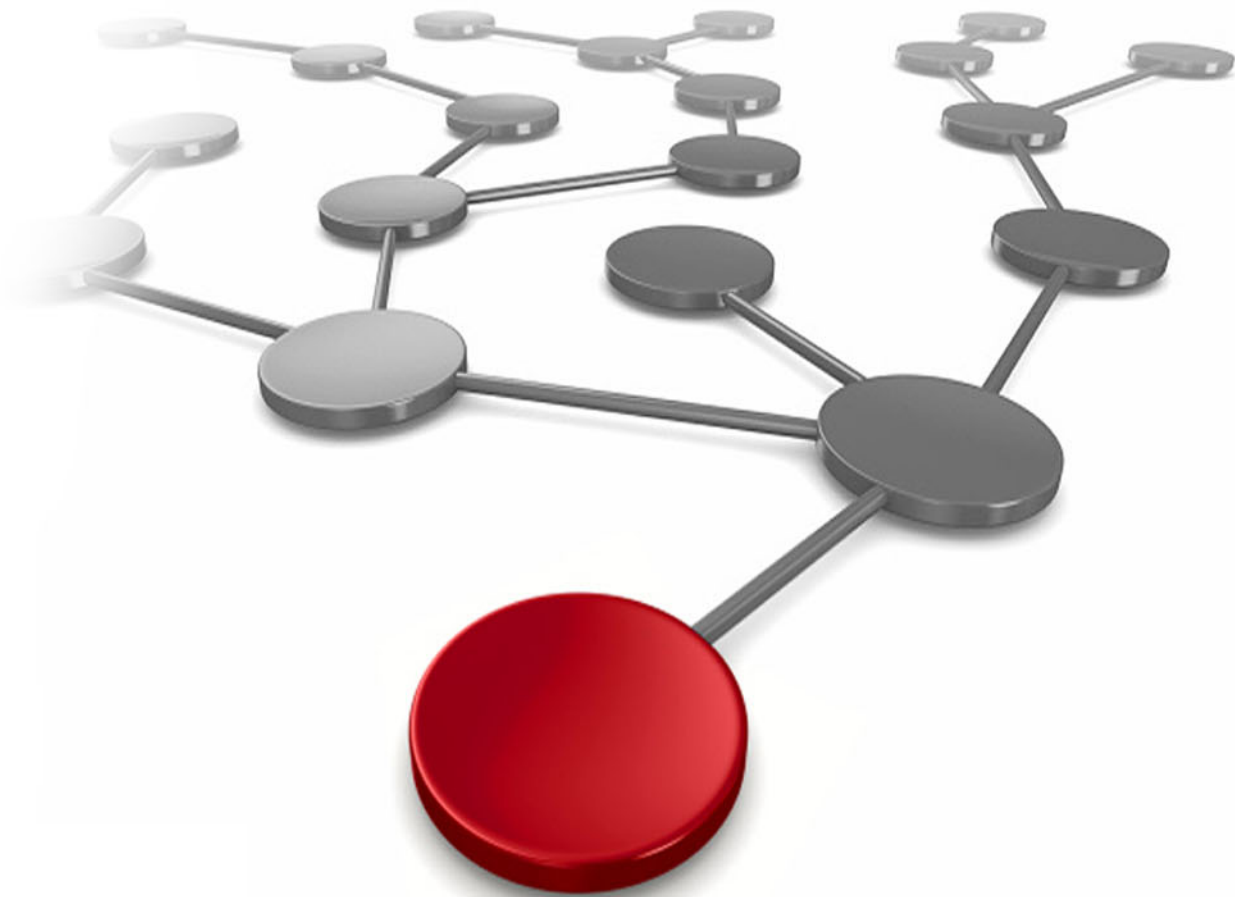


# Enabling IBM Cloud Pak for Multicloud Management to Connect Kubernetes Clusters Using IBM Secure Gateway

Michael Bubel







# Enabling IBM Cloud Pak for Multicloud Management to Connect Kubernetes Clusters Using IBM Secure Gateway

In today's dynamically changing IT landscape, it is highly likely that a company's cloud strategy spans multiple cloud providers. Such a span is known as the *hybrid multi-cloud landscape*.

The challenges that quickly surface in an IT department's list of responsibilities now encompass managing environments that are running on multiple cloud providers. The traditional IT administrators find themselves using individual dashboards for each of the cloud providers to monitor and manage those environments. In turn, each of the cloud provider's dashboards have their own unique features that require a learning curve to become productive.

The traditional IT administrator must now become a specialized hybrid cloud engineer with different hats for each of the cloud providers. This dynamic led to the quick realization of the need for a tool that provides a common dashboard for managing a company's hybrid cloud landscape. IBM® is one such company that quickly recognized this need and used their experience in years of systems management tools created the IBM Cloud® Pak for Multicloud Management.

This paper describes the steps that are required to connect a Kubernetes management dashboard that is provided with the IBM Cloud Pak® for Multicloud Management running on an on-premises private cloud to Kubernetes clusters that are running on public clouds. An IBM Cloud service that is called IBM Secure Gateway is at the core of this connection. The procedure to set up the Kubernetes clusters to use the IBM Secure Gateway service also is described in this paper.

## IBM Cloud Pak for Multicloud Management

The IBM Cloud Pak for Multicloud Management, running on Red Hat OpenShift, provides consistent visibility, governance, and automation from on-premises to any public clouds that are used in the enterprise. Enterprises gain capabilities, such as multicluster management, event management, application management, and infrastructure management. Enterprises can use this Cloud Pak to help increase operational efficiency that is driven by intelligent data, analysis, and predictive golden signals. They also gain built-in support for their Compliance Management. For more information, see the [IBM Cloud Pak for Multicloud Management documentation](#).

# What is included in the IBM Cloud Pak for Multicloud Management

In addition to the default features for managing multicloud environments, the IBM Cloud Pak for Multicloud Management includes the following installable modules that you can add to your cluster to manage applications and infrastructure and automate tasks:

- ▶ Monitoring Module for monitoring the performance and availability of cloud applications in hybrid cloud environments.
- ▶ Terraform and Service Automation Module for cluster security, operating efficiency, and appropriate service level delivery.
- ▶ CloudForms for controlling and managing cloud infrastructures.
- ▶ Red Hat Ansible Tower for running your automation tasks.<sup>1</sup>

## Requirement

Figure 1 shows an example of a typical hybrid cloud requirement. In this scenario, IBM Cloud Pak for Multicloud Management must be deployed in an on-premises cloud managing Kubernetes clusters that is running on IBM public cloud, AWS public cloud, and Azure public cloud.

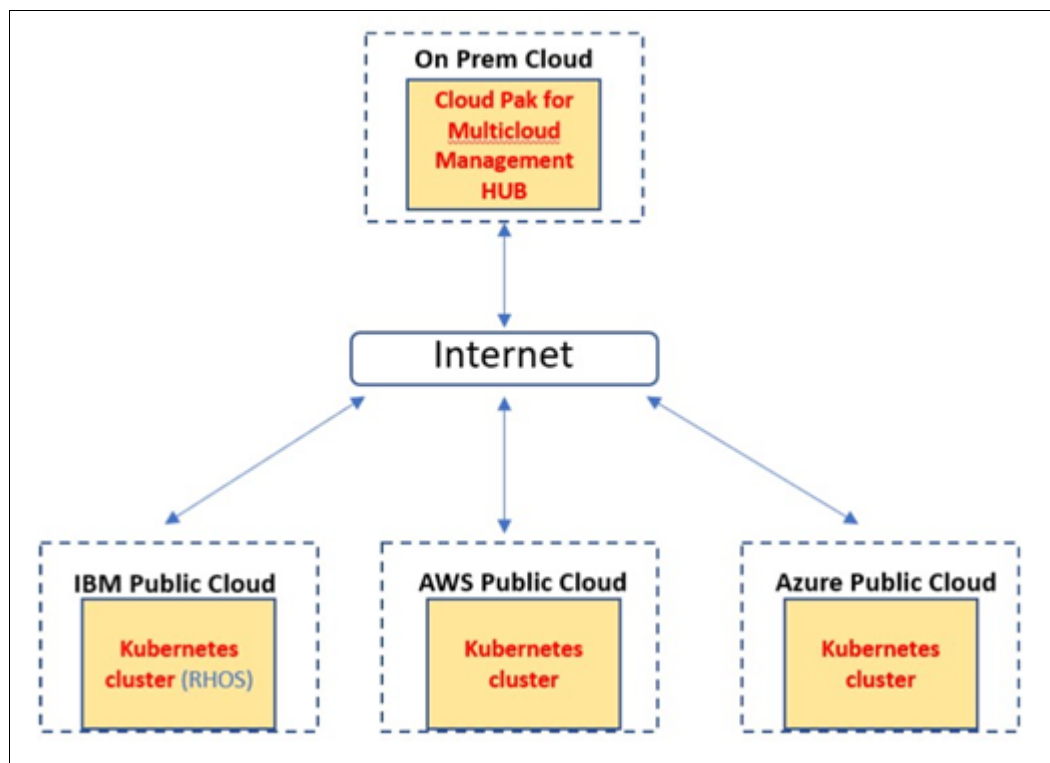


Figure 1 Hybrid cloud requirement example

<sup>1</sup> <https://cloud.ibm.com/catalog/content/ibm-cp-management-2c621076-b5eb-489d-ae92-c63f13d713ef-global#about>

## Challenges

A Hybrid Cloud engineer can be faced with several challenges while trying to connect public and private clouds, including the following examples:

- ▶ Security policies and firewall rules.
- ▶ The network team in your data center does not want a traditional VPN to bridge private on-premises networks to private public cloud networks.
- ▶ Time challenge in setting up a traditional VPN.
- ▶ Potential high cost of third-party solutions.

**Note:** Check with your security regulations department to validate that the solution that is provided here adheres to security regulations.

## Solution

Deploying the IBM Secure Gateway service on the IBM Cloud creates an encrypted secure connection to your on-premises cloud. The Cloud Pak for Multicloud Management hub cluster use this connection to securely communicate with the public cloud Kubernetes clusters.

Figure 2 shows a high-level overview of the IBM Secure Gateway that provides the solution of the secure connection by using the internet by providing controlled public access points.

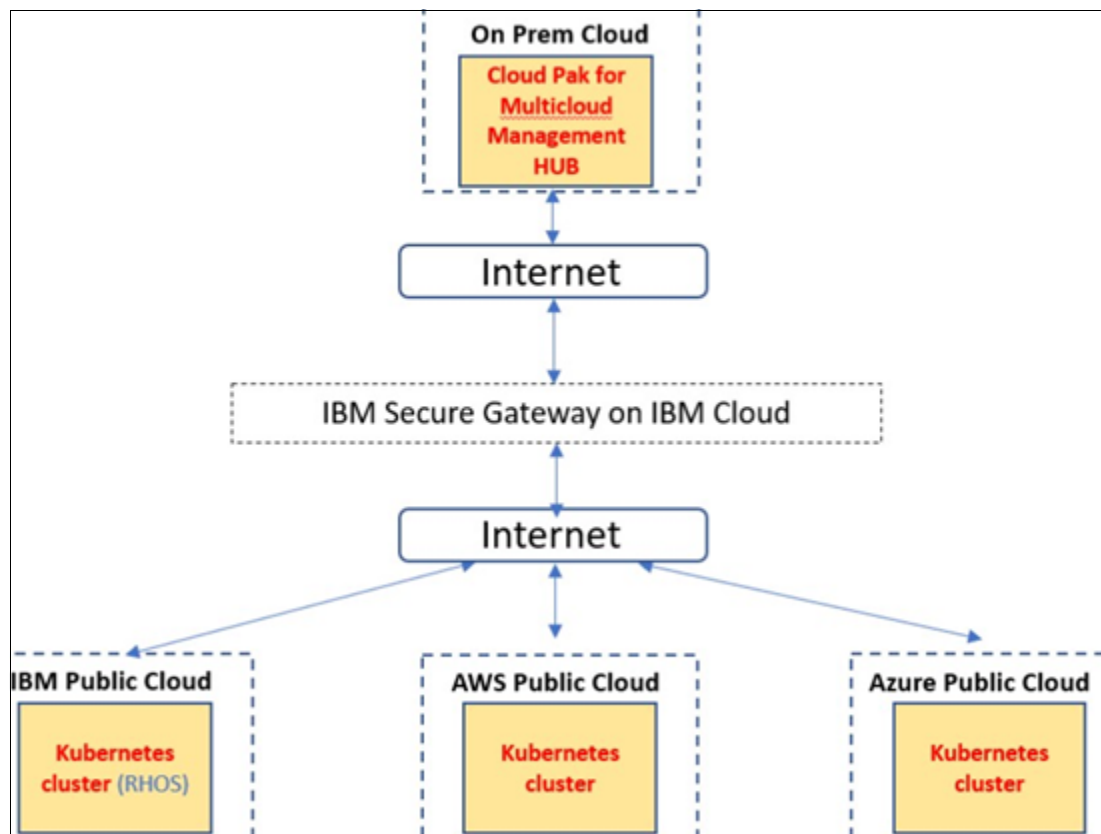


Figure 2 High-level overview of the IBM Secure Gateway

# IBM Secure Gateway Service on IBM Cloud overview

The Secure Gateway Service provides a quick, easy, and secure solution for connecting resources in a protected environment to cloud resources. By deploying the light-weight and natively installed Secure Gateway Client, you can establish a secure, persistent connection between your environment and the cloud through an outbound call.

After the client is connected, you can safely connect your applications and resources by specifying their host and port to create corresponding destinations on the cloud. Rather than bridging your environments at the network level, such as in a traditional VPN that begins with full access and must be limited from the top down, Secure Gateway provides granular access only to the resources that you defined.

For more information about the IBM Secure Gateway, see [this web page](#).

## Purpose

This technology document is indented to describe a step-by-step procedure to set up the IBM Secure Gateway Service on your IBM Public Cloud account. This gateway is used to connect an on-premises Cloud Pak for Multicloud Management HUB cluster to a Kubernetes cluster running on a public cloud.

This document is *not* indented to replace the current IBM Secure Gateway documentation, IBM Cloud Pak for Multicloud Management (MCM) documentation, or the Kubernetes documentations.

## Summary

This procedure features the following overall tasks:

- ▶ Defining the IBM Secure Gateway
- ▶ Adding a cluster to be managed by IBM Cloud Pak for MCM Hub cluster
- ▶ Modifying the klusterlet-bootstrap secret to point to the IBM Secure Gateway

## Requirements

The following requirements must be met:

- ▶ IBM Cloud Account
- ▶ Deployed IBM Cloud Pak for Multicloud Management HUB cluster on-premises:
  - Version 1.2 Enterprise, RedHat OpenShift x86 Linux.
  - Version 1.3 Enterprise, RedHat OpenShift x86 Linux, or IBM Power 8 or Power 9.
- ▶ Kubernetes Clusters on public clouds to be managed: Kubernetes version < 1.16

## Assumptions

It is assumed that the reader is familiar with or proficient in the following subject areas:

- ▶ Cloud Pak for Multicloud Management
- ▶ Kubernetes clusters
- ▶ System administration and networking basics

# Procedure

In this section, the procedure is broken up into three different sections. Each of the sections feature multiple steps to complete each section.

The first section is creating the IBM Secure Gateway. The steps cover logging into the IBM Cloud account, creating a IBM Secure Gateway, configuring the IBM Secure Gateway service connection into your datacenter, and enabling the gateway service to be available for use.

The second section describes adding a cluster to be managed by IBM Cloud Pak for Multicloud Management hub cluster. The steps cover logging into the Multicloud Management hub cluster console, adding a Kubernetes cluster to be managed, and running the cluster import command in the new Kubernetes cluster to be managed.

The final section of this procedure is modifying the klusterlet bootstrap secret to point to the IBM Secure Gateway. The steps describe patching the klusterlet bootstrap secret on the new Kubernetes cluster to be managed.

## Step 1: Defining the IBM Secure Gateway

The IBM Secure Gateway is a service that is on the IBM Public Cloud. This section first describes the process that is used to define the service on your IBM cloud account.

Next, we described how to install the IBM Secure Gateway client on a server in your on-premises cloud, along with configuring the proper access control list (ACL) to enforce the security policy into your on-premises cloud.

Finally, the steps that are required to create the IBM Cloud Secure Gateway Destination, which creates the public access host and port, are described.

This public access host and port are used by the Kubernetes Clusters on the public clouds to communicate with the managing IBM Cloud Pak for Multicloud Management HUB Cluster configured in the on-premises cloud.

Complete the following steps to define the IBM Secure Gateway service on your IBM Cloud account:

1. Log in to your [IBM Cloud Account](#) by using your account credentials.
2. Search the IBM Cloud Catalog for Secure Gateway and select the Secure Gateway from the displayed results (see Figure 3 on page 6).

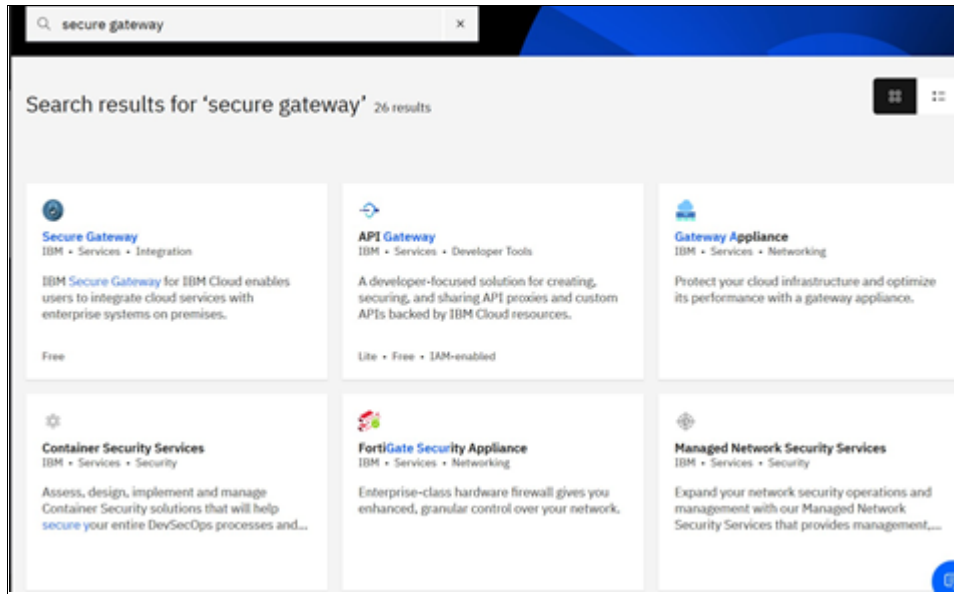


Figure 3 Typical returned results after searching for IBM Secure Gateway in the IBM Cloud Catalog

- After the Secure Gateway services are selected from step 2, the panel that is shown in Figure 4 is displayed.

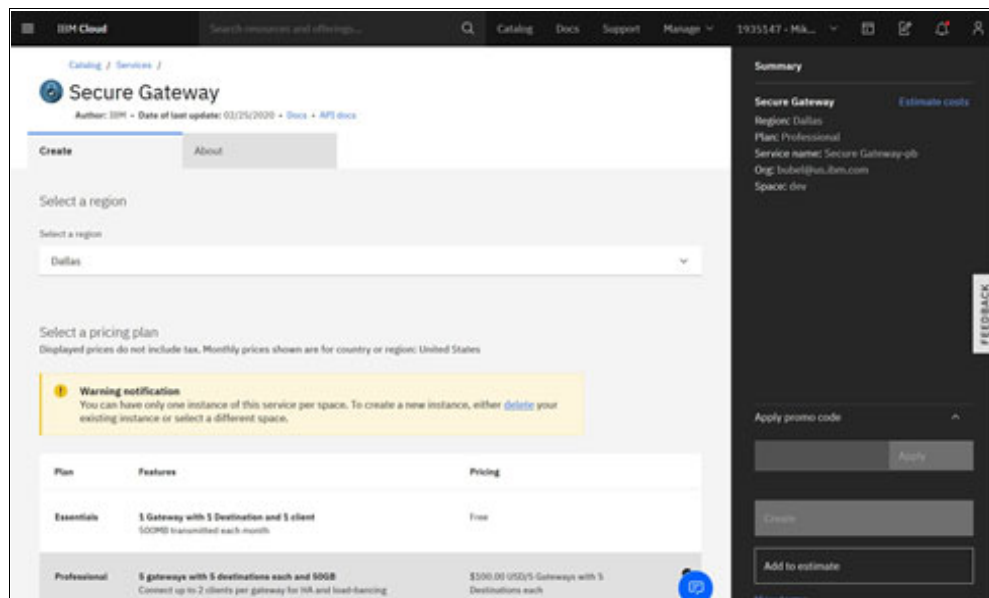


Figure 4 IBM Secure Gateway service configuration landing page

- Add a gateway and enter a meaningful name for the gateway.



5. Select **Connect a Client** (see Figure 5).

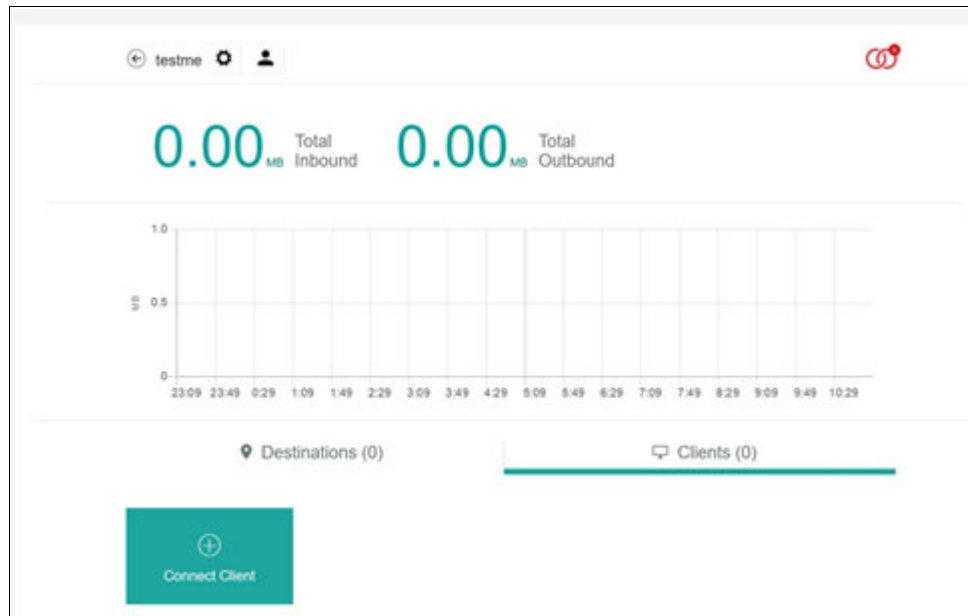


Figure 5 IBM Secure Gateway service to connect a client configuration page

6. Complete the following steps:

- a. Download the secure [gateway client rpm](#).
- b. Install the secure gateway client rpm.

During the installation, you need the GATEWAY ID and the SECURITY TOKEN from the Installer page, as shown in Figure 6.

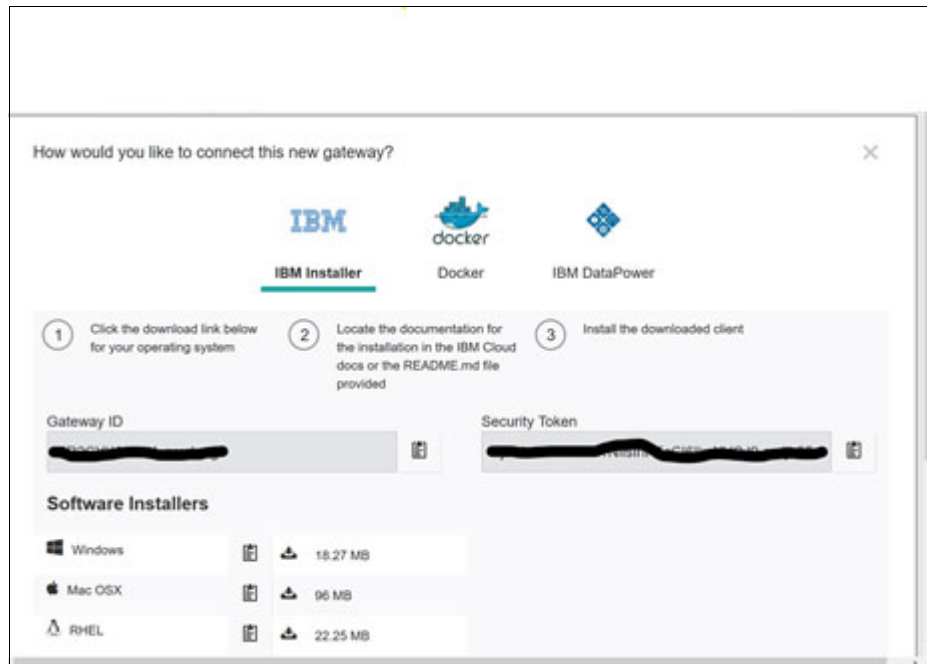


Figure 6 IBM Secure Gateway service to download a client installer options page

c. Configure the secure gateway client.

Setting up the `/etc/acl.file` configures the secure gateway client to allow the `cluster_kube_apiserver_host` and `cluster_kube_apiserver_port` restricted access, which enforces security policies into the on-premises datacenter local network.

This information must be obtained from the IBM Cloud Pak for MCM HUB cluster.

**Note:** The `cluster_kube_apiserver_host` and `cluster_kube_apiserver_port` is how the Kubernetes clusters that are to be managed by the IBM Cloud Pak for Multicloud Management HUB cluster communicate.

You must obtain the host and port information from your HUB cluster (the login session must be authenticated to the HUB cluster).

Run the following command from your authenticated login session:

```
kubectl -n kube-public get configmap ibmcloud-cluster-info -o yaml
```

The output of this command is similar to the output that is shown in Example 1.

*Example 1 Output example*

---

```
apiVersion: v1
data:
  cluster_address: icp-console.pokmcm.joelab.ibm.com
  cluster_ca_domain: icp-console.pokmcm.joelab.ibm.com
  cluster_endpoint: https://icp-management-ingress.kube-system.svc:443
  cluster_kube_apiserver_host: mcm-hub-cluster1.pokmcm.joelab.ibm.com
  cluster_kube_apiserver_port: "8443"
  cluster_name: mycluster
  cluster_router_http_port: "8080"
  cluster_router_https_port: "443"
  edition: Enterprise Edition
  openshift_router_base_domain: pokmcm.joelab.ibm.com
  proxy_address: icp-proxy.pokmcm.joelab.ibm.com
  proxy_ingress_http_port: "80"
  proxy_ingress_https_port: "443"
  version: 3.2.3
kind: ConfigMap
metadata:
  .....
```

---

From the output of the command, make a copy of the following values:

```
cluster_kube_apiserver_host: mcm-hub-cluster1.pokmcm.joelab.ibm.com
cluster_kube_apiserver_port: "8443"
```

Copy the Sample ACL File template from the `/opt/ibm/securegateway/client/` directory to the `/etc/acl.file`:

```
cp /opt/ibm/securegateway/client/SampleACLFile.txt /etc/acl.file
```

Edit the `/etc/acl.file` to configure the secure gateway client to allow `cluster_kube_apiserver_host` and `cluster_kube_apiserver_port` restricted access, which enforces security policies into the on-premises datacenter local network:

```
vi /etc/acl.file
```

The following lines were added to the example:

```
#-----#  
# To allow all ports on a specific host:  
#acl allow cluster_kube_apiserver_host:cluster_kube_apiserver_port  
acl allow mcm-hub-cluster1.pokmcm.joelab.ibm.com:8443  
#-----#
```

When editing your `/etc/acl` file, update the values of `cluster_kube_apiserver_host` and `cluster_kube_apiserver_port` to match your configuration.

- d. Restart the secure gateway client.

```
systemctl restart securegateway_client
```

**Important:** Make note of the Gateway ID and the Security Token that is used during the installation of the client.

## Installer

Several operating systems options are available for the installer, as shown in Figure 7.













Software Installers			
 Windows			18.27 MB
 Mac OSX			96 MB
 RHEL			22.25 MB
 Ubuntu Z-Linux			13.96 MB
 Ubuntu PPC			13.81 MB
 Ubuntu			14.53 MB
 AIX			25.61 MB

Figure 7 Available installers for download options

After the client is connected, you must create an IBM Cloud Secure Gateway Destination. Complete the following steps:

1. Select **On-Premises** (see Figure 8).

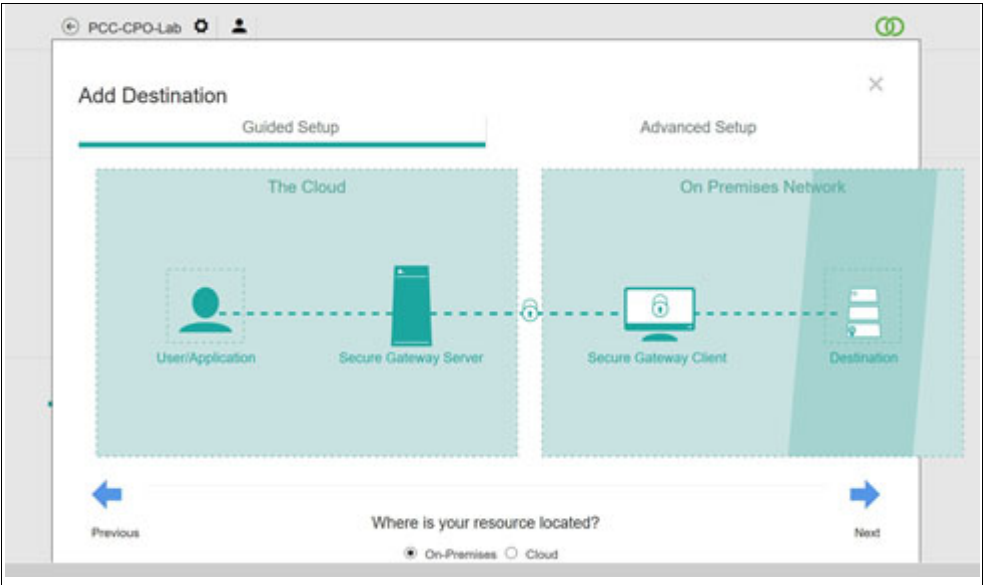


Figure 8 Create gateway destination options

2. Enter the host name of the port (see Figure 9). The host name and port are obtained from the output of the command from step 6, which lists the `cluster_kube_apiserver_host` and the `cluster_kube_apiserver_port`. This is the same information that you added in the `/etc/ac1.file` configuration file for the Secure Gateway client:

```
cluster_kube_apiserver_host: mcm-hub-cluster1.pokmcm.joelab.ibm.com
cluster_kube_apiserver_port: "8443"
```

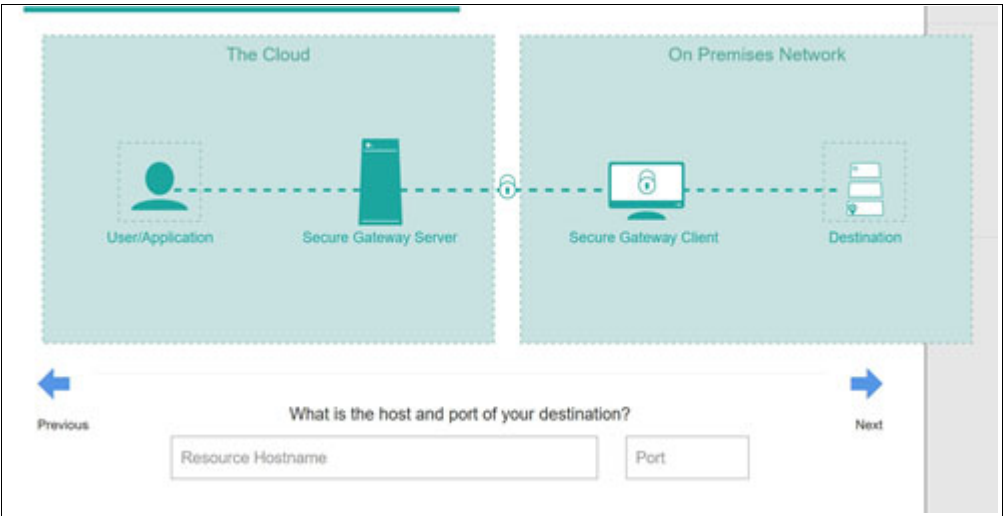


Figure 9 Adding the host and port for the gateway destination requirements

3. Select **TCP** as the protocol to connect to your destination (see Figure 10).

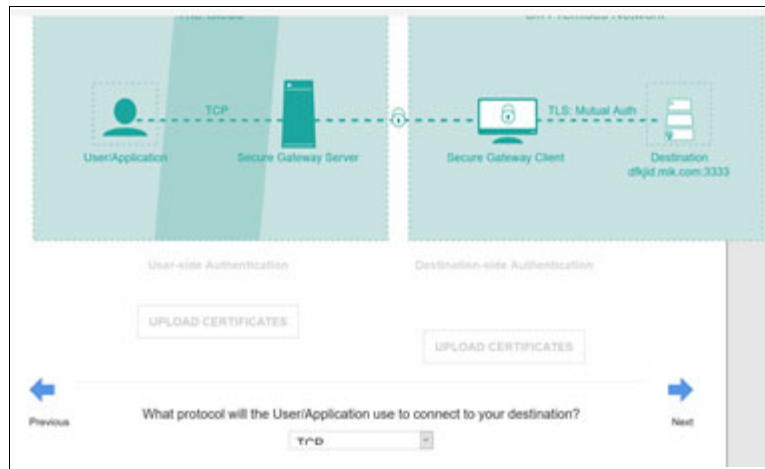


Figure 10 Selecting protocol for communication options

4. Select **None** for the type of authentication for the destination to enforce (see Figure 11).

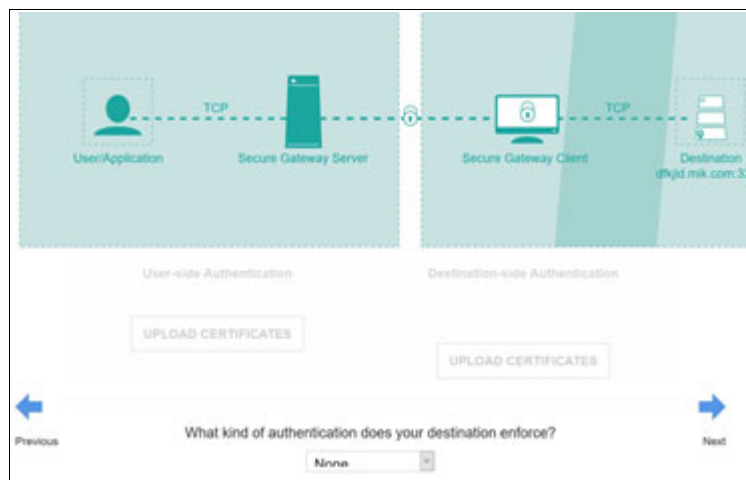


Figure 11 Selecting authentication for communication options

5. As shown in Figure 12, leave the destination private IP tables rules blank.

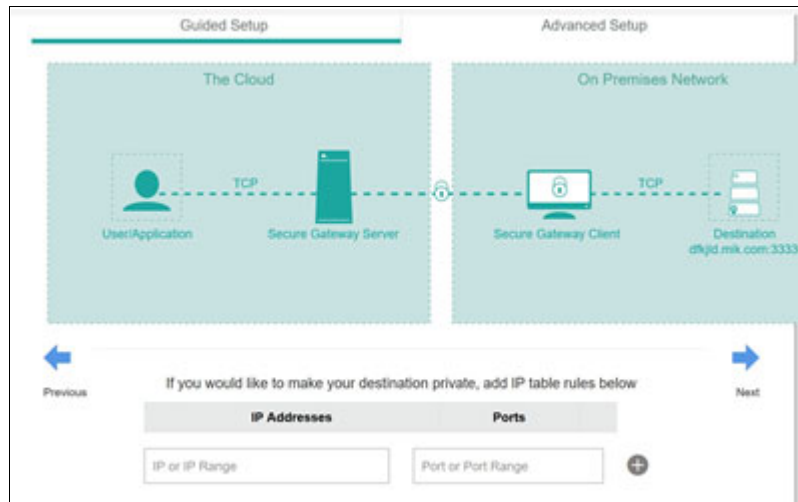


Figure 12 Selecting private IP Tables to configure options

6. Enter a name for the destination (in this example: PPC-MCM-HUB-Cluster).

7. Obtain the Cloud Host and port from the newly created destination.

This information is the public access host name and port into your on-premises data center that the Cloud Pak for MCM hub cluster uses to communicate with the public cloud Kubernetes clusters (see Figure 13).

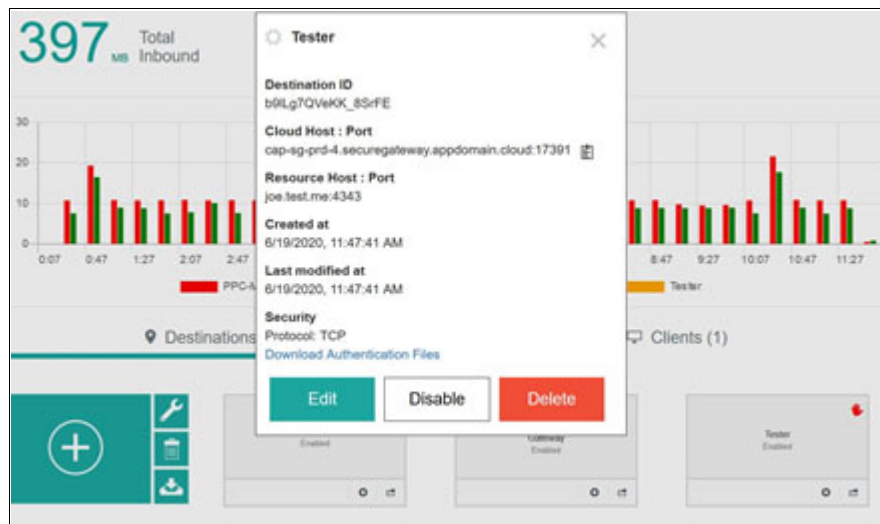


Figure 13 IBM Secure Gateway service after it is created

8. Make note of the Cloud Host and port that is used in Section III.

In this example:

- Cloud Host: cap-sg-prd-4.securegateway.appdomain.cloud
- Cloud Port: 43232

## Step 2: Adding a cluster to be managed by IBM Cloud Pak for Multicloud Management Hub cluster

The IBM Cloud Pak for Multicloud Management Hub cluster provides the single pane of glass that is used to manage multiple Kubernetes clusters on- or off-premises.

The processes of adding a Kubernetes cluster to be managed by the Multicloud Management hub is done by using the management GUI panels in the tool. The cluster view panel in the GUI has the option to add a cluster. We use this process to add the new cluster to be managed.

The output of following the steps to add a Kubernetes cluster to be managed by the hub cluster is an **import** command to be copied and run on the new cluster. The command is run by using **kubect1** commands. To run the **kubect1** commands, the correct kubect1 authentication into the Kubernetes cluster is required.

Complete the following steps:

1. Add a public Kubernetes cluster to be managed by the Cloud Pak for Multicloud Management HUB cluster:
  - a. Log in to the Cloud Pak for Multicloud Management Hub cluster console (see Figure 14).

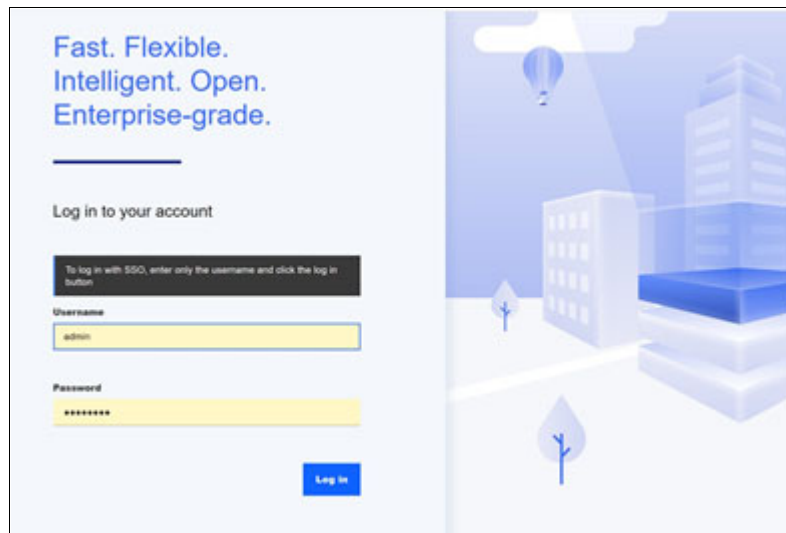


Figure 14 IBM Cloud Pak for Multicloud Management Hub cluster log in window

- b. From the Clusters window, select **Add Cluster** (see Figure 15).

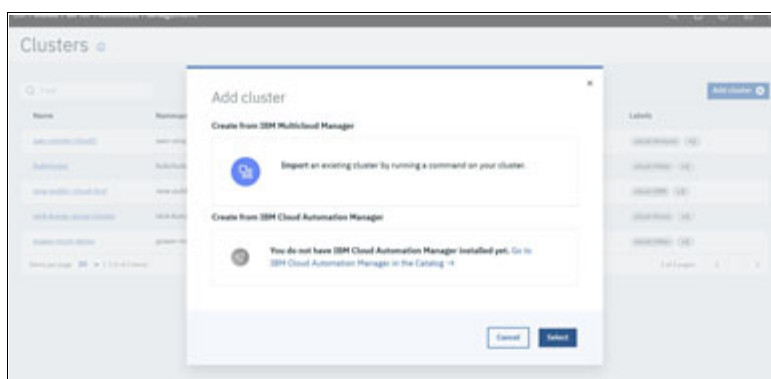


Figure 15 Adding cluster options

- c. Select the **Import an existing cluster by running a command on your cluster** option that is shown in Figure 15.
- d. Enter the cluster name and the namespace for the new cluster (see Figure 16). In this Example:
- Cluster name: `ibm-cloud-app-development`
  - Namespace: `ibm-cloud-app-development`

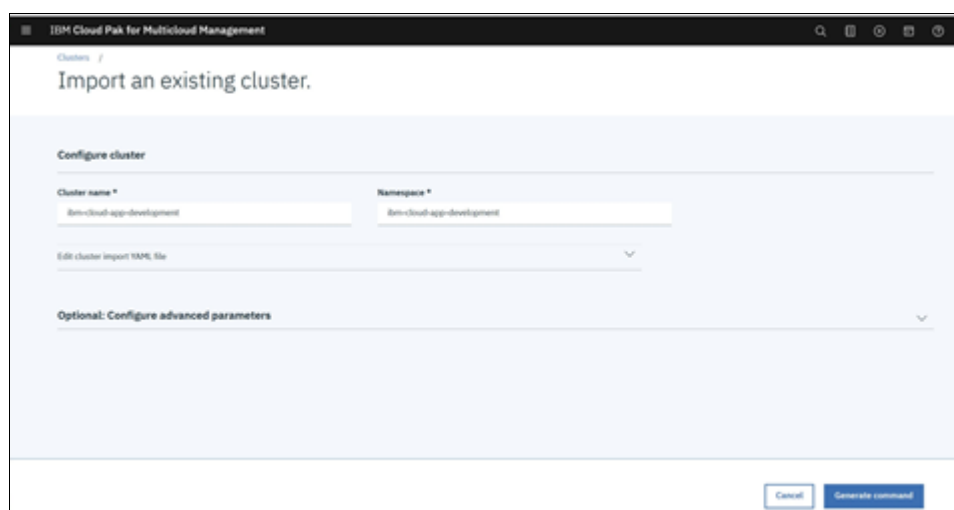


Figure 16 Importing cluster options



- e. Select **Generate Command** and copy the command (see Figure 17).

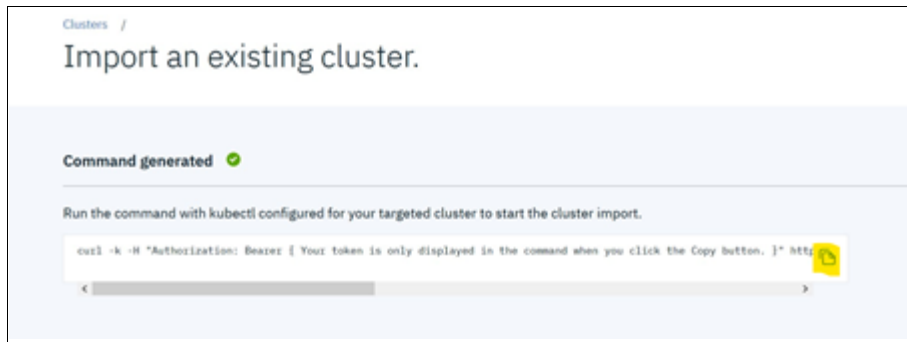


Figure 17 Copy to import an existing cluster command

2. Run the command that you copied on the Kubernetes cluster, which you are importing into the HUB Cluster. (You must be authenticated to the Kubernetes cluster for the import to be successful.)

You must run this command on a server in your on-premises cloud that has simple network access to the Kubernetes cluster to be imported in the public cloud and the IBM Cloud Pak for MCM hub cluster:

- a. Authenticate to the Kubernetes cluster in the public cloud.
- b. Test that a connection exists to the Kubernetes cluster in the public cloud:

Example: **kubectl get nodes**

3. Test that you can access the local HUB Cluster:

Example: **ping icp-console.pokmcm.joelab.ibm.com**

4. Run the **import** command:

Example: `curl -k -H "Authorization: Bearer ffladceedc8f...3a4140794732f732c31" https://icp-console.pokmcm.joelab.ibm.com:443/rcm/v1/clusters/ibm-cloud-app-development/ibm-cloud-app-development/import.yaml | kubectl apply -f -`

**Note:** If the Kubernetes cluster that is being imported is part the on-premises cluster that is hosting the IBM Cloud Pak for MCM hub cluster, the IBM Secure Gateway does not need to be used. In this example, the following command successfully imports the cluster into the HUB and you can skip “Step 3: Modifying the klusterlet-bootstrap secret to point to the IBM Secure Gateway” on page 16:

```
curl -k -H "Authorization: Bearer ffladceedc8f...3a4140794732f732c31"
https://icp-console.pokmcm.joelab.ibm.com:443/rcm/v1/clusters/ibm-cloud-app-dev
elopment/ibm-cloud-app-development/import.yaml | kubectl apply -f -
```

Because the purpose of this publication is to describe how to connect Kubernetes clusters on public cloud to an IBM Cloud Pak for MCM Hub cluster deployed on an on-premises cloud, see “Step 3: Modifying the klusterlet-bootstrap secret to point to the IBM Secure Gateway” on page 16.

### Step 3: Modifying the klusterlet-bootstrap secret to point to the IBM Secure Gateway

The **import** command that was run in “Step 2: Adding a cluster to be managed by IBM Cloud Pak for Multicloud Management Hub cluster” on page 13 to import the Kubernetes cluster running on a public cloud into the IBM Cloud Pak for MCM hub cluster does not successfully import the cluster.

The status of the Kubernetes cluster that is being imported is in Pending State. The reason why the Kubernetes cluster is not successfully imported is that the `cluster_kube_apiserver_host` and `cluster_kube_apiserver_port` are not accessible from the public cloud into your on-premises cluster. The service set-up by the IBM Secure gateway provides access to the `cluster_kube_apiserver_host` and `cluster_kube_apiserver_port` by the Cloud Host and port from the IBM Secure Gateway.

Unfortunately, no mechanism is available to pass the IBM Secure Gateway Cloud Host and port as part of the **import** command.

This section describes the process that is used to patch the klusterlet-bootstrap configuration secret that is set up on the Kubernetes cluster during the **kubectrl import** process.

After the klusterlet-bootstrap configuration secret is patched and new configuration secrets are re-created, the Kubernetes cluster is successfully imported into the IBM Cloud PAK MCM Hub cluster. The communication to manage the imported cluster is done over the IBM Secure Gateway Cloud Host and port.

Complete the following steps:

1. Patch the configuration secret (klusterlet-bootstrap) on the Kubernetes cluster to be imported to point to the Cloud Host and port from the IBM Secure Gateway.

In the previous paragraph, no mechanism is available when the import command (in this example “**curl -k -H "Authorization: Bearer ffladceedc8f...3a4140794732f732c31" https://icp-console.pokmcm.joelab.ibm.com:443/rcm/v1/clusters/ibm-cloud-app-development/ibm-cloud-app-development/import.yaml | kubectrl apply -f -**”) is run to change the following variables:

- `cluster_kube_apiserver_host: mcm-hub-cluster1.pokmcm.joelab.ibm.com`
- `cluster_kube_apiserver_port: "8443"`

On the server, which is authenticated to the Kubernetes cluster to be imported, run the following command:

```
kubectrl get secret klusterlet-bootstrap -n multicloud-endpoint -o jsonpath={.data.kubeconfig} | base64 -d
```

Example output of the command is shown in Figure 18.

```
root@power-mcm-lb:~  
proxy,infra,master 3d v1.11.0+d4cacc0  
t@power-mcm-lb ~]# kubectl get secret klusterlet-bootstrap -n multiclust  
int -o jsonpath={.data.kubeconfig} | base64 -d  
ersion: v1  
ters:  
uster:  
insecure-skip-tls-verify: true  
server: https://power-mcm-lb.powermcm.cpolab.ibm.com:8443  
me: default-cluster  
exts:  
ntext:  
cluster: default-cluster  
namespace: default  
user: default-auth  
me: default-context  
ent-context: default-context  
: Config  
erences: {}  
s:  
me: default-auth  
er:  
token: eyJhbGciOiJSUzI1NiIsImtpZCI6IiJ9.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZp  
dW50Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWVjb3VudC9uYW1lc3BhY2UiOiJwb3dlci  
dWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3N1Y3JldC5uYW1lIjoicG93ZXItaHVlLWJv
```

Figure 18 Command output example

Make note of the “server” stanza. This stanza is the `cluster_kube_apiserver_host` and `cluster_kube_apiserver_port`. This stanza must be changed to the Cloud Host and Port that was saved in “Step 1: Defining the IBM Secure Gateway” on page 5.

The change uses a bash script that uses the sed tool:

- a. Create a simple bash script: `patch-secret.sh`. Carefully replace the following lines with the suitable values from your cluster. In this example:
  - `cluster_kube_apiserver_host` is replaced with `mcm-hub-cluster1.pokmcm.joelab.ibm.com`
  - `cluster_kube_apiserver_port` is replaced with `8443`
  - Cloud Host is replaced with `cap-sg-prd-4.securegateway.appdomain.cloud`
  - Cloud Port is replaced with `43232`

Figure 19 shows an example of the simple bash script `patch-secret.sh` that must be modified to the values of your environment.

```
#!/bin/bash

echo "Getting secret for the klusterlet-bootstrap and changing
kubeconfig"
newSecret=$(kubectl get secret klusterlet-bootstrap -n
multicloud-endpoint -o jsonpath={.data.kubeconfig} | base64 -d | sed
s/cluster_kube_apiserver_host:cluster_kube_apiserver_port/Cloud
Host:Cloud Port/g | base64 -w 0)

if [ -z "$newSecret" ]
then
    echo "Cloud not get the secret something went wrong"
    exit 1
fi

echo "Patching the secret"
echo -e $(echo $newSecret | base64 -d)
kubectl patch secret klusterlet-bootstrap -n multicloud-endpoint
--type='json' -p ' [{"op": "replace", "path": "/data/kubeconfig", "value":
"${newSecret}"} ]'

echo "Deleteing the secrets to let them recreate"
kubectl get secrets -n multicloud-endpoint
sleep 10
kubectl delete secret endpoint-connmgr-cert-store
endpoint-connmgr-hub-kubeconfig -n multicloud-endpoint
```

*Figure 19 Simple bash script patch-secret.sh*

Figure 20 on page 19 shows the `patch-secret.sh` bash script that was updated with the values of `"cluster_kube_apiserver_host"`, `"cluster_kube_apiserver_port"`, `"Cloud Host"`, and `"Cloud Port"` that are specific to our example.

```
#!/bin/bash

echo "Getting secret for the klusterlet-bootstrap and changing kubeconfig"
newSecret=$(kubectl get secret klusterlet-bootstrap -n multicluster-endpoint
-o jsonpath={.data.kubeconfig} | base64 -d | sed s/
mcm-hub-cluster1.pokmcm.joelab.ibm.com:8443/
cap-sg-prd-4.securegateway.appdomain.cloud:43232/g | base64 -w 0)

if [ -z "$newSecret" ]
then
    echo "Cloud not get the secret something went wrong"
    exit 1
fi

echo "Patching the secret"
echo -e $(echo $newSecret | base64 -d)
kubectl patch secret klusterlet-bootstrap -n multicluster-endpoint
--type='json' -p ' [{"op": "replace", "path": "/data/kubeconfig", "value":
"${newSecret}"} ]'

echo "Deleteing the secrets to let them recreate"
kubectl get secrets -n multicluster-endpoint
sleep 10
kubectl delete secret endpoint-connmgr-cert-store
endpoint-connmgr-hub-kubeconfig -n multicluster-endpoint
```

Figure 20 Updated patch-secret.sh bash script

2. Run the script while authenticated to the Kubernetes cluster to be imported.

This script replaces the current value of the HUB cluster server API with the Cloud Host and Cloud Port of the secure gateway. It also deletes the two endpoint secrets (endpoint-connmgr-cert-store and endpoint-connmgr-hub-kubeconfig), which are automatically recreated by using the new value of the HUB cluster server API, which is stored in the klusterlet-bootstrap.

To run the bash script patch-secret.sh that you created, you must change the permissions to run '+x' and then run the script. This process is done from the command line on the server that is authenticated to the Kubernetes cluster to be imported:

```
chmod +x patch-secret.sh
sh ./patch-secret.sh
```

After a few minutes, the Hub Cluster sees that the newly imported Kubernetes clusters were successfully imported and are in the Ready status. The single pane of glass is used for managing multiple Kubernetes clusters across various clouds (see Figure 21 on page 20).

Name	Namespace	Status	Nodes	Kusterlet version	Kubernetes version	Labels
mike-icp-cluster-ibm-public-cloud1	mike-icp-cluster-ibm-public-cloud1	Ready	4	3.3.0	v1.13.9+icp	cloud-ibm v1
power-hub	power-hub	Ready	1	3.3.0	v1.11.0+d4cacc0/hos	cloud-on-prem v1

Figure 21 All imported clusters are displayed as ready

## Conclusion

The steps that are outlined in “Step 1: Defining the IBM Secure Gateway” on page 5, and “Step 2: Adding a cluster to be managed by IBM Cloud Pak for Multicloud Management Hub cluster” on page 13 are standard operating procedures for [setting up an IBM Secure Gateway](#) and [adding a Kubernetes cluster](#) to be managed by an IBM Cloud Pak for Multicloud Management HUB cluster.

At the time of this writing, no facility was available from the IBM Cloud Pak for Multicloud Management to pass the name of the Secure Gateway public host and port as part of the import cluster command to have the server stanza of the kusterlet-bootstrap secret updated with the wanted value. It is likely that facility eventually is included in future releases of the product.

The latest version of the product was released in August 2020 as version 2.0. Red Hat features a similar offering under the name Red Hat Advanced Cluster Management for Kubernetes.

This document focused on version 1.2 and version 1.3 of the IBM Cloud Pak for Multicloud Management, which were the only releases available as the basis for the procedure setup.

## Author

**Mike Bubel** is a Hybrid Cloud Systems Engineer at the Poughkeepsie Competitive Center (PCC).

## Acknowledgements

The author would like to acknowledge the following contributors for their help in producing this publication:

- ▶ Markus Keppeler on Slack, who provided the framework script to change the endpoint-connmgr.
- ▶ Amine Anouja, who provided a tutorial documentation on Secure Gateway/MCM in earlier MCM code releases.
- ▶ The PCC Team in Poughkeepsie and Raleigh: Tom Ambrosio, Bill Lamastro, Craig Harmon, and Faton “Tony” Avdiu. They provided me the confidence, time, and resources needed to complete this document.

# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.


## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

Redbooks (logo) ®  
IBM®

IBM Cloud®  
IBM Cloud Pak®

The following terms are trademarks of other companies:

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Ansible, CloudForms, OpenShift, Red Hat, are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.







REDP-5615-00

ISBN 073845916x

Printed in U.S.A.

Get connected

