



Preparing to install Cloud Pak for Integration on TechZone AWS, Azure and IBM Cloud Environments

This document will walk you through the steps on how to prepare your environment for the purpose of demonstrating the Cloud Pak for Integration on TechZone.

This will include various steps such as:

How to request a demo environment, Logging into your cloud of choice, and Step by Step guide for running the Automation.

Goals for the Demo:

- Prepare your TechZone environment to demo Cloud Pak for Integration AWS, Azure and IBM Cloud which are supported with ROSA, ARO and ROKS OpenShift clusters.
- Deploying the Cloud Pak for Integration capabilities of your choice like Platform UI, API connect, App Connect Enterprise, MQ and Event Streams.

Prerequisites:

- A valid IBM ID that can be used to access
 - TechZone <https://techzone.ibm.com/>
- Install a code editor, we recommend **VSCode**
 - VS Code <https://code.visualstudio.com/>

Licenses and Entitlements

Details on Cloud Pak for Integration licensing available at <https://www.ibm.com/docs/en/cloud-paks/cp-integration/2022.2?topic=planning-licensing>

Obtaining your IBM entitlement API key

To install Cloud Pak for Integration you are required to have an entitlement key that provides access to the software components. After the necessary entitlements have been granted, use the following steps to download the entitlement key and apply it to the automation:

1. Visit the Container Software Library site - <https://myibm.ibm.com/products-services/containerlibrary>
2. Log in with your IBM ID credentials
3. Assuming the entitlements are in place, you will be presented with an entitlement key. Click "Copy key".
4. This will be used in steps subsequently while deploying cloud pak for Integration



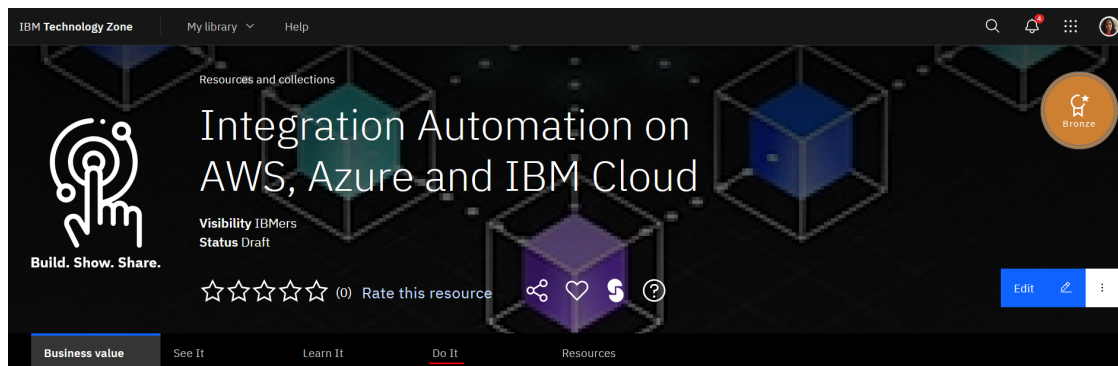
Cloud Pak for Integration Deployment Guide for AWS, Azure, and IBM Cloud

IBM Ecosystem Engineering

Requesting TechZone Environment

1. If you have not already done so, access the TechZone collection for Cloud Pak for Integration Automation for AWS, Azure, and IBM Cloud

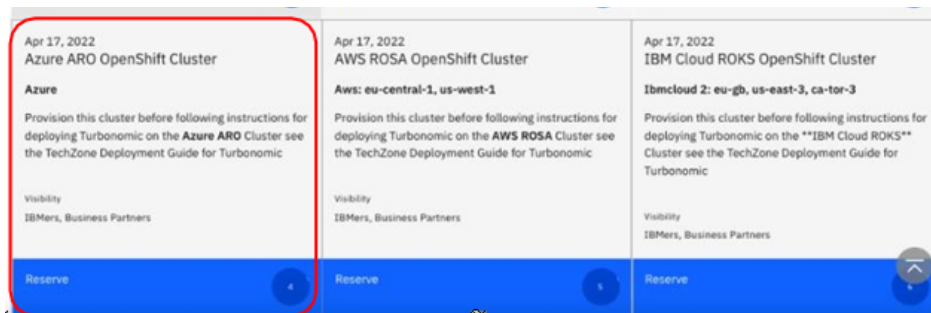
<https://techzone.ibm.com/collection/integration-automation-on-aws-azure-and-ibm-cloud>



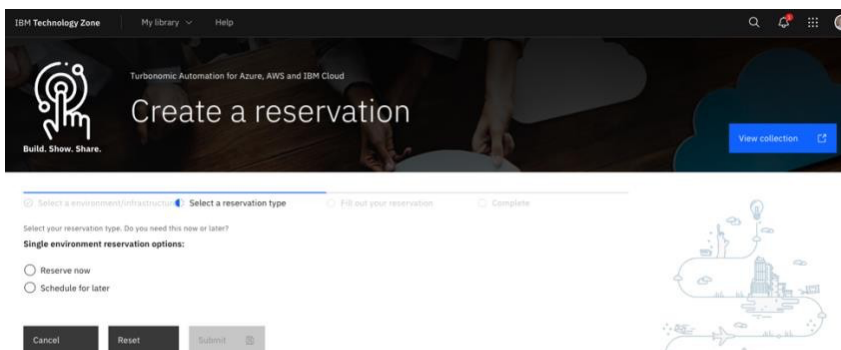
This collection helps with the automation and deployment of Cloud Pak for Integration on AWS, Azure and IBM Cloud. The collection is structured around a set of Journeys to help you to understand how to provision Integration Automation on three cloud platforms AWS, Azure and IBM Cloud.

Reference Architecture refers to the process of combining components, software, hardware, and services into one system. The level of integration increases from the combination of components.

2. The first thing you need to do is to request an access to the demo environment. Click on **Do It** Tab



3. Click on the cloud type of your choice. For example **Azure ARO OpenShift Cluster** environment





Cloud Pak for Integration Deployment Guide for AWS, Azure, and IBM Cloud IBM Ecosystem Engineering

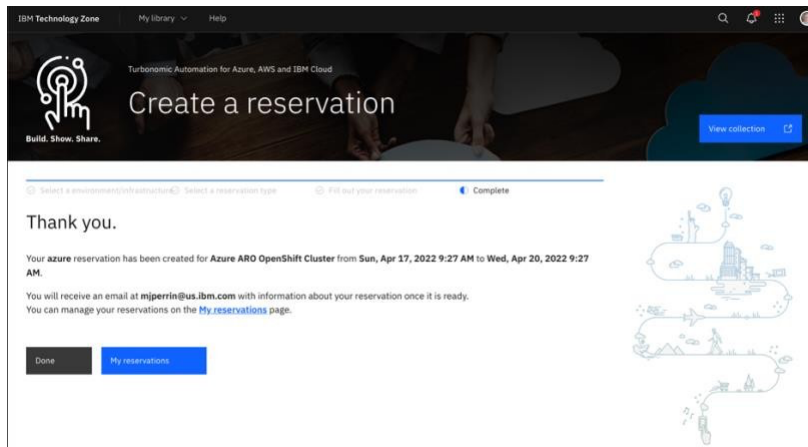
4. Click on the **Reserve now** radio button

5. Complete the Reservation Type form. Please make sure all the fields are complete, you may want to select **Customer Demo** as the Purpose of your request
 - a. As these are real AWS, Azure, and IBM Cloud environment, you will need to use a real customer opportunity value from **IBM Sales Cloud**
<https://w3.ibm.com/w3publisher/ibm-sales-cloud>
6. Enter the **Sales Opportunity Number** as you will not be able to proceed without a valid and live opportunity
7. Select your duration for the environment, and for the size for Cloud Pak for Integration based on the capabilities to be deployed <https://www.ibm.com/docs/en/cloud-paks/cp-integration/2022.2?topic=requirements-compute-resources-development-environments>
8. Click on **Submit** once all the fields are entered correctly
9. Once the **Submit** has completed you will see the following screen with a THANK YOU message to confirm that your reservation is being processed



Cloud Pak for Integration Deployment Guide for AWS, Azure, and IBM Cloud

IBM Ecosystem Engineering

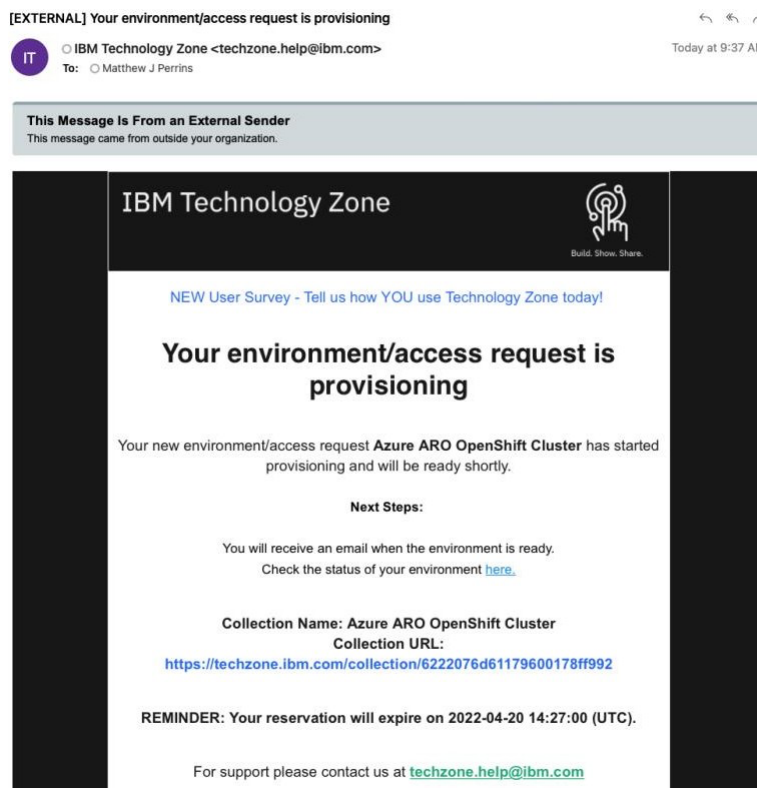


Check your email for confirmation.

Receive Confirmation Email Log in for the First Time

Steps:

1. Once you have reserved the demo environment, you will receive the confirmation email.

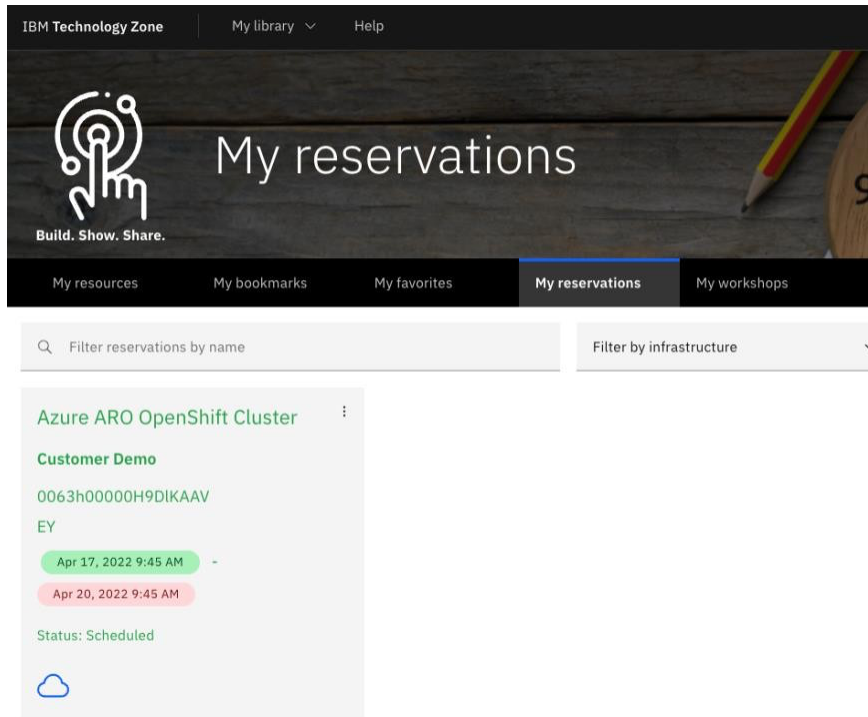


2. It is important that you read the email to follow the instructions on how to log into the Tech Zone Demo Account.



Cloud Pak for Integration Deployment Guide for AWS, Azure, and IBM Cloud IBM Ecosystem Engineering

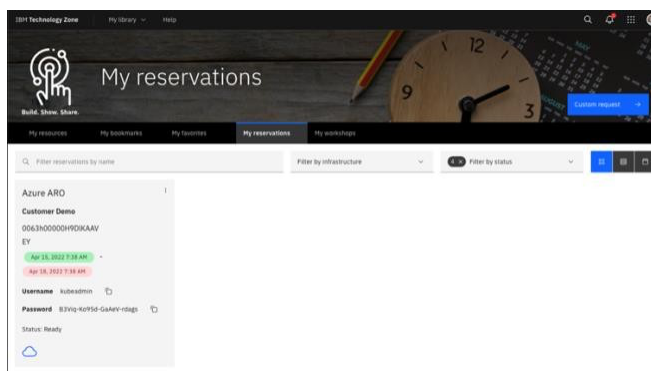
You will receive a second email once your reservation has been processed; Click on **My reservations** to see the status of your reservation.



Log into your TechZone Environment

Steps:

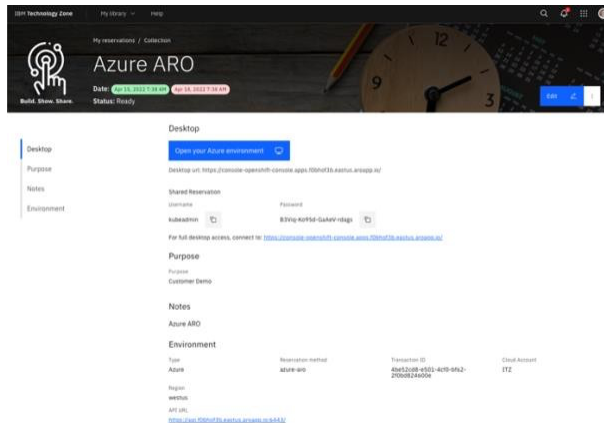
1. Navigate to **TechZone** and **Your Reservations**, click on the reservation you created in the previous steps.



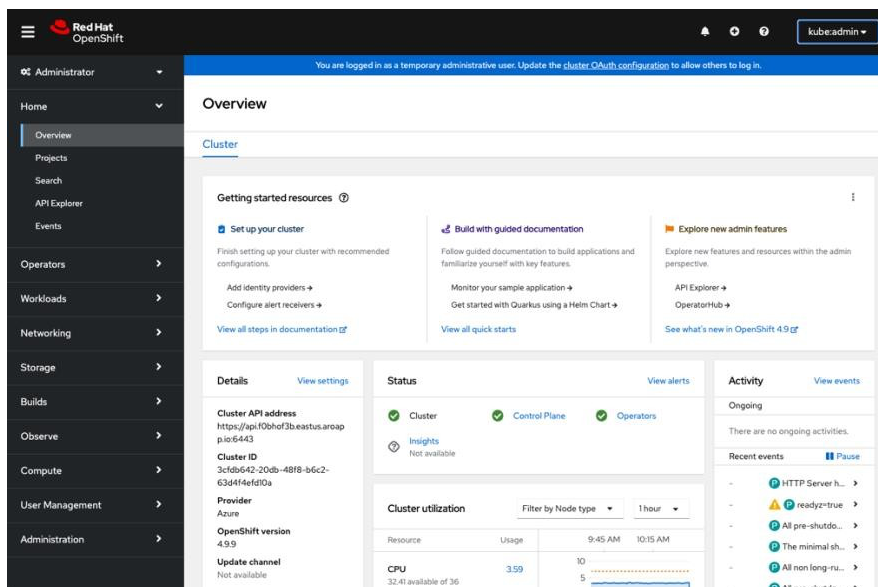
2. You will see the reservation view in detail, copy the **password** and click on the **Open your Azure environment** button.



Cloud Pak for Integration Deployment Guide for AWS, Azure, and IBM Cloud IBM Ecosystem Engineering



3. This will display the OpenShift login screen for the cluster, enter the user ID and password details from the previous reservation screen. Typically, it is **kubeadmin** for the user ID and the system generated password.
4. Once you have logged in you will see the main OpenShift administration screen



5. You have successfully logged into your TechZone environment. You can now start the installation process



Installing Cloud Pak for Integration into your TechZone environment

The installation process will use standard Terraform git repository that has been built using the modules. You need to make Cloud Pak for Integration installation consistent across the three cloud environments AWS, Azure, and IBM Cloud.

Set up the runtime environment

At this time the most reliable way of running this automation is with Terraform on your local machine either through a bootstrapped docker image or Virtual Machine. We provide both a [container image](#) and a virtual machine [cloud-init](#) script that have all the common SRE tools installed.

Supported Runtimes

There are two supported runtimes where the automation is expected to be executed inside of:

1. [Docker Desktop](#) (Container engine)
2. [Multipass](#) (VM)

The Terraform automation can be run from the local operating system, but it is recommended to use either of the runtimes listed above, which provide a consistent and controlled environment, with all dependencies pre-installed. Detailed instructions for downloading and configuring both Docker Desktop and Multipass can be found in [RUNTIMES.md](#)

Set up environment credentials

1. First step is to clone the automation code to your local machine. Run this git command in your favorite command line shell.

```
$ git clone git@github.com:IBM/automation-integration-platform.git
```

2. Navigate into the **automation-integration-platform** folder using your command line.
 - a. The **README.md** has a comprehensive instruction on how to install this into cloud environments other than TechZone. This document focuses on running in a TechZone requested environment.

```
$ cd automation-integration-platform
```

3. The first step is to setup your **credentials.properties** file. This will enable a secure access to your cluster.

First copy the **credentials.template** file to a new file called **credentials.properties**.

```
$ cp credentials.template credentials.properties
```

Next, open the **credentials.properties** file using a code editor.

```
$ code credentials.properties
```



Cloud Pak for Integration Deployment Guide for AWS, Azure, and IBM Cloud

IBM Ecosystem Engineering

Next you need to retrieve login credentials from your OpenShift cluster. From your **OpenShift console** click on top right menu and select **Copy login command** and click on **Display Token**

Your API token is

sha256~uSjFiiAvvc1TBGK4gRhVIbWknF5tVvVxEZ790yyTEno

Log in with this token

```
oc login --token=sha256~uSjFiiAvvc1TBGK4gRhVIbWknF5tVvVxEZ790yyTEno --server=https://api.hr9czz9.eastus.aroapp.io:6443
```

Use this token directly against the API

```
curl -H "Authorization: Bearer sha256~uSjFiiAvvc1TBGK4gRhVIbWknF5tVvVxEZ790yyTEno"
"https://api.hr9czz9.eastus.aroapp.io:6443/apis/user.openshift.io/v1/users/~"
```

[Request another token](#)

[Logout](#)

- Copy the API Token value into the **TF_VAR_cluster_login_token** value and Server URL into the **TF_VAR_server_url** value, only the part starting with https

Log in with this token

```
oc login --token=sha256~mXKmswz5pjackSR-CeUaaILZDN6zW8~WnA9c --server=https://c115-eu-de.containers.cloud.ibm.com:31028
```

- Copy the entitlement key, this can be obtained from visiting the [IBM Container Library](#) and place it in the **TF_VAR_entitlement_key** variable
- By default, the gitops-based deployment will use an in-cluster git repository using Gitea. **Optionally** you can specify GitHub for your GitOps repository, but this is not required.

To enable GitHub:

- Uncomment and set **TF_VAR_gitops_repo_host** to "github.com"
- Uncomment, and add set the **TF_VAR_gitops_repo_username** value to your GitHub User Name
- Uncomment and set your GitHub Personal Access Token value for **TF_VAR_gitops_repo_token**

Configure Storage

Deploying Rook NFS Server on IBM Cloud, Azure & AWS

In Cloud Pak for Integration v2022.2.1, IBM introduced the support for deploying Rook NFS server which would facilitate the deployment of PlatformNavigator using the underlying RWO (ReadWriteOnce) storage. The instruction provided in <https://www.ibm.com/docs/en/cloud-paks/cp-integration/2022.2?topic=ui-deploying-platform-rwo-storage> is automated in this integration-automation asset.

IMPORTANT: Please NOTE. Use this storage in case of PoC/PoTs/Demos. You should use IBM ODF for any real-world customer engagement. These steps will walk you through Rook NFS deployment.



Set up the automation workspace

1. Launch the automation runtime. Ensure the current working directory is 'automation-integration-platform'
 - 1.1. If using *Docker Desktop*, run `./launch.sh`. This will start a container image with the prompt opened in the `/terraform` directory.
 - 1.2. If using *Multipass*, run `mutlipass shell cli-tools` to start the interactive shell, and `cd` into the `/automation/{template}` directory, where `{template}` is the folder you've cloned this repo. Be sure to run `source credentials.properties` once in the shell.
2. Next, we need to set up the working directory for the automation: the *setup-workspace* scripts are responsible for choosing the required module to be deployed on Openshift Cluster. The module we refer here is cater to "GitOps, Storage & Cloud Pak capabilities (PlatformNavigator, APIC, MQ, ACE and EventStreams)".
3. Run the **setup-workspace-with-rook-NFS.sh** script.

```
./setup-workspace-with-rook-NFS.sh [-p {cloud provider}] [-n {prefix name}]
```

where:

- **cloud provider** - the target cloud provider for the deployment (aws, azure, or ibm). This must match the target environment you chose from TechZone

At this stage, We assume you have Openshift Cluster up and running. Following info will help the user in setting up the right workspace.

```
if [OpenShift Cluster is Provisioned on IBM Cloud ]  
./setup-workspace-with-rook-NFS.sh -p ibm
```

```
if [OpenShift Cluster is Provisioned on AWS ]  
./setup-workspace-with-rook-NFS.sh -p aws
```

```
if [OpenShift Cluster is Provisioned on Azure ]  
./setup-workspace-with-rook-NFS.sh -p azure
```

This script will prompt you to select individual Cloud Pak for Integration components that should be deployed. Select the components that you would like deployed into your cluster.

⚠ For a real client engagement or production deployment, you should use the 'odf' storage options in the **setup-workspace-with-odf-or-portworx.sh** script, not the **setup-workspace-with-rook-NFS.sh** script.

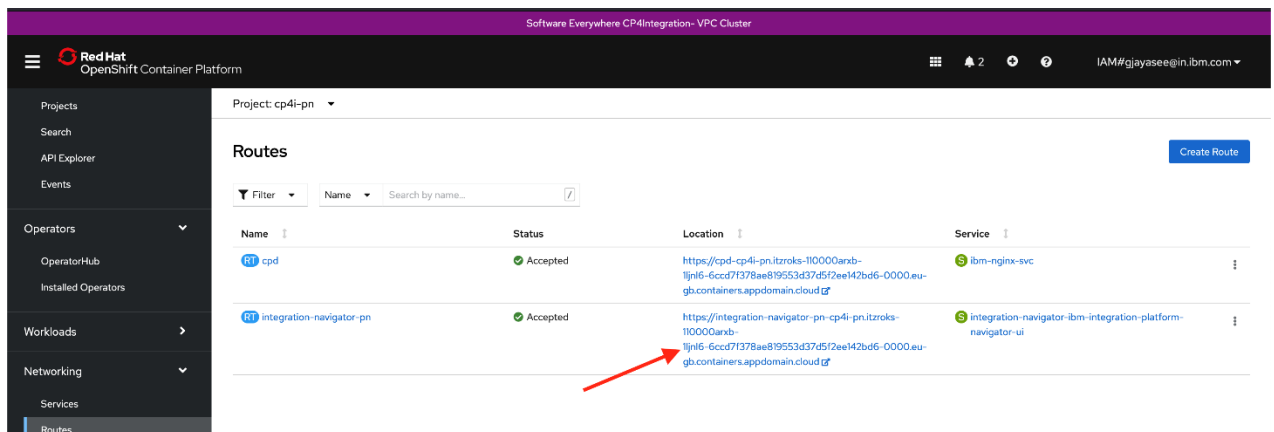
1. This will generate your **terraform.tfvars** file, which contains configuration information for the deployment.
2. Scan through the `terraform.tfvars` file thoroughly and double check the values based on your environment.



3. Save the terraform.tfvars file if you made any changes.

Apply the automation

1. We are now ready to start installing Cloud Pak for Integration. Ensure you are inside the running container or Multipass VM.
2. ⚠ Within the container terminal, change directory to the **/workspaces/current** folder. This folder was populated by the setup-workspace script in the previous step.
3. Run `./apply-all.sh` to kick off the automation. The script will apply each layer in order. Cloud Pak for Integration deployment will run asynchronously in the background and may require up to 90 to 100 minutes to complete.
4. You can check the progress by looking at **Argo CD** on your OpenShift cluster. From the OpenShift user interface, click on the Application menu 3x3 Icon on the header and select **Cluster Argo CD** menu item.
5. Once deployment is complete, go back into the OpenShift cluster user interface and navigate to view Routes for the cp4i-pn namespace. Here you can see the URL to the deployed Platform Navigator instance. Open this url in a new browser window.



6. Navigate to Secrets in the ibm-common-services namespace, and find the platform-auth-idp-credentials secret. Copy the value of password key inside of that secret.
7. Go back to the PlatformNavigator instance that you opened in a separate window. Log in using the username admin with the password copied in the previous step.

THIS CONCLUDES THE CLOUD PAK FOR INTEGRATION INSTALLATION STEPS