Table of Contents

* [OpenShift Container Platform Upgrades Lab](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_openshift_container_platform_upgrades_lab)
* [1. Prepare for Upgrade](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_prepare_for_upgrade)
  + [1.1. Log In](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_log_in)
  + [1.2. Check Cluster Health](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_check_cluster_health)
* [2. Upgrade Nodes, Masters, and Bastion](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_upgrade_nodes_masters_and_bastion)
* [3. Upgrade EFK and Logging Stacks](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_upgrade_efk_and_logging_stacks)
  + [3.1. Upgrade Existing EFK Logging Stack Deployment](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_upgrade_existing_efk_logging_stack_deployment)
  + [3.2. Upgrade Existing Cluster Metrics Deployment](https://www.opentlc.com/labs/ocp_operations/04_01_Upgrade_Lab.html#_upgrade_existing_cluster_metrics_deployment)

OpenShift Container Platform Upgrades Lab

In this lab, you upgrade your OpenShift cluster in-place from an earlier asynchornous errata release (v3.6.0.173.5) to a later release (v3.6.0.173.21). The steps are the same for upgrading to a new minor release.

In an in-place upgrade, the cluster upgrade is performed on all of the hosts in a single, running cluster—first masters and then nodes. Pods are evacuated off nodes and recreated on other running nodes before a node upgrade begins. This helps reduce downtime for user applications.

There are two ways to perform this upgrade:

* Upgrade all of the nodes in one playbook run.
* Upgrade the control plane first, then the nodes.

In this lab, you upgrade all of the nodes in one playbook run.

|  |  |
| --- | --- |
|  | There are detailed upgrade instructions available in the OpenShift documentation [Performing Automated In-place Cluster Upgrades](https://docs.openshift.com/container-platform/3.6/install_config/upgrading/automated_upgrades.html). |

1. Prepare for Upgrade

1.1. Log In

1. Log in to the Bastion host of your cluster and switch to the **root** user (**sudo -i**).
2. Verify that you are logged in to OpenShift as **system:admin** on the Bastion host and all of the master hosts:

[root@bastion ~]*# ansible masters,localhost -m shell -a"oc whoami"*

**Sample Output**

localhost | SUCCESS | rc=0 >>

system:admin

master1.GUID.internal | SUCCESS | rc=0 >>

system:admin

|  |  |
| --- | --- |
|  | Your **GUID** is different from the **GUID** listed in this sample output. The number of nodes may also be different from what is shown. |

1. If you are not **system:admin**, use SSH to remotely connect to the Master and log in as **system:admin**.

1.2. Check Cluster Health

In this section, you check the cluster health.

1. Retrieve the status of the nodes:

[root@bastion ~]*# oc get nodes*

1. Examine the output to make sure that all of the nodes are ready and the non-schedulable labels are correct:

**Sample Output**

NAME STATUS AGE VERSION

infranode1.GUID.internal Ready 2h v1.6.1+5115d708d7

master1.GUID.internal Ready,SchedulingDisabled 2h v1.6.1+5115d708d7

node1.GUID.internal Ready 2h v1.6.1+5115d708d7

node2.GUID.internal Ready 2h v1.6.1+5115d708d7

* + In your cluster, the only non-schedulable node should be the Master.

|  |  |
| --- | --- |
|  | Your **GUID** is different from the **GUID** listed in this sample output. The number of nodes may also be different from what is shown. The Version number may be different as well. |

1. Whether you choose to upgrade all of the hosts or perform your upgrade in phases, make sure that you have the latest version of **atomic-openshift-utils** on each node, including the Bastion host from which you are running the upgrade process.
   * This package also updates the **openshift-ansible-\*** packages, which provide the Ansible Playbooks used for the installation.
2. Make sure that Ansible can reach every host:

ansible all -m ping

1. On the Bastion host, open the **/etc/yum.repos.d/open\_ocp-workshop.repo** file in your favorite text editor and change every occurrence of **{version\_old}** to **{version\_new}**.
   * Expect your final file to look like this:
   * [rhel-7-server-rpms]
   * name=Red Hat Enterprise Linux 7
   * baseurl=http://repos.example.com/repos/ocp/3.6.173.0.21/rhel-7-server-rpms
   * enabled=1
   * gpgcheck=0
   * [rhel-7-server-rh-common-rpms]
   * name=Red Hat Enterprise Linux 7 Common
   * baseurl=http://repos.example.com/repos/ocp/3.6.173.0.21/rhel-7-server-rh-common-rpms
   * enabled=1
   * gpgcheck=0
   * [rhel-7-server-extras-rpms]
   * name=Red Hat Enterprise Linux 7 Extras
   * baseurl=http://repos.example.com/repos/ocp/3.6.173.0.21/rhel-7-server-extras-rpms
   * enabled=1
   * gpgcheck=0
   * [rhel-7-server-optional-rpms]
   * name=Red Hat Enterprise Linux 7 Optional
   * baseurl=http://repos.example.com/repos/ocp/3.6.173.0.21/rhel-7-server-optional-rpms
   * enabled=1
   * gpgcheck=0
   * [rhel-7-server-ose-3.6-rpms]
   * name=Red Hat Enterprise Linux 7 OSE 3.6
   * baseurl=http://repos.example.com/repos/ocp/3.6.173.0.21/rhel-7-server-ose-3.6-rpms
   * enabled=1
   * gpgcheck=0
   * *## Required since OCP 3.5*
   * [rhel-7-fast-datapath-rpms]
   * name=Red Hat Enterprise Linux Fast Datapath (RHEL 7 Server) (RPMs)
   * baseurl=http://repos.example.com/repos/ocp/3.6.173.0.21/rhel-7-fast-datapath-rpms
   * enabled=1

gpgcheck=0

|  |  |
| --- | --- |
|  | The hostname where your repositories are hosted are different for your lab. Do not change your hostname to **repos.example.com**. Your specific version will also be different: v3.6.0.173.21 |

1. Copy the file **open\_ocp-workshop.repo** to **/etc/yum.repos.d** on all of the hosts:

ansible all -m copy -a"src=/etc/yum.repos.d/open\_ocp-workshop.repo dest=/etc/yum.repos.d"

2. Upgrade Nodes, Masters, and Bastion

1. Update your Ansible Playbooks and OpenShift utilities to the latest available version:
2. yum repolist
3. ansible all -m shell -a"yum repolist"

yum -y update atomic-openshift-utils

1. Open the **/etc/ansible/hosts** file with your text editor and make sure the line **docker\_version="1.12.6"** is not commented out.
2. Run the update playbook to upgrade the entire cluster to the latest release:

ansible-playbook -i /etc/ansible/hosts /usr/share/ansible/openshift-ansible/playbooks/byo/openshift-cluster/upgrades/v3\_6/upgrade.yml

* + Expect the upgrade playbook to finish in about 15 minutes.

1. When the playbook is finished, examine the output and expect it to be similar to the following:

**Sample Output**

TASK [openshift\_excluder : Enable openshift excluder] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [master1.GUID.internal]

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

infranode1.GUID.internal : ok=196 changed=20 unreachable=0 failed=0

localhost : ok=26 changed=0 unreachable=0 failed=0

master1.GUID.internal : ok=519 changed=52 unreachable=0 failed=0

node1.GUID.internal : ok=196 changed=20 unreachable=0 failed=0

node2.GUID.internal : ok=196 changed=20 unreachable=0 failed=0

support1.GUID.internal : ok=53 changed=2 unreachable=0 failed=0

* + Again, the **GUID** and the number of hosts are likely to be different in your output.

1. Verify that all of the nodes are running:
2. [root@bastion ~]*# oc get nodes*
3. NAME STATUS AGE VERSION
4. infranode1.upg.internal Ready 13h v1.6.1+5115d708d7
5. master1.upg.internal Ready,SchedulingDisabled 13h v1.6.1+5115d708d7
6. node1.upg.internal Ready 13h v1.6.1+5115d708d7

node2.upg.internal Ready 13h v1.6.1+5115d708d7

1. Verify that all of the nodes have the correct package installed:
2. [root@bastion ~]*# ansible nodes -m shell -a"yum info atomic-openshift-node"|grep Version*
3. Version : 3.6.173.0.21
4. Version : 3.6.173.0.21
5. Version : 3.6.173.0.21

Version : 3.6.173.0.21

|  |  |
| --- | --- |
|  | Your specific version should be v3.6.0.173.21. |

1. Upgrade the OpenShift client on the Bastion host:

yum -y update atomic-openshift-clients

1. Verify the version of the client and server:
2. [root@bastion ~]*# oc version*
3. oc v3.6.173.0.21
4. kubernetes v1.6.1+5115d708d7
5. features: Basic-Auth GSSAPI Kerberos SPNEGO
6. Server https://master.GUID.example.opentlc.com
7. openshift v3.6.173.0.21

kubernetes v1.6.1+5115d708d7

|  |  |
| --- | --- |
|  | Again your specific OpenShift version should be v3.6.0.173.21. |

1. Verify that you are running the expected versions of **docker-registry** and router images:
2. [root@bastion ~]*# oc get -n default dc/docker-registry -o json | grep \"image\"*
3. "image": "openshift3/ose-docker-registry:v3.6.173.0.21",
4. [root@bastion ~]*# oc get -n default dc/router -o json | grep \"image\"*

"image": "openshift3/ose-haproxy-router:v3.6.173.0.21",

|  |  |
| --- | --- |
|  | Your specific Image tag version should be v3.6.0.173.21. |

3. Upgrade EFK and Logging Stacks

3.1. Upgrade Existing EFK Logging Stack Deployment

To upgrade an existing EFK (Elasticsearch + Fluentd + Kibana) logging stack deployment, you must use the provided **/usr/share/ansible/openshift-ansible/playbooks/byo/openshift-cluster/openshift-logging.yml** Ansible Playbook. This is the same playbook you use to deploy logging for the first time on an existing cluster.

1. Run the playbook:

ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/byo/openshift-cluster/openshift-logging.yml

* + Expect to see a successful completion:

**Sample Output**

PLAY [Update Master configs] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

skipping: no hosts matched

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

localhost : ok=12 changed=0 unreachable=0 failed=0

master1.GUID.internal : ok=154 changed=19 unreachable=0 failed=0

1. Verify that all of the logging pods are running:

[root@bastion ~]*# oc get pod -n logging*

**Sample Output**

NAME READY STATUS RESTARTS AGE

logging-curator-1-ls4mn 1/1 Running 0 1h

logging-es-data-master-tvq0assb-1-bpxww 1/1 Running 0 1h

logging-fluentd-9hlqx 1/1 Running 0 6m

logging-fluentd-hg9zg 1/1 Running 0 7m

logging-fluentd-rzw5t 1/1 Running 0 7m

logging-fluentd-sg42x 1/1 Running 0 6m

logging-kibana-2-12vzx 2/2 Running 0 7m

|  |  |
| --- | --- |
|  | Depending on the number of nodes in your cluster, the number of **logging-fluentd** pods may be different. |

3.2. Upgrade Existing Cluster Metrics Deployment

To upgrade an existing cluster metrics deployment, you must use the provided **/usr/share/ansible/openshift-ansible/playbooks/byo/openshift-cluster/openshift-metrics.yml** Ansible Playbook. This is the same playbook you use to deploy metrics for the first time on an existing cluster.

1. Run the playbook:

ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/byo/openshift-cluster/openshift-metrics.yml

* + Expect to see a successful completion:

**Sample Output**

PLAY [OpenShift Metrics] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

skipping: no hosts matched

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

localhost : ok=12 changed=0 unreachable=0 failed=0

master1.GUID.internal : ok=186 changed=37 unreachable=0 failed=0

1. Verify that all of the metrics pods are running:
2. [root@bastion ~]*# oc get pod -n openshift-infra*
3. NAME READY STATUS RESTARTS AGE
4. hawkular-cassandra-1-bdr2w 1/1 Running 0 3m
5. hawkular-metrics-2qhlg 1/1 Running 0 3m

heapster-b1xxw 1/1 Running 0 3m

You successfully upgraded your cluster.

Build Version: be32934b41da1e34c27e21e79bdf57d30c732a3e : Last updated 2018-01-19 05:59:04 EST