OpenShift Container Platform 4.18

Installing on IBM Power Virtual Server

Installing OpenShift Container Platform on IBM Power Virtual Server

Last Updated: 2025-06-04

OpenShift Container Platform 4.18 Installing on IBM Power Virtual Server

Installing OpenShift Container Platform on IBM Power Virtual Server

Legal Notice

Abstract

This document describes how to install OpenShift Container Platform on IBM Power Virtual Server.

Table of Contents

CHAPTER 1. INSTALLATION METHODS

You can install OpenShift Container Platform on IBM Power® Virtual Server using installer-provisioned infrastructure. This process involves using an installation program to provision the underlying infrastructure for your cluster. Installing OpenShift Container Platform on IBM Power® Virtual Server using user-provisioned infrastructure is not supported at this time.

See Installation process for more information about installer-provisioned installation processes.

1.1. INSTALLING A CLUSTER ON INSTALLER-PROVISIONED INFRASTRUCTURE

You can install a cluster on IBM Power® Virtual Server infrastructure that is provisioned by the OpenShift Container Platform installation program by using one of the following methods:

- **Installing a customized cluster on IBM Power® Virtual Server**: You can install a customized cluster on IBM Power® Virtual Server infrastructure that the installation program provisions. The installation program allows for some customization to be applied at the installation stage. Many other customization options are available post-installation.

- **Installing a cluster on IBM Power® Virtual Server into an existing VPC**: You can install OpenShift Container Platform on IBM Power® Virtual Server into an existing Virtual Private Cloud (VPC). You can use this installation method if you have constraints set by the guidelines of your company, such as limits when creating new accounts or infrastructure.

- **Installing a private cluster on IBM Power® Virtual Server**: You can install a private cluster on IBM Power® Virtual Server. You can use this method to deploy OpenShift Container Platform on an internal network that is not visible to the internet.

- **Installing a cluster on IBM Power® Virtual Server in a restricted network**: You can install OpenShift Container Platform on IBM Power® Virtual Server on installer-provisioned infrastructure by using an internal mirror of the installation release content. You can use this method to install a cluster that does not require an active internet connection to obtain the software components.

1.2. CONFIGURING THE CLOUD CREDENTIAL OPERATOR UTILITY

The Cloud Credential Operator (CCO) manages cloud provider credentials as Kubernetes custom resource definitions (CRDs). To install a cluster on IBM Power® Virtual Server, you must set the CCO to **manual** mode as part of the installation process.

To create and manage cloud credentials from outside of the cluster when the Cloud Credential Operator (CCO) is operating in manual mode, extract and prepare the CCO utility (**ccoctl**) binary.

NOTE

The **ccoctl** utility is a Linux binary that must run in a Linux environment.

Prerequisites

- You have access to an OpenShift Container Platform account with cluster administrator access.

- You have installed the OpenShift CLI (**oc**).

Procedure

1. Set a variable for the OpenShift Container Platform release image by running the following command:

   ```
   $ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
   ```

2. Obtain the CCO container image from the OpenShift Container Platform release image by running the following command:

   ```
   $ CCO_IMAGE=$(oc adm release info --image-for='cloud-credential-operator'
   $RELEASE_IMAGE -a ~/.pull-secret)
   ```

   NOTE

   Ensure that the architecture of the **$RELEASE_IMAGE** matches the architecture of the environment in which you will use the **ccoctl** tool.

3. Extract the **ccoctl** binary from the CCO container image within the OpenShift Container Platform release image by running the following command:

   ```
   $ oc image extract $CCO_IMAGE \
     --file="/usr/bin/ccoctl.<rhel_version>" \ 1
     -a ~/.pull-secret
   ```

   1 For **<rhel_version>**, specify the value that corresponds to the version of Red Hat Enterprise Linux (RHEL) that the host uses. If no value is specified, **ccoctl.rhel8** is used by default. The following values are valid:

   - **rhel8**: Specify this value for hosts that use RHEL 8.
   - **rhel9**: Specify this value for hosts that use RHEL 9.

4. Change the permissions to make **ccoctl** executable by running the following command:

   ```
   $ chmod 775 ccoctl.<rhel_version>
   ```

Verification

- To verify that **ccoctl** is ready to use, display the help file. Use a relative file name when you run the command, for example:

  ```
  $ ./ccoctl.rhel9
  ```

  #### Example output

  ```
  OpenShift credentials provisioning tool

  Usage:
    ccoctl [command]

  Available Commands:
  ```

```
  aws         Manage credentials objects for AWS cloud
  azure       Manage credentials objects for Azure
  gcp         Manage credentials objects for Google cloud
  help        Help about any command
  ibmcloud    Manage credentials objects for {ibm-cloud-title}
  nutanix     Manage credentials objects for Nutanix

Flags:
  -h, --help   help for ccoctl

Use "ccoctl [command] --help" for more information about a command.
```

Additional resources

- Rotating API keys

1.3. NEXT STEPS

- Configuring an IBM Cloud® account

CHAPTER 2. CONFIGURING AN IBM CLOUD ACCOUNT

Before you can install OpenShift Container Platform, you must configure an IBM Cloud® account.

2.1. PREREQUISITES

- You have an IBM Cloud® account with a subscription. You cannot install OpenShift Container Platform on a free or on a trial IBM Cloud® account.

2.2. QUOTAS AND LIMITS ON IBM POWER VIRTUAL SERVER

The OpenShift Container Platform cluster uses several IBM Cloud® and IBM Power® Virtual Server components, and the default quotas and limits affect your ability to install OpenShift Container Platform clusters. If you use certain cluster configurations, deploy your cluster in certain regions, or run multiple clusters from your account, you might need to request additional resources for your IBM Cloud® account.

For a comprehensive list of the default IBM Cloud® quotas and service limits, see the IBM Cloud® documentation for Quotas and service limits .

Virtual Private Cloud

Each OpenShift Container Platform cluster creates its own Virtual Private Cloud (VPC). The default quota of VPCs per region is 10. If you have 10 VPCs created, you will need to increase your quota before attempting an installation.

Application load balancer

By default, each cluster creates two application load balancers (ALBs):

- Internal load balancer for the control plane API server
- External load balancer for the control plane API server

You can create additional **LoadBalancer** service objects to create additional ALBs. The default quota of VPC ALBs are 50 per region. To have more than 50 ALBs, you must increase this quota.

VPC ALBs are supported. Classic ALBs are not supported for IBM Power® Virtual Server.

Transit Gateways

Each OpenShift Container Platform cluster creates its own Transit Gateway to enable communication with a VPC. The default quota of transit gateways per account is 10. If you have 10 transit gateways created, you will need to increase your quota before attempting an installation.

Dynamic Host Configuration Protocol Service

There is a limit of one Dynamic Host Configuration Protocol (DHCP) service per IBM Power® Virtual Server instance.

Virtual Server Instances

By default, a cluster creates server instances with the following resources :

- 0.5 CPUs
- 32 GB RAM
- System Type: **s922**

- Processor Type: **uncapped**, **shared**
- Storage Tier: **Tier-3**

The following nodes are created:

- One bootstrap machine, which is removed after the installation is complete
- Three control plane nodes
- Three compute nodes

For more information, see Creating a Power Systems Virtual Server in the IBM Cloud® documentation.

2.3. CONFIGURING DNS RESOLUTION

How you configure DNS resolution depends on the type of OpenShift Container Platform cluster you are installing:

- If you are installing a public cluster, you use IBM Cloud® Internet Services (CIS).
- If you are installing a private cluster, you use IBM Cloud® DNS Services (DNS Services).

2.4. USING IBM CLOUD INTERNET SERVICES FOR DNS RESOLUTION

The installation program uses IBM Cloud® Internet Services (CIS) to configure cluster DNS resolution and provide name lookup for a public cluster.

NOTE

This offering does not support IPv6, so dual stack or IPv6 environments are not possible.

You must create a domain zone in CIS in the same account as your cluster. You must also ensure the zone is authoritative for the domain. You can do this using a root domain or subdomain.

Prerequisites

- You have installed the IBM Cloud® CLI.
- You have an existing domain and registrar. For more information, see the IBM® documentation.

Procedure

1. Create a CIS instance to use with your cluster:
   a. Install the CIS plugin:

   ```
   $ ibmcloud plugin install cis
   ```

   b. Log in to IBM Cloud® by using the CLI:

   ```
   $ ibmcloud login
   ```

   c. Create the CIS instance:

```
$ ibmcloud cis instance-create <instance_name> standard-next 1
```

1 At a minimum, you require a **Standard Next** plan for CIS to manage the cluster subdomain and its DNS records.

NOTE

After you have configured your registrar or DNS provider, it can take up to 24 hours for the changes to take effect.

2. Connect an existing domain to your CIS instance:

   a. Set the context instance for CIS:

   ```
   $ ibmcloud cis instance-set <instance_CRN> 1
   ```

   1 The instance CRN (Cloud Resource Name). For example: **ibmcloud cis instance-set crn:v1:bluemix:public:power-iaas:osa21:a/65b64c1f1c29460d8c2e4bbfbd893c2c:c09233ac-48a5-4ccb-a051-d1cfb3fc7eb5::**

   b. Add the domain for CIS:

   ```
   $ ibmcloud cis domain-add <domain_name> 1
   ```

   1 The fully qualified domain name. You can use either the root domain or subdomain value as the domain name, depending on which you plan to configure.

   NOTE

   A root domain uses the form **openshiftcorp.com**. A subdomain uses the form **clusters.openshiftcorp.com**.

3. Open the CIS web console, navigate to the **Overview** page, and note your CIS name servers. These name servers will be used in the next step.

4. Configure the name servers for your domains or subdomains at the domain's registrar or DNS provider. For more information, see the IBM Cloud® documentation.

2.5. IBM CLOUD IAM POLICIES AND API KEY

To install OpenShift Container Platform into your IBM Cloud® account, the installation program requires an IAM API key, which provides authentication and authorization to access IBM Cloud® service APIs. You can use an existing IAM API key that contains the required policies or create a new one.

For an IBM Cloud® IAM overview, see the IBM Cloud® documentation.

2.5.1. Pre-requisite permissions

Table 2.1. Pre-requisite permissions

Role	Access
Viewer, Operator, Editor, Administrator, Reader, Writer, Manager	Internet Services service in <resource_group> resource group
Viewer, Operator, Editor, Administrator, User API key creator, Service ID creator	IAM Identity Service service
Viewer, Operator, Administrator, Editor, Reader, Writer, Manager, Console Administrator	VPC Infrastructure Services service in <resource_group> resource group
Viewer	Resource Group: Access to view the resource group itself. The resource type should equal **Resource group**, with a value of <your_resource_group_name>.

2.5.2. Cluster-creation permissions

Table 2.2. Cluster-creation permissions

Role	Access
Viewer	<resource_group> (Resource Group Created for Your Team)
Viewer, Operator, Editor, Reader, Writer, Manager	All Identity and IAM enabled services in Default resource group
Viewer, Reader	Internet Services service
Viewer, Operator, Reader, Writer, Manager, Content Reader, Object Reader, Object Writer, Editor	Cloud Object Storage service
Viewer	Default resource group: The resource type should equal **Resource group**, with a value of**Default**. If your account administrator changed your account’s default resource group to something other than Default, use that value instead.
Viewer, Operator, Editor, Reader, Manager	Workspace for IBM Power® Virtual Server service in <resource_group> resource group
Viewer, Operator, Editor, Reader, Writer, Manager, Administrator	Internet Services service in <resource_group> resource group: CIS functional scope string equals reliability
Viewer, Operator, Editor	Transit Gateway service

Role	Access
Viewer, Operator, Editor, Administrator, Reader, Writer, Manager, Console Administrator	VPC Infrastructure Services service <resource_group> resource group

2.5.3. Access policy assignment

In IBM Cloud® IAM, access policies can be attached to different subjects:

- Access group (Recommended)
- Service ID
- User

NOTE

The recommended method is to define IAM access policies in an access group. This helps organize all the access required for OpenShift Container Platform and enables you to onboard users and service IDs to this group. You can also assign access to users and service IDs directly, if desired.

2.5.4. Creating an API key

You must create a user API key or a service ID API key for your IBM Cloud® account.

Prerequisites

- You have assigned the required access policies to your IBM Cloud® account.
- You have attached you IAM access policies to an access group, or other appropriate resource.

Procedure

- Create an API key, depending on how you defined your IAM access policies.
  For example, if you assigned your access policies to a user, you must create a user API key. If you assigned your access policies to a service ID, you must create a service ID API key . If your access policies are assigned to an access group, you can use either API key type. For more information on IBM Cloud® API keys, see Understanding API keys .

2.6. SUPPORTED IBM POWER VIRTUAL SERVER REGIONS AND ZONES

You can deploy an OpenShift Container Platform cluster to the following regions:

- **tor** (Toronto, Canada)
  - **tor01**
- **dal** (Dallas, USA)
  - **dal10**

  - **dal12**
- **eu-de** (Frankfurt, Germany)
  - **eu-de-1**
  - **eu-de-2**
- **lon** (London, UK)
  - **lon04**
- **mad** (Madrid, Spain)
  - **mad02**
  - **mad04**
- **osa** (Osaka, Japan)
  - **osa21**
- **sao** (Sao Paulo, Brazil)
  - **sao01**
  - **sao04**
- **syd** (Sydney, Australia)
  - **syd04**
  - **syd05**
- **wdc** (Washington DC, USA)
  - **wdc06**
  - **wdc07**
- **us-east** (Washington DC, United States)
  - **us-east**
- **us-south** (Dallas, United States)
  - **us-south**

You might optionally specify the IBM Cloud® region in which the installation program creates any VPC components.

NOTE

If you do not specify the region, the installation program selects the region closest to IBM Power Virtual Server zone you are deploying to.

IBM Cloud® supports the following regions:

- **us-east**
- **us-south**
- **eu-de**
- **eu-es**
- **eu-gb**
- **jp-osa**
- **au-syd**
- **br-sao**
- **ca-tor**
- **jp-tok**

2.7. NEXT STEPS

- Creating an IBM Power® Virtual Server workspace

CHAPTER 3. CREATING AN IBM POWER VIRTUAL SERVER WORKSPACE

3.1. CREATING AN IBM POWER VIRTUAL SERVER WORKSPACE

Use the following procedure to create an IBM Power® Virtual Server workspace.

Procedure

1. To create an IBM Power® Virtual Server workspace, complete step 1 to step 5 from the IBM Cloud® documentation for Creating an IBM Power® Virtual Server .

2. After it has finished provisioning, retrieve the 32-character alphanumeric Globally Unique Identifier (GUID) of your new workspace by entering the following command:

```
$ ibmcloud resource service-instance <workspace name>
```

3.2. NEXT STEPS

- Installing a cluster on IBM Power® Virtual Server with customizations

CHAPTER 4. INSTALLING A CLUSTER ON IBM POWER VIRTUAL SERVER WITH CUSTOMIZATIONS

In OpenShift Container Platform version 4.18, you can install a customized cluster on infrastructure that the installation program provisions on IBM Power Virtual Server. To customize the installation, you modify parameters in the **install-config.yaml** file before you install the cluster.

4.1. PREREQUISITES

- You reviewed details about the OpenShift Container Platform installation and update processes.
- You read the documentation on selecting a cluster installation method and preparing it for users.
- You configured an IBM Cloud® account to host the cluster.
- If you use a firewall, you configured it to allow the sites that your cluster requires access to.
- You configured the **ccoctl** utility before you installed the cluster. For more information, see Configuring the Cloud Credential Operator utility.

4.2. INTERNET ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, you require access to the internet to install your cluster.

You must have internet access to:

- Access OpenShift Cluster Manager to download the installation program and perform subscription management. If the cluster has internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Access Quay.io to obtain the packages that are required to install your cluster.
- Obtain the packages that are required to perform cluster updates.

IMPORTANT

If your cluster cannot have direct internet access, you can perform a restricted network installation on some types of infrastructure that you provision. During that process, you download the required content and use it to populate a mirror registry with the installation packages. With some installation types, the environment that you install your cluster in will not require internet access. Before you update the cluster, you update the content of the mirror registry.

4.3. GENERATING A KEY PAIR FOR CLUSTER NODE SSH ACCESS

During an OpenShift Container Platform installation, you can provide an SSH public key to the installation program. The key is passed to the Red Hat Enterprise Linux CoreOS (RHCOS) nodes through their Ignition config files and is used to authenticate SSH access to the nodes. The key is added to the **~/.ssh/authorized_keys** list for the **core** user on each node, which enables password-less authentication.

After the key is passed to the nodes, you can use the key pair to SSH in to the RHCOS nodes as the user **core**. To access the nodes through SSH, the private key identity must be managed by SSH for your local user.

If you want to SSH in to your cluster nodes to perform installation debugging or disaster recovery, you must provide the SSH public key during the installation process. The **./openshift-install gather** command also requires the SSH public key to be in place on the cluster nodes.

IMPORTANT

Do not skip this procedure in production environments, where disaster recovery and debugging is required.

NOTE

You must use a local key, not one that you configured with platform-specific approaches.

Procedure

1. If you do not have an existing SSH key pair on your local machine to use for authentication onto your cluster nodes, create one. For example, on a computer that uses a Linux operating system, run the following command:

```
$ ssh-keygen -t ed25519 -N '' -f <path>/<file_name> 1
```

1 Specify the path and file name, such as **~/.ssh/id_ed25519**, of the new SSH key. If you have an existing key pair, ensure your public key is in the your **~/.ssh** directory.

2. View the public SSH key:

```
$ cat <path>/<file_name>.pub
```

For example, run the following to view the **~/.ssh/id_ed25519.pub** public key:

```
$ cat ~/.ssh/id_ed25519.pub
```

3. Add the SSH private key identity to the SSH agent for your local user, if it has not already been added. SSH agent management of the key is required for password-less SSH authentication onto your cluster nodes, or if you want to use the **./openshift-install gather** command.

NOTE

On some distributions, default SSH private key identities such as **~/.ssh/id_rsa** and **~/.ssh/id_dsa** are managed automatically.

a. If the **ssh-agent** process is not already running for your local user, start it as a background task:

```
$ eval "$(ssh-agent -s)"
```

Example output

```
Agent pid 31874
```

4. Add your SSH private key to the **ssh-agent**:

```
$ ssh-add <path>/<file_name> 1
```

1 Specify the path and file name for your SSH private key, such as **~/.ssh/id_ed25519**

Example output

```
Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program.

4.4. OBTAINING THE INSTALLATION PROGRAM

Before you install OpenShift Container Platform, download the installation file on the host you are using for installation.

Prerequisites

- You have a computer that runs Linux or macOS, with 500 MB of local disk space.

Procedure

1. Go to the Cluster Type page on the Red Hat Hybrid Cloud Console. If you have a Red Hat account, log in with your credentials. If you do not, create an account.

   TIP

   You can also download the binaries for a specific OpenShift Container Platform release .

2. Select your infrastructure provider from the **Run it yourself** section of the page.

3. Select your host operating system and architecture from the dropdown menus under **OpenShift Installer** and click **Download Installer**.

4. Place the downloaded file in the directory where you want to store the installation configuration files.

IMPORTANT

- The installation program creates several files on the computer that you use to install your cluster. You must keep the installation program and the files that the installation program creates after you finish installing the cluster. Both of the files are required to delete the cluster.
- Deleting the files created by the installation program does not remove your cluster, even if the cluster failed during installation. To remove your cluster, complete the OpenShift Container Platform uninstallation procedures for your specific cloud provider.

5. Extract the installation program. For example, on a computer that uses a Linux operating system, run the following command:

```
$ tar -xvf openshift-install-linux.tar.gz
```

6. Download your installation pull secret from Red Hat OpenShift Cluster Manager . This pull secret allows you to authenticate with the services that are provided by the included authorities, including Quay.io, which serves the container images for OpenShift Container Platform components.

TIP

Alternatively, you can retrieve the installation program from the Red Hat Customer Portal, where you can specify a version of the installation program to download. However, you must have an active subscription to access this page.

4.5. EXPORTING THE API KEY

You must set the API key you created as a global variable; the installation program ingests the variable during startup to set the API key.

Prerequisites

- You have created either a user API key or service ID API key for your IBM Cloud® account.

Procedure

- Export your API key for your account as a global variable:

```
$ export IBMCLOUD_API_KEY=<api_key>
```

IMPORTANT

You must set the variable name exactly as specified; the installation program expects the variable name to be present during startup.

4.6. CREATING THE INSTALLATION CONFIGURATION FILE

You can customize the OpenShift Container Platform cluster you install on

Prerequisites

- You have the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

1. Create the **install-config.yaml** file.

   a. Change to the directory that contains the installation program and run the following command:

   ```
   $ ./openshift-install create install-config --dir <installation_directory> 1
   ```

   1 For **<installation_directory>**, specify the directory name to store the files that the installation program creates.

   When specifying the directory:

   - Use an empty directory. Some installation assets, such as bootstrap X.509 certificates, have short expiration intervals, therefore you must not reuse an installation directory. If you want to reuse individual files from another cluster installation, you can copy them into your directory. However, the file names for the installation assets might change between releases. Use caution when copying installation files from an earlier OpenShift Container Platform version.

     NOTE

     Always delete the **~/.powervs** directory to avoid reusing a stale configuration. Run the following command:

     ```
     $ rm -rf ~/.powervs
     ```

   b. At the prompts, provide the configuration details for your cloud:

      i. Optional: Select an SSH key to use to access your cluster machines.

      NOTE

      For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.

      ii. Select **powervs** as the platform to target.

      iii. Select the region to deploy the cluster to.

      iv. Select the zone to deploy the cluster to.

      v. Select the base domain to deploy the cluster to. The base domain corresponds to the public DNS zone that you created for your cluster.

      vi. Enter a descriptive name for your cluster.

2. Modify the **install-config.yaml** file. You can find more information about the available parameters in the "Installation configuration parameters" section.

3. Back up the **install-config.yaml** file so that you can use it to install multiple clusters.

IMPORTANT

The **install-config.yaml** file is consumed during the installation process. If you want to reuse the file, you must back it up now.

Additional resources

- Installation configuration parameters for IBM Power® Virtual Server

4.6.1. Sample customized install-config.yaml file for IBM Power Virtual Server

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

IMPORTANT

This sample YAML file is provided for reference only. You must obtain your **install-config.yaml** file by using the installation program and modify it.

```
apiVersion: v1
baseDomain: example.com
compute: ① ②
- architecture: ppc64le
  hyperthreading: Enabled ③
  name: worker
  platform:
    powervs:
      smtLevel: 8 ④
  replicas: 3
controlPlane: ⑤ ⑥
  architecture: ppc64le
  hyperthreading: Enabled ⑦
  name: master
  platform:
    powervs:
      smtLevel: 8 ⑧
  replicas: 3
metadata:
  creationTimestamp: null
  name: example-cluster-name
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  machineNetwork:
  - cidr: 192.168.0.0/24
  networkType: OVNKubernetes ⑨
  serviceNetwork:
```

```
  - 172.30.0.0/16
platform:
  powervs:
    userID: ibm-user-id
    region: powervs-region
    zone: powervs-zone
    powervsResourceGroup: "ibmcloud-resource-group" ⑩
    serviceInstanceGUID: "powervs-region-service-instance-guid"
    vpcRegion : vpc-region
publish: External
pullSecret: '{"auths": ...}' ⑪
sshKey: ssh-ed25519 AAAA... ⑫
```

① ⑤ If you do not provide these parameters and values, the installation program provides the default value.

② ⑥ The **controlPlane** section is a single mapping, but the compute section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Although both sections currently define a single machine pool, it is possible that OpenShift Container Platform will support defining multiple compute pools during installation. Only one control plane pool is used.

③ ⑦ Whether to enable or disable simultaneous multithreading, or **hyperthreading**. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

IMPORTANT

If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.

④ ⑧ The smtLevel specifies the level of SMT to set to the control plane and compute machines. The supported values are 1, 2, 4, 8, **'off'** and **'on'**. The default value is 8. The smtLevel **'off'** sets SMT to off and smtlevel **'on'** sets SMT to the default value 8 on the cluster nodes.

NOTE

When simultaneous multithreading (SMT), or hyperthreading is not enabled, one vCPU is equivalent to one physical core. When enabled, total vCPUs is computed as: (Thread(s) per core * Core(s) per socket) * Socket(s). The smtLevel controls the threads per core. Lower SMT levels may require additional assigned cores when deploying the cluster nodes. You can do this by setting the **'processors'** parameter in the **install-config.yaml** file to an appropriate value to meet the requirements for deploying OpenShift Container Platform successfully.

⑨ The cluster network plugin to install. The default value **OVNKubernetes** is the only supported value.

⑩ The name of an existing resource group.

⑪ Required. The installation program prompts you for this value.

NOTE

For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.

4.6.2. Configuring the cluster-wide proxy during installation

Production environments can deny direct access to the internet and instead have an HTTP or HTTPS proxy available. You can configure a new OpenShift Container Platform cluster to use a proxy by configuring the proxy settings in the **install-config.yaml** file.

Prerequisites

- You have an existing **install-config.yaml** file.
- You reviewed the sites that your cluster requires access to and determined whether any of them need to bypass the proxy. By default, all cluster egress traffic is proxied, including calls to hosting cloud provider APIs. You added sites to the **Proxy** object's **spec.noProxy** field to bypass the proxy if necessary.

NOTE

The **Proxy** object **status.noProxy** field is populated with the values of the **networking.machineNetwork[].cidr**, **networking.clusterNetwork[].cidr**, and **networking.serviceNetwork[]** fields from your installation configuration.

For installations on Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure, and Red Hat OpenStack Platform (RHOSP), the **Proxy** object **status.noProxy** field is also populated with the instance metadata endpoint (**169.254.169.254**).

Procedure

1. Edit your **install-config.yaml** file and add the proxy settings. For example:

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> ①
  httpsProxy: https://<username>:<pswd>@<ip>:<port> ②
  noProxy: example.com ③
additionalTrustBundle: | ④
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
additionalTrustBundlePolicy: <policy_to_add_additionalTrustBundle> ⑤
```

① A proxy URL to use for creating HTTP connections outside the cluster. The URL scheme must be **http**.

② A proxy URL to use for creating HTTPS connections outside the cluster.

3. A comma-separated list of destination domain names, IP addresses, or other network CIDRs to exclude from proxying. Preface a domain with **.** to match subdomains only. For

4. If provided, the installation program generates a config map that is named **user-ca-bundle** in the **openshift-config** namespace that contains one or more additional CA certificates that are required for proxying HTTPS connections. The Cluster Network Operator then creates a **trusted-ca-bundle** config map that merges these contents with the Red Hat Enterprise Linux CoreOS (RHCOS) trust bundle, and this config map is referenced in the **trustedCA** field of the **Proxy** object. The **additionalTrustBundle** field is required unless the proxy's identity certificate is signed by an authority from the RHCOS trust bundle.

5. Optional: The policy to determine the configuration of the **Proxy** object to reference the **user-ca-bundle** config map in the **trustedCA** field. The allowed values are **Proxyonly** and **Always**. Use **Proxyonly** to reference the **user-ca-bundle** config map only when **http/https** proxy is configured. Use **Always** to always reference the **user-ca-bundle** config map. The default value is **Proxyonly**.

NOTE

The installation program does not support the proxy **readinessEndpoints** field.

NOTE

If the installer times out, restart and then complete the deployment by using the **wait-for** command of the installer. For example:

```
$ ./openshift-install wait-for install-complete --log-level debug
```

2. Save the file and reference it when installing OpenShift Container Platform.

The installation program creates a cluster-wide proxy that is named **cluster** that uses the proxy settings in the provided **install-config.yaml** file. If no proxy settings are provided, a **cluster Proxy** object is still created, but it will have a nil **spec**.

NOTE

Only the **Proxy** object named **cluster** is supported, and no additional proxies can be created.

4.7. MANUALLY CREATING IAM

Installing the cluster requires that the Cloud Credential Operator (CCO) operate in manual mode. While the installation program configures the CCO for manual mode, you must specify the identity and access management secrets for you cloud provider.

You can use the Cloud Credential Operator (CCO) utility (**ccoctl**) to create the required IBM Cloud® resources.

Prerequisites

- You have configured the **ccoctl** binary.
- You have an existing **install-config.yaml** file.

Procedure

1. Edit the **install-config.yaml** configuration file so that it contains the **credentialsMode** parameter set to **Manual**.

   ### Example install-config.yaml configuration file

   ```
   apiVersion: v1
   baseDomain: cluster1.example.com
   credentialsMode: Manual 1
   compute:
   - architecture: ppc64le
     hyperthreading: Enabled
   ```

   1 This line is added to set the **credentialsMode** parameter to **Manual**.

2. To generate the manifests, run the following command from the directory that contains the installation program:

   ```
   $ ./openshift-install create manifests --dir <installation_directory>
   ```

3. From the directory that contains the installation program, set a **$RELEASE_IMAGE** variable with the release image from your installation file by running the following command:

   ```
   $ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
   ```

4. Extract the list of **CredentialsRequest** custom resources (CRs) from the OpenShift Container Platform release image by running the following command:

   ```
   $ oc adm release extract \
     --from=$RELEASE_IMAGE \
     --credentials-requests \
     --included \ 1
     --install-config=<path_to_directory_with_installation_configuration>/install-config.yaml \ 2
     --to=<path_to_directory_for_credentials_requests> 3
   ```

   1 The **--included** parameter includes only the manifests that your specific cluster configuration requires.

   2 Specify the location of the **install-config.yaml** file.

   3 Specify the path to the directory where you want to store the **CredentialsRequest** objects. If the specified directory does not exist, this command creates it.

   This command creates a YAML file for each **CredentialsRequest** object.

   ### Sample CredentialsRequest object

   ```
     apiVersion: cloudcredential.openshift.io/v1
     kind: CredentialsRequest
     metadata:
       labels:
   ```

```
    controller-tools.k8s.io: "1.0"
  name: openshift-image-registry-ibmcos
  namespace: openshift-cloud-credential-operator
spec:
  secretRef:
    name: installer-cloud-credentials
    namespace: openshift-image-registry
  providerSpec:
    apiVersion: cloudcredential.openshift.io/v1
    kind: IBMCloudProviderSpec
    policies:
    - attributes:
      - name: serviceName
        value: cloud-object-storage
      roles:
      - crn:v1:bluemix:public:iam::::role:Viewer
      - crn:v1:bluemix:public:iam::::role:Operator
      - crn:v1:bluemix:public:iam::::role:Editor
      - crn:v1:bluemix:public:iam::::serviceRole:Reader
      - crn:v1:bluemix:public:iam::::serviceRole:Writer
    - attributes:
      - name: resourceType
        value: resource-group
      roles:
      - crn:v1:bluemix:public:iam::::role:Viewer
```

5. Create the service ID for each credential request, assign the policies defined, create an API key, and generate the secret:

```
$ ccoctl ibmcloud create-service-id \
  --credentials-requests-dir=<path_to_credential_requests_directory> \ ①
  --name=<cluster_name> \ ②
  --output-dir=<installation_directory> \ ③
  --resource-group-name=<resource_group_name> ④
```

① Specify the directory containing the files for the component **CredentialsRequest** objects.

② Specify the name of the OpenShift Container Platform cluster.

③ Optional: Specify the directory in which you want the **ccoctl** utility to create objects. By default, the utility creates objects in the directory in which the commands are run.

④ Optional: Specify the name of the resource group used for scoping the access policies.

NOTE

If your cluster uses Technology Preview features that are enabled by the **TechPreviewNoUpgrade** feature set, you must include the **--enable-tech-preview** parameter.

If an incorrect resource group name is provided, the installation fails during the bootstrap phase. To find the correct resource group name, run the following command:

```
$ grep resourceGroup <installation_directory>/manifests/cluster-infrastructure-02-config.yml
```

Verification

- Ensure that the appropriate secrets were generated in your cluster's **manifests** directory.

4.8. DEPLOYING THE CLUSTER

You can install OpenShift Container Platform on a compatible cloud platform.

IMPORTANT

You can run the **create cluster** command of the installation program only once, during initial installation.

Prerequisites

- You have configured an account with the cloud platform that hosts your cluster.
- You have the OpenShift Container Platform installation program and the pull secret for your cluster.
- You have verified that the cloud provider account on your host has the correct permissions to deploy the cluster. An account with incorrect permissions causes the installation process to fail with an error message that displays the missing permissions.

Procedure

- Change to the directory that contains the installation program and initialize the cluster deployment:

```
$ ./openshift-install create cluster --dir <installation_directory> \ 1
    --log-level=info 2
```

1 For **<installation_directory>**, specify the location of your customized **./install-config.yaml** file.

2 To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.

Verification

When the cluster deployment completes successfully:

- The terminal displays directions for accessing your cluster, including a link to the web console and credentials for the **kubeadmin** user.

- Credential information also outputs to **<installation_directory>/.openshift_install.log**.

IMPORTANT

Do not delete the installation program or the files that the installation program creates. Both are required to delete the cluster.

Example output

```
...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export
KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-
console.apps.mycluster.example.com
INFO Login to the console with user: "kubeadmin", and password: "password"
INFO Time elapsed: 36m22s
```

IMPORTANT

- The Ignition config files that the installation program generates contain certificates that expire after 24 hours, which are then renewed at that time. If the cluster is shut down before renewing the certificates and the cluster is later restarted after the 24 hours have elapsed, the cluster automatically recovers the expired certificates. The exception is that you must manually approve the pending **node-bootstrapper** certificate signing requests (CSRs) to recover kubelet certificates. See the documentation for *Recovering from expired control plane certificates* for more information.

- It is recommended that you use Ignition config files within 12 hours after they are generated because the 24-hour certificate rotates from 16 to 22 hours after the cluster is installed. By using the Ignition config files within 12 hours, you can avoid installation failure if the certificate update runs during installation.

4.9. INSTALLING THE OPENSHIFT CLI

You can install the OpenShift CLI (**oc**) to interact with OpenShift Container Platform from a command-line interface. You can install **oc** on Linux, Windows, or macOS.

IMPORTANT

If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.18. Download and install the new version of **oc**.

Installing the OpenShift CLI on Linux

You can install the OpenShift CLI (**oc**) binary on Linux by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.

2. Select the architecture from the **Product Variant** drop-down list.

3. Select the appropriate version from the **Version** drop-down list.

4. Click **Download Now** next to the **OpenShift v4.18 Linux Clients** entry and save the file.

5. Unpack the archive:

   ```
   $ tar xvf <file>
   ```

6. Place the **oc** binary in a directory that is on your **PATH**.
   To check your **PATH**, execute the following command:

   ```
   $ echo $PATH
   ```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

  ```
  $ oc <command>
  ```

Installing the OpenShift CLI on Windows

You can install the OpenShift CLI (**oc**) binary on Windows by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.

2. Select the appropriate version from the **Version** drop-down list.

3. Click **Download Now** next to the **OpenShift v4.18 Windows Client** entry and save the file.

4. Unzip the archive with a ZIP program.

5. Move the **oc** binary to a directory that is on your **PATH**.
   To check your **PATH**, open the command prompt and execute the following command:

   ```
   C:\> path
   ```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

  ```
  C:\> oc <command>
  ```

Installing the OpenShift CLI on macOS

You can install the OpenShift CLI (**oc**) binary on macOS by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the appropriate version from the **Version** drop-down list.
3. Click **Download Now** next to the **OpenShift v4.18 macOS Clients** entry and save the file.

   > NOTE
   >
   > For macOS arm64, choose the **OpenShift v4.18 macOS arm64 Client** entry.

4. Unpack and unzip the archive.
5. Move the **oc** binary to a directory on your PATH.
   To check your **PATH**, open a terminal and execute the following command:

   ```
   $ echo $PATH
   ```

Verification

- Verify your installation by using an **oc** command:

  ```
  $ oc <command>
  ```

4.10. LOGGING IN TO THE CLUSTER BY USING THE CLI

You can log in to your cluster as a default system user by exporting the cluster **kubeconfig** file. The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server. The file is specific to a cluster and is created during OpenShift Container Platform installation.

Prerequisites

- You deployed an OpenShift Container Platform cluster.
- You installed the **oc** CLI.

Procedure

1. Export the **kubeadmin** credentials:

   ```
   $ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
   ```

   1 For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

2. Verify you can run **oc** commands successfully using the exported configuration:

   ```
   $ oc whoami
   ```

   Example output

```
system:admin
```

Additional resources

- Accessing the web console

4.11. TELEMETRY ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, the Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, requires internet access. If your cluster is connected to the internet, Telemetry runs automatically, and your cluster is registered to OpenShift Cluster Manager.

After you confirm that your OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually by using OpenShift Cluster Manager, use subscription watch to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

Additional resources

- About remote health monitoring

4.12. NEXT STEPS

- Customize your cluster
- If necessary, you can opt out of remote health reporting

CHAPTER 5. INSTALLING A CLUSTER ON IBM POWER VIRTUAL SERVER INTO AN EXISTING VPC

In OpenShift Container Platform version 4.18, you can install a cluster into an existing Virtual Private Cloud (VPC) on IBM Cloud®. The installation program provisions the rest of the required infrastructure, which you can then further customize. To customize the installation, you modify parameters in the **install-config.yaml** file before you install the cluster.

5.1. PREREQUISITES

- You reviewed details about the OpenShift Container Platform installation and update processes.
- You read the documentation on selecting a cluster installation method and preparing it for users.
- You configured an IBM Cloud® account to host the cluster.
- If you use a firewall, you configured it to allow the sites that your cluster requires access to.
- You configured the **ccoctl** utility before you installed the cluster. For more information, see Configuring the Cloud Credential Operator utility.

5.2. ABOUT USING A CUSTOM VPC

In OpenShift Container Platform 4.18, you can deploy a cluster using an existing IBM® Virtual Private Cloud (VPC).

Because the installation program cannot know what other components are in your existing subnets, it cannot choose subnet CIDRs and so forth. You must configure networking for the subnets to which you will install the cluster.

5.2.1. Requirements for using your VPC

You must correctly configure the existing VPC and its subnets before you install the cluster. The installation program does not create a VPC or VPC subnet in this scenario.

The installation program cannot:

- Subdivide network ranges for the cluster to use
- Set route tables for the subnets
- Set VPC options like DHCP

NOTE

The installation program requires that you use the cloud-provided DNS server. Using a custom DNS server is not supported and causes the installation to fail.

5.2.2. VPC validation

The VPC and all of the subnets must be in an existing resource group. The cluster is deployed to this resource group.

As part of the installation, specify the following in the **install-config.yaml** file:

- The name of the resource group
- The name of VPC
- The name of the VPC subnet

To ensure that the subnets that you provide are suitable, the installation program confirms that all of the subnets you specify exists.

NOTE

Subnet IDs are not supported.

5.2.3. Isolation between clusters

If you deploy OpenShift Container Platform to an existing network, the isolation of cluster services is reduced in the following ways:

- ICMP Ingress is allowed to the entire network.
- TCP port 22 Ingress (SSH) is allowed to the entire network.
- Control plane TCP 6443 Ingress (Kubernetes API) is allowed to the entire network.
- Control plane TCP 22623 Ingress (MCS) is allowed to the entire network.

5.3. INTERNET ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, you require access to the internet to install your cluster.

You must have internet access to:

- Access OpenShift Cluster Manager to download the installation program and perform subscription management. If the cluster has internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Access Quay.io to obtain the packages that are required to install your cluster.
- Obtain the packages that are required to perform cluster updates.

IMPORTANT

If your cluster cannot have direct internet access, you can perform a restricted network installation on some types of infrastructure that you provision. During that process, you download the required content and use it to populate a mirror registry with the installation packages. With some installation types, the environment that you install your cluster in will not require internet access. Before you update the cluster, you update the content of the mirror registry.

5.4. GENERATING A KEY PAIR FOR CLUSTER NODE SSH ACCESS

During an OpenShift Container Platform installation, you can provide an SSH public key to the installation program. The key is passed to the Red Hat Enterprise Linux CoreOS (RHCOS) nodes through their Ignition config files and is used to authenticate SSH access to the nodes. The key is added to the **~/.ssh/authorized_keys** list for the **core** user on each node, which enables password-less authentication.

After the key is passed to the nodes, you can use the key pair to SSH in to the RHCOS nodes as the user **core**. To access the nodes through SSH, the private key identity must be managed by SSH for your local user.

If you want to SSH in to your cluster nodes to perform installation debugging or disaster recovery, you must provide the SSH public key during the installation process. The **./openshift-install gather** command also requires the SSH public key to be in place on the cluster nodes.

IMPORTANT

Do not skip this procedure in production environments, where disaster recovery and debugging is required.

NOTE

You must use a local key, not one that you configured with platform-specific approaches.

Procedure

1. If you do not have an existing SSH key pair on your local machine to use for authentication onto your cluster nodes, create one. For example, on a computer that uses a Linux operating system, run the following command:

```
$ ssh-keygen -t ed25519 -N '' -f <path>/<file_name> 1
```

1 Specify the path and file name, such as **~/.ssh/id_ed25519**, of the new SSH key. If you have an existing key pair, ensure your public key is in the your **~/.ssh** directory.

NOTE

If you plan to install an OpenShift Container Platform cluster that uses the RHEL cryptographic libraries that have been submitted to NIST for FIPS 140-2/140-3 Validation on only the **x86_64**, **ppc64le**, and **s390x** architectures, do not create a key that uses the **ed25519** algorithm. Instead, create a key that uses the **rsa** or **ecdsa** algorithm.

2. View the public SSH key:

```
$ cat <path>/<file_name>.pub
```

For example, run the following to view the **~/.ssh/id_ed25519.pub** public key:

```
$ cat ~/.ssh/id_ed25519.pub
```

3. Add the SSH private key identity to the SSH agent for your local user, if it has not already been added. SSH agent management of the key is required for password-less SSH authentication onto your cluster nodes, or if you want to use the **./openshift-install gather** command.

NOTE

On some distributions, default SSH private key identities such as ~/**.ssh/id_rsa** and ~/**.ssh/id_dsa** are managed automatically.

a. If the **ssh-agent** process is not already running for your local user, start it as a background task:

```
$ eval "$(ssh-agent -s)"
```

Example output

```
Agent pid 31874
```

NOTE

If your cluster is in FIPS mode, only use FIPS-compliant algorithms to generate the SSH key. The key must be either RSA or ECDSA.

4. Add your SSH private key to the **ssh-agent**:

```
$ ssh-add <path>/<file_name> 1
```

1 Specify the path and file name for your SSH private key, such as ~/**.ssh/id_ed25519**

Example output

```
Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program.

5.5. OBTAINING THE INSTALLATION PROGRAM

Before you install OpenShift Container Platform, download the installation file on the host you are using for installation.

Prerequisites

- You have a computer that runs Linux or macOS, with 500 MB of local disk space.

Procedure

1. Go to the Cluster Type page on the Red Hat Hybrid Cloud Console. If you have a Red Hat account, log in with your credentials. If you do not, create an account.

TIP

You can also download the binaries for a specific OpenShift Container Platform release .

2. Select your infrastructure provider from the **Run it yourself** section of the page.

3. Select your host operating system and architecture from the dropdown menus under **OpenShift Installer** and click **Download Installer**.

4. Place the downloaded file in the directory where you want to store the installation configuration files.

IMPORTANT

- The installation program creates several files on the computer that you use to install your cluster. You must keep the installation program and the files that the installation program creates after you finish installing the cluster. Both of the files are required to delete the cluster.

- Deleting the files created by the installation program does not remove your cluster, even if the cluster failed during installation. To remove your cluster, complete the OpenShift Container Platform uninstallation procedures for your specific cloud provider.

5. Extract the installation program. For example, on a computer that uses a Linux operating system, run the following command:

```
$ tar -xvf openshift-install-linux.tar.gz
```

6. Download your installation pull secret from Red Hat OpenShift Cluster Manager . This pull secret allows you to authenticate with the services that are provided by the included authorities, including Quay.io, which serves the container images for OpenShift Container Platform components.

TIP

Alternatively, you can retrieve the installation program from the Red Hat Customer Portal, where you can specify a version of the installation program to download. However, you must have an active subscription to access this page.

5.6. EXPORTING THE API KEY

You must set the API key you created as a global variable; the installation program ingests the variable during startup to set the API key.

Prerequisites

- You have created either a user API key or service ID API key for your IBM Cloud® account.

Procedure

- Export your API key for your account as a global variable:

```
$ export IBMCLOUD_API_KEY=<api_key>
```

IMPORTANT

You must set the variable name exactly as specified; the installation program expects the variable name to be present during startup.

5.7. CREATING THE INSTALLATION CONFIGURATION FILE

You can customize the OpenShift Container Platform cluster you install on

Prerequisites

- You have the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

1. Create the **install-config.yaml** file.

   a. Change to the directory that contains the installation program and run the following command:

   ```
   $ ./openshift-install create install-config --dir <installation_directory> 1
   ```

   1 For **<installation_directory>**, specify the directory name to store the files that the installation program creates.

   When specifying the directory:

   - Use an empty directory. Some installation assets, such as bootstrap X.509 certificates, have short expiration intervals, therefore you must not reuse an installation directory. If you want to reuse individual files from another cluster installation, you can copy them into your directory. However, the file names for the installation assets might change between releases. Use caution when copying installation files from an earlier OpenShift Container Platform version.

     NOTE

     Always delete the **~/.powervs** directory to avoid reusing a stale configuration. Run the following command:

     ```
     $ rm -rf ~/.powervs
     ```

   b. At the prompts, provide the configuration details for your cloud:

      i. Optional: Select an SSH key to use to access your cluster machines.

NOTE

For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.

ii. Select **powervs** as the platform to target.

iii. Select the region to deploy the cluster to.

iv. Select the zone to deploy the cluster to.

v. Select the base domain to deploy the cluster to. The base domain corresponds to the public DNS zone that you created for your cluster.

vi. Enter a descriptive name for your cluster.

2. Modify the **install-config.yaml** file. You can find more information about the available parameters in the "Installation configuration parameters" section.

3. Back up the **install-config.yaml** file so that you can use it to install multiple clusters.

IMPORTANT

The **install-config.yaml** file is consumed during the installation process. If you want to reuse the file, you must back it up now.

Additional resources

- Installation configuration parameters for IBM Power® Virtual Server

5.7.1. Minimum resource requirements for cluster installation

Each cluster machine must meet the following minimum requirements:

Table 5.1. Minimum resource requirements

Machine	Operating System	vCPU [1]	Virtual RAM	Storage	Input/Output Per Second (IOPS)[2]
Bootstrap	RHCOS	4	16 GB	100 GB	300
Control plane	RHCOS	4	16 GB	100 GB	300
Compute	RHCOS, RHEL 8.6 and later [3]	2	8 GB	100 GB	300

1. One vCPU is equivalent to one physical core when simultaneous multithreading (SMT), or Hyper-Threading, is not enabled. When enabled, use the following formula to calculate the corresponding ratio: (threads per core × cores) × sockets = vCPUs.

2. OpenShift Container Platform and Kubernetes are sensitive to disk performance, and faster storage is recommended, particularly for etcd on the control plane nodes which require a 10 ms p99 fsync duration. Note that on many cloud platforms, storage size and IOPS scale together, so you might need to over-allocate storage volume to obtain sufficient performance.

3. As with all user-provisioned installations, if you choose to use RHEL compute machines in your cluster, you take responsibility for all operating system life cycle management and maintenance, including performing system updates, applying patches, and completing all other required tasks. Use of RHEL 7 compute machines is deprecated and has been removed in OpenShift Container Platform 4.10 and later.

NOTE

For OpenShift Container Platform version 4.18, RHCOS is based on RHEL version 9.4, which updates the micro-architecture requirements. The following list contains the minimum instruction set architectures (ISA) that each architecture requires:

- x86-64 architecture requires x86-64-v2 ISA
- ARM64 architecture requires ARMv8.0-A ISA
- IBM Power architecture requires Power 9 ISA
- s390x architecture requires z14 ISA

For more information, see Architectures (RHEL documentation).

If an instance type for your platform meets the minimum requirements for cluster machines, it is supported to use in OpenShift Container Platform.

Additional resources

- Optimizing storage

5.7.2. Sample customized install-config.yaml file for IBM Power Virtual Server

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

IMPORTANT

This sample YAML file is provided for reference only. You must obtain your **install-config.yaml** file by using the installation program and modify it.

```
apiVersion: v1
baseDomain: example.com
compute: 1 2
- architecture: ppc64le
  hyperthreading: Enabled 3
  name: worker
  platform:
    powervs:
      smtLevel: 8 4
  replicas: 3
```

```
controlPlane: 5 6
  architecture: ppc64le
  hyperthreading: Enabled 7
  name: master
  platform:
    powervs:
      smtLevel: 8 8
  replicas: 3
metadata:
  creationTimestamp: null
  name: example-cluster-existing-vpc
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14 9
    hostPrefix: 23
  machineNetwork:
  - cidr: 192.168.0.0/24
  networkType: OVNKubernetes 10
  serviceNetwork:
  - 172.30.0.0/16
platform:
  powervs:
    userID: ibm-user-id
    powervsResourceGroup: "ibmcloud-resource-group"
    region: powervs-region
    vpcRegion : vpc-region
    vpcName: name-of-existing-vpc 11
    zone: powervs-zone
    serviceInstanceGUID: "powervs-region-service-instance-guid"
credentialsMode: Manual
publish: External 12
pullSecret: '{"auths": ...}' 13
fips: false
sshKey: ssh-ed25519 AAAA... 14
```

1 5 If you do not provide these parameters and values, the installation program provides the default value.

2 6 The **controlPlane** section is a single mapping, but the compute section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Both sections currently define a single machine pool. Only one control plane pool is used.

3 7 Whether to enable or disable simultaneous multithreading, or **hyperthreading**. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

4 8 The smtLevel specifies the level of SMT to set to the control plane and compute machines. The supported values are 1, 2, 4, 8, **'off'** and **'on'**. The default value is 8. The smtLevel **'off'** sets SMT to off and smtlevel **'on'** sets SMT to the default value 8 on the cluster nodes.

NOTE

When simultaneous multithreading (SMT), or hyperthreading is not enabled, one vCPU is equivalent to one physical core. When enabled, total vCPUs is computed as (Thread(s) per core * Core(s) per socket) * Socket(s). The smtLevel controls the threads per core. Lower SMT levels may require additional assigned cores when deploying the cluster nodes. You can do this by setting the **'processors'** parameter in the **install-config.yaml** file to an appropriate value to meet the requirements for deploying OpenShift Container Platform successfully.

9 The machine CIDR must contain the subnets for the compute machines and control plane machines.

10 The cluster network plugin for installation. The supported value is **OVNKubernetes**.

11 Specify the name of an existing VPC.

12 Specify how to publish the user-facing endpoints of your cluster.

13 Required. The installation program prompts you for this value.

14 Provide the **sshKey** value that you use to access the machines in your cluster.

IMPORTANT

If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.

NOTE

For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.

5.7.3. Configuring the cluster-wide proxy during installation

Production environments can deny direct access to the internet and instead have an HTTP or HTTPS proxy available. You can configure a new OpenShift Container Platform cluster to use a proxy by configuring the proxy settings in the **install-config.yaml** file.

Prerequisites

- You have an existing **install-config.yaml** file.
- You reviewed the sites that your cluster requires access to and determined whether any of them need to bypass the proxy. By default, all cluster egress traffic is proxied, including calls to hosting cloud provider APIs. You added sites to the **Proxy** object's **spec.noProxy** field to bypass the proxy if necessary.

NOTE

The **Proxy** object **status.noProxy** field is populated with the values of the **networking.machineNetwork[].cidr**, **networking.clusterNetwork[].cidr**, and **networking.serviceNetwork[]** fields from your installation configuration.

For installations on Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure, and Red Hat OpenStack Platform (RHOSP), the **Proxy** object **status.noProxy** field is also populated with the instance metadata endpoint (**169.254.169.254**).

Procedure

1. Edit your **install-config.yaml** file and add the proxy settings. For example:

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> 1
  httpsProxy: https://<username>:<pswd>@<ip>:<port> 2
  noProxy: example.com 3
additionalTrustBundle: | 4
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
additionalTrustBundlePolicy: <policy_to_add_additionalTrustBundle> 5
```

1 A proxy URL to use for creating HTTP connections outside the cluster. The URL scheme must be **http**.

2 A proxy URL to use for creating HTTPS connections outside the cluster.

3 A comma-separated list of destination domain names, IP addresses, or other network CIDRs to exclude from proxying. Preface a domain with **.** to match subdomains only. For example, **.y.com** matches **x.y.com**, but not **y.com**. Use * to bypass the proxy for all destinations.

4 If provided, the installation program generates a config map that is named **user-ca-bundle** in the **openshift-config** namespace that contains one or more additional CA certificates that are required for proxying HTTPS connections. The Cluster Network Operator then creates a **trusted-ca-bundle** config map that merges these contents with the Red Hat Enterprise Linux CoreOS (RHCOS) trust bundle, and this config map is referenced in the **trustedCA** field of the **Proxy** object. The **additionalTrustBundle** field is required unless the proxy's identity certificate is signed by an authority from the RHCOS trust bundle.

5 Optional: The policy to determine the configuration of the **Proxy** object to reference the **user-ca-bundle** config map in the **trustedCA** field. The allowed values are **Proxyonly** and **Always**. Use **Proxyonly** to reference the **user-ca-bundle** config map only when **http/https** proxy is configured. Use **Always** to always reference the **user-ca-bundle** config map. The default value is **Proxyonly**.

NOTE

The installation program does not support the proxy **readinessEndpoints** field.

NOTE

If the installer times out, restart and then complete the deployment by using the **wait-for** command of the installer. For example:

```
$ ./openshift-install wait-for install-complete --log-level debug
```

2. Save the file and reference it when installing OpenShift Container Platform.

The installation program creates a cluster-wide proxy that is named **cluster** that uses the proxy settings in the provided **install-config.yaml** file. If no proxy settings are provided, a **cluster Proxy** object is still created, but it will have a nil **spec**.

NOTE

Only the **Proxy** object named **cluster** is supported, and no additional proxies can be created.

5.8. MANUALLY CREATING IAM

Installing the cluster requires that the Cloud Credential Operator (CCO) operate in manual mode. While the installation program configures the CCO for manual mode, you must specify the identity and access management secrets for you cloud provider.

You can use the Cloud Credential Operator (CCO) utility (**ccoctl**) to create the required IBM Cloud® resources.

Prerequisites

- You have configured the **ccoctl** binary.
- You have an existing **install-config.yaml** file.

Procedure

1. Edit the **install-config.yaml** configuration file so that it contains the **credentialsMode** parameter set to **Manual**.

   Example install-config.yaml configuration file

   ```
   apiVersion: v1
   baseDomain: cluster1.example.com
   credentialsMode: Manual 1
   compute:
   - architecture: ppc64le
     hyperthreading: Enabled
   ```

   1 This line is added to set the **credentialsMode** parameter to **Manual**.

2. To generate the manifests, run the following command from the directory that contains the installation program:

```
$ ./openshift-install create manifests --dir <installation_directory>
```

3. From the directory that contains the installation program, set a **$RELEASE_IMAGE** variable with the release image from your installation file by running the following command:

```
$ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
```

4. Extract the list of **CredentialsRequest** custom resources (CRs) from the OpenShift Container Platform release image by running the following command:

```
$ oc adm release extract \
  --from=$RELEASE_IMAGE \
  --credentials-requests \
  --included \ 1
  --install-config=<path_to_directory_with_installation_configuration>/install-config.yaml \ 2
  --to=<path_to_directory_for_credentials_requests> 3
```

1 The **--included** parameter includes only the manifests that your specific cluster configuration requires.

2 Specify the location of the **install-config.yaml** file.

3 Specify the path to the directory where you want to store the **CredentialsRequest** objects. If the specified directory does not exist, this command creates it.

This command creates a YAML file for each **CredentialsRequest** object.

Sample CredentialsRequest object

```
apiVersion: cloudcredential.openshift.io/v1
kind: CredentialsRequest
metadata:
  labels:
    controller-tools.k8s.io: "1.0"
  name: openshift-image-registry-ibmcos
  namespace: openshift-cloud-credential-operator
spec:
  secretRef:
    name: installer-cloud-credentials
    namespace: openshift-image-registry
  providerSpec:
    apiVersion: cloudcredential.openshift.io/v1
    kind: IBMCloudProviderSpec
    policies:
    - attributes:
      - name: serviceName
        value: cloud-object-storage
      roles:
      - crn:v1:bluemix:public:iam::::role:Viewer
      - crn:v1:bluemix:public:iam::::role:Operator
      - crn:v1:bluemix:public:iam::::role:Editor
      - crn:v1:bluemix:public:iam::::serviceRole:Reader
      - crn:v1:bluemix:public:iam::::serviceRole:Writer
- attributes:
  - name: resourceType
    value: resource-group
  roles:
  - crn:v1:bluemix:public:iam::::role:Viewer
```

5. Create the service ID for each credential request, assign the policies defined, create an API key, and generate the secret:

```
$ ccoctl ibmcloud create-service-id \
  --credentials-requests-dir=<path_to_credential_requests_directory> \ ①
  --name=<cluster_name> \ ②
  --output-dir=<installation_directory> \ ③
  --resource-group-name=<resource_group_name> ④
```

① Specify the directory containing the files for the component **CredentialsRequest** objects.

② Specify the name of the OpenShift Container Platform cluster.

③ Optional: Specify the directory in which you want the **ccoctl** utility to create objects. By default, the utility creates objects in the directory in which the commands are run.

④ Optional: Specify the name of the resource group used for scoping the access policies.

NOTE

If your cluster uses Technology Preview features that are enabled by the **TechPreviewNoUpgrade** feature set, you must include the **--enable-tech-preview** parameter.

If an incorrect resource group name is provided, the installation fails during the bootstrap phase. To find the correct resource group name, run the following command:

```
$ grep resourceGroup <installation_directory>/manifests/cluster-infrastructure-02-config.yml
```

Verification

- Ensure that the appropriate secrets were generated in your cluster's **manifests** directory.

5.9. DEPLOYING THE CLUSTER

You can install OpenShift Container Platform on a compatible cloud platform.

IMPORTANT

You can run the **create cluster** command of the installation program only once, during initial installation.

Prerequisites

- You have configured an account with the cloud platform that hosts your cluster.
- You have the OpenShift Container Platform installation program and the pull secret for your cluster.
- You have verified that the cloud provider account on your host has the correct permissions to deploy the cluster. An account with incorrect permissions causes the installation process to fail with an error message that displays the missing permissions.

Procedure

- Change to the directory that contains the installation program and initialize the cluster deployment:

```
$ ./openshift-install create cluster --dir <installation_directory> \ ①
    --log-level=info ②
```

① For **<installation_directory>**, specify the location of your customized **./install-config.yaml** file.

② To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.

Verification

When the cluster deployment completes successfully:

- The terminal displays directions for accessing your cluster, including a link to the web console and credentials for the **kubeadmin** user.
- Credential information also outputs to **<installation_directory>/.openshift_install.log**.

IMPORTANT

Do not delete the installation program or the files that the installation program creates. Both are required to delete the cluster.

Example output

```
...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export
KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-
console.apps.mycluster.example.com
INFO Login to the console with user: "kubeadmin", and password: "password"
INFO Time elapsed: 36m22s
```

IMPORTANT

- The Ignition config files that the installation program generates contain certificates that expire after 24 hours, which are then renewed at that time. If the cluster is shut down before renewing the certificates and the cluster is later restarted after the 24 hours have elapsed, the cluster automatically recovers the expired certificates. The exception is that you must manually approve the pending **node-bootstrapper** certificate signing requests (CSRs) to recover kubelet certificates. See the documentation for *Recovering from expired control plane certificates* for more information.

- It is recommended that you use Ignition config files within 12 hours after they are generated because the 24-hour certificate rotates from 16 to 22 hours after the cluster is installed. By using the Ignition config files within 12 hours, you can avoid installation failure if the certificate update runs during installation.

5.10. INSTALLING THE OPENSHIFT CLI

You can install the OpenShift CLI (**oc**) to interact with OpenShift Container Platform from a command-line interface. You can install **oc** on Linux, Windows, or macOS.

IMPORTANT

If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.18. Download and install the new version of **oc**.

Installing the OpenShift CLI on Linux

You can install the OpenShift CLI (**oc**) binary on Linux by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.

2. Select the architecture from the **Product Variant** drop-down list.

3. Select the appropriate version from the **Version** drop-down list.

4. Click **Download Now** next to the **OpenShift v4.18 Linux Clients** entry and save the file.

5. Unpack the archive:

```
$ tar xvf <file>
```

6. Place the **oc** binary in a directory that is on your **PATH**.
   To check your **PATH**, execute the following command:

```
$ echo $PATH
```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

```
$ oc <command>
```

Installing the OpenShift CLI on Windows

You can install the OpenShift CLI (**oc**) binary on Windows by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the appropriate version from the **Version** drop-down list.
3. Click **Download Now** next to the **OpenShift v4.18 Windows Client** entry and save the file.
4. Unzip the archive with a ZIP program.
5. Move the **oc** binary to a directory that is on your **PATH**.
   To check your **PATH**, open the command prompt and execute the following command:

   ```
   C:\> path
   ```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

  ```
  C:\> oc <command>
  ```

Installing the OpenShift CLI on macOS

You can install the OpenShift CLI (**oc**) binary on macOS by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the appropriate version from the **Version** drop-down list.
3. Click **Download Now** next to the **OpenShift v4.18 macOS Clients** entry and save the file.

   NOTE

   For macOS arm64, choose the **OpenShift v4.18 macOS arm64 Client** entry.

4. Unpack and unzip the archive.
5. Move the **oc** binary to a directory on your PATH.
   To check your **PATH**, open a terminal and execute the following command:

   ```
   $ echo $PATH
   ```

Verification

- Verify your installation by using an **oc** command:

  ```
  $ oc <command>
  ```

5.11. LOGGING IN TO THE CLUSTER BY USING THE CLI

You can log in to your cluster as a default system user by exporting the cluster **kubeconfig** file. The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server. The file is specific to a cluster and is created during OpenShift Container Platform installation.

Prerequisites

- You deployed an OpenShift Container Platform cluster.
- You installed the **oc** CLI.

Procedure

1. Export the **kubeadmin** credentials:

   ```
   $ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
   ```

   1 For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

2. Verify you can run **oc** commands successfully using the exported configuration:

   ```
   $ oc whoami
   ```

   Example output

   ```
   system:admin
   ```

Additional resources

- Accessing the web console

5.12. TELEMETRY ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, the Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, requires internet access. If your cluster is connected to the internet, Telemetry runs automatically, and your cluster is registered to OpenShift Cluster Manager.

After you confirm that your OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually by using OpenShift Cluster Manager, use subscription watch to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

Additional resources

- About remote health monitoring

5.13. NEXT STEPS

- Customize your cluster
- Optional: Opt out of remote health reporting

CHAPTER 6. INSTALLING A PRIVATE CLUSTER ON IBM POWER VIRTUAL SERVER

In OpenShift Container Platform version 4.18, you can install a private cluster into an existing VPC and IBM Power® Virtual Server Workspace. The installation program provisions the rest of the required infrastructure, which you can further customize. To customize the installation, you modify parameters in the **install-config.yaml** file before you install the cluster.

6.1. PREREQUISITES

- You reviewed details about the OpenShift Container Platform installation and update processes.
- You read the documentation on selecting a cluster installation method and preparing it for users.
- You configured an IBM Cloud® account to host the cluster.
- If you use a firewall, you configured it to allow the sites that your cluster requires access to.
- You configured the **ccoctl** utility before you installed the cluster. For more information, see Configuring the Cloud Credential Operator utility.

6.2. PRIVATE CLUSTERS

You can deploy a private OpenShift Container Platform cluster that does not expose external endpoints. Private clusters are accessible from only an internal network and are not visible to the internet.

By default, OpenShift Container Platform is provisioned to use publicly-accessible DNS and endpoints. A private cluster sets the DNS, Ingress Controller, and API server to private when you deploy your cluster. This means that the cluster resources are only accessible from your internal network and are not visible to the internet.

IMPORTANT

If the cluster has any public subnets, load balancer services created by administrators might be publicly accessible. To ensure cluster security, verify that these services are explicitly annotated as private.

To deploy a private cluster, you must:

- Use existing networking that meets your requirements.
- Create a DNS zone using IBM Cloud® DNS Services and specify it as the base domain of the cluster. For more information, see "Using IBM Cloud® DNS Services to configure DNS resolution".
- Deploy from a machine that has access to:
  - The API services for the cloud to which you provision.
  - The hosts on the network that you provision.

- The internet to obtain installation media.

You can use any machine that meets these access requirements and follows your company’s guidelines. For example, this machine can be a bastion host on your cloud network or a machine that has access to the network through a VPN.

6.3. PRIVATE CLUSTERS IN IBM POWER VIRTUAL SERVER

To create a private cluster on IBM Power® Virtual Server, you must provide an existing private Virtual Private Cloud (VPC) and subnets to host the cluster. The installation program must also be able to resolve the DNS records that the cluster requires. The installation program configures the Ingress Operator and API server for only internal traffic.

The cluster still requires access to internet to access the IBM Cloud® APIs.

The following items are not required or created when you install a private cluster:

- Public subnets
- Public network load balancers, which support public Ingress
- A public DNS zone that matches the **baseDomain** for the cluster

You will also need to create an IBM® DNS service containing a DNS zone that matches your **baseDomain**. Unlike standard deployments on Power VS which use IBM® CIS for DNS, you must use IBM® DNS for your DNS service.

6.3.1. Limitations

Private clusters on IBM Power® Virtual Server are subject only to the limitations associated with the existing VPC that was used for cluster deployment.

6.4. REQUIREMENTS FOR USING YOUR VPC

You must correctly configure the existing VPC and its subnets before you install the cluster. The installation program does not create a VPC or VPC subnet in this scenario.

The installation program cannot:

- Subdivide network ranges for the cluster to use
- Set route tables for the subnets
- Set VPC options like DHCP

NOTE

The installation program requires that you use the cloud-provided DNS server. Using a custom DNS server is not supported and causes the installation to fail.

6.4.1. VPC validation

The VPC and all of the subnets must be in an existing resource group. The cluster is deployed to this resource group.

As part of the installation, specify the following in the **install-config.yaml** file:

- The name of the resource group
- The name of VPC
- The name of the VPC subnet

To ensure that the subnets that you provide are suitable, the installation program confirms that all of the subnets you specify exists.

NOTE

Subnet IDs are not supported.

6.4.2. Isolation between clusters

If you deploy OpenShift Container Platform to an existing network, the isolation of cluster services is reduced in the following ways:

- ICMP Ingress is allowed to the entire network.
- TCP port 22 Ingress (SSH) is allowed to the entire network.
- Control plane TCP 6443 Ingress (Kubernetes API) is allowed to the entire network.
- Control plane TCP 22623 Ingress (MCS) is allowed to the entire network.

6.5. INTERNET ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, you require access to the internet to install your cluster.

You must have internet access to:

- Access OpenShift Cluster Manager to download the installation program and perform subscription management. If the cluster has internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Access Quay.io to obtain the packages that are required to install your cluster.
- Obtain the packages that are required to perform cluster updates.

IMPORTANT

If your cluster cannot have direct internet access, you can perform a restricted network installation on some types of infrastructure that you provision. During that process, you download the required content and use it to populate a mirror registry with the installation packages. With some installation types, the environment that you install your cluster in will not require internet access. Before you update the cluster, you update the content of the mirror registry.

6.6. GENERATING A KEY PAIR FOR CLUSTER NODE SSH ACCESS

During an OpenShift Container Platform installation, you can provide an SSH public key to the installation program. The key is passed to the Red Hat Enterprise Linux CoreOS (RHCOS) nodes through their Ignition config files and is used to authenticate SSH access to the nodes. The key is added to the **~/.ssh/authorized_keys** list for the **core** user on each node, which enables password-less authentication.

After the key is passed to the nodes, you can use the key pair to SSH in to the RHCOS nodes as the user **core**. To access the nodes through SSH, the private key identity must be managed by SSH for your local user.

If you want to SSH in to your cluster nodes to perform installation debugging or disaster recovery, you must provide the SSH public key during the installation process. The **./openshift-install gather** command also requires the SSH public key to be in place on the cluster nodes.

IMPORTANT

Do not skip this procedure in production environments, where disaster recovery and debugging is required.

NOTE

You must use a local key, not one that you configured with platform-specific approaches.

Procedure

1. If you do not have an existing SSH key pair on your local machine to use for authentication onto your cluster nodes, create one. For example, on a computer that uses a Linux operating system, run the following command:

```
$ ssh-keygen -t ed25519 -N '' -f <path>/<file_name> 1
```

1 Specify the path and file name, such as **~/.ssh/id_ed25519**, of the new SSH key. If you have an existing key pair, ensure your public key is in the your **~/.ssh** directory.

2. View the public SSH key:

```
$ cat <path>/<file_name>.pub
```

For example, run the following to view the **~/.ssh/id_ed25519.pub** public key:

```
$ cat ~/.ssh/id_ed25519.pub
```

3. Add the SSH private key identity to the SSH agent for your local user, if it has not already been added. SSH agent management of the key is required for password-less SSH authentication onto your cluster nodes, or if you want to use the **./openshift-install gather** command.

NOTE

On some distributions, default SSH private key identities such as **~/.ssh/id_rsa** and **~/.ssh/id_dsa** are managed automatically.

a. If the **ssh-agent** process is not already running for your local user, start it as a background task:

```
$ eval "$(ssh-agent -s)"
```

Example output

```
Agent pid 31874
```

4. Add your SSH private key to the **ssh-agent**:

```
$ ssh-add <path>/<file_name> 1
```

1 Specify the path and file name for your SSH private key, such as **~/.ssh/id_ed25519**

Example output

```
Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program.

6.7. OBTAINING THE INSTALLATION PROGRAM

Before you install OpenShift Container Platform, download the installation file on the host you are using for installation.

Prerequisites

- You have a computer that runs Linux or macOS, with 500 MB of local disk space.

Procedure

1. Go to the Cluster Type page on the Red Hat Hybrid Cloud Console. If you have a Red Hat account, log in with your credentials. If you do not, create an account.

   TIP

   You can also download the binaries for a specific OpenShift Container Platform release .

2. Select your infrastructure provider from the **Run it yourself** section of the page.

3. Select your host operating system and architecture from the dropdown menus under **OpenShift Installer** and click **Download Installer**.

4. Place the downloaded file in the directory where you want to store the installation configuration files.

IMPORTANT

- The installation program creates several files on the computer that you use to install your cluster. You must keep the installation program and the files that the installation program creates after you finish installing the cluster. Both of the files are required to delete the cluster.
- Deleting the files created by the installation program does not remove your cluster, even if the cluster failed during installation. To remove your cluster, complete the OpenShift Container Platform uninstallation procedures for your specific cloud provider.

5. Extract the installation program. For example, on a computer that uses a Linux operating system, run the following command:

   ```
   $ tar -xvf openshift-install-linux.tar.gz
   ```

6. Download your installation pull secret from Red Hat OpenShift Cluster Manager . This pull secret allows you to authenticate with the services that are provided by the included authorities, including Quay.io, which serves the container images for OpenShift Container Platform components.

TIP

Alternatively, you can retrieve the installation program from the Red Hat Customer Portal, where you can specify a version of the installation program to download. However, you must have an active subscription to access this page.

6.8. EXPORTING THE API KEY

You must set the API key you created as a global variable; the installation program ingests the variable during startup to set the API key.

Prerequisites

- You have created either a user API key or service ID API key for your IBM Cloud® account.

Procedure

- Export your API key for your account as a global variable:

  ```
  $ export IBMCLOUD_API_KEY=<api_key>
  ```

IMPORTANT

You must set the variable name exactly as specified; the installation program expects the variable name to be present during startup.

6.9. MANUALLY CREATING THE INSTALLATION CONFIGURATION FILE

Installing the cluster requires that you manually create the installation configuration file.

Prerequisites

- You have an SSH public key on your local machine to provide to the installation program. The key will be used for SSH authentication onto your cluster nodes for debugging and disaster recovery.
- You have obtained the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

1. Create an installation directory to store your required installation assets in:

   ```
   $ mkdir <installation_directory>
   ```

   IMPORTANT

   You must create a directory. Some installation assets, like bootstrap X.509 certificates have short expiration intervals, so you must not reuse an installation directory. If you want to reuse individual files from another cluster installation, you can copy them into your directory. However, the file names for the installation assets might change between releases. Use caution when copying installation files from an earlier OpenShift Container Platform version.

2. Customize the sample **install-config.yaml** file template that is provided and save it in the **<installation_directory>**.

   NOTE

   You must name this configuration file **install-config.yaml**.

3. Back up the **install-config.yaml** file so that you can use it to install multiple clusters.

   IMPORTANT

   The **install-config.yaml** file is consumed during the next step of the installation process. You must back it up now.

Additional resources

- Installation configuration parameters for IBM Power® Virtual Server

6.9.1. Minimum resource requirements for cluster installation

Each cluster machine must meet the following minimum requirements:

Table 6.1. Minimum resource requirements

Machine	Operating System	vCPU [1]	Virtual RAM	Storage	Input/Output Per Second (IOPS)[2]
Bootstrap	RHCOS	2	16 GB	100 GB	300
Control plane	RHCOS	2	16 GB	100 GB	300
Compute	RHCOS	2	8 GB	100 GB	300

1. One vCPU is equivalent to one physical core when simultaneous multithreading (SMT), or Hyper-Threading, is not enabled. When enabled, use the following formula to calculate the corresponding ratio: (threads per core × cores) × sockets = vCPUs.

2. OpenShift Container Platform and Kubernetes are sensitive to disk performance, and faster storage is recommended, particularly for etcd on the control plane nodes. Note that on many cloud platforms, storage size and IOPS scale together, so you might need to over-allocate storage volume to obtain sufficient performance.

NOTE

For OpenShift Container Platform version 4.18, RHCOS is based on RHEL version 9.4, which updates the micro-architecture requirements. The following list contains the minimum instruction set architectures (ISA) that each architecture requires:

- x86-64 architecture requires x86-64-v2 ISA
- ARM64 architecture requires ARMv8.0-A ISA
- IBM Power architecture requires Power 9 ISA
- s390x architecture requires z14 ISA

For more information, see Architectures (RHEL documentation).

If an instance type for your platform meets the minimum requirements for cluster machines, it is supported to use in OpenShift Container Platform.

Additional resources

- Optimizing storage

6.9.2. Sample customized install-config.yaml file for IBM Power Virtual Server

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

IMPORTANT

This sample YAML file is provided for reference only. You must obtain your **install-config.yaml** file by using the installation program and modify it.

_

```
apiVersion: v1
baseDomain: example.com
compute: ① ②
- architecture: ppc64le
  hyperthreading: Enabled ③
  name: worker
  platform:
    powervs:
      smtLevel: 8 ④
  replicas: 3
controlPlane: ⑤ ⑥
  architecture: ppc64le
  hyperthreading: Enabled ⑦
  name: master
  platform:
    powervs:
      smtLevel: 8 ⑧
  replicas: 3
metadata:
  creationTimestamp: null
  name: example-private-cluster-name
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14 ⑨
    hostPrefix: 23
  machineNetwork:
  - cidr: 192.168.0.0/24
  networkType: OVNKubernetes ⑩
  serviceNetwork:
  - 172.30.0.0/16
platform:
  powervs:
    userID: ibm-user-id
    powervsResourceGroup: "ibmcloud-resource-group"
    region: powervs-region
    vpcName: name-of-existing-vpc ⑪
    vpcRegion : vpc-region
    zone: powervs-zone
    serviceInstanceGUID: "powervs-region-service-instance-guid"
publish: Internal ⑫
pullSecret: '{"auths": ...}' ⑬
sshKey: ssh-ed25519 AAAA... ⑭
```

① ⑤ If you do not provide these parameters and values, the installation program provides the default value.

② ⑥ The **controlPlane** section is a single mapping, but the compute section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Both sections currently define a single machine pool. Only one control plane pool is used.

③ ⑦ Whether to enable or disable simultaneous multithreading, or **hyperthreading**. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

4 8 The smtLevel specifies the level of SMT to set to the control plane and compute machines. The supported values are 1, 2, 4, 8, **'off'** and **'on'**. The default value is 8. The smtLevel **'off'** sets SMT to off and smtlevel **'on'** sets SMT to the default value 8 on the cluster nodes.

NOTE

When simultaneous multithreading (SMT), or hyperthreading is not enabled, one vCPU is equivalent to one physical core. When enabled, total vCPUs is computed as (Thread(s) per core * Core(s) per socket) * Socket(s). The smtLevel controls the threads per core. Lower SMT levels may require additional assigned cores when deploying the cluster nodes. You can do this by setting the **'processors'** parameter in the **install-config.yaml** file to an appropriate value to meet the requirements for deploying OpenShift Container Platform successfully.

9 The machine CIDR must contain the subnets for the compute machines and control plane machines.

10 The cluster network plugin to install. The default value **OVNKubernetes** is the only supported value.

11 Specify the name of an existing VPC.

12 Specify how to publish the user-facing endpoints of your cluster. Set publish to **Internal** to deploy a private cluster.

13 Required. The installation program prompts you for this value.

14 Provide the **sshKey** value that you use to access the machines in your cluster.

IMPORTANT

If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.

NOTE

For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.

6.9.3. Configuring the cluster-wide proxy during installation

Production environments can deny direct access to the internet and instead have an HTTP or HTTPS proxy available. You can configure a new OpenShift Container Platform cluster to use a proxy by configuring the proxy settings in the **install-config.yaml** file.

Prerequisites

- You have an existing **install-config.yaml** file.

- You reviewed the sites that your cluster requires access to and determined whether any of them need to bypass the proxy. By default, all cluster egress traffic is proxied, including calls to hosting cloud provider APIs. You added sites to the **Proxy** object's **spec.noProxy** field to bypass the proxy if necessary.

NOTE

The **Proxy** object **status.noProxy** field is populated with the values of the **networking.machineNetwork[].cidr**, **networking.clusterNetwork[].cidr**, and **networking.serviceNetwork[]** fields from your installation configuration.

For installations on Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure, and Red Hat OpenStack Platform (RHOSP), the **Proxy** object **status.noProxy** field is also populated with the instance metadata endpoint (**169.254.169.254**).

Procedure

1. Edit your **install-config.yaml** file and add the proxy settings. For example:

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> 1
  httpsProxy: https://<username>:<pswd>@<ip>:<port> 2
  noProxy: example.com 3
additionalTrustBundle: | 4
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
additionalTrustBundlePolicy: <policy_to_add_additionalTrustBundle> 5
```

1 A proxy URL to use for creating HTTP connections outside the cluster. The URL scheme must be **http**.

2 A proxy URL to use for creating HTTPS connections outside the cluster.

3 A comma-separated list of destination domain names, IP addresses, or other network CIDRs to exclude from proxying. Preface a domain with **.** to match subdomains only. For example, **.y.com** matches **x.y.com**, but not **y.com**. Use * to bypass the proxy for all destinations.

4 If provided, the installation program generates a config map that is named **user-ca-bundle** in the **openshift-config** namespace that contains one or more additional CA certificates that are required for proxying HTTPS connections. The Cluster Network Operator then creates a **trusted-ca-bundle** config map that merges these contents with the Red Hat Enterprise Linux CoreOS (RHCOS) trust bundle, and this config map is referenced in the **trustedCA** field of the **Proxy** object. The **additionalTrustBundle** field is required unless the proxy's identity certificate is signed by an authority from the RHCOS trust bundle.

5 Optional: The policy to determine the configuration of the **Proxy** object to reference the **user-ca-bundle** config map in the **trustedCA** field. The allowed values are **Proxyonly** and **Always**. Use **Proxyonly** to reference the **user-ca-bundle** config map only when **http/https** proxy is configured. Use **Always** to always reference the **user-ca-bundle** config map. The default value is **Proxyonly**.

> NOTE
>
> The installation program does not support the proxy **readinessEndpoints** field.

> NOTE
>
> If the installer times out, restart and then complete the deployment by using the **wait-for** command of the installer. For example:
>
> ```
> $ ./openshift-install wait-for install-complete --log-level debug
> ```

2. Save the file and reference it when installing OpenShift Container Platform.

The installation program creates a cluster-wide proxy that is named **cluster** that uses the proxy settings in the provided **install-config.yaml** file. If no proxy settings are provided, a **cluster** **Proxy** object is still created, but it will have a nil **spec**.

> NOTE
>
> Only the **Proxy** object named **cluster** is supported, and no additional proxies can be created.

6.10. MANUALLY CREATING IAM

Installing the cluster requires that the Cloud Credential Operator (CCO) operate in manual mode. While the installation program configures the CCO for manual mode, you must specify the identity and access management secrets for you cloud provider.

You can use the Cloud Credential Operator (CCO) utility (**ccoctl**) to create the required IBM Cloud® resources.

Prerequisites

- You have configured the **ccoctl** binary.
- You have an existing **install-config.yaml** file.

Procedure

1. Edit the **install-config.yaml** configuration file so that it contains the **credentialsMode** parameter set to **Manual**.

   Example install-config.yaml configuration file

   ```
   apiVersion: v1
   baseDomain: cluster1.example.com
   credentialsMode: Manual 1
   compute:
   - architecture: ppc64le
     hyperthreading: Enabled
   ```

   1 This line is added to set the **credentialsMode** parameter to **Manual**.

2. To generate the manifests, run the following command from the directory that contains the installation program:

   ```
   $ ./openshift-install create manifests --dir <installation_directory>
   ```

3. From the directory that contains the installation program, set a **$RELEASE_IMAGE** variable with the release image from your installation file by running the following command:

   ```
   $ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
   ```

4. Extract the list of **CredentialsRequest** custom resources (CRs) from the OpenShift Container Platform release image by running the following command:

   ```
   $ oc adm release extract \
     --from=$RELEASE_IMAGE \
     --credentials-requests \
     --included \ 1
     --install-config=<path_to_directory_with_installation_configuration>/install-config.yaml \ 2
     --to=<path_to_directory_for_credentials_requests> 3
   ```

   1 The **--included** parameter includes only the manifests that your specific cluster configuration requires.

   2 Specify the location of the **install-config.yaml** file.

   3 Specify the path to the directory where you want to store the **CredentialsRequest** objects. If the specified directory does not exist, this command creates it.

   This command creates a YAML file for each **CredentialsRequest** object.

   Sample CredentialsRequest object

   ```
   apiVersion: cloudcredential.openshift.io/v1
   kind: CredentialsRequest
   metadata:
     labels:
       controller-tools.k8s.io: "1.0"
     name: openshift-image-registry-ibmcos
     namespace: openshift-cloud-credential-operator
   spec:
     secretRef:
       name: installer-cloud-credentials
       namespace: openshift-image-registry
     providerSpec:
       apiVersion: cloudcredential.openshift.io/v1
       kind: IBMCloudProviderSpec
       policies:
       - attributes:
         - name: serviceName
           value: cloud-object-storage
         roles:
         - crn:v1:bluemix:public:iam::::role:Viewer
         - crn:v1:bluemix:public:iam::::role:Operator
         - crn:v1:bluemix:public:iam::::role:Editor
   ```

```
      - crn:v1:bluemix:public:iam::::serviceRole:Reader
      - crn:v1:bluemix:public:iam::::serviceRole:Writer
    - attributes:
      - name: resourceType
        value: resource-group
      roles:
      - crn:v1:bluemix:public:iam::::role:Viewer
```

5. Create the service ID for each credential request, assign the policies defined, create an API key, and generate the secret:

```
$ ccoctl ibmcloud create-service-id \
  --credentials-requests-dir=<path_to_credential_requests_directory> \ 1
  --name=<cluster_name> \ 2
  --output-dir=<installation_directory> \ 3
  --resource-group-name=<resource_group_name> 4
```

1 Specify the directory containing the files for the component **CredentialsRequest** objects.

2 Specify the name of the OpenShift Container Platform cluster.

3 Optional: Specify the directory in which you want the **ccoctl** utility to create objects. By default, the utility creates objects in the directory in which the commands are run.

4 Optional: Specify the name of the resource group used for scoping the access policies.

NOTE

If your cluster uses Technology Preview features that are enabled by the **TechPreviewNoUpgrade** feature set, you must include the **--enable-tech-preview** parameter.

If an incorrect resource group name is provided, the installation fails during the bootstrap phase. To find the correct resource group name, run the following command:

```
$ grep resourceGroup <installation_directory>/manifests/cluster-infrastructure-02-config.yml
```

Verification

- Ensure that the appropriate secrets were generated in your cluster's **manifests** directory.

6.11. DEPLOYING THE CLUSTER

You can install OpenShift Container Platform on a compatible cloud platform.

IMPORTANT

You can run the **create cluster** command of the installation program only once, during initial installation.

Prerequisites

- You have configured an account with the cloud platform that hosts your cluster.
- You have the OpenShift Container Platform installation program and the pull secret for your cluster.
- You have verified that the cloud provider account on your host has the correct permissions to deploy the cluster. An account with incorrect permissions causes the installation process to fail with an error message that displays the missing permissions.

Procedure

- Change to the directory that contains the installation program and initialize the cluster deployment:

```
$ ./openshift-install create cluster --dir <installation_directory> \ 1
    --log-level=info 2
```

1 For **<installation_directory>**, specify the location of your customized **./install-config.yaml** file.

2 To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.

Verification

When the cluster deployment completes successfully:

- The terminal displays directions for accessing your cluster, including a link to the web console and credentials for the **kubeadmin** user.
- Credential information also outputs to **<installation_directory>/.openshift_install.log**.

IMPORTANT

Do not delete the installation program or the files that the installation program creates. Both are required to delete the cluster.

Example output

```
...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export
KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-
console.apps.mycluster.example.com
INFO Login to the console with user: "kubeadmin", and password: "password"
INFO Time elapsed: 36m22s
```

IMPORTANT

- The Ignition config files that the installation program generates contain certificates that expire after 24 hours, which are then renewed at that time. If the cluster is shut down before renewing the certificates and the cluster is later restarted after the 24 hours have elapsed, the cluster automatically recovers the expired certificates. The exception is that you must manually approve the pending **node-bootstrapper** certificate signing requests (CSRs) to recover kubelet certificates. See the documentation for *Recovering from expired control plane certificates* for more information.

- It is recommended that you use Ignition config files within 12 hours after they are generated because the 24-hour certificate rotates from 16 to 22 hours after the cluster is installed. By using the Ignition config files within 12 hours, you can avoid installation failure if the certificate update runs during installation.

6.12. INSTALLING THE OPENSHIFT CLI

You can install the OpenShift CLI (**oc**) to interact with OpenShift Container Platform from a command-line interface. You can install **oc** on Linux, Windows, or macOS.

IMPORTANT

If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.18. Download and install the new version of **oc**.

Installing the OpenShift CLI on Linux

You can install the OpenShift CLI (**oc**) binary on Linux by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.

2. Select the architecture from the **Product Variant** drop-down list.

3. Select the appropriate version from the **Version** drop-down list.

4. Click **Download Now** next to the **OpenShift v4.18 Linux Clients** entry and save the file.

5. Unpack the archive:

```
$ tar xvf <file>
```

6. Place the **oc** binary in a directory that is on your **PATH**.
To check your **PATH**, execute the following command:

```
$ echo $PATH
```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

```
$ oc <command>
```

Installing the OpenShift CLI on Windows

You can install the OpenShift CLI (**oc**) binary on Windows by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the appropriate version from the **Version** drop-down list.
3. Click **Download Now** next to the **OpenShift v4.18 Windows Client** entry and save the file.
4. Unzip the archive with a ZIP program.
5. Move the **oc** binary to a directory that is on your **PATH**.
   To check your **PATH**, open the command prompt and execute the following command:

   ```
   C:\> path
   ```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

  ```
  C:\> oc <command>
  ```

Installing the OpenShift CLI on macOS

You can install the OpenShift CLI (**oc**) binary on macOS by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the appropriate version from the **Version** drop-down list.
3. Click **Download Now** next to the **OpenShift v4.18 macOS Clients** entry and save the file.

   NOTE

   For macOS arm64, choose the **OpenShift v4.18 macOS arm64 Client** entry.

4. Unpack and unzip the archive.
5. Move the **oc** binary to a directory on your PATH.
   To check your **PATH**, open a terminal and execute the following command:

   ```
   $ echo $PATH
   ```

Verification

- Verify your installation by using an **oc** command:

```
$ oc <command>
```

6.13. LOGGING IN TO THE CLUSTER BY USING THE CLI

You can log in to your cluster as a default system user by exporting the cluster **kubeconfig** file. The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server. The file is specific to a cluster and is created during OpenShift Container Platform installation.

Prerequisites

- You deployed an OpenShift Container Platform cluster.
- You installed the **oc** CLI.

Procedure

1. Export the **kubeadmin** credentials:

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

1 For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

2. Verify you can run **oc** commands successfully using the exported configuration:

```
$ oc whoami
```

Example output

```
system:admin
```

Additional resources

- Accessing the web console

6.14. TELEMETRY ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, the Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, requires internet access. If your cluster is connected to the internet, Telemetry runs automatically, and your cluster is registered to OpenShift Cluster Manager.

After you confirm that your OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually by using OpenShift Cluster Manager, use subscription watch to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

Additional resources

- About remote health monitoring

6.15. NEXT STEPS

- Customize your cluster
- Optional: Opt out of remote health reporting

CHAPTER 7. INSTALLING A CLUSTER ON IBM POWER VIRTUAL SERVER IN A DISCONNECTED ENVIRONMENT

In OpenShift Container Platform 4.18, you can install a cluster on IBM Cloud® in a restricted network by creating an internal mirror of the installation release content on an existing Virtual Private Cloud (VPC) on IBM Cloud®.

7.1. PREREQUISITES

- You reviewed details about the OpenShift Container Platform installation and update processes.
- You read the documentation on selecting a cluster installation method and preparing it for users.
- You configured an IBM Cloud® account to host the cluster.
- You mirrored the images for a disconnected installation to your registry and obtained the **imageContentSources** data for your version of OpenShift Container Platform.

  IMPORTANT

  Because the installation media is on the mirror host, you can use that computer to complete all installation steps.

- You have an existing VPC in IBM Cloud®. When installing a cluster in a restricted network, you cannot use the installer-provisioned VPC. You must use a user-provisioned VPC that satisfies one of the following requirements:
  - Contains the mirror registry
  - Has firewall rules or a peering connection to access the mirror registry hosted elsewhere
- If you use a firewall, you configured it to allow the sites that your cluster requires access to.
- You configured the **ccoctl** utility before you installed the cluster. For more information, see Configuring the Cloud Credential Operator utility.

7.2. ABOUT INSTALLATIONS IN RESTRICTED NETWORKS

In OpenShift Container Platform 4.18, you can perform an installation that does not require an active connection to the internet to obtain software components. Restricted network installations can be completed using installer-provisioned infrastructure or user-provisioned infrastructure, depending on the cloud platform to which you are installing the cluster.

NOTE

For installer-provisioned infrastructure in OpenShift Container Platform 4.18, you need to deploy your restricted network cluster in OpenShift Container Platform 4.16 and upgrade it to OpenShift Container Platform 4.18.

If you choose to perform a restricted network installation on a cloud platform, you still require access to its cloud APIs. Some cloud functions, like Amazon Web Service’s Route 53 DNS and IAM services, require internet access. Depending on your network, you might require less internet access for an installation on bare metal hardware, Nutanix, or on VMware vSphere.

To complete a restricted network installation, you must create a registry that mirrors the contents of the OpenShift image registry and contains the installation media. You can create this registry on a mirror host, which can access both the internet and your closed network, or by using other methods that meet your restrictions.

7.2.1. Additional limits

Clusters in restricted networks have the following additional limitations and restrictions:

- The **ClusterVersion** status includes an **Unable to retrieve available updates** error.
- By default, you cannot use the contents of the Developer Catalog because you cannot access the required image stream tags.

7.3. ABOUT USING A CUSTOM VPC

In OpenShift Container Platform 4.18, you can deploy a cluster into the subnets of an existing IBM® Virtual Private Cloud (VPC).

7.3.1. Requirements for using your VPC

You must correctly configure the existing VPC and its subnets before you install the cluster. The installation program does not create a VPC or VPC subnet in this scenario.

The installation program cannot:

- Subdivide network ranges for the cluster to use
- Set route tables for the subnets
- Set VPC options like DHCP

NOTE

The installation program requires that you use the cloud-provided DNS server. Using a custom DNS server is not supported and causes the installation to fail.

7.3.2. VPC validation

The VPC and all of the subnets must be in an existing resource group. The cluster is deployed to this resource group.

As part of the installation, specify the following in the **install-config.yaml** file:

- The name of the resource group
- The name of VPC
- The name of the VPC subnet

To ensure that the subnets that you provide are suitable, the installation program confirms that all of the subnets you specify exists.

NOTE

Subnet IDs are not supported.

7.3.3. Isolation between clusters

If you deploy OpenShift Container Platform to an existing network, the isolation of cluster services is reduced in the following ways:

- ICMP Ingress is allowed to the entire network.
- TCP port 22 Ingress (SSH) is allowed to the entire network.
- Control plane TCP 6443 Ingress (Kubernetes API) is allowed to the entire network.
- Control plane TCP 22623 Ingress (MCS) is allowed to the entire network.

7.4. INTERNET ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, you require access to the internet to obtain the images that are necessary to install your cluster.

You must have internet access to:

- Access OpenShift Cluster Manager to download the installation program and perform subscription management. If the cluster has internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Access Quay.io to obtain the packages that are required to install your cluster.
- Obtain the packages that are required to perform cluster updates.

7.5. GENERATING A KEY PAIR FOR CLUSTER NODE SSH ACCESS

During an OpenShift Container Platform installation, you can provide an SSH public key to the installation program. The key is passed to the Red Hat Enterprise Linux CoreOS (RHCOS) nodes through their Ignition config files and is used to authenticate SSH access to the nodes. The key is added to the **~/.ssh/authorized_keys** list for the **core** user on each node, which enables password-less authentication.

After the key is passed to the nodes, you can use the key pair to SSH in to the RHCOS nodes as the user **core**. To access the nodes through SSH, the private key identity must be managed by SSH for your local user.

If you want to SSH in to your cluster nodes to perform installation debugging or disaster recovery, you must provide the SSH public key during the installation process. The **./openshift-install gather** command also requires the SSH public key to be in place on the cluster nodes.

IMPORTANT

Do not skip this procedure in production environments, where disaster recovery and debugging is required.

NOTE

You must use a local key, not one that you configured with platform-specific approaches.

Procedure

1. If you do not have an existing SSH key pair on your local machine to use for authentication onto your cluster nodes, create one. For example, on a computer that uses a Linux operating system, run the following command:

   ```
   $ ssh-keygen -t ed25519 -N '' -f <path>/<file_name> 1
   ```

   1 Specify the path and file name, such as **~/.ssh/id_ed25519**, of the new SSH key. If you have an existing key pair, ensure your public key is in the your **~/.ssh** directory.

2. View the public SSH key:

   ```
   $ cat <path>/<file_name>.pub
   ```

   For example, run the following to view the **~/.ssh/id_ed25519.pub** public key:

   ```
   $ cat ~/.ssh/id_ed25519.pub
   ```

3. Add the SSH private key identity to the SSH agent for your local user, if it has not already been added. SSH agent management of the key is required for password-less SSH authentication onto your cluster nodes, or if you want to use the **./openshift-install gather** command.

   NOTE

   On some distributions, default SSH private key identities such as **~/.ssh/id_rsa** and **~/.ssh/id_dsa** are managed automatically.

   a. If the **ssh-agent** process is not already running for your local user, start it as a background task:

      ```
      $ eval "$(ssh-agent -s)"
      ```

      #### Example output

      ```
      Agent pid 31874
      ```

4. Add your SSH private key to the **ssh-agent**:

   ```
   $ ssh-add <path>/<file_name> 1
   ```

   1 Specify the path and file name for your SSH private key, such as **~/.ssh/id_ed25519**

   #### Example output

   ```
   Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
   ```

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program.

7.6. EXPORTING THE API KEY

You must set the API key you created as a global variable; the installation program ingests the variable during startup to set the API key.

Prerequisites

- You have created either a user API key or service ID API key for your IBM Cloud® account.

Procedure

- Export your API key for your account as a global variable:

  ```
  $ export IBMCLOUD_API_KEY=<api_key>
  ```

IMPORTANT

You must set the variable name exactly as specified; the installation program expects the variable name to be present during startup.

7.7. CREATING THE INSTALLATION CONFIGURATION FILE

You can customize the OpenShift Container Platform cluster you install on

Prerequisites

- You have the OpenShift Container Platform installation program and the pull secret for your cluster. For a restricted network installation, these files are on your mirror host.
- You have the **imageContentSources** values that were generated during mirror registry creation.
- You have obtained the contents of the certificate for your mirror registry.
- You have retrieved a Red Hat Enterprise Linux CoreOS (RHCOS) image and uploaded it to an accessible location.

Procedure

1. Create the **install-config.yaml** file.
   a. Change to the directory that contains the installation program and run the following command:

      ```
      $ ./openshift-install create install-config --dir <installation_directory> 1
      ```

      1 For **<installation_directory>**, specify the directory name to store the files that the installation program creates.

When specifying the directory:

- Use an empty directory. Some installation assets, such as bootstrap X.509 certificates, have short expiration intervals, therefore you must not reuse an installation directory. If you want to reuse individual files from another cluster installation, you can copy them into your directory. However, the file names for the installation assets might change between releases. Use caution when copying installation files from an earlier OpenShift Container Platform version.

NOTE

Always delete the **~/.powervs** directory to avoid reusing a stale configuration. Run the following command:

```
$ rm -rf ~/.powervs
```

b. At the prompts, provide the configuration details for your cloud:

i. Optional: Select an SSH key to use to access your cluster machines.

NOTE

For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.

ii. Select **powervs** as the platform to target.

iii. Select the region to deploy the cluster to.

iv. Select the zone to deploy the cluster to.

v. Select the base domain to deploy the cluster to. The base domain corresponds to the public DNS zone that you created for your cluster.

vi. Enter a descriptive name for your cluster.

2. Edit the **install-config.yaml** file to give the additional information that is required for an installation in a restricted network.

a. Update the **pullSecret** value to contain the authentication information for your registry:

```
pullSecret: '{"auths":{"<mirror_host_name>:5000": {"auth": "<credentials>","email": "you@example.com"}}}'
```

For **<mirror_host_name>**, specify the registry domain name that you specified in the certificate for your mirror registry, and for **<credentials>**, specify the base64-encoded user name and password for your mirror registry.

b. Add the **additionalTrustBundle** parameter and value.

```
additionalTrustBundle: |
  -----BEGIN CERTIFICATE-----
```

```
ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
  -----END CERTIFICATE-----
```

The value must be the contents of the certificate file that you used for your mirror registry. The certificate file can be an existing, trusted certificate authority, or the self-signed certificate that you generated for the mirror registry.

c. Define the network and subnets for the VPC to install the cluster in under the parent **platform.powervs** field:

```
vpcName: <existing_vpc>
vpcSubnets: <vpcSubnet>
```

For **platform.powervs.vpcName**, specify the name for the existing IBM Cloud®. For **platform.powervs.vpcSubnets**, specify the existing subnets.

d. Add the image content resources, which resemble the following YAML excerpt:

```
imageContentSources:
- mirrors:
  - <mirror_host_name>:5000/<repo_name>/release
  source: quay.io/openshift-release-dev/ocp-release
- mirrors:
  - <mirror_host_name>:5000/<repo_name>/release
  source: registry.redhat.io/ocp/release
```

For these values, use the **imageContentSources** that you recorded during mirror registry creation.

e. Optional: Set the publishing strategy to **Internal**:

```
publish: Internal
```

By setting this option, you create an internal Ingress Controller and a private load balancer.

3. Make any other modifications to the **install-config.yaml** file that you require.
   For more information about the parameters, see "Installation configuration parameters".

4. Back up the **install-config.yaml** file so that you can use it to install multiple clusters.

IMPORTANT

The **install-config.yaml** file is consumed during the installation process. If you want to reuse the file, you must back it up now.

Additional resources

- Installation configuration parameters for IBM Power® Virtual Server

7.7.1. Minimum resource requirements for cluster installation

Each cluster machine must meet the following minimum requirements:

Table 7.1. Minimum resource requirements

Machine	Operating System	vCPU [1]	Virtual RAM	Storage	Input/Output Per Second (IOPS)[2]
Bootstrap	RHCOS	2	16 GB	100 GB	300
Control plane	RHCOS	2	16 GB	100 GB	300
Compute	RHCOS	2	8 GB	100 GB	300

1. One vCPU is equivalent to one physical core when simultaneous multithreading (SMT), or Hyper-Threading, is not enabled. When enabled, use the following formula to calculate the corresponding ratio: (threads per core × cores) × sockets = vCPUs.

2. OpenShift Container Platform and Kubernetes are sensitive to disk performance, and faster storage is recommended, particularly for etcd on the control plane nodes. Note that on many cloud platforms, storage size and IOPS scale together, so you might need to over-allocate storage volume to obtain sufficient performance.

NOTE

For OpenShift Container Platform version 4.18, RHCOS is based on RHEL version 9.4, which updates the micro-architecture requirements. The following list contains the minimum instruction set architectures (ISA) that each architecture requires:

- x86-64 architecture requires x86-64-v2 ISA
- ARM64 architecture requires ARMv8.0-A ISA
- IBM Power architecture requires Power 9 ISA
- s390x architecture requires z14 ISA

For more information, see Architectures (RHEL documentation).

If an instance type for your platform meets the minimum requirements for cluster machines, it is supported to use in OpenShift Container Platform.

Additional resources

- Optimizing storage

7.7.2. Sample customized install-config.yaml file for IBM Power Virtual Server

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

IMPORTANT

This sample YAML file is provided for reference only. You must obtain your **install-config.yaml** file by using the installation program and modify it.

–

```
apiVersion: v1
baseDomain: example.com 1
controlPlane: 2 3
  hyperthreading: Enabled 4
  name: master
  platform:
    powervs:
      smtLevel: 8 5
  replicas: 3
compute: 6 7
- hyperthreading: Enabled 8
  name: worker
  platform:
    powervs:
      smtLevel: 8 9
    ibmcloud: {}
  replicas: 3
metadata:
  name: example-restricted-cluster-name 10
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14 11
    hostPrefix: 23
  machineNetwork:
  - cidr: 10.0.0.0/16 12
  networkType: OVNKubernetes 13
  serviceNetwork:
  - 192.168.0.0/24
platform:
  powervs:
    userid: ibm-user-id
    powervsResourceGroup: "ibmcloud-resource-group" 14
    region: "powervs-region"
    vpcRegion: "vpc-region"
    vpcName: name-of-existing-vpc 15
    vpcSubnets: 16
       - name-of-existing-vpc-subnet
    zone: "powervs-zone"
    serviceInstanceID: "service-instance-id"
publish: Internal
credentialsMode: Manual
pullSecret: '{"auths":{"<local_registry>": {"auth": "<credentials>","email": "you@example.com"}}}' 17
sshKey: ssh-ed25519 AAAA... 18
additionalTrustBundle: | 19
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
imageContentSources: 20
- mirrors:
  - <local_registry>/<local_repository_name>/release
  source: quay.io/openshift-release-dev/ocp-release
```

```
- mirrors:
  - <local_registry>/<local_repository_name>/release
  source: quay.io/openshift-release-dev/ocp-v4.0-art-dev
```

1 10 Required.

2 6 If you do not provide these parameters and values, the installation program provides the default value.

3 7 The **controlPlane** section is a single mapping, but the **compute** section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Only one control plane pool is used.

4 8 Enables or disables simultaneous multithreading, also known as Hyper-Threading. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

IMPORTANT

If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance. Use larger machine types, such as **n1-standard-8**, for your machines if you disable simultaneous multithreading.

5 9 The smtLevel specifies the level of SMT to set to the control plane and compute machines. The supported values are 1, 2, 4, 8, **'off'** and **'on'**. The default value is 8. The smtLevel **'off'** sets SMT to off and smtlevel **'on'** sets SMT to the default value 8 on the cluster nodes.

NOTE

When simultaneous multithreading (SMT), or hyperthreading is not enabled, one vCPU is equivalent to one physical core. When enabled, total vCPUs is computed as (Thread(s) per core * Core(s) per socket) * Socket(s). The smtLevel controls the threads per core. Lower SMT levels may require additional assigned cores when deploying the cluster nodes. You can do this by setting the **'processors'** parameter in the **install-config.yaml** file to an appropriate value to meet the requirements for deploying OpenShift Container Platform successfully.

11 The machine CIDR must contain the subnets for the compute machines and control plane machines.

12 The CIDR must contain the subnets defined in **platform.ibmcloud.controlPlaneSubnets** and **platform.ibmcloud.computeSubnets**.

13 The cluster network plugin to install. The default value **OVNKubernetes** is the only supported value.

14 The name of an existing resource group. The existing VPC and subnets should be in this resource group. The cluster is deployed to this resource group.

15 Specify the name of an existing VPC.

16

Specify the name of the existing VPC subnet. The subnets must belong to the VPC that you specified. Specify a subnet for each availability zone in the region.

17 For **<local_registry>**, specify the registry domain name, and optionally the port, that your mirror registry uses to serve content. For example, registry.example.com or registry.example.com:5000. For **<credentials>**, specify the base64-encoded user name and password for your mirror registry.

18 You can optionally provide the **sshKey** value that you use to access the machines in your cluster.

19 Provide the contents of the certificate file that you used for your mirror registry.

20 Provide the **imageContentSources** section from the output of the command to mirror the repository.

> NOTE
>
> For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.

7.7.3. Configuring the cluster-wide proxy during installation

Production environments can deny direct access to the internet and instead have an HTTP or HTTPS proxy available. You can configure a new OpenShift Container Platform cluster to use a proxy by configuring the proxy settings in the **install-config.yaml** file.

Prerequisites

- You have an existing **install-config.yaml** file.
- You reviewed the sites that your cluster requires access to and determined whether any of them need to bypass the proxy. By default, all cluster egress traffic is proxied, including calls to hosting cloud provider APIs. You added sites to the **Proxy** object's **spec.noProxy** field to bypass the proxy if necessary.

> NOTE
>
> The **Proxy** object **status.noProxy** field is populated with the values of the **networking.machineNetwork[].cidr**, **networking.clusterNetwork[].cidr**, and **networking.serviceNetwork[]** fields from your installation configuration.
>
> For installations on Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure, and Red Hat OpenStack Platform (RHOSP), the **Proxy** object **status.noProxy** field is also populated with the instance metadata endpoint (**169.254.169.254**).

Procedure

1. Edit your **install-config.yaml** file and add the proxy settings. For example:

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> 1
  httpsProxy: https://<username>:<pswd>@<ip>:<port> 2
  noProxy: example.com 3
additionalTrustBundle: | 4
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
additionalTrustBundlePolicy: <policy_to_add_additionalTrustBundle> 5
```

1. A proxy URL to use for creating HTTP connections outside the cluster. The URL scheme must be **http**.

2. A proxy URL to use for creating HTTPS connections outside the cluster.

3. A comma-separated list of destination domain names, IP addresses, or other network CIDRs to exclude from proxying. Preface a domain with **.** to match subdomains only. For example, **.y.com** matches **x.y.com**, but not **y.com**. Use * to bypass the proxy for all destinations.

4. If provided, the installation program generates a config map that is named **user-ca-bundle** in the **openshift-config** namespace that contains one or more additional CA certificates that are required for proxying HTTPS connections. The Cluster Network Operator then creates a **trusted-ca-bundle** config map that merges these contents with the Red Hat Enterprise Linux CoreOS (RHCOS) trust bundle, and this config map is referenced in the **trustedCA** field of the **Proxy** object. The **additionalTrustBundle** field is required unless the proxy's identity certificate is signed by an authority from the RHCOS trust bundle.

5. Optional: The policy to determine the configuration of the **Proxy** object to reference the **user-ca-bundle** config map in the **trustedCA** field. The allowed values are **Proxyonly** and **Always**. Use **Proxyonly** to reference the **user-ca-bundle** config map only when **http/https** proxy is configured. Use **Always** to always reference the **user-ca-bundle** config map. The default value is **Proxyonly**.

NOTE

The installation program does not support the proxy **readinessEndpoints** field.

NOTE

If the installer times out, restart and then complete the deployment by using the **wait-for** command of the installer. For example:

```
$ ./openshift-install wait-for install-complete --log-level debug
```

2. Save the file and reference it when installing OpenShift Container Platform.

The installation program creates a cluster-wide proxy that is named **cluster** that uses the proxy settings in the provided **install-config.yaml** file. If no proxy settings are provided, a **cluster Proxy** object is still created, but it will have a nil **spec**.

NOTE

Only the **Proxy** object named **cluster** is supported, and no additional proxies can be created.

7.8. MANUALLY CREATING IAM

Installing the cluster requires that the Cloud Credential Operator (CCO) operate in manual mode. While the installation program configures the CCO for manual mode, you must specify the identity and access management secrets for you cloud provider.

You can use the Cloud Credential Operator (CCO) utility (**ccoctl**) to create the required IBM Cloud® resources.

Prerequisites

- You have configured the **ccoctl** binary.
- You have an existing **install-config.yaml** file.

Procedure

1. Edit the **install-config.yaml** configuration file so that it contains the **credentialsMode** parameter set to **Manual**.

   Example install-config.yaml configuration file

   ```
   apiVersion: v1
   baseDomain: cluster1.example.com
   credentialsMode: Manual 1
   compute:
   - architecture: ppc64le
     hyperthreading: Enabled
   ```

   1 This line is added to set the **credentialsMode** parameter to **Manual**.

2. To generate the manifests, run the following command from the directory that contains the installation program:

   ```
   $ ./openshift-install create manifests --dir <installation_directory>
   ```

3. From the directory that contains the installation program, set a **$RELEASE_IMAGE** variable with the release image from your installation file by running the following command:

   ```
   $ RELEASE_IMAGE=$(./openshift-install version | awk '/release image/ {print $3}')
   ```

4. Extract the list of **CredentialsRequest** custom resources (CRs) from the OpenShift Container Platform release image by running the following command:

   ```
   $ oc adm release extract \
     --from=$RELEASE_IMAGE \
     --credentials-requests \
     --included \ 1
     --install-config=<path_to_directory_with_installation_configuration>/install-config.yaml \ 2
     --to=<path_to_directory_for_credentials_requests> 3
   ```

1. The **--included** parameter includes only the manifests that your specific cluster configuration requires.

2. Specify the location of the **install-config.yaml** file.

3. Specify the path to the directory where you want to store the **CredentialsRequest** objects. If the specified directory does not exist, this command creates it.

This command creates a YAML file for each **CredentialsRequest** object.

Sample CredentialsRequest object

```
apiVersion: cloudcredential.openshift.io/v1
kind: CredentialsRequest
metadata:
  labels:
    controller-tools.k8s.io: "1.0"
  name: openshift-image-registry-ibmcos
  namespace: openshift-cloud-credential-operator
spec:
  secretRef:
    name: installer-cloud-credentials
    namespace: openshift-image-registry
  providerSpec:
    apiVersion: cloudcredential.openshift.io/v1
    kind: IBMCloudProviderSpec
    policies:
    - attributes:
      - name: serviceName
        value: cloud-object-storage
      roles:
      - crn:v1:bluemix:public:iam::::role:Viewer
      - crn:v1:bluemix:public:iam::::role:Operator
      - crn:v1:bluemix:public:iam::::role:Editor
      - crn:v1:bluemix:public:iam::::serviceRole:Reader
      - crn:v1:bluemix:public:iam::::serviceRole:Writer
    - attributes:
      - name: resourceType
        value: resource-group
      roles:
      - crn:v1:bluemix:public:iam::::role:Viewer
```

5. Create the service ID for each credential request, assign the policies defined, create an API key, and generate the secret:

```
$ ccoctl ibmcloud create-service-id \
 --credentials-requests-dir=<path_to_credential_requests_directory> \ 1
 --name=<cluster_name> \ 2
 --output-dir=<installation_directory> \ 3
 --resource-group-name=<resource_group_name> 4
```

1. Specify the directory containing the files for the component **CredentialsRequest** objects.

2. Specify the name of the OpenShift Container Platform cluster.

3 Optional: Specify the directory in which you want the **ccoctl** utility to create objects. By default, the utility creates objects in the directory in which the commands are run.

4 Optional: Specify the name of the resource group used for scoping the access policies.

NOTE

If your cluster uses Technology Preview features that are enabled by the **TechPreviewNoUpgrade** feature set, you must include the **--enable-tech-preview** parameter.

If an incorrect resource group name is provided, the installation fails during the bootstrap phase. To find the correct resource group name, run the following command:

```
$ grep resourceGroup <installation_directory>/manifests/cluster-infrastructure-02-config.yml
```

Verification

- Ensure that the appropriate secrets were generated in your cluster's **manifests** directory.

7.9. DEPLOYING THE CLUSTER

You can install OpenShift Container Platform on a compatible cloud platform.

IMPORTANT

You can run the **create cluster** command of the installation program only once, during initial installation.

Prerequisites

- You have configured an account with the cloud platform that hosts your cluster.
- You have the OpenShift Container Platform installation program and the pull secret for your cluster.
- You have verified that the cloud provider account on your host has the correct permissions to deploy the cluster. An account with incorrect permissions causes the installation process to fail with an error message that displays the missing permissions.

Procedure

- Change to the directory that contains the installation program and initialize the cluster deployment:

```
$ ./openshift-install create cluster --dir <installation_directory> \ 1
  --log-level=info 2
```

1 For **<installation_directory>**, specify the location of your customized **./install-config.yaml** file.

2 To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.

Verification

When the cluster deployment completes successfully:

- The terminal displays directions for accessing your cluster, including a link to the web console and credentials for the **kubeadmin** user.
- Credential information also outputs to **<installation_directory>/.openshift_install.log**.

IMPORTANT

Do not delete the installation program or the files that the installation program creates. Both are required to delete the cluster.

Example output

```
...
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export
KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-
console.apps.mycluster.example.com
INFO Login to the console with user: "kubeadmin", and password: "password"
INFO Time elapsed: 36m22s
```

IMPORTANT

- The Ignition config files that the installation program generates contain certificates that expire after 24 hours, which are then renewed at that time. If the cluster is shut down before renewing the certificates and the cluster is later restarted after the 24 hours have elapsed, the cluster automatically recovers the expired certificates. The exception is that you must manually approve the pending **node-bootstrapper** certificate signing requests (CSRs) to recover kubelet certificates. See the documentation for *Recovering from expired control plane certificates* for more information.
- It is recommended that you use Ignition config files within 12 hours after they are generated because the 24-hour certificate rotates from 16 to 22 hours after the cluster is installed. By using the Ignition config files within 12 hours, you can avoid installation failure if the certificate update runs during installation.

7.10. INSTALLING THE OPENSHIFT CLI

You can install the OpenShift CLI (**oc**) to interact with OpenShift Container Platform from a command-line interface. You can install **oc** on Linux, Windows, or macOS.

IMPORTANT

If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.18. Download and install the new version of **oc**.

Installing the OpenShift CLI on Linux

You can install the OpenShift CLI (**oc**) binary on Linux by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the architecture from the **Product Variant** drop-down list.
3. Select the appropriate version from the **Version** drop-down list.
4. Click **Download Now** next to the **OpenShift v4.18 Linux Clients** entry and save the file.
5. Unpack the archive:

```
$ tar xvf <file>
```

6. Place the **oc** binary in a directory that is on your **PATH**.
   To check your **PATH**, execute the following command:

```
$ echo $PATH
```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

```
$ oc <command>
```

Installing the OpenShift CLI on Windows

You can install the OpenShift CLI (**oc**) binary on Windows by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the appropriate version from the **Version** drop-down list.
3. Click **Download Now** next to the **OpenShift v4.18 Windows Client** entry and save the file.
4. Unzip the archive with a ZIP program.
5. Move the **oc** binary to a directory that is on your **PATH**.
   To check your **PATH**, open the command prompt and execute the following command:

```
C:\> path
```

Verification

- After you install the OpenShift CLI, it is available using the **oc** command:

```
C:\> oc <command>
```

Installing the OpenShift CLI on macOS

You can install the OpenShift CLI (**oc**) binary on macOS by using the following procedure.

Procedure

1. Navigate to the OpenShift Container Platform downloads page on the Red Hat Customer Portal.
2. Select the appropriate version from the **Version** drop-down list.
3. Click **Download Now** next to the **OpenShift v4.18 macOS Clients** entry and save the file.

   NOTE

   For macOS arm64, choose the **OpenShift v4.18 macOS arm64 Client** entry.

4. Unpack and unzip the archive.
5. Move the **oc** binary to a directory on your PATH.
   To check your **PATH**, open a terminal and execute the following command:

   ```
   $ echo $PATH
   ```

Verification

- Verify your installation by using an **oc** command:

  ```
  $ oc <command>
  ```

7.11. LOGGING IN TO THE CLUSTER BY USING THE CLI

You can log in to your cluster as a default system user by exporting the cluster **kubeconfig** file. The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server. The file is specific to a cluster and is created during OpenShift Container Platform installation.

Prerequisites

- You deployed an OpenShift Container Platform cluster.
- You installed the **oc** CLI.

Procedure

1. Export the **kubeadmin** credentials:

   ```
   $ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
   ```

   1 For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

2. Verify you can run **oc** commands successfully using the exported configuration:

```
$ oc whoami
```

Example output

```
system:admin
```

Additional resources

- Accessing the web console

7.12. DISABLING THE DEFAULT OPERATORHUB CATALOG SOURCES

Operator catalogs that source content provided by Red Hat and community projects are configured for OperatorHub by default during an OpenShift Container Platform installation. In a restricted network environment, you must disable the default catalogs as a cluster administrator.

Procedure

- Disable the sources for the default catalogs by adding **disableAllDefaultSources: true** to the **OperatorHub** object:

```
$ oc patch OperatorHub cluster --type json \
    -p '[{"op": "add", "path": "/spec/disableAllDefaultSources", "value": true}]'
```

TIP

Alternatively, you can use the web console to manage catalog sources. From the **Administration → Cluster Settings → Configuration → OperatorHub** page, click the **Sources** tab, where you can create, update, delete, disable, and enable individual sources.

7.13. TELEMETRY ACCESS FOR OPENSHIFT CONTAINER PLATFORM

In OpenShift Container Platform 4.18, the Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, requires internet access. If your cluster is connected to the internet, Telemetry runs automatically, and your cluster is registered to OpenShift Cluster Manager.

After you confirm that your OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually by using OpenShift Cluster Manager, use subscription watch to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

Additional resources

- About remote health monitoring

7.14. NEXT STEPS

- Customize your cluster
- Optional: Opt out of remote health reporting
- Optional: Registering your disconnected cluster

CHAPTER 8. UNINSTALLING A CLUSTER ON IBM POWER VIRTUAL SERVER

You can remove a cluster that you deployed to IBM Power® Virtual Server.

8.1. REMOVING A CLUSTER THAT USES INSTALLER-PROVISIONED INFRASTRUCTURE

You can remove a cluster that uses installer-provisioned infrastructure from your cloud.

NOTE

After uninstallation, check your cloud provider for any resources not removed properly, especially with User Provisioned Infrastructure (UPI) clusters. There might be resources that the installer did not create or that the installer is unable to access.

Prerequisites

- You have a copy of the installation program that you used to deploy the cluster.
- You have the files that the installation program generated when you created your cluster.
- You have configured the **ccoctl** binary.
- You have installed the IBM Cloud® CLI and installed or updated the VPC infrastructure service plugin. For more information see "Prerequisites" in the IBM Cloud® CLI documentation .

Procedure

1. If the following conditions are met, this step is required:
   - The installer created a resource group as part of the installation process.
   - You or one of your applications created persistent volume claims (PVCs) after the cluster was deployed.

   In which case, the PVCs are not removed when uninstalling the cluster, which might prevent the resource group from being successfully removed. To prevent a failure:

   a. Log in to the IBM Cloud® using the CLI.

   b. To list the PVCs, run the following command:

   ```
   $ ibmcloud is volumes --resource-group-name <infrastructure_id>
   ```

   For more information about listing volumes, see the IBM Cloud® CLI documentation .

   c. To delete the PVCs, run the following command:

   ```
   $ ibmcloud is volume-delete --force <volume_id>
   ```

   For more information about deleting volumes, see the IBM Cloud® CLI documentation .

2. Export the API key that was created as part of the installation process.

```
$ export IBMCLOUD_API_KEY=<api_key>
```

NOTE

You must set the variable name exactly as specified. The installation program expects the variable name to be present to remove the service IDs that were created when the cluster was installed.

3. From the directory that contains the installation program on the computer that you used to install the cluster, run the following command:

```
$ ./openshift-install destroy cluster \
--dir <installation_directory> --log-level info ① ②
```

① For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

② To view different details, specify **warn**, **debug**, or **error** instead of **info**.

NOTE

- You must specify the directory that contains the cluster definition files for your cluster. The installation program requires the **metadata.json** file in this directory to delete the cluster.
- You might have to run the **openshift-install destroy** command up to three times to ensure a proper cleanup.

4. Remove the manual CCO credentials that were created for the cluster:

```
$ ccoctl ibmcloud delete-service-id \
    --credentials-requests-dir <path_to_credential_requests_directory> \
    --name <cluster_name>
```

NOTE

If your cluster uses Technology Preview features that are enabled by the **TechPreviewNoUpgrade** feature set, you must include the **--enable-tech-preview** parameter.

5. Optional: Delete the **<installation_directory>** directory and the OpenShift Container Platform installation program.

CHAPTER 9. INSTALLATION CONFIGURATION PARAMETERS FOR IBM POWER VIRTUAL SERVER

Before you deploy an OpenShift Container Platform on IBM Power® Virtual Server, you provide parameters to customize your cluster and the platform that hosts it. When you create the **install-config.yaml** file, you provide values for the required parameters through the command line. You can then modify the **install-config.yaml** file to customize your cluster further.

9.1. AVAILABLE INSTALLATION CONFIGURATION PARAMETERS FOR IBM POWER VIRTUAL SERVER

The following tables specify the required, optional, and IBM Power Virtual Server-specific installation configuration parameters that you can set as part of the installation process.

NOTE

After installation, you cannot modify these parameters in the **install-config.yaml** file.

9.1.1. Required configuration parameters

Required installation configuration parameters are described in the following table:

Table 9.1. Required parameters

Parameter	Description	Values
apiVersion:	The API version for the **install-config.yaml** content. The current version is **v1**. The installation program may also support older API versions.	String
baseDomain:	The base domain of your cloud provider. The base domain is used to create routes to your OpenShift Container Platform cluster components. The full DNS name for your cluster is a combination of the **baseDomain** and **metadata.name** parameter values that uses the **<metadata.name>.<baseDomain>** format.	A fully-qualified domain or subdomain name, such as **example.com**.
metadata:	Kubernetes resource **ObjectMeta**, from which only the **name** parameter is consumed.	Object

Parameter	Description	Values
metadata: name:	The name of the cluster. DNS records for the cluster are all subdomains of **{{.metadata.name}}.{{.baseDomain}}**.	String of lowercase letters, hyphens (**-**), and periods (**.**), such as **dev**.
platform:	The configuration for the specific platform upon which to perform the installation: **aws**, **baremetal**, **azure**, **gcp**, **ibmcloud**, **nutanix**, **openstack**, **powervs**, **vsphere**, or **{}**. For additional information about **platform.<platform>** parameters, consult the table for your specific platform that follows.	Object

pullSecret:	Get a pull secret from Red Hat OpenShift Cluster Manager to authenticate downloading container images for OpenShift Container Platform components from services such as Quay.io.	{
		  "auths":{
		    "cloud.openshift.com":{
		      "auth":"b3Blb=",
		      "email":"you@example.com"
		    },
		    "quay.io":{
		      "auth":"b3Blb=",
		      "email":"you@example.com"
		    }
		  }
		}

platform:
  powervs:
    userID:
	The UserID is the login for the user's IBM Cloud® account.	String. For example, **existing_user_id**.

platform:
  powervs:
    powervsResourceGroup:
	The PowerVSResourceGroup is the resource group in which IBM Power® Virtual Server resources are created. If using an existing VPC, the existing VPC and subnets should be in this resource group.	String. For example, **existing_resource_group**.

platform:
  powervs:
    region:
	Specifies the IBM Cloud® colo region where the cluster will be created.	String. For example, **existing_region**.

Parameter	Description	Values

platform:
  powervs:
    zone:
	Specifies the IBM Cloud® colo region where the cluster will be created.	String. For example, **existing_zone**.

9.1.2. Network configuration parameters

You can customize your installation configuration based on the requirements of your existing network infrastructure. For example, you can expand the IP address block for the cluster network or provide different IP address blocks than the defaults.

Only IPv4 addresses are supported.

Table 9.2. Network parameters

Parameter	Description	Values
networking:	The configuration for the cluster network.	Object NOTE You cannot modify parameters specified by the **networking** object after installation.
networking: networkType:	The Red Hat OpenShift Networking network plugin to install.	The default value is **OVNKubernetes**.
networking: clusterNetwork:	The IP address blocks for pods. The default value is **10.128.0.0/14** with a host prefix of **/23**. If you specify multiple IP address blocks, the blocks must not overlap.	An array of objects. For example: networking: clusterNetwork: - cidr: 10.128.0.0/14 hostPrefix: 23

networking:
  clusterNetwork:
    cidr:
	Required if you use **networking.clusterNetwork**. An IP address block.
	An IPv4 network.
		An IP address block in Classless Inter-Domain Routing (CIDR) notation. The prefix length for an IPv4 block is between **0** and **32**.

Parameter	Description	Values

`networking:`
`  clusterNetwork:`
`    hostPrefix:`
	The subnet prefix length to assign to each individual node. For example, if **hostPrefix** is set to **23** then each node is assigned a **/23** subnet out of the given **cidr**. A **hostPrefix** value of **23** provides 510 (2^(32 - 23) - 2) pod IP addresses.	A subnet prefix.
		The default value is **23**.

`networking:`
`  serviceNetwork:`
	The IP address block for services. The default value is **172.30.0.0/16**.
	The OVN-Kubernetes network plugins supports only a single IP address block for the service network.
		An array with an IP address block in CIDR format. For example:
		`networking:`
		`  serviceNetwork:`
		`   - 172.30.0.0/16`

`networking:`
`  machineNetwork:`
	The IP address blocks for machines.	An array of objects. For example:
		`networking:`
		`  machineNetwork:`
		`  - cidr: 10.0.0.0/16`

`networking:`
`  machineNetwork:`
`    cidr:`
	Required if you use **networking.machineNetwork**. An IP address block. The default value is **10.0.0.0/16** for all platforms other than libvirt and IBM Power® Virtual Server. For libvirt, the default value is **192.168.126.0/24**. For IBM Power® Virtual Server, the default value is **192.168.0.0/24**.	An IP network block in CIDR notation.
		For example, **192.168.0.0/24**.
		NOTE
		Set the **networking.machineNetwork** to match the CIDR that the preferred NIC resides in.

`networking:`
`  ovnKubernetesConfig:`
`    ipv4:`
`  internalJoinSubnet:`
	Configures the IPv4 join subnet that is used internally by **ovn-kubernetes**. This subnet must not overlap with any other subnet that OpenShift Container Platform is using, including the node network. The size of the subnet must be larger than the number of nodes. You cannot change the value after installation.	An IP network block in CIDR notation. The default value is **100.64.0.0/16**.

9.1.3. Optional configuration parameters

Optional installation configuration parameters are described in the following table:

Table 9.3. Optional parameters

Parameter	Description	Values
`additionalTrustBundle:`	A PEM-encoded X.509 certificate bundle that is added to the nodes' trusted certificate store. This trust bundle may also be used when a proxy has been configured.	String
`capabilities:`	Controls the installation of optional core cluster components. You can reduce the footprint of your OpenShift Container Platform cluster by disabling optional components. For more information, see the "Cluster capabilities" page in *Installing*.	String array

`capabilities:`
`baselineCapabilitySet:`
	Selects an initial set of optional capabilities to enable. Valid values are **None**, **v4.11**, **v4.12** and **vCurrent**. The default value is **vCurrent**.	String

`capabilities:`
`additionalEnabledCapabilities:`
	Extends the set of optional capabilities beyond what you specify in **baselineCapabilitySet**. You may specify multiple capabilities in this parameter.	String array

`cpuPartitioningMode:`	Enables workload partitioning, which isolates OpenShift Container Platform services, cluster management workloads, and infrastructure pods to run on a reserved set of CPUs. Workload partitioning can only be enabled during installation and cannot be disabled after installation. While this field enables workload partitioning, it does not configure workloads to use specific CPUs. For more information, see the *Workload partitioning* page in the *Scalability and Performance* section.	**None** or **AllNodes**. **None** is the default value.
`compute:`	The configuration for the machines that comprise the compute nodes.	Array of **MachinePool** objects.

Parameter	Description	Values
compute: architecture:	Determines the instruction set architecture of the machines in the pool. Currently, heteregeneous clusters are not supported, so all pools must specify the same architecture. Valid values are **ppc64le** (the default).	String
compute: hyperthreading:	Whether to enable or disable simultaneous multithreading, or **hyperthreading**, on compute machines. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. IMPORTANT If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.	**Enabled** or **Disabled**
compute: smtLevel:	The SMTLevel specifies the level of SMT to set to the control plane and compute machines. Valid values are 1, 2, 3, 4, 5, 6, 7, 8, **off**, and **on**.	String
compute: name:	Required if you use **compute**. The name of the machine pool.	**worker**
compute: platform:	Required if you use **compute**. Use this parameter to specify the cloud provider to host the worker machines. This parameter value must match the **controlPlane.platform** parameter value. Example usage, **compute.platform.powervs.sysType**.	**aws**, **azure**, **gcp**, **ibmcloud**, **nutanix**, **openstack**, **powervs**, **vsphere**, or **{}**
compute: replicas:	The number of compute machines, which are also known as worker machines, to provision.	A positive integer greater than or equal to **2**. The default value is **3**.

Parameter	Description	Values
featureSet:	Enables the cluster for a feature set. A feature set is a collection of OpenShift Container Platform features that are not enabled by default. For more information about enabling a feature set during installation, see "Enabling features using feature gates".	String. The name of the feature set to enable, such as **TechPreviewNoUpgrade**.
controlPlane:	The configuration for the machines that comprise the control plane.	Array of **MachinePool** objects.
controlPlane: architecture:	Determines the instruction set architecture of the machines in the pool. Currently, heterogeneous clusters are not supported, so all pools must specify the same architecture. Valid values are **ppc64le** (the default).	String
controlPlane: hyperthreading:	Whether to enable or disable simultaneous multithreading, or **hyperthreading**, on control plane machines. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. IMPORTANT If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.	**Enabled** or **Disabled**
controlPlane: name:	Required if you use **controlPlane**. The name of the machine pool.	**master**

Parameter	Description	Values
controlPlane: platform:	Required if you use **controlPlane**. Use this parameter to specify the cloud provider that hosts the control plane machines. This parameter value must match the **compute.platform** parameter value. Example usage, **controlPlane.platform.powervs.processors**.	**aws**, **azure**, **gcp**, **ibmcloud**, **nutanix**, **openstack**, **powervs**, **vsphere**, or **{}**
controlPlane: replicas:	The number of control plane machines to provision.	Supported values are **3**, or **1** when deploying single-node OpenShift.
credentialsMode:	The Cloud Credential Operator (CCO) mode. If no mode is specified, the CCO dynamically tries to determine the capabilities of the provided credentials, with a preference for mint mode on the platforms where multiple modes are supported. NOTE Not all CCO modes are supported for all cloud providers. For more information about CCO modes, see the "Managing cloud provider credentials" entry in the *Authentication and authorization* content.	**Mint**, **Passthrough**, **Manual** or an empty string (**""**).
imageContentSources:	Sources and repositories for the release-image content.	Array of objects. Includes a **source** and, optionally, **mirrors**, as described in the following rows of this table.
imageContentSources: source:	Required if you use **imageContentSources**. Specify the repository that users refer to, for example, in image pull specifications.	String
imageContentSources: mirrors:	Specify one or more repositories that may also contain the same images.	Array of strings

Parameter	Description	Values
publish:	How to publish or expose the user-facing endpoints of your cluster, such as the Kubernetes API, OpenShift routes.	**Internal** or **External**. The default value is **External**. Setting this field to **Internal** is not supported on non-cloud platforms.
sshKey:	The SSH key to authenticate access to your cluster machines. NOTE For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.	For example, **sshKey: ssh-ed25519 AAAA..**.
platform: powervs: vpcRegion:	Specifies the IBM Cloud® region in which to create VPC resources.	String. For example, **existing_vpc_region**.
platform: powervs: vpcSubnets:	Specifies existing subnets (by name) where cluster resources will be created.	String. For example, **powervs_region_example_subnet**.
platform: powervs: vpcName:	Specifies the IBM Cloud® name.	String. For example, **existing_vpcName**.
platform: powervs: serviceInstanceGUID:	Specifies the ID of the Power IAAS instance created from the IBM Cloud® Catalog.	String. For example, **existing_service_instance_GUID**.

Parameter	Description	Values

`platform:`
`  powervs:`
`clusterOSImage:`
	Specifies a pre-created IBM Power® Virtual Server boot image that overrides the default image for cluster nodes.	String. For example, **existing_cluster_os_image**.

`platform:`
`  powervs:`
`defaultMachinePlatform:`
	Specifies the default configuration used when installing on IBM Power® Virtual Server for machine pools that do not define their own platform configuration.	String. For example, **existing_machine_platform**.

`platform:`
`  powervs:`
`    memoryGiB:`
	Specifies the size of a virtual machine's memory, in GB.	The valid integer must be an integer number of GB that is at least 2 and no more than 64, depending on the machine type.

`platform:`
`  powervs:`
`    procType:`
	Defines the processor sharing model for the instance.	The valid values are Capped, Dedicated, and Shared.

`platform:`
`  powervs:`
`    processors:`
	Defines the processing units for the instance.	The number of processors must be from .5 to 32 cores. The processors must be in increments of .25.

`platform:`
`  powervs:`
`    sysType:`
	Defines the system type for the instance.	The system type must be **e980**, **s922**, **e1080**, or **s1022**. The available system types depend on the zone you want to target.

`platform:`
`  powervs:`
`    tgName:`
	Defines the name of an existing Transit Gateway.	String. For example, **existing_tgName**.