** Ansible variable. For example:

```
openshift_master_named_certificates=[{"certfile":
"/path/to/custom1.crt", "keyfile": "/path/to/custom1.key","cafile":
"/path/to/ca.crt"}]
```

(**BZ#1360754**)

Backup and Redeploy Cluster Certificates

Administrators can now backup and redeploy cluster certificates using the following Ansible playbook:

```
$ ansible-playbook -i <inventory_file> \
    /usr/share/ansible/openshift-ansible/playbooks/byo/openshift-
cluster/redeploy-certificates.yml
```

By default, the playbook retains the current OpenShift Enterprise CA. To replace the CA with a generated or custom CA:

```
$ ansible-playbook -i <inventory_file> \
    /usr/share/ansible/openshift-ansible/playbooks/byo/openshift-
cluster/redeploy-certificates.yml \
    --extra-vars "openshift_certificates_redeploy_ca=true"
```

(**BZ#1275648**)

2.7.6.3. Bug Fixes

BZ#1365379

The installer now enables the **`NoVolumeZoneConflict`** scheduler policy by default. This policy restricts that with pods with persistent volumes (PVs) be scheduled in the availability zone where its PV is located.

BZ#1329455

Previously, the quick installer had issues adding new nodes to existing clusters in certain configurations. This bug fix updates the installer to properly identify these configurations and allow new nodes to be added as expected.

BZ#1356463

If Docker 1.8.2 is installed on a host before starting an OpenShift Enterprise 3.2 installation, Ansible now reports the following message: "Cannot upgrade Docker to greater than or equal to 1.10, please upgrade or remove Docker manually, or use the Docker upgrade playbook if OpenShift is already installed."

BZ#1316378

Previously when attempting to perform an initial installation on all new hosts, the quick installer incorrectly detected an "Installed environment" if a stand-alone load balancer host was included. This bug fix updates the installer to properly identify these configurations and allow the installation to continue as expected.

BZ#1357801

Previously, the installer failed when defining customized router certificate files using the **`openshift_hosted_router_certificate`** Ansible variable. This bug fix ensures that the certificate contents are properly checked when using this variable, and as a result this issue no longer occurs.

BZ#1358101

The installation options for OpenShift Enterprise 3.1, OpenShift Enterprise 3.0, and Atomic Enterprise Platform have been removed from the OpenShift Enterprise 3.2 version of the quick installer. To install OpenShift Enterprise 3.1 or 3.0, use the **atomic-openshift-utils** package from the respective product version's repository.

BZ#1358723

Previously, the installer failed when using the **`openshift_hosted_router_replicas`** Ansible variable. This bug fix ensures that the number of router replicas can be set using this variable, and as a result this issue no longer occurs.

BZ#1357751

A section of the upgrade process was incorrectly running on RHEL Atomic Host systems during upgrades, which would fail due to a missing **`repoquery`** command. This bug fix modifies the upgrade process to skip this section that attempts to upgrade Docker, as this is not possible on RHEL Atomic Host. As a result, upgrades now complete successfully on RHEL Atomic Host systems.

2.7.7. RHSA-2016:1853 - OpenShift Enterprise 3.2.1.15 security and bug fix update

Issued: 2016-09-12

OpenShift Enterprise release 3.2.1.15 is now available. The list of packages, container images, and bug fixes included in the update are documented in the RHSA-2016:1853 advisory.

2.7.7.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated Cluster Upgrades for instructions.

2.7.8. RHSA-2016:2064 - OpenShift Enterprise 3.2.1.17 security update

Issued: 2016-10-17

OpenShift Enterprise release 3.2.1.17 is now available. The list of packages and security fixes included in the update are documented in the RHSA-2016:2064 advisory. The list of container images included in the update are documented in the RHBA-2016:2065 advisory.

2.7.8.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated Cluster Upgrades for instructions.

2.7.9. RHSA-2016:2915 - OpenShift Enterprise 3.2.1.21 security and bug fix update

Issued: 2016-12-07

OpenShift Enterprise release 3.2.1.21 is now available. The list of packages and bug fixes included in the update are documented in the RHSA-2016:2915 advisory. The list of container images included in the update are documented in the RHBA-2016:2916 advisory.

2.7.9.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.10. RHBA-2017:0199 - OpenShift Enterprise 3.2.1.23 bug fix update

Issued: 2017-01-26

OpenShift Enterprise release 3.2.1.23 is now available. The list of packages and bug fixes included in the update are documented in the RHBA-2017:0199 advisory. The list of container images included in the update are documented in the RHBA-2017:0204 advisory.

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

2.7.10.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.10.2. Bug Fixes

BZ#1412830

The extended certificate validation code (now enabled by default) would not allow some certificates that should be considered valid. This caused self-signed, expired, or not yet current certificates that were otherwise well-formed to be rejected. This bug fix changes the extended validation to allow those cases. As a result, those types of certificates are now allowed.

BZ#1404106

The **atomic-openshift-excluder** and **atomic-openshift-docker-excluder** packages did not properly configure yum to exclude the relevant packages. The excluder scripts have been updated to ensure the proper yum configuration is modified ensuring that the appropriate packages are excluded from yum operations.

2.7.11. RHBA-2017:0289 - OpenShift Enterprise 3.2.1.26 bug fix update

Issued: 2017-02-22

OpenShift Enterprise release 3.2.1.26 is now available. The list of packages included in the update are documented in the RHBA-2017:0289 advisory. The list of container images included in the update are documented in the RHBA-2017:0290 advisory.

The container images in this release have been updated using the **`rhel:7.3-66`** and **`jboss-base-7/jdk8:1.3-6`** base images, where applicable.

2.7.11.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.12. RHSA-2017:0448 - ansible and openshift-ansible Security and Bug Fix Update

Issued: 2017-03-06

OpenShift Enterprise security and bug fix advisory RHSA-2017:0448, providing updated **atomic-openshift-utils**, **ansible**, and **openshift-ansible** packages that fix several bugs and a security issue, is now available.

The security issue is documented in the advisory. However, space precluded documenting all of the non-security 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.

2.7.12.1. 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
```

2.7.12.2. Bug Fixes

This release fixes bugs for the following components:

Installer

- A change in Ansible 2.2.1.0 caused problems with the OpenShift Enterprise 3.2 plabooks. This bug fix works around the change. ensuring that playbooks run correctly on Ansible 2.2.1.0 and newer. (**BZ#1419533**)
- An Ansible 2.2.1.0 compatibility issue has been fixed in the quick installer. (**BZ#1421059**)
- When executing the installer on a remote host that is also included in the inventory, the firewall configuration could potentially cause the installer to hang. A 10 second delay has been added after reseting the firewall to avoid this problem from occurring. (**BZ#1379189**)
- Network Manager previously reset the **`net.ipv4.ip_forward`** parameter, causing OpenShift Enterprise to lose certain functionality. This bug fix ensures the installer sets the **`sysctl`** parameter on the system level, and as a result Network Manager restarts no longer interfere with the installation process. (**BZ#1415067**)

- A certificate expiry checker has been added to the installer tools. (**BZ#1417680**)

2.7.13. RHBA-2017:0512 - OpenShift Enterprise 3.2.1.28 bug fix update

Issued: 2017-03-15

OpenShift Enterprise release 3.2.1.28 is now available. The list of packages included in the update are documented in the RHBA-2017:0512 advisory. The list of container images included in the update are documented in the RHBA-2017:0513 advisory.

The container images in this release have been updated using the **`rhel:7.3-74`** and **`jboss-base-7/jdk8:1.3-10`** base images, where applicable.

2.7.13.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.13.2. Bug Fixes

This release fixes bugs for the following components:

Kubernetes

- This bug fix addresses an issue where the OpenShift Container Platform node logs a panic with a nil deference during volume teardown. (**BZ#1425301**)

Metrics

- The Heapster password was being set via a property value, so the password could be leaked by processes such as **`ps`**. This bug fix ensures the password is now being set via a system property. As a result, the password is no longer leaked by such processes. (**BZ#1427544**)
- The passwords for Hawkular Metrics were being set via a property value, so the passwords could be leaked by processes such as **`ps`**. This bug fix ensures the passwords are now being set via a property file. As a result, the passwords are no longer leaked by such processes. (**BZ#1417652**)

2.7.14. RHBA-2017:0865 - OpenShift Enterprise 3.2.1.30 bug fix update

Issued: 2017-04-04

OpenShift Enterprise release 3.2.1.30 is now available. The list of packages included in the update are documented in the RHBA-2017:0865 advisory. The list of container images included in the update are documented in the RHBA-2017:0866 advisory.

The container images in this release have been updated using the **`rhel:7.3-74`** base image, where applicable.

2.7.14.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.15. RHBA-2017:0989 - OpenShift Enterprise 3.2.1.31-2 bug fix update

Issued: 2017-04-19

OpenShift Enterprise release 3.2.1.31-2 is now available. The list of packages and bug fixes included in the update are documented in the RHBA-2017:0989 advisory. The list of container images included in the update are documented in the RHBA-2017:0990 advisory.

The container images in this release have been updated using the **`rhel:7.3-74`** base image, where applicable.

2.7.15.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.16. RHBA-2017:1129 - OpenShift Enterprise 3.2.1.31-4 bug fix update

Issued: 2017-04-26

OpenShift Enterprise release 3.2.1.31-4 is now available. The list of packages amd bug fixes 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.

The container images in this release have been updated using the **`rhel:7.3-74`** base image, where applicable.

2.7.16.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.17. RHBA-2017:1425 - OpenShift Enterprise 3.2.1.34 bug fix update

Issued: 2017-06-15

OpenShift Enterprise release 3.2.1.34 is now available. The packages 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 images for this release in the advisory. See the following sections for notes on upgrading and details on the images included in this release.

2.7.17.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

2.7.17.2. Images

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

```
openshift3/openvswitch:v3.2.1.34-1
openshift3/ose-pod:v3.2.1.34-1
rhel7/pod-infrastructure:v3.2.1.34-1
```

```
openshift3/ose:v3.2.1.34-1
openshift3/ose-docker-registry:v3.2.1.34-1
openshift3/ose-keepalived-ipfailover:v3.2.1.34-1
openshift3/ose-recycler:v3.2.1.34-1
openshift3/ose-f5-router:v3.2.1.34-1
openshift3/ose-deployer:v3.2.1.34-1
openshift3/node:v3.2.1.34-1
openshift3/ose-sti-builder:v3.2.1.34-1
openshift3/ose-docker-builder:v3.2.1.34-1
openshift3/ose-haproxy-router:v3.2.1.34-1
openshift3/logging-auth-proxy:3.2.1-14
openshift3/logging-deployment:3.2.1-14
openshift3/logging-elasticsearch:3.2.1-17
openshift3/logging-fluentd:3.2.1-12
openshift3/logging-kibana:3.2.1-12
openshift3/metrics-cassandra:3.2.1-14
openshift3/metrics-hawkular-metrics:3.2.1-13
openshift3/metrics-deployer:3.2.1-11
openshift3/metrics-heapster:3.2.1-12
```

2.7.18. RHBA-2017:1494 - OpenShift Enterprise 3.2.1.34-3 images update

Issued: 2017-06-19

OpenShift Enterprise release 3.2.1.34-3 is now available with images that have been rebuilt using the latest base images. The list of container images included in the update are documented in the RHBA-2017:1494 advisory.

2.7.18.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated In-place Cluster Upgrades for instructions.

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

Issued: 2017-06-29

OpenShift Enterprise 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.

2.7.19.1. 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
```

2.7.19.2. Bug Fixes

- 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 Enterprise 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#1462283**)

- 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#1464546**)

- 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#1460972**)

- If etcd 3.x or later was running on the host, a v3 snapshot database must be backed up as part of the backup process. If this directory is not included in the backup, then etcd failed to restore the backup even though v3 data was not used. This bug fix amends the etcd backup steps to ensure that the v3 snapshot database is included in backups. (**BZ#1440303**)

2.7.19.3. 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#1463775**)

2.7.20. RHSA-2018:3742 - OpenShift Enterprise 3.2.1.34 security and bug fix update

Issued: 2018-12-04

OpenShift Enterprise release 3.2.1.34 is now available. The list of packages, container images, and bug fixes included in the update are documented in the RHSA-2018:3742 advisory. The list of container images included in the update are documented in the RHBA-2018:3741 advisory.

2.7.20.1. Upgrading

To upgrade an existing OpenShift Enterprise 3.1 or 3.2 cluster to the latest 3.2 release, use the automated upgrade playbook. See Performing Automated 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 OPENSHIFT ENTERPRISE 2 AND OPENSHIFT ENTERPRISE 3

4.1. OVERVIEW

OpenShift Enterprise 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. THU JUN 29 2017

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:1666 - atomic-openshift-utils Bug Fix and Enhancement Update.

5.2. THU JUN 22 2017

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:1494 - OpenShift Container Platform 3.2.1.34-3 Images Update.
	Added issued dates for all Asynchronous Errata Updates. (**BZ#1463721**)

5.3. WED JUN 14 2017

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:1129 - OpenShift Container Platform 3.2.1.34 Bug Fix Update.

5.4. TUE APR 25 2017

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:0989 - OpenShift Container Platform 3.2.1.31-2 Bug Fix Update and RHBA-2017:1129 - OpenShift Container Platform 3.2.1.31-4 Bug Fix Update.

5.5. THU APR 06 2017

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:0865 - OpenShift Container Platform 3.2.1.30 bug fix update.

5.6. WED MAR 15 2017

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:0512 - OpenShift Enterprise 3.2.1.28 bug fix update.

5.7. TUE MAR 07 2017

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHSA-2017:0448 - ansible and openshift-ansible Security and Bug Fix Update.

5.8. THU FEB 23 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:0289 - OpenShift Enterprise 3.2.1.26 bug fix update.

5.9. THU JAN 26 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHBA-2017:0199 - OpenShift Enterprise 3.2.1.23 bug fix update.

5.10. THU DEC 08 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHSA-2016:2915 - OpenShift Enterprise 3.2.1.21 security and bug fix update.

5.11. MON OCT 31 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHSA-2016:2064 - OpenShift Enterprise 3.2.1.17 security update.

5.12. TUE SEP 20 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for RHSA-2016:1853 - OpenShift Enterprise 3.2.1.15 security and bug fix update.

5.13. TUE AUG 23 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Clarified in the DNS Changes section that the OpenShift Enterprise 3.2 DNS changes are not automatically applied to 3.1 clusters during upgrade.

5.14. THU AUG 18 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Updated the Asynchronous Errata Updates section and added a subsection for RHBA-2016:1639.

5.15. THU AUG 11 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Updated the Asynchronous Errata Updates section and added release notes for OpenShift Enterprise 3.2.1.13, including notes on upgrading.

5.16. WED JUL 20 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Updated the Asynchronous Errata Updates section and added release notes for OpenShift Enterprise 3.2.1.9, including notes on upgrading.

5.17. TUE JUL 05 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Updated the Asynchronous Errata Updates section and added release notes for OpenShift Enterprise 3.2.1.4, including notes on upgrading.

5.18. THU JUN 30 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Updated the Upgrading section in the OpenShift Enterprise 3.2.1.1 release notes to remove a caveat about containerized hosts and to add a note about the ***v3_1_to_v3_2*** upgrade playbook.

5.19. MON JUN 27 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Updated the Asynchronous Errata Updates section and added release notes for OpenShift Enterprise 3.2.1.1, including notes on upgrading and details on the enhancements, bug fixes, and known issues included in the release.

5.20. TUE JUN 07 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Updated the Known Issues section to declare support for containerized upgrades as of the RHBA-2016:1208 advisory.
	Updated the Asynchronous Errata Updates section and added a subsection for RHBA-2016:1208.

5.21. FRI JUN 03 2016

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Removed an incorrect support claim regarding storage drivers for the integrated Docker registry.

5.22. THU MAY 12 2016

OpenShift Enterprise 3.2 initial release.

Affected Topic	Description of Change
OpenShift Enterprise 3.2 Release Notes	Added release notes for initial release.