OpenShift Container Platform 4.17

Validation and troubleshooting

Validating and troubleshooting an OpenShift Container Platform installation

Last Updated: 2024-12-05

OpenShift Container Platform 4.17 Validation and troubleshooting

Validating and troubleshooting an OpenShift Container Platform installation

Legal Notice

Abstract

This document describes how to validate and troubleshoot an OpenShift Container Platform installation.

Table of Contents

CHAPTER 1. VALIDATING AN INSTALLATION 3
1.1. REVIEWING THE INSTALLATION LOG 3
1.2. VIEWING THE IMAGE PULL SOURCE 3
1.3. GETTING CLUSTER VERSION, STATUS, AND UPDATE DETAILS 4
1.4. VERIFYING THAT A CLUSTER USES SHORT-TERM CREDENTIALS 6
1.5. QUERYING THE STATUS OF THE CLUSTER NODES BY USING THE CLI 7
1.6. REVIEWING THE CLUSTER STATUS FROM THE OPENSHIFT CONTAINER PLATFORM WEB CONSOLE 8
1.7. REVIEWING THE CLUSTER STATUS FROM RED HAT OPENSHIFT CLUSTER MANAGER 9
1.8. CHECKING CLUSTER RESOURCE AVAILABILITY AND UTILIZATION 10
1.9. LISTING ALERTS THAT ARE FIRING 11
1.10. NEXT STEPS 11

CHAPTER 2. TROUBLESHOOTING INSTALLATION ISSUES 13
2.1. PREREQUISITES 13
2.2. GATHERING LOGS FROM A FAILED INSTALLATION 13
2.3. MANUALLY GATHERING LOGS WITH SSH ACCESS TO YOUR HOST(S) 14
2.4. MANUALLY GATHERING LOGS WITHOUT SSH ACCESS TO YOUR HOST(S) 15
2.5. GETTING DEBUG INFORMATION FROM THE INSTALLATION PROGRAM 15
2.6. REINSTALLING THE OPENSHIFT CONTAINER PLATFORM CLUSTER 16

CHAPTER 1. VALIDATING AN INSTALLATION

You can check the status of an OpenShift Container Platform cluster after an installation by following the procedures in this document.

1.1. REVIEWING THE INSTALLATION LOG

You can review a summary of an installation in the OpenShift Container Platform installation log. If an installation succeeds, the information required to access the cluster is included in the log.

Prerequisites

- You have access to the installation host.

Procedure

- Review the **.openshift_install.log** log file in the installation directory on your installation host:

```
$ cat <install_dir>/.openshift_install.log
```

Example output

Cluster credentials are included at the end of the log if the installation is successful, as outlined in the following example:

```
...
time="2020-12-03T09:50:47Z" level=info msg="Install complete!"
time="2020-12-03T09:50:47Z" level=info msg="To access the cluster as the system:admin
user when using 'oc', run 'export KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'"
time="2020-12-03T09:50:47Z" level=info msg="Access the OpenShift web-console here:
https://console-openshift-console.apps.mycluster.example.com"
time="2020-12-03T09:50:47Z" level=info msg="Login to the console with user: \"kubeadmin\",
and password: \"password\""
time="2020-12-03T09:50:47Z" level=debug msg="Time elapsed per stage:"
time="2020-12-03T09:50:47Z" level=debug msg="    Infrastructure: 6m45s"
time="2020-12-03T09:50:47Z" level=debug msg="Bootstrap Complete: 11m30s"
time="2020-12-03T09:50:47Z" level=debug msg=" Bootstrap Destroy: 1m5s"
time="2020-12-03T09:50:47Z" level=debug msg=" Cluster Operators: 17m31s"
time="2020-12-03T09:50:47Z" level=info msg="Time elapsed: 37m26s"
```

1.2. VIEWING THE IMAGE PULL SOURCE

For clusters with unrestricted network connectivity, you can view the source of your pulled images by using a command on a node, such as **crictl images**.

However, for disconnected installations, to view the source of pulled images, you must review the CRI-O logs to locate the **Trying to access** log entry, as shown in the following procedure. Other methods to view the image pull source, such as the **crictl images** command, show the non-mirrored image name, even though the image is pulled from the mirrored location.

Prerequisites

- You have access to the cluster as a user with the **cluster-admin** role.

Procedure

- Review the CRI-O logs for a master or worker node:

```
$  oc adm node-logs <node_name> -u crio
```

Example output

The **Trying to access** log entry indicates where the image is being pulled from.

```
...
Mar 17 02:52:50 ip-10-0-138-140.ec2.internal crio[1366]: time="2021-08-05
10:33:21.594930907Z" level=info msg="Pulling image: quay.io/openshift-release-dev/ocp-
release:4.10.0-ppc64le" id=abcd713b-d0e1-4844-ac1c-474c5b60c07c
name=/runtime.v1alpha2.ImageService/PullImage
Mar 17 02:52:50 ip-10-0-138-140.ec2.internal crio[1484]: time="2021-03-17
02:52:50.194341109Z" level=info msg="Trying to access \"li0317gcp1.mirror-
registry.qe.gcp.devcluster.openshift.com:5000/ocp/release@sha256:1926eae7cacb9c00f142ec
98b00628970e974284b6ddaf9a6a086cb9af7a6c31\""
Mar 17 02:52:50 ip-10-0-138-140.ec2.internal crio[1484]: time="2021-03-17
02:52:50.226788351Z" level=info msg="Trying to access \"li0317gcp1.mirror-
registry.qe.gcp.devcluster.openshift.com:5000/ocp/release@sha256:1926eae7cacb9c00f142ec
98b00628970e974284b6ddaf9a6a086cb9af7a6c31\""
...
```

The log might show the image pull source twice, as shown in the preceding example.

If your **ImageContentSourcePolicy** object lists multiple mirrors, OpenShift Container Platform attempts to pull the images in the order listed in the configuration, for example:

```
Trying to access \"li0317gcp1.mirror-
registry.qe.gcp.devcluster.openshift.com:5000/ocp/release@sha256:1926eae7cacb9c00f142ec
98b00628970e974284b6ddaf9a6a086cb9af7a6c31\"
Trying to access \"li0317gcp2.mirror-
registry.qe.gcp.devcluster.openshift.com:5000/ocp/release@sha256:1926eae7cacb9c00f142ec
98b00628970e974284b6ddaf9a6a086cb9af7a6c31\"
```

1.3. GETTING CLUSTER VERSION, STATUS, AND UPDATE DETAILS

You can view the cluster version and status by running the **oc get clusterversion** command. If the status shows that the installation is still progressing, you can review the status of the Operators for more information.

You can also list the current update channel and review the available cluster updates.

Prerequisites

- You have access to the cluster as a user with the **cluster-admin** role.
- You have installed the OpenShift CLI (**oc**).

Procedure

1. Obtain the cluster version and overall status:

```
$ oc get clusterversion
```

Example output

```
NAME      VERSION   AVAILABLE   PROGRESSING   SINCE   STATUS
version   4.6.4     True        False         6m25s   Cluster version is 4.6.4
```

The example output indicates that the cluster has been installed successfully.

2. If the cluster status indicates that the installation is still progressing, you can obtain more detailed progress information by checking the status of the Operators:

```
$ oc get clusteroperators.config.openshift.io
```

3. View a detailed summary of cluster specifications, update availability, and update history:

```
$ oc describe clusterversion
```

4. List the current update channel:

```
$ oc get clusterversion -o jsonpath='{.items[0].spec}{"\n"}'
```

Example output

```
{"channel":"stable-4.6","clusterID":"245539c1-72a3-41aa-9cec-72ed8cf25c5c"}
```

5. Review the available cluster updates:

```
$ oc adm upgrade
```

Example output

```
Cluster version is 4.6.4

Updates:

VERSION IMAGE
4.6.6   quay.io/openshift-release-dev/ocp-release@sha256:c7e8f18e8116356701bd23ae3a23fb9892dd5ea66c8300662ef30563d7104f39
```

Additional resources

- See Querying Operator status after installation for more information about querying Operator status if your installation is still progressing.
- See Troubleshooting Operator issues for information about investigating issues with Operators.
- See Updating a cluster using the web console for more information on updating your cluster.
- See Understanding update channels and releases for an overview about update release channels.

1.4. VERIFYING THAT A CLUSTER USES SHORT-TERM CREDENTIALS

You can verify that a cluster uses short-term security credentials for individual components by checking the Cloud Credential Operator (CCO) configuration and other values in the cluster.

Prerequisites

- You deployed an OpenShift Container Platform cluster using the Cloud Credential Operator utility (**ccoctl**) to implement short-term credentials.
- You installed the OpenShift CLI (**oc**).
- You are logged in as a user with **cluster-admin** privileges.

Procedure

- Verify that the CCO is configured to operate in manual mode by running the following command:

```
$ oc get cloudcredentials cluster \
  -o=jsonpath={.spec.credentialsMode}
```

  The following output confirms that the CCO is operating in manual mode:

  #### Example output

```
Manual
```

- Verify that the cluster does not have **root** credentials by running the following command:

```
$ oc get secrets \
  -n kube-system <secret_name>
```

  where **<secret_name>** is the name of the root secret for your cloud provider.

Platform	Secret name
Amazon Web Services (AWS)	**aws-creds**
Microsoft Azure	**azure-credentials**
Google Cloud Platform (GCP)	**gcp-credentials**

  An error confirms that the root secret is not present on the cluster.

  #### Example output for an AWS cluster

```
Error from server (NotFound): secrets "aws-creds" not found
```

- Verify that the components are using short-term security credentials for individual components by running the following command:

  –

```
$ oc get authentication cluster \
  -o jsonpath \
  --template='{ .spec.serviceAccountIssuer }'
```

This command displays the value of the **.spec.serviceAccountIssuer** parameter in the cluster **Authentication** object. An output of a URL that is associated with your cloud provider indicates that the cluster is using manual mode with short-term credentials that are created and managed from outside of the cluster.

- Azure clusters: Verify that the components are assuming the Azure client ID that is specified in the secret manifests by running the following command:

```
$ oc get secrets \
  -n openshift-image-registry installer-cloud-credentials \
  -o jsonpath='{.data}'
```

An output that contains the **azure_client_id** and **azure_federated_token_file** felids confirms that the components are assuming the Azure client ID.

- Azure clusters: Verify that the pod identity webhook is running by running the following command:

```
$ oc get pods \
  -n openshift-cloud-credential-operator
```

Example output

```
NAME                                        READY   STATUS    RESTARTS   AGE
cloud-credential-operator-59cf744f78-r8pbq  2/2     Running   2          71m
pod-identity-webhook-548f977b4c-859lz       1/1     Running   1          70m
```

1.5. QUERYING THE STATUS OF THE CLUSTER NODES BY USING THE CLI

You can verify the status of the cluster nodes after an installation.

Prerequisites

- You have access to the cluster as a user with the **cluster-admin** role.
- You have installed the OpenShift CLI (**oc**).

Procedure

1. List the status of the cluster nodes. Verify that the output lists all of the expected control plane and compute nodes and that each node has a **Ready** status:

```
$ oc get nodes
```

Example output

```
NAME                 STATUS   ROLES    AGE   VERSION
```

```
compute-1.example.com        Ready    worker   33m   v1.30.3
control-plane-1.example.com  Ready    master   41m   v1.30.3
control-plane-2.example.com  Ready    master   45m   v1.30.3
compute-2.example.com        Ready    worker   38m   v1.30.3
compute-3.example.com        Ready    worker   33m   v1.30.3
control-plane-3.example.com  Ready    master   41m   v1.30.3
```

2. Review CPU and memory resource availability for each cluster node:

```
$ oc adm top nodes
```

Example output

```
NAME                         CPU(cores)   CPU%   MEMORY(bytes)   MEMORY%
compute-1.example.com        128m         8%     1132Mi          16%
control-plane-1.example.com  801m         22%    3471Mi          23%
control-plane-2.example.com  1718m        49%    6085Mi          40%
compute-2.example.com        935m         62%    5178Mi          75%
compute-3.example.com        111m         7%     1131Mi          16%
control-plane-3.example.com  942m         26%    4100Mi          27%
```

Additional resources

- See Verifying node health for more details about reviewing node health and investigating node issues.

1.6. REVIEWING THE CLUSTER STATUS FROM THE OPENSHIFT CONTAINER PLATFORM WEB CONSOLE

You can review the following information in the **Overview** page in the OpenShift Container Platform web console:

- The general status of your cluster
- The status of the control plane, cluster Operators, and storage
- CPU, memory, file system, network transfer, and pod availability
- The API address of the cluster, the cluster ID, and the name of the provider
- Cluster version information
- Cluster update status, including details of the current update channel and available updates
- A cluster inventory detailing node, pod, storage class, and persistent volume claim (PVC) information
- A list of ongoing cluster activities and recent events

Prerequisites

- You have access to the cluster as a user with the **cluster-admin** role.

Procedure

- In the **Administrator** perspective, navigate to **Home → Overview**.

1.7. REVIEWING THE CLUSTER STATUS FROM RED HAT OPENSHIFT CLUSTER MANAGER

From the OpenShift Container Platform web console, you can review detailed information about the status of your cluster on OpenShift Cluster Manager.

Prerequisites

- You are logged in to OpenShift Cluster Manager.
- You have access to the cluster as a user with the **cluster-admin** role.

Procedure

1. Go to the **Cluster List** list in OpenShift Cluster Manager and locate your OpenShift Container Platform cluster.
2. Click the **Overview** tab for your cluster.
3. Review the following information about your cluster:
   - vCPU and memory availability and resource usage
   - The cluster ID, status, type, region, and the provider name
   - Node counts by node type
   - Cluster version details, the creation date of the cluster, and the name of the cluster owner
   - The life cycle support status of the cluster
   - Subscription information, including the service level agreement (SLA) status, the subscription unit type, the production status of the cluster, the subscription obligation, and the service level

     TIP

     To view the history for your cluster, click the **Cluster history** tab.
4. Navigate to the **Monitoring** page to review the following information:
   - A list of any issues that have been detected
   - A list of alerts that are firing
   - The cluster Operator status and version
   - The cluster's resource usage
5. Optional: You can view information about your cluster that Red Hat Insights collects by navigating to the **Overview** menu. From this menu you can view the following information:
   - Potential issues that your cluster might be exposed to, categorized by risk level

- Health-check status by category

Additional resources

- See Using Insights to identify issues with your cluster for more information about reviewing potential issues with your cluster.

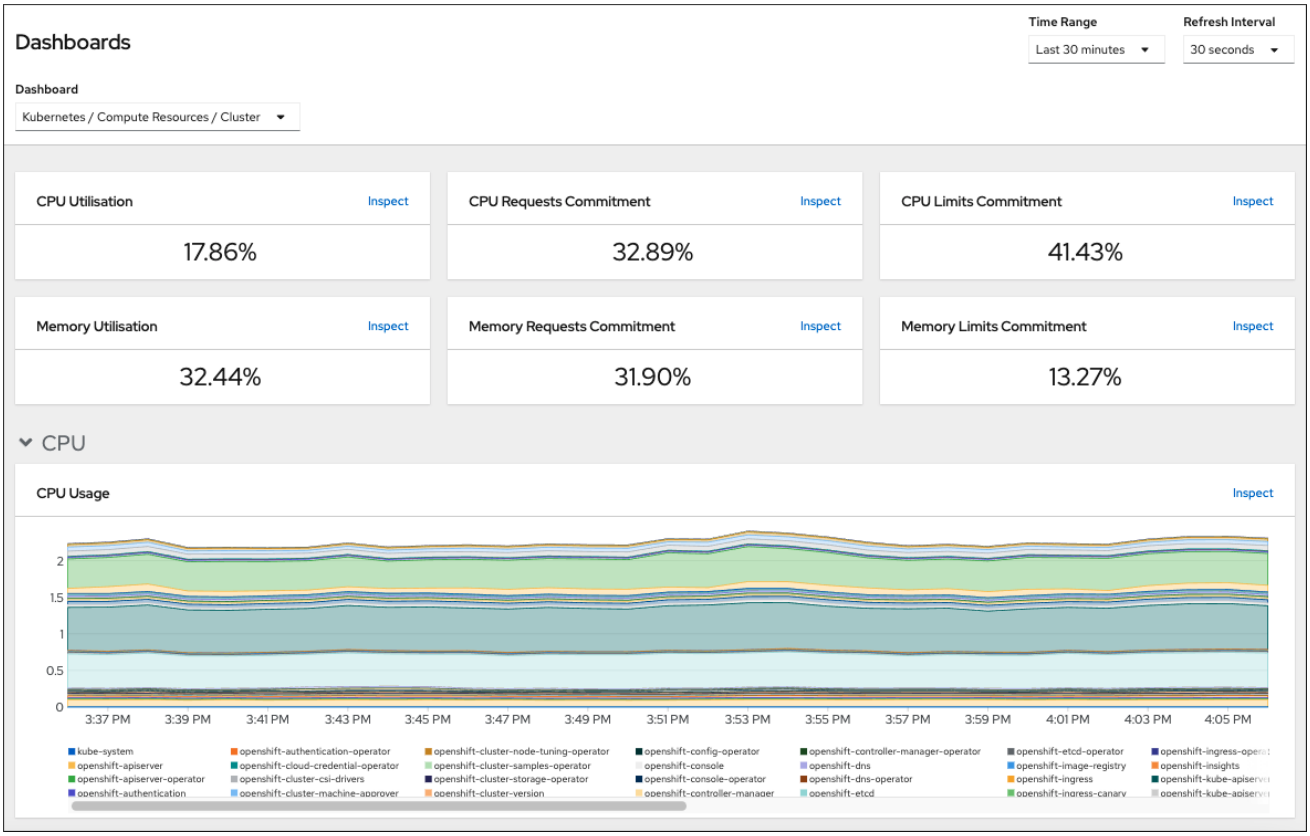
1.8. CHECKING CLUSTER RESOURCE AVAILABILITY AND UTILIZATION

OpenShift Container Platform provides a comprehensive set of monitoring dashboards that help you understand the state of cluster components.

In the **Administrator** perspective, you can access dashboards for core OpenShift Container Platform components, including:

- etcd
- Kubernetes compute resources
- Kubernetes network resources
- Prometheus
- Dashboards relating to cluster and node performance

Figure 1.1. Example compute resources dashboard



Prerequisites

- You have access to the cluster as a user with the **cluster-admin** role.

Procedure

1. In the **Administrator** perspective in the OpenShift Container Platform web console, navigate to **Observe → Dashboards**.
2. Choose a dashboard in the **Dashboard** list. Some dashboards, such as the **etcd** dashboard, produce additional sub-menus when selected.
3. Optional: Select a time range for the graphs in the **Time Range** list.
   - Select a pre-defined time period.
   - Set a custom time range by selecting **Custom time range** in the **Time Range** list.
     a. Input or select the **From** and **To** dates and times.
     b. Click **Save** to save the custom time range.
4. Optional: Select a **Refresh Interval**.
5. Hover over each of the graphs within a dashboard to display detailed information about specific items.

Additional resources

- See Monitoring overview for more information about the OpenShift Container Platform monitoring stack.

1.9. LISTING ALERTS THAT ARE FIRING

Alerts provide notifications when a set of defined conditions are true in an OpenShift Container Platform cluster. You can review the alerts that are firing in your cluster by using the Alerting UI in the OpenShift Container Platform web console.

Prerequisites

- You have access to the cluster as a user with the **cluster-admin** role.

Procedure

1. In the **Administrator** perspective, navigate to the **Observe → Alerting → Alerts** page.
2. Review the alerts that are firing, including their **Severity**, **State**, and **Source**.
3. Select an alert to view more detailed information in the **Alert Details** page.

Additional resources

- See Managing alerts for further details about alerting in OpenShift Container Platform.

1.10. NEXT STEPS

- See Troubleshooting installations if you experience issues when installing your cluster.

- After installing OpenShift Container Platform, you can further expand and customize your cluster.

CHAPTER 2. TROUBLESHOOTING INSTALLATION ISSUES

To assist in troubleshooting a failed OpenShift Container Platform installation, you can gather logs from the bootstrap and control plane machines. You can also get debug information from the installation program. If you are unable to resolve the issue using the logs and debug information, see Determining where installation issues occur for component-specific troubleshooting.

NOTE

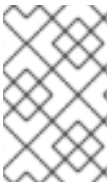
If your OpenShift Container Platform installation fails and the debug output or logs contain network timeouts or other connectivity errors, review the guidelines for configuring your firewall. Gathering logs from your firewall and load balancer can help you diagnose network-related errors.

2.1. PREREQUISITES

- You attempted to install an OpenShift Container Platform cluster and the installation failed.

2.2. GATHERING LOGS FROM A FAILED INSTALLATION

If you gave an SSH key to your installation program, you can gather data about your failed installation.



NOTE

You use a different command to gather logs about an unsuccessful installation than to gather logs from a running cluster. If you must gather logs from a running cluster, use the **oc adm must-gather** command.

Prerequisites

- Your OpenShift Container Platform installation failed before the bootstrap process finished. The bootstrap node is running and accessible through SSH.
- The **ssh-agent** process is active on your computer, and you provided the same SSH key to both the **ssh-agent** process and the installation program.
- If you tried to install a cluster on infrastructure that you provisioned, you must have the fully qualified domain names of the bootstrap and control plane nodes.

Procedure

1. Generate the commands that are required to obtain the installation logs from the bootstrap and control plane machines:
   - If you used installer-provisioned infrastructure, change to the directory that contains the installation program and run the following command:

```
$ ./openshift-install gather bootstrap --dir <installation_directory> 1
```

1 **installation_directory** is the directory you specified when you ran **./openshift-install create cluster**. This directory contains the OpenShift Container Platform definition files that the installation program creates.

For installer-provisioned infrastructure, the installation program stores information about the cluster, so you do not specify the hostnames or IP addresses.

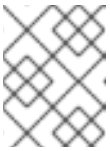
- If you used infrastructure that you provisioned yourself, change to the directory that contains the installation program and run the following command:

```
$ ./openshift-install gather bootstrap --dir <installation_directory> \ 1
  --bootstrap <bootstrap_address> \ 2
  --master <master_1_address> \ 3
  --master <master_2_address> \ 4
  --master <master_3_address> 5
```

1 For **installation_directory**, specify the same directory you specified when you ran **./openshift-install create cluster**. This directory contains the OpenShift Container Platform definition files that the installation program creates.

2 **<bootstrap_address>** is the fully qualified domain name or IP address of the cluster's bootstrap machine.

3 4 5 For each control plane, or master, machine in your cluster, replace **<master_*_address>** with its fully qualified domain name or IP address.



NOTE

A default cluster contains three control plane machines. List all of your control plane machines as shown, no matter how many your cluster uses.

Example output

```
INFO Pulling debug logs from the bootstrap machine
INFO Bootstrap gather logs captured here "<installation_directory>/log-bundle-
<timestamp>.tar.gz"
```

If you open a Red Hat support case about your installation failure, include the compressed logs in the case.

2.3. MANUALLY GATHERING LOGS WITH SSH ACCESS TO YOUR HOST(S)

Manually gather logs in situations where **must-gather** or automated collection methods do not work.

IMPORTANT

By default, SSH access to the OpenShift Container Platform nodes is disabled on the Red Hat OpenStack Platform (RHOSP) based installations.

Prerequisites

- You must have SSH access to your host(s).

Procedure

1. Collect the **bootkube.service** service logs from the bootstrap host using the **journalctl** command by running:

```
$ journalctl -b -f -u bootkube.service
```

2. Collect the bootstrap host's container logs using the podman logs. This is shown as a loop to get all of the container logs from the host:

```
$ for pod in $(sudo podman ps -a -q); do sudo podman logs $pod; done
```

3. Alternatively, collect the host's container logs using the **tail** command by running:

```
# tail -f /var/lib/containers/storage/overlay-containers/*/userdata/ctr.log
```

4. Collect the **kubelet.service** and **crio.service** service logs from the master and worker hosts using the **journalctl** command by running:

```
$ journalctl -b -f -u kubelet.service -u crio.service
```

5. Collect the master and worker host container logs using the **tail** command by running:

```
$ sudo tail -f /var/log/containers/*
```

2.4. MANUALLY GATHERING LOGS WITHOUT SSH ACCESS TO YOUR HOST(S)

Manually gather logs in situations where **must-gather** or automated collection methods do not work.

If you do not have SSH access to your node, you can access the systems journal to investigate what is happening on your host.

Prerequisites

- Your OpenShift Container Platform installation must be complete.
- Your API service is still functional.
- You have system administrator privileges.

Procedure

1. Access **journald** unit logs under **/var/log** by running:

```
$ oc adm node-logs --role=master -u kubelet
```

2. Access host file paths under **/var/log** by running:

```
$ oc adm node-logs --role=master --path=openshift-apiserver
```

2.5. GETTING DEBUG INFORMATION FROM THE INSTALLATION PROGRAM

You can use any of the following actions to get debug information from the installation program.

- Look at debug messages from a past installation in the hidden **.openshift_install.log** file. For example, enter:

```
$ cat ~/<installation_directory>/.openshift_install.log 1
```

1 For **installation_directory**, specify the same directory you specified when you ran **./openshift-install create cluster**.

- Change to the directory that contains the installation program and re-run it with **--log-level=debug**:

```
$ ./openshift-install create cluster --dir <installation_directory> --log-level debug 1
```

1 For **installation_directory**, specify the same directory you specified when you ran **./openshift-install create cluster**.

2.6. REINSTALLING THE OPENSHIFT CONTAINER PLATFORM CLUSTER

If you are unable to debug and resolve issues in the failed OpenShift Container Platform installation, consider installing a new OpenShift Container Platform cluster. Before starting the installation process again, you must complete thorough cleanup. For a user-provisioned infrastructure (UPI) installation, you must manually destroy the cluster and delete all associated resources. The following procedure is for an installer-provisioned infrastructure (IPI) installation.

Procedure

1. Destroy the cluster and remove all the resources associated with the cluster, including the hidden installer state files in the installation directory:

```
$ ./openshift-install destroy cluster --dir <installation_directory> 1
```

1 **installation_directory** is the directory you specified when you ran **./openshift-install create cluster**. This directory contains the OpenShift Container Platform definition files that the installation program creates.

2. Before reinstalling the cluster, delete the installation directory:

```
$ rm -rf <installation_directory>
```

3. Follow the procedure for installing a new OpenShift Container Platform cluster.

Additional resources

- Installing an OpenShift Container Platform cluster