



OCP_4.8_and_CP4I_2021.4_installation_guide_V1.4

For

DASHEN BANK

Sudan Street,

Infront of National Bank of Ethiopia,

Addis Ababa, Ethiopia.

By

EIDIKO SYSTEMS INTEGRATORS (PVT) LTD

Suite 1,

MJR Prashant Hills Raidurgam,

Hyderabad, Telangana 500032.

Powered by IBM



Document Information

| | |
|----------------------------------|---|
| Title | OCP_4.8_and_CP4I_2021.4_installation_guide_V1.4 |
| Author | Upender Kuncham |
| File Ref. | |
| Document Template version | 1.10 |

Revision History

| Version | Status | Date | Author / Editor | Details Of Change |
|---------|---------|------|--|--|
| 1.0 | Updated | | Upender Kuncham | Draft Proposal |
| 1.1 | Updated | | ShashiVardhan | Initial Version |
| 1.2 | Updated | | Ahmed Azraq | Ibm Review |
| 1.3 | Updated | | ShashiVardhan Naresh Uppala Gupta Rajesh | Updated the Document based on Comments by Ahmed and added the required fields. |
| 1.4 | Updated | | Ahmed Azraq | Added more review Comments |
| 1.5 | Updated | | ShashiVardhan | Final Documentation |

| | | | | |
|--|--|--|-------------------------------|--|
| | | | Naresh Uppala Gupta Rajesh | |
|--|--|--|-------------------------------|--|



Document Control

Distribution List

| Name | Organization | Role | Email |
|---|--|--------------------------------|--|
| Molla Berhie | Dashen Bank | Senior Manager, IT Projects | molla.berhie@dashenbanks.com |
| | | | |
| | | | |
| Ratan Sripurapu | Eidiko Systems Integrators Pvt.Ltd. | Technical Architect | ratan.sripurapu@eidiko.com |
| Upender Kuncham | Eidiko Systems Integrators Pvt.Ltd. | Project Lead | upender.kuncham@eidiko.com |
| Ibadur Rehman | Eidiko Systems Integrators Pvt.Ltd. | Team Lead | ibadurrehman.shaik@eidiko-india.com |
| T. ShashiVardhan Naresh Uppala Gupta Rajesh | Eidiko Systems Integrators Pvt.Ltd. | Openshift | shashivardhan.tulshannagi@eidiko.com nareshbabu.uppala@eidiko-india.com rajesh.kandregula@eidiko-india.com |
| | | | |
| | | | |



Table of Contents

| | |
|--|----|
| Version History | 1 |
| 1. Introduction..... | 6 |
| 2.OpenShift Container Platform installation overview | 6 |
| 3.System Requirement..... | 7 |
| 3.1Machine List..... | 7 |
| 4.Preparation for OCP Installation..... | 9 |
| 4.1 Creating the user-provisioned infrastructure..... | 9 |
| 4.1.1 Configure DHCP or set static IP addresses on each node..... | 9 |
| 4.1.2 Provision the required load balancers..... | 10 |
| 4.1.3 Configure the ports of your machines..... | 10 |
| 4.1.4 Configure DNS..... | 11 |
| 4.1.5 Downloading the required files for installation..... | 11 |
| 4.1.6 Configuring other required services..... | 13 |
| 5. OCP Installation..... | 17 |
| 5.1 Installing ansible in helper node..... | 17 |
| 5.2 Installation steps..... | 18 |
| 6. Add another worker node..... | 35 |
| 7.Remove worker node..... | 36 |
| 8. Troubleshooting of various issues we are faced during the installation of ocp4.8 cluster..... | 37 |
| 9. OCS configuration..... | 38 |
| 9.1 Installing Local Storage Operator..... | 39 |
| 9.2 Installing OpenShift Container Storage using local storage devices..... | 40 |
| 9.2.1 through web console..... | 40 |
| 9.3 Creating OpenShift Container Storage cluster on bare metal..... | 41 |

| | |
|-----------------------------------|----|
| 10. Configure Image Registry..... | 48 |
|-----------------------------------|----|



| | |
|---|----|
| 11. Installing Cluster Logging..... | 49 |
| 11.1. Installing OpenShift Logging using the web console..... | 50 |
| 11.2. Post-installation tasks..... | 59 |
| 12. Installing and configuring CP4I..... | 60 |
| 12.1 Installing IBM Cloud Pak foundational services by using the console..... | 61 |
| 12.1.1. Prerequisites..... | 61 |
| 12.1.2. Installing the foundational services operator..... | 61 |
| 12.1.3. Setting the hardware profile..... | 62 |
| 12.1.4. Installing foundational services in your cluster..... | 63 |
| 12.1.5. Creating the OperandRequest instance..... | 64 |
| 12.1.6. Verifying the installation..... | 66 |
| 12.2 Installing IBM Cloud Pak foundational services by using the CLI..... | 69 |
| 12.2.1 Installing the Cloud Pak for Integration Operator..... | 69 |
| 12.2.1.1 Deploying the Platform Navigator..... | 70 |
| 12.2.1.2. Creating the Cloud Pak instance..... | 71 |
| 12.2.1.3. Configuring in the Form view..... | 72 |
| 12.2.1.4. Getting the admin password with the OpenShift Console..... | 73 |
| 12.2.1.5. Logging in to Platform Navigator..... | 73 |
| 12.2.2 API Management Deployment..... | 75 |
| 12.2.2.1. Deploying an instance of API Connect..... | 75 |
| 12.2.2.2. Configuring your API Connect instance..... | 77 |
| 12.2.3. Creating App-connect Instance..... | 78 |
| 12.2.3.1. Creating an instance from the IBM Cloud Pak Platform UI..... | 78 |
| 13. Caching of Images..... | 80 |
| 14. Installation of DataPowerGateway..... | 80 |
| 14.1. Steps to follow the Installation Process..... | 80 |

14.2. Mention below points are the major steps to be followed for the installation of DataPower....81



| | |
|--|-----|
| 15. Upgrading DataPower fix pack from 10.0.1.0 to 10.0.1.5..... | 84 |
| 15.1. Procedure..... | 84 |
| 15.2. HighAvailability Case..... | 85 |
| 16. Configuring DataPower API Gateway..... | 89 |
| 16.1 Steps to configure DataPower API Gateway..... | 89 |
| 16.2. Register the gateway service in the API Connect Cloud Manager console..... | 107 |

1. Introduction

This document will guide you the installation of OpenShift cluster, ODF and CP4I, for better understanding incorporating the required URL's during the installation procedure.

2. OpenShift Container Platform installation overview

The OpenShift Container Platform installation program offers you flexibility. You can use the installation program to deploy a cluster on infrastructure that the installation program provisions and the cluster maintains or deploy a cluster on infrastructure that you prepare and maintain.

These two basic types of OpenShift Container Platform clusters are frequently called installer-provisioned infrastructure clusters and user-provisioned infrastructure clusters.

Both types of clusters have the following characteristics:

- Highly available infrastructure with no single points of failure is available by default
- Administrators maintain control over what updates are applied and when

In the Dashen Bank we are followed with a User Provisioned Infrastructure process.

To know more about the OpenShift Installation overview please refer to the following link.

<https://docs.openshift.com/container-platform/4.8/architecture/architecture-installation.html>



3. System Requirement:

3.1 Machine List:

The following list of machines configured in Dashen Bank environment

| Role | Hostname | IP | Physical Machine Mapping | OS Version | vCPU | RAM (GB) | Disk 1 (GB) | Disk 2 Storage (GB) | Comments |
|----------------|---|---------------|--------------------------|------------|------|----------|-------------|---------------------|-------------------------------|
| Bastion Node | install.cloudpakprod.dashenbank.local | 192.168.18.16 | CAT2-1 | Linux | 4 | 16 | 120 | | |
| Bootstrap Node | bootstrap.cloudpakprod.dashenbank.local | 192.168.18.17 | CAT2-1 | RHC OS | 4 | 16 | 100 | | Temp only during installation |
| Master Node 01 | master01.cloudpakprod.dashenbank.local | 192.168.18.21 | CAT2-1 | RHC OS | 4 | 16 | 100 | | |
| Master Node 02 | master02.cloudpakprod.dashenbank.local | 192.168.18.22 | CAT1-1 | RHC OS | 4 | 16 | 100 | | |

| | | | | | | | | | |
|----------------|--|---------------|--------|--------|----|----|-----|--|--|
| Master Node 03 | master03.cloudpakprod.d ashenbank.local | 192.168.18.23 | CAT1_2 | RHC OS | 4 | 16 | 100 | | |
| Storage | storage01.cloudpakprod.d ashenbank.local | 192.168.18.31 | CAT1_1 | RHC OS | 12 | 26 | 100 | | |



Node 01

| | | | | | | | | | |
|-----------------|--|---------------|--------|--------|----|----|-----|--|--|
| Storage Node 02 | storage02.cloudpakprod.d ashenbank.local | 192.168.18.32 | CAT1_2 | RHC OS | 12 | 26 | 100 | | |
| Storage Node 03 | storage03.cloudpakprod.d ashenbank.local | 192.168.18.33 | CAT1_3 | RHC OS | 12 | 26 | 100 | | |
| Worker Node 01 | worker01.cloudpakprod.d ashenbank.local | 192.168.18.41 | CAT1_1 | RHC OS | 16 | 64 | 100 | | |
| Worker Node 02 | worker02.cloudpakprod.d ashenbank.local | 192.168.18.42 | CAT1_2 | RHC OS | 16 | 64 | 100 | | |
| Worker Node 03 | worker03.cloudpakprod.d ashenbank.local | 192.168.18.43 | CAT1_2 | RHC OS | 16 | 64 | 100 | | |

| | | | | | | | | | |
|-----------------|--|---------------|-----------|--------|----|----|-----|--------------------------|--|
| Worker Node 04 | worker04.cloudpakprod.dashenbank.local | 192.168.18.44 | CAT2 1 | RHC OS | 16 | 64 | 100 | | |
| Worker Node 05 | worker05.cloudpakprod.dashenbank.local | 192.168.18.45 | CAT2 1 | RHC OS | 16 | 64 | 100 | | |
| APIC Gateway 01 | datapower01.prod.dashenbank.local | 10.0.20.10/11 | CAT1 1 | RHC OS | 4 | 16 | 120 | Virtual IP is 10.0.20.18 | |
| APIC Gateway 02 | datapower02.prod.dashenbank.local | 10.0.20.12/13 | CAT1 3 | RHC OS | 4 | 16 | 120 | | |
| APIC Gateway 03 | datapower03.prod.dashenbank.local | 10.0.20.14/15 | CAT2 1 | RHC OS | 4 | 16 | 120 | | |

4. Preparation for OCP Installation

4.1 Creating the user-provisioned infrastructure

Before you deploy an OpenShift Container Platform cluster that uses user-provisioned infrastructure, you must create the underlying infrastructure.

Prerequisites:

- 1 Review the [OpenShift Container Platform 4.x Tested Integrations](#) page before you create the supporting infrastructure for your cluster.

4.1.1 Configure DHCP or set static IP addresses on each node.

Configure DHCP (Dynamic Host Configuration Protocol) or set static IP addresses on each node:

DHCP server is configured to provide persistent IP addresses and host names to the cluster machines.

During the initial boot, the machines require either a DHCP server or that static IP addresses be set on each host in the cluster to establish a network connection, which allows them to download their Ignition config files.

If DHCP is configured then you don't need to set static ip's on machines.

To configure the DHCP follow the link

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/sec-dhcp_configuring-server



NOTE-1: If DHCP is not configured then update the static ip's on each machine. Here we have used static IPs.

NOTE-2: In the Dashen Bank DHCP configuration has been done by using the *ansible-playbook* command which ensures Provisioning of the load balancers, Configuring ports and configuring DNS.

4.1.2 Provision the required load balancers.

Before you install OpenShift Container Platform, you must provision two load balancers that meet the following requirements:

API load balancer: Provides a common endpoint for users, both human and machine, to interact with and configure the platform.

Application Ingress load balancer: Provides an Ingress point for application traffic flowing in from outside the cluster.

To know more about Load Balancers please follow the link below.

https://docs.openshift.com/container-platform/4.8/installing/installing_bare_metal/installing-restricted-networks-bare-metal.html

NOTE-1: In the Dashen Bank provisioning load balancers has been done by using the *ansible-playbook* command which ensures DHCP configuration, provisioning of the load balancers, configuring ports and configuring DNS

4.1.3 Configure the ports for your machines.

OpenShift infrastructure components communicate with each other using ports, which are communication endpoints that are identifiable for specific processes or services. Ensure the following ports required by OpenShift are open between hosts, for example if you have a firewall in your environment. Some ports are optional depending on your configuration and usage.

To know more about ports please follow the link below.

https://docs.openshift.com/enterprise/3.1/install_config/install/prerequisites.html



Now then, go to Network Access and check about ports.

NOTE-1: In the Dashen Bank configuring the ports has been done by using the **ansible-playbook** command which ensures DHCP configuration, provisioning of the load balancers,

configuring ports and configuring DNS

4.1.4 Configure DNS:

DNS stands for Domain Name System, translates hostnames or URLs into IP addresses. For example, if we type www.dashenbank.local in browser, the DNS server translates the domain name into its associated ip address. Since the IP addresses are hard to remember all time, DNS servers are used to translate the hostnames like www.dashenbank.local to 173.xxx.xx.xxx. So it makes it easy to remember the domain names instead of its IP address.

To know more about DNS configuration please follow the link below.

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/configuring_and_managing_networking/manually-configuring-the-etc_resolv-conf-file_configuring-and-managing-networking

NOTE-1: In the Dashen Bank configuring the ports has been done by using the **ansible-playbook** command which ensures DHCP configuration, provisioning of the load balancers, configuring ports and configuring DNS

4.1.5 Downloading the required files for installation:

In order to install the OpenShift we need the following files to be downloaded and kept in the installation machine.

OpenShift-installer:

Extract the tar file from the following URL

https://mirror.openshift.com/pub/openshift-v4/x86_64/clients/ocp/4.8.2/openshift-client-linux-4.8.2.tar.gz

Bios files:



Extract the raw.gz file from the following URL

https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.8/4.8.2/rhcos-4.8.2-x86_64-metal.x86_64.raw.gz

For simplicity, rename the downloaded file into bios.raw.gz.

oc and kubectl commands files:

Extract the tar.gz file from the following URL

https://mirror.openshift.com/pub/openshift-v4/x86_64/clients/ocp/4.8.2/openshift-install-linux-4.8.2.tar.gz

Red Hat CoreOS ISO Image:

Get the ISO image file from the following URL

https://mirror.openshift.com/pub/openshift-v4/x86_64/dependencies/rhcos/4.8/4.8.2/rhcos-4.8.2-x86_64-live.x86_64.iso

NOTE: Verify the version which you are downloading before starting further steps. We have faced an issue here as we have gone through 4.9 instead of the 4.8 OCP version.

After installing the above 4 files you need to copy those files into different paths.

Copy **oc**, **kubectl**, & **openshift-installer** into **/usr/local/bin/**

Now, check the version of the oc client by following command,

```
$ oc version
```

Copy **bios.raw.gz** file into **/var/www/html/install**



4.1.6 Configuring other required services:

NTP Configuration:

NTP (Network Time Protocol) is for keeping hosts in sync with the world clock. Time synchronization is important for time sensitive operations, such as log keeping and time stamps, and is highly recommended for Kubernetes, which OpenShift Container Platform is built on.

NTP must be configured in your system.

NOTE:

1. We have configured the local bastion node as NTP server as there is an issue with the main NTP server of Dashen Bank. Once it is resolved we can configure the main NTP server provided by the DasheBank.

2. We have configured the NTP by using below procedure:
step1: install NTP by following command,

firstly, check ntp is configured or not:

```
$ timedatectl
```

If ntp is not configured follow below commands to configure ntp.

```
# timedatectl set-ntp true  
# yum install chrony  
# systemctl enable chronyd -now  
$ vi /etc/chrony.conf
```

Please, see below chrony.conf file that we have used for NTP server configuration:



Use public servers from the pool.ntp.org project.

Please consider joining the pool (<http://www.pool.ntp.org/join.html>).

pool 2.centos.pool.ntp.org iburst

Record the rate at which the system clock gains/losses time.

driftfile /var/lib/chrony/drift

Allow the system clock to be stepped in the first three updates

if its offset is larger than 1 second.

makestep 1.0 3

Enable kernel synchronization of the real-time clock (RTC).

rtcsync

Enable hardware timestamping on all interfaces that support it.

#hwtimestamp *

Increase the minimum number of selectable sources required to adjust

the system clock.

#minsources 2



```
# Allow NTP client access from the local network.
```

```
#allow 192.168.0.0/16
```

```
allow 192.168.18.0/26
```

```
# Serve time even if not synchronized to a time source.
```

```
#local stratum 10
```

```
#Server 172.30.1.59 ## These 2 are the Main NTP server, when it is active, shift to the main  
server. #Server 172.26.14.59  
server 192.168.18.16
```

```
# Specify file containing keys for NTP authentication.
```

```
keyfile /etc/chrony.keys
```

```
# Get TAI-UTC offset and leap seconds from the system tz database.
```

```
leapsetc right/UTC
```

```
# Specify directory for log files.
```

```
logdir /var/log/chrony
```

```
# Select which information is logged.
```

```
#log measurements statistics tracking
```

To check the NTP configuration,

```
$ chronyc sources
```

Now, for more information please follow the below link:

```
https://docs.openshift.com/container-platform/4.8/installing/install\_config/installing  
customizing.html#installation-special-config-chrony\_installing-customizing
```

Firewall settings:

Firewalld is enabled by default, disable it



```
$ systemctl stop firewalld.service
```

```
$ systemctl disable firewalld.service
```

```
$ systemctl stop firewalld.service
```

```
$ systemctl disable firewalld.service
```

If it cannot be disabled, please add the following firewall rules.

53/tcp and 53/udp: DNS server
22623/tcp: OCP machine config server
6443/tcp: OCP API server
80/tcp: OCP ingress http
443/tcp: OCP ingress https

For example, use the commands below to modify firewalld settings to allow DNS server.

```
firewall-cmd --add-port=53/tcp --zone=internal --permanent  
firewall-cmd --add-port=53/tcp --zone=public --permanent  
firewall-cmd --add-port=53/udp --zone=internal --permanent  
firewall-cmd --add-port=53/udp --zone=public -permanent  
firewall-cmd -reload
```

To know more about the firewalld, follow the below link:

```
https://access.redhat.com/documentation/en-us/red\_hat\_enterprise\_linux/7/html/security\_guide/bh\_stopping\_firewalld
```

Named, sshd, HAProxy service must be active:

HAProxy (High Availability Proxy) is open source proxy and load balancing server software. It provides high availability at the network (TCP) and application (HTTP/S) layers, improving speed and performance by distributing workload across multiple servers. HAProxy runs on Linux, FreeBSD and Solaris operating systems. HA-Proxy is used for the non-production environment. For the production and DR environments, Dashen Bank shall provide load balancers.



To install HAProxy:

NOTE-1: In the Dashen Bank installing and configuring HA-Proxy have been done through the ansible script, so this step can be skipped.

```
$ yum install HAProxy  
$ systemctl status HAProxy  
$ systemctl start HAProxy
```

The Secure Shell Daemon application (SSH daemon or sshd) is the daemon program for ssh. This program is an alternative to rlogin and rsh and provides encrypted communications between two untrusted hosts over an insecure network.

To check and start sshd Service:

```
$ systemctl status sshd  
$ systemctl start sshd
```

After completion of the above steps, we must change the hostname, by running following commands and change in the respective fields.

```
$ vi /etc/hosts --- change hostname in hosts  
$ vi /etc/hostname --- change hostname in hostname  
$ reboot
```

To know more about hostname click on following link:

<https://www.cyberciti.biz/faq/rhel-8-change-hostname-computer-name-command/>

5.OCP Installation

5.1 Installing ansible in helper node:

If you use ansible command then no need to configure the prerequisites (DNS, DHCP, Load Balancer, Configure the ports). Ansible command by default configures all the prerequisites.



It will install all prerequisites needed for OpenShift installation.

Hence we install ansible for default configurations.

```
$ yum -y install ansible  
And to check its version:  
$ ansible -version  
$ yum install -y git
```

NOTE: The ansible version of 2.9.23 or above only is accepted.

Problems faced during ansible script installation:

1. The version of OC is mismatched and installed OCP4.9 instead of 4.8. Hence, we downloaded the required version of openshift installer files ie., 4.8.2.

5.2 Installation steps:

Please follow the below steps for the installation

We need helper node directory, we can get it by following git clone command,

```
$ git clone https://github.com/RedHatOfficial/ocp4-helpernode
```

Every OpenShift Container Platform 4.x pattern includes a helper node. The helper node manages a number of services that are used to install, configure, and access an OpenShift cluster.

The services include the following:

- **HA Proxy server** - provides load balancing and proxy services
- **HTTP server** - RHCOS nodes download the ignition config files from the HTTP server
- **DNS server** - provides IP address name resolution
- **NFS server** - provides storage for the OpenShift image repository
- **DHCP server** - provides static IP address assignment based on the RHCOS node's MAC address

NOTE: To know more about helper node, please go through the below link:

```
https://www.ibm.com/docs/en/psw/2.3.2.0?topic=patterns-openshift-container-platform-helper-node
```

Then go to helper node directory and create one vars.yaml file



```
$ cd ocp4-helpernode/
```

```
$ touch vars.yaml
```



Place the below details in the vars.yaml:

```
staticips: true
helper:
  name: "cloudpakprod"
  ipaddr: "192.168.18.16"
  networkifacename: "ens192"
  dns:
    domain: "dashedbank.local"
    clusterid: "cloudpakprod"
    forwarder1: "8.8.4.4"
    forwarder2: "8.8.8.8"
    forwarder3: "192.168.1.1"
  bootstrap:
    name: "bootstrap"
    ipaddr: "192.168.18.17"
  masters:
    - name: "master01"
      ipaddr: "192.168.18.21"
    - name: "master02"
      ipaddr: "192.168.18.22"
    - name: "master03"
      ipaddr: "192.168.18.23"
  workers:
    - name: "storage01"
      ipaddr: "192.168.18.31"
    - name: "storage02"
      ipaddr: "192.168.18.32"
    - name: "storage03"
      ipaddr: "192.168.18.33"
    - name: "worker01"
      ipaddr: "192.168.18.41"
    - name: "worker02"
      ipaddr: "192.168.18.42"
    - name: "worker03"
      ipaddr: "192.168.18.43"
    - name: "worker04"
      ipaddr: "192.168.18.44"
    - name: "worker05"
```

```
ipaddr: "192.168.18.45"
```



Problems faced during installation:

Change the URLs in ocp4-helpernode/vars/main.yml to reflect the target version (OpenShift 4.8). This was configured to OpenShift 4.9 in one of the installations for Dashen Bank and it resulted in installing OpenShift 4.9 instead.

NOTE: Before moving on to further steps, please check whether these static IP addresses are not being used or not.

Now run the ansible-playbook command to configure DNS, DHCP, Load balancer, & default Ports:

```
$ ansible-playbook -e @vars.yaml tasks/main.yml
```

After running the ansible-playbook command all the packages will be downloaded automatically which are required for OpenShift installation.

Now as a sanity check, check the oc version again.

```
$ oc version
```

Output:

```
Client Version: 4.8.29
```

```
Kubernetes Version: v1.21.6+bb8d50a
```

Place helper ip (Installer Node) address in /etc/resolv.conf:

```
$ vi /etc/resolv.conf
```

Search for the cp4i-uat.dashenbank.local &

Add 192.168.18.16(helper IP) in it.

Now, verify DNS is configured or not, for every nodes which are to be in cluster, by following command:



```
$ dig bootstrapname.hostname  
$ dig mastername.hostname  
$ dig workernamae.hostname
```

After using dig command, the output if you get ANSWER as 1 then dns is configured if ANSWER is 0 then dns is not configured and you have an error.

Generating SSH:

Generate an SSH private key and add it to the agent. If you want to perform installation debugging or disaster recovery on your cluster, you must provide an SSH key to both your ssh-agent and the installation program. You can use this key to access the bootstrap machine in a public cluster to troubleshoot installation issues.

Generating ssh key, evaluating it and starting the ssh-agent process as background using below commands.

```
$ ssh-keygen -t rsa -b 4096  
$ eval "$(ssh-agent -s)"  
$ ssh-add
```

Creating an *ocp* directory and a file install-config.yaml in it:

```
$ mkdir workspace  
$ mkdir cloudpakprod  
$ cd /workspace/cloudpakprod/  
$ touch install-config.yaml
```



Place the below details in the install-config.yaml:

```
apiVersion: v1
baseDomain: cloudpakprod.dashenbank.local
compute:
- hyperthreading: Enabled
  name: worker
  replicas: 0 # Must be set to 0 for User Provisioned Installation as worker nodes will be
manually deployed.
controlPlane:
hyperthreading: Enabled
name: master
replicas: 3
metadata:
name: cloudpakprod
networking:
clusterNetwork:
- cidr: 10.128.0.0/14
hostPrefix: 23
networkType: OpenShiftSDN
serviceNetwork:
- 172.30.0.0/16
platform:
none: {}
fips: false
pullSecret: '<add REDHAT licensed Pull-secret>'
sshKey: "<place the ssh key which we have generated in the above steps>"
```

Note: We have faced an issue while using pullsecret which expired and we have started from this point again after clearing all the things we have done with the expired pullsecret.

To get install-config.yaml file and more details about this use below link,

https://docs.openshift.com/container-platform/4.8/installing/installing_bare_metal/installing-restricted-networks-bare-metal.html#installation-bare-metal-config-yaml_installing-restricted-networks-bare-metal



In the above yaml file take pull secret from Red Hat login account by following URL,

```
https://cloud.redhat.com/openshift/install/pull-secret
```

Take the ssh key from /root/.ssh/id_rsa.pub.

Creating manifest and ignition files:

To modify some cluster definition files and manually start the cluster machines, you must generate the Kubernetes manifest and Ignition config files so that the cluster needs to make its machines. The installation configuration file transforms into the Kubernetes manifest. The manifests wrap into the Ignition configuration files, which are later used to create the cluster.

NOTE: To know more about manifest and ignition files click on the following link.

```
https://docs.openshift.com/container-platform/4.8/installing/installing\_bare\_metal/installing-restricted-networks-bare-metal.html#installation-user-infra-generate-k8s-manifest-ignition\_installing-restricted-networks-bare-metal
```

Create the Manifest files

```
$ openshift-install create manifests --dir=<installation_directory>
```

```
$ openshift-install create ignition-configs --dir=<installation_directory>
```



Now, run the following commands to configure NTP.

```
cat << EOF | base64 -w0
server 192.168.18.16 iburst
driftfile /var/lib/chrony/drift
makestep 1.0 3
rtcsync
logdir /var/log/chrony
EOF
```

```
cat << EOF >
/workspace/cloudpakprod/openshift/99_masters-chrony-configuration.yaml
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
labels:
    machineconfiguration.openshift.io/role: master
name: masters-chrony-configuration
spec:
config:
ignition:
config: {}
security:
tls: {}
timeouts: {}
version: 3.1.0
networkd: {}
passwd: {}
storage:
```



files:

- contents:

source: data:text/plain;charset=utf

8;base64,c2VydmVyIDEvLjE3MS41Ny45OCBpYnVyc3QKZHJpZnRmaWxIIC92YXlvbGliL2Nocm9ueS9kcmImdAp
tY\WtIc3RlcCAxLjAgMwpydGNzeW5jCmxvZ2RpciAvdmFyL2xvZy9jaHJvbnnkK

mode: 420

overwrite: true

path: /etc/chrony.conf

osImageURL: ""

EOF



```
cat << EOF >
/workspace/cloudpakprod/openshift/99_workers-chrony-configuration.yaml
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
labels:
    machineconfiguration.openshift.io/role: worker
name: workers-chrony-configuration
spec:
config:
ignition:
config: {}
security:
tls: {}
timeouts: {}
version: 3.1.0
    networkd: {}
    passwd: {}
    storage:
files:
- contents:
source: data:text/plain;charset=utf
8;base64,c2VydmVyIDEwLjE3MS41Ny45OCBpYnVyc3QKZHJpZnRmaWxIIC92YXlvbGliL2Nocm9ueS9kcmlmdAptY
W|tlc3RlcCAxLjAgMwpydGNzeW5jCmxvZ2RpciAvdmFyL2xvZy9jaHJvbnnKK
mode: 420
overwrite: true
path: /etc/chrony.conf
osImageURL: ""
EOF
```

Remove the machines/machineset definitions if any since it is not needed in UPI installation in case it is not removed already.



```
$ rm -f  
/workspace/cloudpakprod/openshift/99_openshift-cluster-api_master-machines-*yaml  
/workspace/cloudpakprod/openshift/99_openshift-cluster-api_worker-machineset-*yaml
```

Create the Ignition files

```
$ openshift-install create ignition-configs --dir=<installation_directory>
```

Then check (auth bootstrap.ign master.ign metadata.json worker.ign) these files or obtained or not in ocp directory.

Give full permissions to bootstrap.ign, master.ign, metadata.json, & worker.ign.

```
$ chmod -R 544 bootstrap.ign master.ign metadata.json worker.ign
```

Then copy these (bootstrap.ign master.ign metadata.json worker.ign) into /var/www/html/install directory.

Go to the browser and check all 6 files are there or not by using the following URL [whatever helper node IP you have when you are installing, replace that IP with the following IP, in our case we have used the following IP]

```
http://192.168.18.16:8080/install
```

Once the firewall is checked, if it is running, stop the firewall.

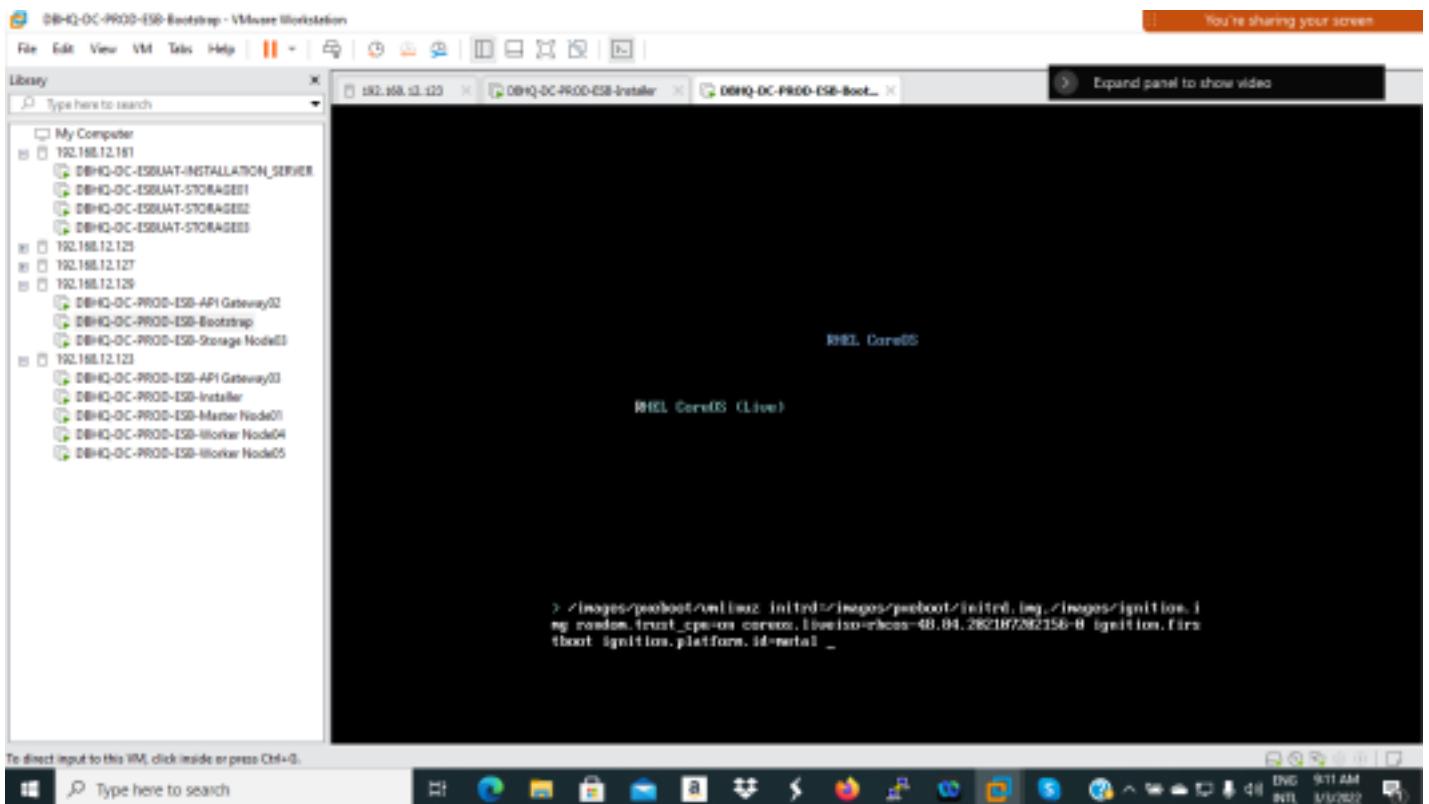
```
$ systemctl status firewalld  
$ systemctl stop firewalld
```

Then go to the bare metal and check the following details:

1. Check whether hyperthreading is enabled and use virtual cores accordingly such as for masternodes and bootstrap $2*2=4$ vCPUs, for worker nodes $8*2=12$ vCPUs, and for the storage nodes $6*2=12$ vCPUs. 2. Now, start the bootstrap machine and to enter into it, right click on the mouse and use the tab at a time so that you'll be in the machine to run the kernel command.



For the reference please find the below picture.



Run the kernels as shown below for the masters first after bootstrap.



```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/bootstrap.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.17::192.168.1.1:255.255.252.0:bootstrap.cloudpakprod.dashenbank.local::no
ne nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/master.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.21::192.168.1.1:255.255.252.0:master01.cloudpakprod.dashenbank.local::no
ne nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/master.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.22::192.168.1.1:255.255.252.0:master02.cloudpakprod.dashenbank.local::no
ne nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/master.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.23::192.168.1.1:255.255.252.0:master03.cloudpakprod.dashenbank.local::no
ne nameserver=192.168.18.16
```

NOTE: To know more about kernel commands click on:

https://docs.openshift.com/container-platform/4.8/installing/installing_bare_metal/installing-restricted-networks-bare-metal.html#creating-machines-bare-metal-restricted-network

Creating the cluster:

To create the OpenShift Container Platform cluster, you wait for the bootstrap process to complete on the



machines that you provisioned by using the Ignition config files that you generated with the installation program.

```
$ openshift-install --dir=/workspace/cloudpakprod/ wait-for bootstrap-complete --log-level=info
```

NOTE: To know more click on the link below.

https://docs.openshift.com/container-platform/4.8/installing/installing_bare_metal/installing-restricted-networks-bare-metal.html#installation-installing-bare-metal_installing-restricted-networks-bare-metal

Now, please run the following commands in bootstrap and masters respectively by connecting to that nodes through *ssh* to check the logs of each while waiting for bootstrapping is completed.

\$ ssh core@192.168.18.17 or by hostname (bootstrap machine)

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

#

```
## $ ssh core@192.168.18.21 or by hostname (master node1)
```

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

#

\$ ssh core@192.168.18.22 or by hostname (master node2)

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

Now, once the installation of ocp is done,

Logging in to the cluster by using the CLI:



To export the kubeadmin credentials,

```
$ export KUBECONFIG=/workspace/cloudpakprod/auth/kubeconfig
```

To verify you can run oc commands successfully using the exported configuration.

```
$ oc whoami
```

NOTE: To know more on how to verify click on:

https://docs.openshift.com/container-platform/4.8/installing/installing_bare_metal/installing-restricted-networks-bare-metal.html#cli-logging-in-kubeadmin_installing-restricted-networks-bare-metal

Approving the certificate signing requests for your machines:

To check if all the nodes are in the ready state or not.

```
$ oc get nodes
```

After successfully installing the ocp4.8 cluster, add remaining nodes such as worker nodes along with storage nodes by following the same procedure on the bare metal. Run the following kernel commands as well.

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.41::192.168.1.1:255.255.252.0:worker01.cloudpakprod.dashenbank.localname::none
nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.42::192.168.1.1:255.255.252.0:worker02.cloudpakprod.dashenbank.localname::none
nameserver=192.168.18.16
```



```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.41::192.168.1.1:255.255.252.0:worker03.cloudpakprod.dashenbank.localname::no
ne nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.44::192.168.1.1:255.255.252.0:worker04.cloudpakprod.dashenbank.localname::no
ne nameserver=192.168.18.1
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.45::192.168.1.1:255.255.252.0:worker05.cloudpakprod.dashenbank.localname::no
ne nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.31::192.168.1.1:255.255.252.0:storage01.cloudpakprod.dashenbank.localname::no
ne nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.32::192.168.1.1:255.255.252.0:storage02.cloudpakprod.dashenbank.localname::no
ne nameserver=192.168.18.16
```

```
coreos.inst.install_dev=sda
coreos.inst.ignition_url=http://192.168.18.16:8080/install/worker.ign
coreos.inst.insecure_url=http://192.168.18.16:8080/install/bios.raw.gz
ip=192.168.18.33::192.168.1.1:255.255.252.0:storage03.cloudpakprod.dashenbank.localname::no
ne nameserver=192.168.18.16
```

To see the CSRs status, if any CSRs are pending then approve them by using below command.



```
$ oc get csr
```

To approve all pending csr's,

```
$ oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}\n{{end}}{{end}}' | xargs  
--no-run-if-empty oc adm certificate approve
```

NOTE: To know more click on,

```
https://docs.openshift.com/container-platform/4.8/installing/installing\_bare\_metal/installing-restricted-networks-bare-metal.html#installation-approve-csrs\_installing-restricted-networks-bare-metal
```

watch the cluster components come online:

```
$ watch -n5 oc get clusteroperators
```

NOTE: Must and should all Clusters are to be in AVAILABLE state are = **True**, PROGRESSING are = **False**, DEGRADED are = **False**

Please check the link below for further information.

```
https://docs.openshift.com/container-platform/4.8/installing/installing\_bare\_metal/installing-restricted-networks-bare-metal.html#installation-operators-config\_installing-restricted-networks-bare-metal
```

```
$ oc get pods --all-namespaces
```

To view all pods

To create new project,

```
$ oc create new-project PROJECT-NAME
```

To get all the projects,



```
$ oc get projects
```

Login into the cluster through Console & CLI for other users:

First you need to update nameserver helper ip on your system in /etc/resolv.conf.

Through CLI:

```
$ oc login https://api.clusterid.eidikointernal.com:6443
```

Through web-console:

```
https://console-openshift-console.apps.clusteris.eidikointernal.com
```

Cleanup:

- 1.Power off and delete the bootstrap VM.
- 2.Remove bootstrap server address from HAProxy configuration (/etc/haproxy/haproxy.cfg), then reload HAProxy.

```
systemctl reload haproxy
```

NOTE: Here, when we tried with different ways to get the main NTP server connection, we lost worker5 as the network team forcefully took the IP of the node and hence it is disconnected from the cluster. Hence we added worker5 as a new worker node into the cluster by the following procedure.

6. Add another worker node

For adding the worker node into the cluster , Please refer to the following links.

<https://access.redhat.com/solutions/4246261> (cluster created less than 24 hours)

<https://access.redhat.com/solutions/4799921> (cluster created more than 24 hours)

If the environment has been running over 24 hours, the certificate should have been regenerated.

Go to the installation folder (e.g. /root/ocp46), retrieve the certificate using the command below.

```
openssl s_client -connect api-int.ocp.example.com:22623 -showcerts
```



The certificate is a block of text starting with "BEGIN CERTIFICATE" and ending at "END CERTIFICATE". Save it in a file, e.g. named "api-int.pem".

Certificate chain

0 s:/CN=api-int.ocp.example.com
i:/OU=openshift/CN=root-ca

-----BEGIN CERTIFICATE-----

```
MIIDVTCCAjBgAwIBAgIJJ1J14NcSgIwDQYJKoZIhvcNAQELBQAwJjESMBAGA1UE  
CxMJB3B1bnNoaWE0MRAwDgYDVQQBwdyb290LNnhMB4XDThwMDIyNjA1MDgxM1oX  
DTMwMDIyMeALMDgxNvow1jEgMB4GA1UEAxMXTKRpLW1udC5vY3AuZXhhbXBsZS5j  
p20wgEiMA0GCSqGSIb3DQEBAQUAA1B0wAwggEKAoIBAQCTQWeV80Bebw9n3Crm  
pVh90Xrd+jqPvbv5DLrcfa2XekXo+vx+pJUq7r4Njd2Jz/MTwmL/P01yUPRSfxi  
wyKNNXingQpeY0tktstatR/tAIEnPHvAq7MD5490Rt11TU++yvsZLrg1zm2JvWlr/f  
cItOUuYkNgFefqtE0ZMbgDjAKve2AiaMaRI1hNDqSTFVIbgRsyVYM822R3CRo11v  
DaMSOqbV9R4C+Ui7iCfcw/Nyb4q25Rpk3oIMTTm3OYdXeC11Yqn9ky2QXjc3BI  
a54V9/0ENAJLla6GzbIkquuN1SERugNgQ8ITE3RCFPXc8HMa/bBaxVEBywVZtHGzN  
EX5jAqNBARGjgYowgYcwEwYDVR01BawCgYIKwTBBQHawEwDAYDVR0TAQH/BAIw  
ADAdBqNVHQ4EFqQU/QxUnuW3ULBaMP1qfuFxq3cjn4wHwYDVR0jBBgwFoAU/QxU  
nUeM3ULBaMP1qfuFxq3cjn4wIgYDVR0RBBswGYIXYKRpLW1udC5vY3AuZXhhbXBs  
ZS5jb20wDQYJKoZIhvcNAQELBQAdgqEBADpKpBa9ZJ+45jH+azfHNnPQvCxshgw  
unBgf7MN9ZEkevPIVgT1bxoa8j78RMRAk+t7+5uEcVXMFR2J4x1QEK6LEUB7jD  
JIruh/6orfkFy39i95h9u8yM8i4cuE+/OyxuN6XXUntv8xnF0V4NIJnPD3XGB  
+7mnarD2+0c9LSn40+MiHEMTYJXpnTMFDsgggJaaREo=YxOKpNLFn9sdVwZdhGU  
aUqRaEZTS5NSvCC8D4Mjx8XIIoAxdjvKJUNedlnVThy/bei019qsyok3cgG5+I  
SGUCeZET4K/LiJ41GCF2f1VeHu1DgAt6kph/dSVivoTaXJ+7tBeZNM=
```

-----END CERTIFICATE-----

Server certificate

subject=/CN=api-int.ocp.example.com

issuer=/OU=openshift/CN=root-ca

Encode the certificate using base64 with the "--wrap=0" option.

```
base64 --wrap=0 ./api-int.pem 1> ./api.int.base64
```

Just in case, make a backup of worker.ign.

```
cp worker.ign worker.ign.bak
```

Open worker.ign with a text editor, replace the value followed by "base64" using the content of api.int.base64.

Finally follow the steps in [Create worker VM](#) to add another worker node. Also don't forget to approve the CSRs.

Make sure that vCPUs and ram are provided according to the requirements.

7. Remove worker node

For removing the worker node from the cluster , Please refer to the following link.

```
https://docs.openshift.com/container-platform/4.8/nodes/nodes/nodes-working.html
```



8. Troubleshooting of various issues we faced during the installation of OCP4.8 cluster

The following are the issues where we faced certain blocking and we troubleshooted accordingly.

1. Pullsecret:

We used a pullsecret that expired and the images couldn't be pulled by the masters and bootstrap from the respective sites during the installation of OCP after running the kernel commands.

We re-installed the cluster from step **5.2** by providing a temporary free licensed pullsecret of TADELE. Later we changed the pullsecret after getting Ateneh's licensed account for pullsecret by following the below procedure.

<https://access.redhat.com/solutions/4902871>

2. Incorrect Version of OCP installer files:

When we downloaded it's a 4.8.2 version of ocp installer files but, as there were previous files of 4.9, the final version overridden it to 4.9. Hence, we have re-initiated the procedure from step **4.1.5** making sure this time by removing the old files and downloading the required installer files.

3. vars.yaml configuration:

During the vars.yaml, we have provided the fully qualified domain name (FQDN), hence we didn't get the hostname as required. So, we re-initiated the procedure from **Then go to the bare metal and check the following details** in the **STEP 5**.

4. NTP Configuration:

The provided NTP server by dashen bank wasn't working and all the nodes are not synchronized during the installation. Hence, we configured the NTP locally ie., in the bastion node and configured it as the NTP server and repeated the steps from **5**.

Steps to configure the NTP please refer to the section of **4.1.6**.

5. Kernel Command FQDN:

During the kernel initialization, we haven't provided the fully qualified domain name (FQDN), hence we didn't get the hostname as required. So, we re-initiated the procedure from **Then go to the bare metal and check the following details** in the **STEP 5**.

6. Internet Connectivity:

Before the final try, we have seen that the internet bandwidth was very low and the masters couldn't be able to form a cluster as they weren't able to download the required files from the respective sites of the internet. Hence, we finally fixed the issue and completed all the cluster setup without any interruptions.



9. OCS configuration

Red Hat OpenShift Container Storage is a provider of agnostic persistent storage for OpenShift Container Platform supporting file, block, and object storage, either in-house or in hybrid clouds. As a Red Hat storage solution, Red Hat OpenShift Container Storage is completely integrated with OpenShift Container Platform for deployment, management, and monitoring.

Red Hat OpenShift Container Storage provides its own documentation library. The complete set of Red Hat OpenShift Container Storage documentation identified below is available at:

```
https://access.redhat.com/documentation/en-us/red\_hat\_openshift\_container\_storage/4.8/html-single/deploying\_and\_managing\_openshift\_container\_storage\_using\_redhat\_openstack\_platform/index#verifying\_openshift-container-storage-deployment\_osp
```

Before going further, we added storage nodes to a label called infra, which is because only storage related tasks are being performed by storage nodes. Use below commands for adding labels for 3 storage nodes.



```
$ oc label node storage01.cloudpakprod.dashenbank.local node
role.kubernetes.io/infra=""  
  
$ oc label node storage01.cloudpakprod.dashenbank.local
cluster.ocs.openshift.io/openshift-storage=""  
  
$ oc adm taint node storage01.cloudpakprod.dashenbank.local
node.ocs.openshift.io/storage="true":NoSchedule
#####
## #####  
  
$ oc label node storage02.cloudpakprod.dashenbank.local node
role.kubernetes.io/infra=""  
  
$ oc label node storage02.cloudpakprod.dashenbank.local
cluster.ocs.openshift.io/openshift-storage=""  
  
$ oc adm taint node storage02.cloudpakprod.dashenbank.local
node.ocs.openshift.io/storage="true":NoSchedule
#####
## #####  
  
$ oc label node storage03.cloudpakprod.dashenbank.local node-
```

```
role.kubernetes.io/infra=""  
  
$ oc label node storage03.cloudpakprod.dashenbank.local
cluster.ocs.openshift.io/openshift-storage=""  
  
$ oc adm taint node storage03.cloudpakprod.dashenbank.local
node.ocs.openshift.io/storage="true":NoSchedule
```

9.1 Installing Local Storage Operator:

Use this procedure to install the Local Storage Operator from the Operator Hub before creating OpenShift Container Storage clusters on local storage devices.

- Log in to the OpenShift Web Console.



- Click **Operators** → **Operator Hub**.
- Search for **Local Storage Operator** from the list of operators and click on it.
- Click **Install**.

Set the following options on the **Install Operator** page:

- 1) Channel as stable-4.8
- 2) Installation Mode as A specific namespace on the cluster
- 3) Installed Namespace as Operator recommended namespace openshift-local-storage.
- 4) Approval Strategy as Automatic
- 5) Click Install.

Verify that the Local Storage Operator shows the Status as Succeeded.

A screenshot of the Red Hat OpenShift Container Platform web console. The left sidebar shows navigation links like 'Administrator', 'Operators', 'Installed Operators' (which is selected), 'Workloads', 'Networking', 'Storage', 'Builds', 'Monitoring', 'Compute', 'User Management', and 'Administration'. The main content area shows the 'Installed Operators' page for the 'openshift-local-storage' project. It lists the 'Local Storage' operator, version 4.8.0-202202070834 provided by Red Hat. The 'Subscription' tab is selected, showing details such as 'Update channel: stable', 'Update approval: Automatic', 'Upgrade status: Up to date', and '0 installing'. Other tabs include 'Details', 'YAML', 'Events', 'All instances', 'Local Volume', 'Local Volume Set', and 'Local Volume Discovery'. The top right corner shows the user is logged in as 'Subadmin'.

9.2 Installing OpenShift Container Storage using local storage devices:

OpenShift Container Platform using local storage devices provides you with the option to create internal cluster resources. This will result in the internal provisioning of the base services, which helps to make additional storage classes available to applications.

9.2.1 Procedure: Through web console :



1. Click **Operators** → **OperatorHub** in the left pane of the OpenShift Web Console.
2. Use **Filter by keyword** text box or the filter list to search for OpenShift Container Storage from the list of operators.
3. Click **OpenShift Container Storage**.
4. On the **OpenShift Container Storage operator** page, click **Install**.
5. On the **Install Operator** page, ensure the following options are selected by default:
 - a. Channel as **eus-4.8**
 - b. Installation Mode as **A specific namespace on the cluster**
 - c. Installed Namespace as **Operator recommended namespace openshift-storage**. If Namespace `openshift-storage` does not exist, it will be created during the operator installation.
 - d. Select **Enable operator recommended cluster monitoring on this namespace** checkbox as this is required for cluster monitoring.
 - e. Select **Approval Strategy** as **Automatic** or **Manual**. Approval Strategy is set to **Automatic** by default.

Verification steps:

- + Verify that **OpenShift Container Storage** Operator shows a green tick indicating successful installation. + Click **View Installed Operators in namespace `openshift-storage`** link to verify that OpenShift Container Storage Operator shows the **Status** as Succeeded on the Installed Operators dashboard.

A screenshot of the OpenShift Web Console interface. The left sidebar shows navigation options like Home, Operators (with OperatorHub and Installed Operators selected), Workloads, Networking, Storage, Builds, Monitoring, Compute, User Management, and Administration. The main content area has a header "Project: openshift-storage" and "You are logged in as a temporary administrative user. Update the cluster's auth configuration to allow others to log in." Below this, it says "Installed Operators > Operator details". A card for "OpenShift Container Storage 4.8.8 provided by Red Hat" is shown, with tabs for Details, YAML, Subscription (selected), Events, All instances, Storage Cluster, Block Pool, Backing Store, Namespace Store, and Bucket Class. Under Subscription details, there are fields for Update channel (set to eus-4.8), Update approval (set to Automatic), and Upgrade status (set to up-to-date). Other sections include Name (ocs-operator), Namespace (openshift-storage), Labels (specular.com.operator.openshift-storage), Created (Feb 18, 2020, 12:10 PM), and a summary section with Installed version (ocs-operator v4.8.8), Starting version (ocs-operator v4.8.8), CatalogSource (redhat-operators, Healthy), and InstallPlan (install-ocs).

9.3 Creating OpenShift Container Storage cluster on bare metal



By following the procedure below, we have installed OCS.

Prerequisites:

Here, we have 3 storage nodes with the same storage type and size attached to each node (has a 1000GB SSD hard drive) to use local storage devices on bare metal.

Procedure:

1. Log into the OpenShift Web Console.

2. Click **Operators** → **Installed Operators** to view all the installed operators.

Ensure that the **Project** selected is **openshift-storage**.

3. Click **OpenShift Container Storage**.

4. Click **Create Instance** link of Storage Cluster.

5. Select **Internal-Attached devices** for the **Select Mode**. By default, Internal is selected. 6. Create a storage cluster using the wizard that includes disk discovery, storage class creation, and storage cluster creation.

7. You are prompted to install the Local Storage Operator if it is not already installed. Click **Install**

8. And install the operator as described in [Installing Local Storage Operator](#).

Discover disks: You can discover a list of potentially usable disks on the selected nodes. Block disks and partitions that are not in use and available for provisioning persistent volumes (PVs) are discovered.

Choose the Select nodes to discover disks from a subset of the available nodes of the **following**:

● **All nodes** to discover disks in all the nodes.

● **Select nodes** to discover disks from a subset of the available nodes.

NOTE: If the nodes selected do not match the OpenShift Container Storage cluster requirement of an aggregated 30 CPUs and 72 GiB of RAM, a minimal cluster will be deployed.

9. Click **Next**.

10. Create **Storage class**

a. Enter the **Local Volume Set Name**.

b. Enter the **Storage Class Name** as **ocs-storagecluster**.

c. The nodes selected for disk discovery in the previous step are displayed in the **Filter Disks By section**.



A screenshot of the Red Hat OpenShift Container Platform web console. The URL in the address bar is https://console-openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/openshift-storage/clusterservicecatalog. The page shows the "Storage Cluster" tab selected under the "Installed Operators" section. A single storage cluster named "aci-storagecluster" is listed, showing it is a StorageCluster kind, has a status of "Phase: Ready", and was last updated on Feb 18, 2022, at 1:46 PM. The left sidebar includes sections for Home, Operators, Operator Hub, Workloads, Networking, Storage, Builds, Monitoring, Compute, and User Management. The top navigation bar shows the user is logged in as a temporary administrative user.

Select the nodes:

Disks on selected nodes to select a subset of the nodes for which you discovered the devices.

Verification steps:

- To verify that the final Status of the installed storage cluster shows as Phase: Ready with a green tick mark.
- Click Operators→Installed Operators→Storage Cluster link to view the storage cluster installation status.

A screenshot of the Red Hat OpenShift Container Platform web console. The left sidebar shows navigation options like Home, Operators, OperatorHub, Workloads, Pods, Deployments, DeploymentConfigs, StatefulSets, Secrets, ConfigMaps, CronJobs, Jobs, DaemonSets, and Replications. The main content area is titled 'Storage Clusters' under the 'Installed Operators' section. It shows a table with one row: 'ocs-storagecluster' of kind 'StorageCluster', status 'Phase: Ready', and last updated on Feb 18, 2022, at 1:46 PM. A 'Create Storage Cluster' button is visible at the top right of the table area.

| Name | Kind | Status | Labels | Last updated |
|--------------------|----------------|--------------|-----------|-----------------------|
| ocs-storagecluster | StorageCluster | Phase: Ready | No labels | Feb 18, 2022, 1:46 PM |

Alternatively, when you are on the Operator Details tab, you can click on the StorageCluster tab to view the status.

- To verify if flexible scaling is enabled on your storage cluster, perform the following steps (for arbiter mode, flexible scaling is disabled):
 1. Click ocs-storage cluster in the Storage Cluster tab.
 2. In the YAML tab, search for the keys `flexibleScaling` in spec section and `failureDomain` in status section. If `flexible scaling` is true and `failureDomain` is set to host, the flexible scaling feature is enabled
- To verify that all components for OpenShift Container Storage are successfully installed, see [Verifying your OpenShift Container Storage installation](#).
- To verify that OpenShift Container Storage is successfully installed, see [Verifying your OpenShift Container Storage installation](#).

Verifying The OpenShift Container Storage cluster is healthy:

To verify that the cluster of OpenShift Container Storage is healthy, follow the steps in the procedure.



Procedure:

1. Click Storage → Overview and click the Block and File tab.
2. In the Status card, verify that *Storage Cluster* and *Data Resiliency* has a green tick mark.
3. In the Details card, verify that the cluster information is displayed.

Verifying that the OpenShift Container Storage specific storage classes exist

To verify the storage classes exists in the cluster:

Procedure:

- Click Storage → Storage Classes from the OpenShift Web Console.
- Verify that the following storage classes are created with the OpenShift Container Storage cluster creation:
 - ocs-storagecluster-ceph-rbd
 - ocs-storagecluster-cephfs
 - openshift-storage.noobaa.io
 - ocs-storagecluster-ceph-rgw



Activities Firefox Feb 26 11:19

https://console-openshift-console.apps.cloudpakprod.dashenbank.local/ocs-dashboards 80%

Centos Wiki Documentation Forums Other Bookmarks

Red Hat OpenShift Container Platform

Administrator

Home

Overview Projects Search API Explorer Events

Operators Operator Hub Installed Operators

Workloads Networking Storage

Overviews Persistentolumes

Details

Service Name: OpenShift Container Storage Cluster Name: ocs-storagecluster Provider: None Mode: Internal Version: ocs-operator:4.6.0

Status

Storage Cluster Data Resiliency

Raw Capacity

Used: 654.2 GiB Available: 2.28 TiB

654.2 GiB Used of 2.93 TiB

Activity

11:05 AM: dash-err-restart...
11:04 AM: csi-ibdplug-prov...
11:04 AM: csi-cephplugin-prov...
11:04 AM: csi-ibdplug-prov...
11:03 AM: csi-cephplugin-prov...
11:03 AM: csi-ibdplug-prov...
11:03 AM: Created container cil...
11:03 AM: Container Image Tagl...
11:03 AM: Error updating fil...

The screenshot shows the Red Hat OpenShift Container Platform interface with the OCS dashboard selected. The left sidebar has sections for Home, Overview, Projects, Search, API Explorer, Events, Operators, Workloads, Networking, and Storage. The Storage section is expanded, showing Persistentvolumes and PersistentVolumeClaims. The main content area shows the OCS dashboard with tabs for Details, Status, Raw Capacity, Used Capacity Breakdown, and Activity. The Status tab shows the Storage Cluster and Data Resiliency as healthy. The Raw Capacity section shows 654.2 GiB used out of 2.93 TiB. The Used Capacity Breakdown chart shows usage by namespace. The Activity log lists numerous events from the previous 24 hours, mostly related to CSI plugins and container images.



Activities Firefox Feb 26 11:25

IBM Pod IBM IBM Inbox S DAT Dash Skype IBM Web Post Conf IBM > + ×

Centos Wiki Documentation Forums Other Bookmarks

Red Hat OpenShift Container Platform kubernetes.kubeadmin

Administrator

Home Operators OperatorHub Installed Operators Workloads Networking Storage Overview PersistentVolumes PersistentVolumeClasses StorageClasses VolumeSnapshots VolumeSnapshotClasses VolumeSnapshotContents Client Plugins

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

StorageClasses

Name Search by name...

| Name | Provisioner | Reclaim policy |
|---------------------------------------|---------------------------------------|----------------|
| localblock | kubernetes.io/no-provisioner | Delete |
| oci-storagecluster-cephfs | openshift-storage.cephfs.ceph.com | Delete |
| oci-storagecluster-ceph-rbd - Default | openshift-storage.rbd.csi.ceph.com | Delete |
| oci-storagecluster-ceph-rgw | openshift-storage.ceph.rook.io/bucket | Delete |
| openshift-storage.rnbd.in | openshift-storage.rnbd.in/rbd | Delete |

Create StorageClass



10. Configure Image Registry

Creating a RWX PVC using Cephfs (rook-cephfs) storage class

```
cat << EOF > /workspace/cloudpakprod/image-registry-storage.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: image-registry-storage
  namespace: openshift-image-registry
spec:
  accessModes:
    - ReadWriteMany
  resources:
  requests:
    storage: 100Gi
  storageClassName: ocs-storagecluster-cephfs
EOF
oc create -f /workspace/cloudpakprod/image-registry-storage.yaml
```

Update the image registry operator configuration. Update the specs to use the created PVC. Update spec.storage : {}, and Update spec.managementState from "Removed" to "Managed".

```
oc edit configs.imageregistry.operator.openshift.io
```



```
Activities Terminal Mar 1 08:29
root@install:/workspace/cloudpakkprod/auth
File Edit View Search Terminal Delete stale/orphan images in Red Hat OpenShift Data Foundation to free up storage
root@install:/workspace/cloudpakkprod/auth
core@master01:~
```

Please edit the object below. Lines beginning with a '#' will be ignored,
and an empty file will abort the edit. If an error occurs while saving this file will be
responded with the relevant failures.

```
#apiVersion: imageregistry.operator.openshift.io/v1
kind: Config
metadata:
  creationTimestamp: "2022-02-17T19:45:00Z"
  finalizers:
  - imageregistry.operator.openshift.io/finalizer
  generation: 4
  name: cluster
  resourceVersion: "13250015"
  uid: 5652937b-828c-499a-a43c-4ce6ec0wa8d6
spec:
  defaultRoute: true
  httpSecret: 42b6daeaafa944488a147cd49693465bcfc0238cf456d6cb3d7dee4da58d9194a9364c1b2881032b666f516c7da58c254311dd7d1c1e667aa984c998ead1086e
  logLevel: Normal
  managementState: Managed
  observedConfig: null
  operatorLogLevel: Normal
  proxy: {}
  replicas: 1
  requests:
    read:
      maxWaitInQueue: 8s
    write:
      maxWaitInQueue: 8s
  rolloutStrategy: RollingUpdate
  storage:
    managementState: Unmanaged
    pvc:
      claim: image-registry-storage
      unsupportedConfigOverrides: null
  "/tmp/oc-edit-019ag.yaml" 93L, 2643C
```

Enable default route for image registry.

```
oc patch configs.imageregistry.operator.openshift.io/cluster --type merge -p '{"spec":{"defaultRoute":true}}'
```

Wait several minutes. Then login to the cluster again.

Login to the registry using podman.

```
podman login default-route-openshift-image-registry.apps.ocp.example.com -u admin -p `oc whoami -t`  
--tls verify=false
```

11. Installing cluster logging

We have installed OpenShift Logging by deploying the OpenShift Elasticsearch and Red Hat OpenShift Logging Operators. The OpenShift Elasticsearch Operator creates and manages the Elasticsearch cluster used by OpenShift Logging. The Red Hat OpenShift Logging Operator creates and manages the components of the logging stack.

The process for deploying OpenShift Logging to OpenShift Container Platform involves:



- Reviewing the [OpenShift Logging storage considerations](#).
- Installing the OpenShift Elasticsearch Operator and Red Hat OpenShift Logging Operator using the OpenShift Container Platform [web console](#) or [CLI](#).

11.1. Installing OpenShift Logging using the web console:

You can use the OpenShift Container Platform web console to install the OpenShift Elasticsearch and Red Hat OpenShift Logging Operators.

Note: If you do not want to use the default Elasticsearch log store, you can remove the internal Elasticsearch logStore and Kibana visualization components from the ClusterLogging custom resource (CR). Removing these components is optional but saves resources. For more information, see [Removing unused components if you do not use the default Elasticsearch log store](#).

Prerequisites:

- Ensure that you have the necessary persistent storage for Elasticsearch. Note that each Elasticsearch node requires its own storage volume.

Elasticsearch is a memory-intensive application. By default, OpenShift Container Platform installs three Elasticsearch nodes with memory requests and limits of 16 GB. This initial set of three OpenShift Container Platform nodes might not have enough memory to run Elasticsearch within your cluster. If you experience memory issues that are related to Elasticsearch, add more Elasticsearch nodes to your cluster rather than increasing the memory on existing nodes.

Procedure:

To install the OpenShift Elasticsearch Operator and Red Hat OpenShift Logging Operator using the OpenShift Container Platform web console:

- 1) Install the OpenShift Elasticsearch Operator:
 - a) In the OpenShift Container Platform web console, click **Operators** → **OperatorHub**.
 - b) Choose **OpenShift Elasticsearch Operator** from the list of available Operators, and click **Install**.
 - c) Ensure that the **All namespaces on the cluster** are selected under **Installation Mode**.
 - d) Ensure that **openshift-operators-redhat** is selected under **Installed Namespace**.

You must specify the openshift-operators-redhat namespace. The openshift-operators namespace might contain Community Operators, which are untrusted and could publish a metric with the same name as an OpenShift Container Platform metric, which would cause conflicts.

- e) Select **Enable operator recommended cluster monitoring on this namespace**.



This option sets the openshift.io/cluster-monitoring: "true" label in the Namespace object. You must select this option to ensure that cluster monitoring scrapes the openshift-operators-redhat namespace.

f) Select **stable-5.x** as the **Update Channel**.

g) Select an **Approval Strategy**.

i) The **Automatic** strategy allows Operator Lifecycle Manager (OLM) to automatically update the Operator when a new version is available.

ii) The **Manual** strategy requires a user with appropriate credentials to approve the Operator update.

h) Click **Install**.

i) Verify that the OpenShift Elasticsearch Operator installed by switching to the **Operators → Installed Operators** page.

j) Ensure that **OpenShift Elasticsearch Operator** is listed in all projects with a **Status of Succeeded**.

A screenshot of the Red Hat OpenShift Container Platform web console. The URL in the address bar is https://console-openshift-console.apps.cloudpakkprod.dashenbank.local/k8s/ns/openshift-operators-redhat/. The page shows the 'Installed Operators' section under the 'Operators' menu. A single operator, 'elasticsearch-operator', is listed. The 'Subscription' tab is selected, showing details like 'Update channel: stable-5.3', 'Update approval: Automatic', and 'Upgrade status: Up to date'. The 'Installed' status is shown as '0 installing'. The 'Details' tab shows the operator's name as 'elasticsearch-operator', namespace as 'openshift-operators-redhat', and labels including 'operator-lifecycle-manager=operator-elasticsearch-operator-redhat'. The 'Logs' tab shows the log output: 'Created at: Fri Jun 21, 2024, 8:40 AM'. The 'Logs' section is currently empty. The 'Logs' tab has a red dot indicating new logs.



2) Install the Red Hat OpenShift Logging Operator:

- a) In the OpenShift Container Platform web console, click **Operators** → **OperatorHub**.
- b) Choose **Red Hat OpenShift Logging** from the list of available Operators, and click **Install**.
- c) Ensure that the **A specific namespace on the cluster** is selected under **Installation Mode**.
- d) Ensure that the Operator **recommended namespace** is **openshift-logging** under **Installed Namespace**.
- e) Select **Enable operator recommended cluster monitoring on this namespace**.

This option sets the openshift.io/cluster-monitoring: "true" label in the Namespace object. You must select this option to ensure that cluster monitoring scrapes the openshift-logging namespace.

- f) Select **stable-5.x** as the **Update Channel**.
- g) Select an **Approval Strategy**.
 - i) The **Automatic** strategy allows Operator Lifecycle Manager (OLM) to automatically update the Operator when a new version is available.
 - ii) The **Manual** strategy requires a user with appropriate credentials to approve the Operator update.
- h) Click **Install**.
- i) Verify that the Red Hat OpenShift Logging Operator installed by switching to the **Operators** → **Installed Operators** page.



A screenshot of the Red Hat OpenShift console interface. The top navigation bar shows the URL as https://console-openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/openshift-logging/operators. The left sidebar has sections for Administrator, Home, Overview, Projects, Search, API Explorer, Events, Operators (with 'Installed Operators' selected), Workloads, Pods, Deployments, DeploymentConfigs, StatefulSets, and Secrets. The main content area is titled 'Project: openshift-logging'. It shows a single installed operator named 'Red Hat OpenShift Logging 5.3.4-0 provided by Red Hat'. Below this, there are tabs for Details, YAML, Subscription (which is selected), Events, All instances, Cluster Logging, and Cluster Log Forwarder. Under 'Subscription details', it shows the update channel as 'stable-53', update approval as 'Automatic', and upgrade status as 'Installed' (Up to date, 0 installing). It also lists the namespace as 'cluster-logging', starting version as 'cluster-logging 5.3.4-0', catalog source as 'redhat-operators' (status 'Healthy'), and install plan as 'install-c7n26'. A note at the top says 'You are logged in as a temporary administrative user. Update the cluster-wide configuration to allow others to log in.'

- j) Ensure that **Red Hat OpenShift Logging** is listed in the **openshift-logging** project with a **Status of Succeeded**.

If the Operator does not appear as installed, to troubleshoot further:

- i) Switch to the **Operators** → **Installed Operators** page and inspect the **Status** column for any errors or failures.
- ii) Switch to the **Workloads** → **Pods** page and check the logs in any pods in the **openshift-logging** project that are reporting issues.



3) Create an OpenShift Logging instance:

- a) Switch to the **Administration** → **Custom Resource Definitions** page.
- b) On the **Custom Resource Definitions** page, click **ClusterLogging**.
- c) On the **Custom Resource Definition details** page, select **View Instances** from the **Actions** menu.
- d) On the **ClusterLoggings** page, click **Create ClusterLogging**.

You might have to refresh the page to load the data.

- e) In the YAML field, replace the code with the following:

```
apiVersion: "logging.openshift.io/v1"
kind: "ClusterLogging"
metadata:
  name: "instance"
  namespace: "openshift-logging"
spec:
  managementState: "Managed"
  logStore:
    type: "elasticsearch"
    retentionPolicy:
      application:
        maxAge: 1d
      infra:
        maxAge: 2d
      audit:
        maxAge: 2d
    elasticsearch:
      nodeCount: 3
      resources:
        limits:
          memory: 16Gi
        requests:
          cpu: 500m
          memory: 16Gi
      storage:
        storageClassName: "ocs-storagecluster-ceph-rbd"
        size: "100G"
        redundancyPolicy: "SingleRedundancy"
      visualization:
        type: "kibana"
        kibana:
          resources:
            limits:
              memory: 736Mi
            requests:
              cpu: 100m
              memory: 736Mi
            replicas: 1
            curation:
```



```
type: "curator"
curator:
resources:
limits:
memory: 256Mi
requests:
cpu: 100m
memory: 256Mi
schedule: "30 3 * * *"
collection:
logs:
type: "fluentd"
fluentd:
resources:
limits:
memory: 736Mi
requests:
cpu: 100m
memory: 736Mi
```

NOTE: Please follow the below instructions for further reference.

1. The name must be an instance.
2. The OpenShift Logging management state. In some cases, if you change the OpenShift Logging defaults, you must set this to Unmanaged. However, an unmanaged deployment does not receive updates until OpenShift Logging is placed back into a managed state.
3. Settings for configuring Elasticsearch. Using the CR, you can configure shard replication policy and persistent storage.



4. Specify the length of time that Elasticsearch should retain each log source. Enter an integer and a time designation: weeks(w), hours(h/H), minutes(m) and seconds(s). For example, 7d for seven days. Logs older than the maxAge are deleted. You must specify a retention policy for each log source or the Elasticsearch indices will not be created for that source.

5.Specify the number of Elasticsearch nodes. See the note that follows this list.

6.Enter the name of an existing storage class for Elasticsearch storage. For best performance, specify a storage class that allocates block storage. If you do not specify a storage class, OpenShift Logging uses ephemeral storage.

7.Specify the CPU and memory requests for Elasticsearch as needed. If you leave these values blank, the OpenShift Elasticsearch Operator sets default values that should be sufficient for most deployments. The default values are 16Gi for the memory request and 1 for the CPU request.

8.Specify the CPU and memory requests for the Elasticsearch proxy as needed. If you leave these values blank, the OpenShift Elasticsearch Operator sets default values that should be sufficient for most deployments. The default values are 256Mi for the memory request and 100m for the CPU request.

9.Settings for configuring Kibana. Using the CR, you can scale Kibana for redundancy and configure the CPU and memory for your Kibana nodes. For more information, see [Configuring the log visualizer](#).

10.Settings for configuring Fluentd. Using the CR, you can configure Fluentd CPU and memory limits. For more information, see [Configuring Fluentd](#).

f. Click **Create**. This creates the OpenShift Logging components, the Elasticsearch custom resource and components, and the Kibana interface.

Now, verify the instance is created by the following commands:



```
$ oc get deployment
```

output:

```
NAME READY UP-TO-DATE AVAILABLE AGE
cluster-logging-operator 1/1 1 1 8d
elasticsearch-cdm-krsbwdf7-1 1/1 1 1 7d23h
elasticsearch-cdm-krsbwdf7-2 1/1 1 1 7d23h
elasticsearch-cdm-krsbwdf7-3 1/1 1 1 7d23h
kibana 1/1 1 1 7d23h
```

The number of primary shards for the index templates is equal to the number of Elasticsearch data nodes..

4) Verify the install:

a) Switch to the **Workloads** → **Pods** page.

b) Select the **openshift-logging** project.

You should see several pods for OpenShift Logging, Elasticsearch, Fluentd, and Kibana similar to the following list:

- i) cluster-logging-operator-cb795f8dc-xkckc
- ii) elasticsearch-cdm-b3nqzchd-1-5c6797-67kfz
- iii) elasticsearch-cdm-b3nqzchd-2-6657f4-wtprv
- iv) elasticsearch-cdm-b3nqzchd-3-588c65-clg7g
- v) fluentd-2c7dg
- vi) fluentd-9z7kk
- vii) fluentd-br7r2
- viii) fluentd-fn2sb
- ix) fluentd-pb2f8
- x) fluentd-zqgqx
- xi) kibana-7fb4fd4cc9-bvt4p



11.2. Post-installation tasks

If you plan to use Kibana, you must [manually create your Kibana index patterns and visualizations](#) to explore and visualize data in Kibana.

If your cluster network provider enforces network isolation, [allow network traffic between the projects that contain the OpenShift Logging operators](#).

Defining Kibana index patterns

An index pattern defines the Elasticsearch indices that you want to visualize. To explore and visualize data in Kibana, you must create an index pattern.

Prerequisites

- A user must have the cluster-admin role, the cluster-reader role, or both roles to view the infra and audit indices in Kibana. The default kubeadmin user has proper permissions to view these indices.

If you can view the pods and logs in the default, kube- and openshift- projects, you should be able to access these indices. You can use the following command to check if the current user has appropriate permissions:

```
$ oc auth can-i get pods/log -n <project>
```

Output:

yes

Elasticsearch documents must be indexed before you can create index patterns. This is done automatically, but it might take a few minutes in a new or updated cluster.

Procedure:

To define index patterns and create visualizations in Kibana:

1. In the OpenShift Container Platform console, click the Application Launcher and select **Logging**.

2. Create your Kibana index patterns by clicking **Management** → **Index Patterns** → **Create index pattern**:



- Each user must manually create index patterns when logging into Kibana the first time to see logs for their projects. Users must create an index pattern named app and use the @timestamp time field to view their container logs.
- Each admin user must create index patterns when logged into Kibana the first time for the app, infra, and audit indices using the @timestamp time field.

3. Create Kibana Visualizations from the new index patterns.

For the further information use below link:

<https://docs.openshift.com/container-platform/4.8/logging/cluster-logging-deploying.html>

12. Installing and configuring CP4I

We are installing IBM Cloud Pak® for Integration on an internet-connected cluster using the OpenShift web console, you can follow these steps to get up-and-running quickly with IBM Cloud Pak for Integration. This procedure covers all phases of installation up to, but not including, deploying instances of capabilities.

This task must be performed by a [Cluster Administrator](#).

CP4I installation:

You can use express installation if your deployment meets the following requirements and criteria:

Cluster requirements

Please find the required information through the below link:

<https://www.ibm.com/docs/en/cloud-paks/cp-integration/2021.4?topic=installing-online-installation-openshift-web-console>

Storage requirements

Please find the required information through the below link:

<https://www.ibm.com/docs/en/cloud-paks/cp-integration/2021.4?topic=installing-online-installation-openshift-web-console>

web-console



12.1 Installing IBM Cloud Pak foundational services by using the console:

You can use the IBM Cloud Pak foundational services operator to install foundational services in your cluster. Use the following procedure to install foundational services version 3.15.x.

12.1.1. Prerequisites:

An OpenShift Container Platform cluster must be installed. For the supported OpenShift Container Platform versions, see [Supported OpenShift versions and platforms](#).

12.1.2. Installing the foundational services operator:

You must complete these tasks from your OpenShift cluster console to install the latest version of foundational services operator.

Create the foundational services catalog source.

- A. Log in to your OpenShift cluster console.
- B. Click the plus icon. You see the **Import YAML** dialog box.
- C. Create the IBM Cloud Pak foundational services CatalogSource.

To create the CatalogSource, paste the following definition in the YAML dialog box:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-operator-catalog
  namespace: openshift-marketplace
spec:
  displayName: ibm-operator-catalog
  publisher: IBM Content
  sourceType: grpc
  image: icr.io/cpopen(ibm-operator-catalog
  updateStrategy:
    registryPoll:
      interval: 45m
```

D.Click **Create**. The catalog source ibm-operator-catalog is created.

E.Verify that the source container is running.

```
oc -n openshift-marketplace get pod | grep ibm-operator-catalog
```



1. Create a namespace for the IBM Cloud Pak foundational services operator. You need a dedicated namespace for the operator.
 - a. From the navigation pane, click **Home > Projects**. The **Projects** page is displayed.
 - b. Click **Create Project**. A **Create Project** area is displayed.
 - c. Enter details of the namespace that you are creating. For example, you can specify **common-service** as the name.
 - d. Click **Create**. The namespace for your IBM Cloud Pak foundational services operator is created.
2. Install the IBM Cloud Pak foundational services operator that you created.
 - a. From the navigation pane, click **Operators > OperatorHub**. The **OperatorHub** page is displayed. b. In the **All Items** field, enter IBM Cloud Pak foundational services. The **IBM Cloud Pak foundational services** operator is displayed.
 - c. Click the **IBM Cloud Pak foundational services** tile. The **IBM Cloud Pak foundational services** window is displayed.
 - d. Click **Install**. You see the **Install Operator** page.
 - e. Set **Installation Mode** to the specific namespace that you created for the IBM Cloud Pak foundational services operator. For example, common-service.
 - f. Set **Update Channel** to the v3 version.
 - g. Set **Approval Strategy** to Automatic.
 - h. Click **Install**. After a few minutes, the **IBM Cloud Pak foundational services** operator and the **Operand Deployment Lifecycle Manager** operator are installed, and you can see these operators on the **Installed Operators** page.

The IBM Cloud Pak foundational services operator creates the ibm-common-services namespace, or the custom namespace that you specified in the configmap, and installs the Operand Deployment Lifecycle Manager operator and the IBM NamespaceScope Operator in the namespace.

Optionally, configure the foundational services webhook. For more information, see [IBM Cloud Pak foundational services webhook](#).

12.1.3. Setting the hardware profile:

Set the hardware requirements profile based on the workloads in your cluster. For more information about the profiles, see [Hardware requirements and recommendations for foundational services](#).

The default profile is starterset. You can change the profile to small, medium, or large, if required.

Note: For versions 3.5.x and 3.6.x, if you change the profile to large or medium, you cannot scale down the profile to small after installation. However, you can switch between large and medium, or scale up a profile after installation.

1. From the navigation pane, click **Home > Search**.
2. From the **Project** drop-down list, select ibm-common-services.
3. From the **Resources** drop-down list, select CommonService.



4. Click the common-service resource.
5. Select the YAML tab.
6. Update the spec.size parameter. We in production provided 'size' parameter as medium.

```
apiVersion: operator.ibm.com/v3
kind: CommonService
metadata:
  name: common-service
  namespace: ibm-common-services
spec:
  size: medium
```

Click **Save**.

12.1.4. Installing foundational services in your cluster

1. From the navigation pane, click **Operators > Installed Operators**.
2. From the **Project** drop-down list, select the ibm-common-services namespace. You see the **IBM Cloud Pak foundational services** operator and the **Operand Deployment Lifecycle Manager** operator. **Note:** You must install and configure foundational services in the ibm-common-services namespace.

The **IBM Cloud Pak foundational services** operator provides the CommonService custom resource. If required, you can customize the service definitions by editing the custom resource. For more information, see [Configuring foundational services by using the CommonService custom resource](#).

1. Obtain the entitlement key that is assigned to your ID.

- a. Log in to [Container Software Library](#) by using the IBMid and password that are associated with the entitled software.
- b. In the **Entitlement keys** section, click **Copy key** to copy the entitlement key to the clipboard.
- c. Copy the key to a safe place. You need the key to create your pull secret.
- d. (Optional) Verify the validity of the key by logging in to the IBM Entitled Registry by using a container tool.

```
docker login cp.icr.io --username cp --password entitlement_key
```

2. Log in to your OpenShift Container Platform cluster by using the oc login command.

3. Create a Docker registry secret named ibm-entitlement-key and add the pull secret to the namespace where you are installing foundational services.



```
oc create secret docker-registry ibm-entitlement-key \
--docker-username=cp \
--docker-password=<entitlement_key> \
--docker-server=cp.icr.io \
--namespace=<target_namespace>
```

These are the variables that you need to designate:

- <entitlement_key> is the value of your entitlement key that you copied earlier from [Container Software Library](#).
- <target_namespace> is the namespace where you are installing foundational services in your cluster. The default namespace that you can use here is ibm-common-services.

12.1.5. Creating the OperandRequest instance:

To create the instance, click **Operand Deployment Lifecycle Manager**, select the **OperandRequest** tab, and click **Create OperandRequest**. On the **Create OperandRequest** page, select a configuration mode: **Form View** or **YAML View**.

Creating the OperandRequest instance by using the YAML View:

Complete these steps to use the YAML to create the OperandRequest instance.

1. On the **Create OperandRequest** page, select **YAML View**.
2. Paste the following content in the YAML editor:



```
apiVersion: operator.ibm.com/v1alpha1
kind: OperandRequest
metadata:
  name: common-service
  namespace: ibm-common-services
  labels:
    app.kubernetes.io/instance: operand-deployment-lifecycle-manager
    app.kubernetes.io/managed-by: operand-deployment-lifecycle-manager
    app.kubernetes.io/name: odlm
spec:
  requests:
    - operands:
        - name: ibm-cert-manager-operator
        - name: ibm-mongodb-operator
        - name: ibm-iam-operator
        - name: ibm-monitoring-grafana-operator
        - name: ibm-healthcheck-operator
        - name: ibm-management-ingress-operator
        - name: ibm-licensing-operator
        - name: ibm-commonui-operator
        - name: ibm-events-operator
        - name: ibm-ingress-nginx-operator(deprecated)
        - name: ibm-auditlogging-operator
        - name: ibm-platform-api-operator
        - name: ibm-zen-operator
  registry: common-service
```

1. In the spec.requests.operands section, retain the operator names of the services that you want to install in your cluster. You can remove the services that you do not want. For a list of foundational services that you can install, see [IBM Cloud Pak foundational services operators and versions](#).
2. Add bindings to access a service. This step is optional. For more information, see [Accessing the services](#).



3. Click **Create**.

You can see the list of services that are installed in your cluster on the **Operators > Installed Operators** page.

12.1.6. Verifying the installation:

Checking all the pods are up and running in ibm-common-services namespace.

A screenshot of a Firefox browser window showing the Red Hat OpenShift Container Platform web interface. The URL is https://console.openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/ibm-common-services/pods. The page displays a table of pods in the 'ibm-common-services' namespace. The table includes columns for Name, Status, Ready, Restarts, Owner, Memory, CPU, and Created. All pods listed are in a 'Running' state, with 'Completed' or 'Running' status. The 'Created' column shows dates ranging from Feb 22, 2022, at 3:10 PM to 4:20 PM. The sidebar on the left shows other workloads like Deployments, StatefulSets, Secrets, ConfigMaps, CronJobs, Jobs, DaemonSets, ReplicaSets, and ReplicationControllers, as well as Networking and Storage sections.



checking all the operators in the ibm-common-services namespace:

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: ibm-common-services

Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the [Understanding Operators documentation](#). Or create an Operator and ClusterServiceVersion using the [Operator SDK](#).

| Name | Managed Namespaces | Status | Last updated | Provided APIs |
|-------------------------------------|---------------------|-------------------------|------------------------|--|
| OpenShift Elasticsearch Operator | All Namespaces | Succeeded Up to date | Feb 26, 2022, 4:10 AM | Elasticsearch Kibana |
| IBM Audit Logging Operator | ibm-common-services | Succeeded Up to date | Feb 27, 2022, 11:56 PM | Common Audit Audit Logging Audit Policy |
| IBM Cloud Pak foundational services | ibm-common-services | Succeeded Up to date | Feb 26, 2022, 4:19 AM | CommonService |
| IBM IAM Operator | ibm-common-services | Succeeded Up to date | Feb 26, 2022, 3:45 AM | Authentication OIDCClientWatcher Pap PolicyController View 3 more... |
| IBM Licensing Operator | ibm-common-services | Succeeded | Feb 26, 2022, 3:51 AM | IBM License Service |



Checking if all the PVCs are bounded.

Activities Firefox ▾ Mar 1 09:12

Custo Chrome Del Insta Pe Onlin OCP Skype Insta Sent OCP Log Log Log UAT + ×

https://console.openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/ibm-common-services/persist 80% Other Bookmarks

Centos Wiki Documentation Forums kube:admin ▾

Red Hat OpenShift Container Platform

Installed Operators

Workloads Networking Storage

Overview PersistentVolumes PersistentVolumeClaims

StorageClasses VolumeSnapshots VolumeSnapshotClasses VolumeSnapshotContents Object Buckets Object Bucket Claims

Builds Monitoring Compute

You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

PersistentVolumeClaims

Create PersistentVolumeClaim

| Name | Status | PersistentVolumes | Capacity | Used | StorageClass |
|------------------------------|--------|---|----------|-----------|--------------------------------|
| PVC mongodbd1r-icp-mongodb-0 | Bound | PV pvc-9212175a-8d84-4ffb-abad-fbe3a69b6e61 | 20 GiB | 430 MiB | SC ocs-storagecluster-ceph-rbd |
| PVC mongodbd1r-icp-mongodb-1 | Bound | PV pvc-76ae7d4c-68bc-487e-9a34-al0ae16c53c6 | 20 GiB | 426.2 MiB | SC ocs-storagecluster-ceph-rbd |
| PVC mongodbd1r-icp-mongodb-2 | Bound | PV pvc-3ed85af3-3625-45e9-bed4-aa6584b570d4 | 20 GiB | 428.8 MiB | SC ocs-storagecluster-ceph-rbd |
| PVC must-gather-pvc | Bound | PV pvc-99fd87dd-0a8b-46ea-a3df-eb9765765b99 | 5 GiB | 19.99 MiB | SC ocs-storagecluster-ceph-rbd |



Check operator status:

1. From the navigation pane, click **Home > Overview**. The cluster status is displayed.
2. Click **Operators** on the status card. The operator statuses are displayed.

The screenshot shows the Red Hat OpenShift Container Platform dashboard. On the left, there's a navigation sidebar with options like Home, Overview, Projects, Search, API Explorer, Events, Operators, Workloads, Networking, and Storage. The 'Overview' tab is selected. In the main content area, there's a 'Getting started resources' section with links to 'Set up your cluster', 'Build with guided documentation', and 'Explore new admin features'. Below this is a 'Status' section with cards for 'Cluster' (green checkmark), 'Control Plane' (green checkmark), 'Storage' (green checkmark), and 'Insights' (yellow warning icon). A tooltip for 'Operator status' is open, showing 'Operators (25 installed)' and 'IBM Automation Foundation ... 1 project installing'. At the bottom of the status section, there are 'Update cluster' and 'View details' buttons. To the right of the status section, there's an 'Activity' section showing recent events like 'The minimal shutdown...' and 'Received signal to term...'. The top of the page shows browser tabs and a URL: <https://console-openshift-console.apps.cloudpakprod.dashenbank.local/dashboards>.

12.2 Installing IBM Cloud Pak foundational services by using the CLI:

For further information, use below link:

<https://www.ibm.com/docs/en/cfps?topic=315-installing-foundational-services-by-using-cli>

12.2.1 Installing the Cloud Pak for Integration Operator

Open the Red Hat OpenShift Operator Hub. Switch to the Administrator perspective, if you're not there already. In the left navigation menu of the OpenShift console, click Operators.

- 1 Search for and select the **IBM Cloud Pak for Integration operator**.
- 2 Click **Install**.
- 3 Select the latest update channel.
- 4 You can choose to install Cloud Pak for Integration in one namespace or for all namespaces on your

cluster. If in doubt, choose the All namespaces on the cluster installation mode, and accept the default Installed Namespace.

5 Select the Automatic approval strategy.

6 Click Install.

Obtaining your entitlement key

Obtain an entitlement key from [IBM Container Library](#). You will use this key in the next section.

Click Get an entitlement key.

1 Copy the entitlement key presented to a safe place so you can use it in [Adding a pull secret to a namespace](#).

2 (Optional) Verify the validity of the key by logging in to the IBM Entitled Registry using a container tool.

```
docker login cp.icr.io --username cp --password <entitlement_key>
```

Creating a namespace for the cp4i installation:

```
$ oc new-project cp4i-prod
```

Adding the pull secret to the namespace:

To install Cloud Pak for Integration in its own namespace, create a docker registry secret named ibm entitlement-key, using the following command. On the third line, replace your_entitlement_key with the value of your entitlement key from the previous section. In the last line, replace target_namespace with the name of your target namespace:

```
oc create secret docker-registry ibm-entitlement-key --docker-username=cp --docker-password=<entitlementkey>
--docker-server=cp.icr.io --namespace=cp4i-prod
```

For further information, please follow this link below.

<https://www.ibm.com/docs/en/cloud-paks/cp-integration/2021.4?topic=installing-online-installation-openshift-web-console>

12.2.1.1 Deploying the Platform Navigator:

To configure and deploy the Platform Navigator:

Open the OpenShift web console.

1 In the navigation panel, click Operators > Installed Operators.

2 Click **IBM Cloud Pak for Integration Platform Navigator**.

3 Click **Platform Navigator**.

4 Click Create Platform Navigator. Make sure the Form view is selected. You will use the form to



configure Platform Navigator.

A screenshot of the Red Hat OpenShift Container Platform web console. The left sidebar shows navigation options like Home, Operators, Workloads, Networking, and Storage. The main content area shows the 'IBM Cloud Pak for Integration Platform Navigator' operator details. It includes tabs for Details, YAML, Subscription (which is selected), Events, and Platform Navigator. The Subscription details section shows an update channel of v5.2, automatic approval, and an upgrade status of 'Up to date'. The operator is installed in the 'cp4i-production' project under the 'ibm-integration-platform-navigator' namespace. Labels include 'operators.coreos.com/ibm-integration-platform-navigator.cp4i-production'. The operator was created on February 22, 2022, at 4:19 PM. The right side of the screen shows other administrative details like installed version (v5.2.0), starting version, catalog source, and install plan.

12.2.1.2. Creating the Cloud Pak instance:

Log into the OpenShift web console with your OpenShift cluster admin credentials.

1. Change **Project** to the desired project/namespace name. Click the drop-down arrow and select your project name from the list. The project name in the example screens below is cp4i.
2. In the navigation panel, click **Operators > Installed Operators**.
3. In the list on the Installed Operators panel, find and click **IBM Cloud Pak for Integration Platform Navigator**.
4. Click the **Platform Navigator** tab.
5. Click **Create PlatformNavigator**. The **Create Platform Navigator** panel opens, and offers two methods for configuring the resource; the **Form view** and the **YAML view**. The **Form view** is selected by default:



12.2.1.3. Configuring in the Form view:

The Form view opens a form that lets you view or modify the resource configuration.

1. Change **Project** to your project (namespace) name. Click the drop-down arrow and select your project name from the list. The project name in the example screen below is cp4i.
2. In the **Name** field, enter a name for the new instance or leave the default.
3. Next to **License**, click the arrow to expand the license acceptance section.
4. Set **License Accept** to **true** if you accept the [Cloud Pak for Integration license agreement](#).

A screenshot of a Firefox browser window showing the Red Hat OpenShift Container Platform web interface. The URL is https://console.openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/cp4i-production/clusterservice. The page shows the configuration for an 'Installed Operator' named 'platform-navigator'. The 'Name' field is set to 'platform-navigator' and the 'Labels' field contains 'app=frontend'. Under the 'License' section, there is a note about accepting the Cloud Pak for Integration license. A checkbox labeled 'accept' is checked, with a tooltip explaining it selects the appropriate License ID. The 'License LI' dropdown is set to 'L-RJON-C7QG3S'. The 'Replicas' field shows a value of '2'. The 'MQ Dashboard' toggle switch is set to 'true'. The left sidebar shows navigation links for Home, Projects, Search, API Explorer, Events, Operators, Workloads, Networking, and Storage.

For License LI, see [Licensing](#) to determine the license. Next, click the expand icon and select the license from the drop-down list.

Specify the Storage class: Click to expand the Storage section, then click to open a list of options for Select Storage Class. Choose a storage class that supports ReadWriteMany (RWX) and allows read and write

access to non-root users. Supported storage providers include ibmc-file-gold-gid, OpenShift Container Storage, Spectrum, and Portworx. In the case of Dashen Bank, use ocs-storagecluster-cephfs storage class.



A screenshot of a Firefox browser window showing the Red Hat OpenShift Container Platform UI. The URL is https://console-openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/cp4i-production/clusterservice. The user is logged in as a temporary administrative user. The left sidebar shows navigation options like Home, Projects, Search, API Explorer, Events, Operators, Workloads, Networking, Storage, and PersistentVolumes. The Operators section has 'Installed Operators' selected. A modal dialog is open for creating a new instance of the Platform Navigator. The 'Project' dropdown is set to 'cp4i-production'. Under 'Replicas', there is a field with the value '2'. Below it, a note says 'The desired number of replica pods for the Platform Navigator.' Under 'MQ Dashboard', a toggle switch is set to 'true'. Under 'Version', a dropdown menu shows '2021.4.1'. Under 'Storage', a 'Storage Class' dropdown is set to 'ocp-storagecluster-cephfs'. There is also an 'Advanced configuration' link. At the bottom of the modal are 'Create' and 'Cancel' buttons. The status bar at the top right shows 'Feb 26 11:45' and the user 'kube:admin'.

Click **Create**.

Your instance of Platform Navigator is added to the list of instances in the current project (namespace).

12.2.1.4. Getting the admin password with the OpenShift Console:

To get the admin password in the OpenShift Console UI:

Log into the OpenShift cluster as a [Cluster Administrator](#).

- 1 To switch to the **ibm-common-services** project, click the arrow to expand the Project list. 2 In the navigation menu, click Workloads to expand the selections, then select Secrets. Open the **platform-auth-idp-credentials** secret.
- 3 At the bottom of the Details page, in the Data section, click Reveal Values to get and copy the value for **admin_password**.
- 4 Save this password to use when logging in to Platform Navigator.

12.2.1.5. Logging in to Platform Navigator:

When the Platform Navigator status changes to Ready, you can access the Platform Navigator UI.

- I Open the OpenShift web console.



- I In the navigation panel, click Operators > **Installed Operators**.
- II In the Installed Operators pane, click **IBM Cloud Pak for Integration Platform Navigator**.
- III Click **Platform Navigator**.

Click the installed instance link from the list, then click the link under Platform Navigator UI.

A screenshot of a Firefox browser window showing the Red Hat OpenShift console interface. The URL is https://console-openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/cp4i-production/clusterservice. The page displays the details of an installed operator named 'platform-navigator' within the 'cp4i-production' project. The left sidebar shows the navigation menu with 'Installed Operators' selected. The main content area shows the operator's name, namespace (cp4i-production), and status (Pending). It also includes sections for Metadata (Cloud Pak UI endpoint), Annotations, Created at (Feb 22, 2022, 4:23 PM), and Owner (No owner).

At the login screen, click **OpenShift authentication** and log in with your OpenShift cluster admin credentials



Log in to IBM Cloud Pak | Administration Hub

Select your authentication type:

[OpenShift authentication](#)

[IBM provided credentials \(admin only\)](#)

If you are unable to log in with your OpenShift credentials, click **IBM provided credentials (admin only)** and:

- For **Username**, enter admin.
- For **Password**, enter the IBM admin password you obtain in [Getting the initial admin password](#).

Platform UI opens. You can now create users and permissions by clicking **Manage users**. For detailed information, see [Adding users in the IBM Cloud Pak Platform UI](#) and the other topics in the [Identity and access management](#) section.

<https://www.ibm.com/docs/en/cloud-paks/cp-integration/2021.4?topic=installing-deploying-cloud-pak-integration-using-openshift-console>

12.2.2 API Management Deployment:

For more information regarding API-connect Deployment, please refer to the link below:

<https://www.ibm.com/docs/en/cloud-paks/cp-integration/2021.4?topic=capabilities-api-management-deployment>

12.2.2.1. Deploying an instance of API Connect:

Complete the following steps to deploy the API Connect capability:



Attention: When you deploy the API Connect operator, the DataPower operator is also deployed. The DataPower Gateway is a required component of API Connect. If you delete the DataPower operator, the API Connect installation will fail.

Log in to the IBM Cloud Pak for Integration instance (IBM Automation Platform home page). Use the authentication method and credentials set by your administrator.

1. Click **IBM Automation> Administration > Integration instances**.
2. On the home page, click **Create instance**.
3. Select the **API management** tile and click **Next**.
4. On the Create an instance for API management page, select “**API Endpoints - Three Node – Production**” deployment type from the below list and click **Next**:
 - **API Endpoints - One Node - Minimum** represents a nonproduction deployment, which installs 1 node for each of the API Connect subsystems.
 - **API Endpoints - Three Node - Production** represents a production deployment, which installs 3 nodes for each of the API Connect subsystems.
 - **API Endpoints - Three Node - Minimum Gateway** represents a production deployment with a minimum requirement of 1 CPU per gateway instead of 4 CPUs.

On the next page, fill in the configuration fields.

A screenshot of a Firefox browser window showing the Red Hat OpenShift Container Platform UI. The URL is https://console.openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/cp4i-production/clusterservice. The page displays the configuration for the 'cp4i-production' project, specifically for the API Connect instance. The left sidebar shows navigation options like Home, Projects, Search, API Explorer, Events, Operators, Workloads, Networking, and Storage. The main content area shows license acceptance (True), license metric (VIRTUAL_PROCESSOR_CORE), license use (production), and license ID (L-RJON-C7BJ42). It also lists deployment profile (n3xc14.m48), product version (10.0.4.0-ifix1-54), storage class (ocs-storagecluster-ceph-rbd), supported endpoint types (None), and analytics subsystem details. A message at the top states: "You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in."

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: cp4i-production

License

Licence acceptance: True

License metric: VIRTUAL_PROCESSOR_CORE

License use: production

License ID: L-RJON-C7BJ42

Deployment profile: n3xc14.m48

Product version: 10.0.4.0-ifix1-54

Storage class: ocs-storagecluster-ceph-rbd

Supported endpoint types: None

Analytics subsystem

Database backups: None

Message queue: Unsupported

Name: None

Storage: Unsupported

For more information, please follow the below link:

<https://www.ibm.com/docs/en/cloud-paks/cp-integration/2021.4?topic=capabilities-api-management-deployment>

Click Create.

12.2.2.2. Configuring your API Connect instance:

Complete basic configuration settings for your instance of API Connect.

1. In the navigation UI, click > Administration > Integration instances.
2. Click the next to the API Connect instance name, and select Cloud Manager.
3. Log in to Cloud Manager with the Common Services user registry, using the same user name and password that you used for logging in to the Cloud Pak for Integration instance (IBM Automation UI). Starting with Cloud Pak for Integration 2021.1, users must log in to the Cloud Pak for Integration platform in order to access the Cloud Manager and API Manager user interfaces in API Connect.



Activities Firefox ▾ Feb 26 11:58

IBM (M Inbox Red apic- IBM C x UAT Dash Skype Insta Insta Post Conf IBM how IBM I > + x

← → C ⌂ https://cpd-cp4i-production.apps.cloudpakprod.dashenbank.local/integration/apis/cp4i-production/apic-prod/admin/ ⌂

Centos Wiki Documentation Forums Other Bookmarks

IBM Cloud Pak | Automation

Cloud Manager cp4i-production | apic-prod admin

Hello!

Welcome to Cloud Manager

The dashboard features a central illustration of three people (two men, one woman) interacting with various cloud components like servers and databases. Below the illustration are four main navigation buttons:

- Configure cloud (gear icon)
- Configure topology (cog icon)
- Manage resources (grid icon)
- Manage organizations (people icon)

For more information, please follow the below link:

<https://www.ibm.com/docs/en/cloud-paks/cp-integration/2021.4?topic=capabilities-api-management-deployment>

The API Connect instance requires further configuration before you can begin creating APIs. For information on what to do next with the API management features, which are provided by API Connect, see [Managing your APIs](#).

12.2.3. Creating App-connect Instance:

Please follow the link below for future reference and information about APP-connect installation in CP4I.

https://www.ibm.com/docs/en/app-connect/containers_eus?topic=instances-dashboard-reference

12.2.3.1. Creating an instance from the IBM Cloud Pak Platform UI:

To create an App Connect Dashboard instance from the IBM Cloud Pak Platform UI, complete the following steps:

From a browser window, log in to the IBM Cloud Pak Platform UI.



1. Tip: You can use the generated URL for a deployed IBM Cloud Pak for Integration Platform Navigator instance

to access the IBM Cloud Pak Platform UI.

The Platform UI home page opens with cards and navigation menu options that provide access to the instances and other resources that you are authorized to create, manage, or use. For information about completing administration tasks (such as user management or platform customization) from this page, see [Platform UI](#) in the IBM Cloud Pak foundational services documentation.

2. From the navigation menu , expand Administration and click Integration instances.
3. From the "Integration instances" page, click Create an instance.
4. To create an App Connect Dashboard instance from the "**Create an instance**" page, click the Integration dashboard tile and click **Next**.
5. From the "**Create an integration dashboard**" page, click a tile to select the production type of instance want to create from the below options:
 - Quick start: Deploy a development dashboard with one replica pod.
 - Production: Deploy a production dashboard with multiple replica pods for resilience and high availability.
6. Click Next. A "UI form" view opens with the minimum configuration required to create the **instance**. 7. Click **Create**. You are redirected to the "Integration instances" page. An entry for the instance is shown in the table with an initial status of Pending, which you can click to check the progress of the deployment. When the deployment completes, the status changes to Ready.

A screenshot of a web browser displaying the Red Hat OpenShift Container Platform OperatorHub interface. The URL in the address bar is https://console-openshift-console.apps.cloudpakprod.dashenbank.local/k8s/ns/cp4i-production/clusterservice. The page shows a form for creating an integration instance. The left sidebar has a navigation menu with items like Administrator, Home, Operators, OperatorHub (which is selected), Installed Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, User Management, and Administration. The main content area shows the following fields:

- Project: cp4i-production
- Owner: No owner
- Channel or version: 11.0.0.14-rl-eus
- Storage:
 - s3 bucket: None
 - Claim Name: None
 - Storage class: ocs-storagecluster-cephfs
- License:
 - Accept: Accept
 - License Li: L-KSBM-C5JEHP
 - License use: CloudPakForIntegrationProduction
- Replicas: 2 pods
- Use Common Services: True
- Log Format: basic
- Log Level: info

Users with the required permission can access this Dashboard instance and [use it to deploy Designer and Toolkit](#)



[integrations to integration servers](#).

For more information, please follow the below link:

https://www.ibm.com/docs/en/app-connect/containers_cd?topic=resources-dashboard-reference

13. Caching of Images

As requested by Dashen Bank to cache images for all the worker nodes incase of any internet interruptions, to maintain cluster in steady state, followed below procedure:

Login into each work node and run the following command to download the ACE integration images.

```
$ podman login -u cp -p <ibm-entitlementkey> cp.icr.io
```

```
$ podman pull cp.icr.io/cp/appc/ace-server
```

```
prod@sha256:891841a6a7e6996bd23fb8910972cd73005d2ffb507a1e8adea4319b322536fa
```

NOTE: Now, login into all the remaining 4 worker nodes and follow the same procedure for caching of images.

14. Installation of DataPower Gateway in DMZ

Initially we have to import the datapower idg ova file(10.0.1.5) in **vmware**. And that file belongs to the production department.

Link to ova file for DataPower 10.0.1.5 is below:

<https://www.ibm.com/support/pages/fix-packs-datapower-gateway-version-1001x#10.0.1.5>

Once to complete the import we have to configure the VMware hardware properties such as RAM size,CPU size and network adapter etc.

We are using 3 DataPower nodes for the high availability scenario in the DMZ zone.

14.1. Steps to follow the installation process:



First we need to give credentials for logging such as login id and password.

After that we have to choose the options yes/no for our WEBGUI DataPower configuration. Now follow the procedure of installing DataPower node by enabling below provided details.

Note: All these steps will be done in the CLI mode itself when you log in into the datapower by using credentials that we are going to use further.

Enable secure backup mode ?

Select yes

Confirm secure backup mode ?

Select yes

Enable common criteria compatibility mode ?

Select no

Please enter a new password

To use your own password

Please re enter the password to confirm

Confirm password again

Do you want to run the install wizard ?

Select yes

14.2. Mention below points are the major steps to be followed for the installation of DataPower:

1. Do you want to configure network interfaces ?

Select yes



Do you have this information?

Select yes

Do you want to configure the eth0 interface ?

Select yes

Do you want to enable DHCP?

Select yes

Do you want to configure the eth1 interface?

Select yes

Do you want to enable DHCP?

Select yes

Do you want to configure the eth2 interface?

Select yes

Do you want to enable DHCP?

Select yes

Do you want to configure the eth3 interface?

Select yes

Do you want to enable DHCP?

Select yes

2. Do you want to configure network services ?

Select yes

Do you want to configure DNS?



Select no

3. Do you want to define a unique system identifier ?

Select yes

--Enter the name and confirm

4. Do you want to configure remote management access ?

Select yes

Do you have this information?

Select yes

Do you want to enable SSH?

Select yes

Enter the local IP address[0 for all]

Select 0

Enter the port number[22]

Select [22] default

Do you want to enable WEBGUI access?

Select yes

Enter the port number

Select 9090(default)

Note : if you want to configure the backup user for resetting the password select "yes" for step 5 otherwise select "no"

5. Do you want to configure user account that can reset password

Select yes

6. Do you want to configure the RAID array?

Select no



7. Do you want to review the current configuration?

Select yes

`Do you want to save the current configuration?`

Select yes

`Override previously saved configuration?`

Select yes

`ctrl+G -->openInput & ctrl+alt-->closeInput`

- And to view all configuration of datapower gateway just give a command

`show int`

- It shows the information of the network and ip address.
- Later we have to send the request from any browser with an ip address, port number along with security:
`https:10.0.20.10:9090`
- After that we need to access a license agreement for that. Once complete credentials will get WEBGUI datapower gateway interface.

NOTE: As we have discussed and agreed on the DataPower Highly availability scenario, repeat the installation procedure for 2 more nodes and install them.

15. Upgrading DataPower fix pack from 10.0.1.5 to 10.0.4.0 & High Availability scenario

Upgrading DataPower fix pack from 10.0.1.5 to 10.0.4.0 as we have API-connect, which is installed in the OpenShift, is of version 10.0.4.0.

15.1. Procedure:

Prerequisites:



1. Here we want Idg DataPower script file for the desired version of (idg 10.0.4.0)

2. Internet connectivity

Procedure to upgrade DataPower fix pack :

- Login into datapower idg webgui
- Go to the default domain
- From the control panel select system control.
- In system control select boot image, here we need to upload data power upgrade fix pack script file(10.0.1.5).
- Then we have to accept the license and click on the boot image.

NOTE: As we have discussed and agreed on the DataPower Highly availability scenario, repeat the upgradation procedure for 2 more nodes and upgrade them.

15.2. High Availability Case:

Now, to set up highly availability case with the installed 3 DataPower nodes, follow the below procedure:

Step 1: Create a Virtual IP address having connectivity with all three DataPower nodes.

Step 2: Login into one of the WEBGUI interface of the DataPower nodes ie., <https://10.0.20.10:9090> by placing this in the browser.

Step 3: Navigate to Network → Ethernet Interface, Click on the eth0



Activities Firefox Mar 1 10:34

IBM Data dashenba Log In IBM Cloud IBM Cloud ibm de Config The se How t The Q Error Using Error dashenba + ×

← → ⌂ ⌂ https://10.0.20.10:9090/list/EthernetInterface ☆

Centos Wiki Documentation Forums Other Bookmarks

DataPower Gateway admin @ 10.0.20.10:9090 3/1/2022, 5:33:39 AM (EST) Domain: default ▾ Save Configuration Logout IBM

Control Panel
Blueprint Console (deprecated)

Search

>Status Services Network

Interface

- Ethernet Interface
- VLAN Interface
- Link Aggregation Interface
- Network Settings
- Host Alias
- DNS Settings
- NTP Service

Management

Other

Administration Objects

Firmware: IDG.10.0.4.0
Build: 337832
Delivery type: Continuous delivery
IBM DataPower Gateway
Copyright IBM Corporation 1999-2021
View License Agreement

The running configuration of the device contains unsaved changes. [Review changes.](#)

Configure Ethernet Interface

[Refresh List](#)

| Name | Status | Op-State | Logs | Administrative state | Comments |
|------|--------|----------|------|----------------------|----------|
| eth0 | saved | up | Logs | enabled | |
| eth1 | saved | up | Logs | enabled | |
| eth2 | saved | down | Logs | disabled | |
| eth3 | saved | down | Logs | disabled | |

Add



Step 4: Navigate to the “Standby Control”

The screenshot shows the DataPower Gateway interface in Firefox. The URL is https://10.0.20.10:9090/configure/EthernetInterface/eth0. The page title is "Configure Ethernet Interface". The "Standby control" tab is selected. A message at the top says "The running configuration of the device contains unsaved changes. Review changes." The left sidebar shows a tree view of network objects under "Network". The main form has fields for "Group number" (1), "Primary virtual IP address" (10.0.20.18), "Priority" (100), and "Secondary virtual IP addresses" (empty). Buttons include "Apply", "Cancel", "Delete", "Undo", "Start packet capture", "Stop packet capture", "Disable hardware offload", and "Yield standby".

Step 5: Configure the virtual IP and set priority based on the performance of the nodes.

Step 6: Apply the changes and save the configuration.

NOTE: Now, repeat the same procedure for the remaining 2 DataPower nodes and place the priority level accordingly.

After configuring the remaining nodes, now navigate to the “standby status” and verify all the nodes.

Activities Firefox Mar 1 10:42

IBM Data dashenba Log In IBM Cloud IBM Cloud ibm da Config The se How t The Q Error Using Error dashenba + x

Centos Wiki Documentation Forums Other Bookmarks

DataPower Gateway admin @ 10.0.20.10:9090 3/1/2022, 5:42:07 AM (EST) Domain: default Save Configuration Logout IBM

Control Panel Blueprint Console (deprecated)

Search

Status View Logs Main Configuration System IP-Network

- DNS Cached Hosts
- DNS Search Domains
- DNS Servers
- DNS Static Hosts
- IGMP Status
- IP address status
- IP Multicast Status
- Link Aggregation Member Status
- Link Aggregation Status
- Link status
- Load Balancer Status
- ND Cache Table
- Network Interfaces
- Routing Table
- Rx Packet Throughput
- Rx Throughput
- Standby Status
- TCP Port Status
- TCP Port Summary
- Tx Packet Throughput

The running configuration of the device contains unsaved changes. Review changes.

Standby Status

Refresh Status Help

| ifIndex | Type | Name | Group | Virtual IP address | Priority | State | Preemption state | VIP owner | Self-balancing | Distribution algorithm |
|---------|----------|------|-------|--------------------|----------|--------|------------------|------------|----------------|------------------------|
| 4 | Ethernet | eth0 | 1 | 10.0.20.18 | 100 | Listen | off | 10.0.20.14 | | |

Activities Firefox Mar 1 10:43

IBM Data dashenba Log In IBM Cloud IBM Cloud ibm da Config The se How t The Q Error Using Error dashenba + x

Centos Wiki Documentation Forums Other Bookmarks

DataPower Gateway admin @ dashenbank_prod_02 3/1/2022, 5:42:43 AM (EST) Domain: default Save Configuration Logout IBM

Control Panel Blueprint Console (deprecated)

Search

Status View Logs Main Configuration System IP-Network

- DNS Cached Hosts
- DNS Search Domains
- DNS Servers
- DNS Static Hosts
- IGMP Status
- IP address status
- IP Multicast Status
- Link Aggregation Member Status
- Link Aggregation Status
- Link status
- Load Balancer Status
- ND Cache Table
- Network Interfaces
- Routing Table
- Rx Packet Throughput
- Rx Throughput
- Standby Status
- TCP Port Status
- TCP Port Summary
- Tx Packet Throughput

Standby Status

Refresh Status Help

| ifIndex | Type | Name | Group | Virtual IP address | Priority | State | Preemption state | VIP owner | Self-balancing | Distribution algorithm |
|---------|----------|------|-------|--------------------|----------|---------|------------------|------------|----------------|------------------------|
| 4 | Ethernet | eth0 | 1 | 10.0.20.18 | 100 | Standby | off | 10.0.20.14 | | |

The screenshot shows the DataPower Gateway Control Panel. The left sidebar contains a search bar and links to Status, View Logs, Main, Configuration, System, and IP-Network sections. The IP-Network section is expanded, showing sub-links for DNS Cached Hosts, DNS Search Domains, DNS Servers, DNS Static Hosts, IGMP Status, IP address status, IP Multicast Status, Link Aggregation Member Status, Link Aggregation Status, Link status, Load Balancer Status, ND Cache Table, Network Interfaces, Routing Table, Rx Packet Throughput, Rx Throughput, Standby Status, TCP Port Status, TCP Port Summary, and Tx Packet Throughput.

The main content area is titled "Standby Status" and contains a table with the following data:

| ifIndex | Type | Name | Group | Virtual IP address | Priority | State | Preemption state | VIP owner | Self-balancing | Distribution algorithm |
|---------|----------|------|-------|--------------------|----------|--------|------------------|------------|----------------|------------------------|
| 4 | Ethernet | eth0 | 1 | 10.0.20.18 | 100 | Active | off | 10.0.20.14 | | |

16. Configuring DataPower API Gateway

- Configuring a gateway service with a single gateway server. The lowest-level configuration objects are created first, then used in other configuration objects.
- Adding gateways to configure a peering environment is similar to creating the first gateway, and is recommended for resiliency in a production environment.
- A minimum of three gateway servers in a gateway service is recommended for high availability. •

See below link for more information about configuring additional gateways for peering.

<https://www.ibm.com/docs/en/datapower-gateway/10.0.x?topic=gateway-peering>

16.1. Steps to Configure DataPower API Gateway:

- Open the DataPower WebGUI interface. Most of the configuration procedure is done in the DataPower WebGUI interface, not in the Blueprint Console.
 - Enable the XML management interface in the default domain. The XML management interface is optional for DataPower API Gateway. If enabled, this interface allows you to send status and configuration requests to the DataPower appliance through a standard SOAP interface, using SOAP messages.
 - Search for the XML management interface in the navigation search bar, and select it.



- b. Set the Administrative state to enabled.
- c. You can specify a different port number if you do not want to use the default of 5550.
- d. Select Apply to make the changes

e. Save changes to the default domain by selecting Save Configuration.

A screenshot of a Firefox browser window showing the DataPower Gateway XML Management Interface. The URL is https://10.0.20.10:9090/configure/MgmtInterface. The page title is "Configure XML Management Interface". On the left, there's a navigation sidebar with sections like Control Panel, Blueprint Console (deprecated), Status, Services, Network, Interface, Management, Telnet Service, SSH Service, Web Management Service, XML Management Interface, REST Management Interface, Other, Administration, and Objects. A message at the top says "The running configuration of the device contains unsaved changes. Review changes." The main form has tabs for Main, Advanced, and SLM, with Main selected. It includes fields for Local address (0.0.0.0), Port number (5550), Access control list (xml-mgmt), Comments, and Enabled services (SOAP management URI, SOAP configuration management, SOAP configuration management (v2004), AMP endpoint, SLM endpoint, WS-Management endpoint, WSDM endpoint, and LDAP authentication (deprecated)). Buttons for Apply, Cancel, Undo, Export, View Log, View Status, and Help are present.

2. Enable the REST management interface in the default domain. The REST management interface is required if you want to enable the Trace feature in the assembly Test tab in the API Manager.
 - a. Search for the REST management interface in the navigation search bar, and select it.
 - b. Set the Administrative state to enabled.
 - c. You can specify a different port number if you do not want to use the default of 5554.
 - d. Select Apply to make the changes



e. Save changes to the default domain by selecting Save Configuration.

A screenshot of a Firefox browser window showing the IBM DataPower Gateway Control Panel. The URL is https://10.0.20.10:9090/configure/RestMgmtInterface. The page title is "Configure REST Management Interface". On the left, there's a navigation tree with "Control Panel" selected, showing categories like Status, Services, Network, Interface, Management, and Objects. A message at the top says "The running configuration of the device contains unsaved changes. [Review changes.](#)". The main form has sections for "Main", "REST Management Interface [up]", and "Comments". It includes fields for "Local address" (0.0.0.0), "Port Number" (5554), "Custom TLS server type" (Server Profile), "Custom TLS server profile" (none), "Access Control List" (rest-mgmt), and "Comments". Buttons for "Apply", "Cancel", and "Undo" are at the top of the form. At the bottom right are links for "Export", "View Log", "View Status", and "Help". The status bar at the bottom shows "Domain: default" and "Save Configuration".

3. Create an application domain. This domain receives your traffic.

- a. Search for Application domain in the navigation search bar, and select it.
- b. Select Add to create the application domain.
- c. Enter a unique name for your domain.
- d. Ensure that enabled is selected for the Administrative state.
- e. Ensure that the default domain is listed in the Visible application domain list.
- f. Select Apply.
- g. Change to your new application domain by selecting Domain in the menu bar, and selecting the domain that you created.



h. Select Save changes and switch domains.

i. All of the remaining steps on the DataPower gateway must be done in the application domain that you created.

j. Save changes to the domain by selecting Save Configuration.

A screenshot of a Firefox browser window showing the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/Domain/dashenbank_prod_01. The page title is "DataPower Gateway". On the left, there's a navigation sidebar with sections like Control Panel, Blueprint Console (deprecated), Status, Services, Network, Administration, Access, Device, Storage Devices, Debug, Miscellaneous, and Objects. The main content area shows the "Configure Application Domain" screen for the domain "dashenbank_prod_01". It has tabs for Main and Configuration, with Main selected. Under Main, there are buttons for Apply, Cancel, Delete, and Undo. A message box says "The running configuration of the device contains unsaved changes. Review changes." Below that, there are fields for Administrative state (set to enabled), Comments (empty), and Visible application domains (set to default). There are also sections for File permission to the local directory and File-monitoring of the local directory, each with several checkboxes. At the bottom right, there are links for Export, View Log, View Status, Help, Restart Domain, Reset Domain, Quiesce, and Unquiesce.

4. Ensure that your deployment includes an NTP server to synchronize time between each of the DataPower Gateways .

5. Ensure that you have set a unique System Identifier for each v10 DataPower gateway. 6.

Upload your private crypto key file to the domain.

a. Search for the Crypto key in the navigation search bar, and select it.

b. Select Add to create a key object.

c. Create a unique name for the key object in the Name field.

d. Select Upload.



e. Browse for the key file (which must be a .pem or .p12 file) and select it.

f. If you want to rename it, enter a new name for the file.

g. Select Upload to move it to the server in the cert:// folder.

h. Select Apply to save the changes.

A screenshot of a Firefox browser window showing the DataPower Gateway Control Panel. The URL is https://10.0.20.10:9090/configure/CryptoKey/Dashen_KEY. The page title is "Configure Crypto Key". On the left, there's a navigation tree with "Control Panel" expanded, showing "Blueprint Console (deprecated)" and "Objects" with various sub-options like "Network Settings", "Protocol Handlers", etc. A search bar is at the top left. A message bar at the top right says "The running configuration of the device contains unsaved changes. [Review changes.](#)". The main content area has tabs "Main" and "Advanced". Under "Main", there's a section for "Crypto Key: Dashen_KEY [up]". It shows "Administrative state" as "enabled" (radio button selected). Below that is a "File name" field with "cert://" dropdown and "digicert_star_dashenbanksc_com.pem" selected. There are "Upload...", "Fetch...", "Edit...", and "View..." buttons. A "Password alias" field contains "Dashen_password". At the bottom of the page are "Apply", "Cancel", "Delete", and "Undo" buttons, along with links for "Export", "View Log", "View Status", "Help", and "Convert Crypto Key Object".

7. Upload your crypto certificate file to the domain.

a. Search for a Crypto certificate in the navigation search bar, and select it.

b. Select Add to create a certificate object.

c. Create a unique name for the certificate object in the Name field.

d. Select Upload....

e. Browse for the key file (which must be a .pem or .p12 file) and select it.

f. If you want to rename it, enter a new name for the file.



g. Select Upload to move it to the server in the cert:// folder.

h. Select Apply to save the changes.

| Name | Status | Op-State | Logs | File name |
|-------------|--------|----------|------|--|
| Dashen_CA | saved | up | | cert:///digicert_star_dashenbanksc_com.pfx |
| Dashen_Cert | saved | up | | cert:///digicert_star_dashenbanksc_com.pem |

8. Associate the Crypto key with the Crypto certificate by setting the Identification credential. a. Search for Crypto Identification Credentials in the navigation search bar, and select it. b. Select Add.

c. Enter a name for your credential.

d. Ensure that the Administrative state has a value of enabled.

e. In the Crypto Key field, select the name of the key object that you created from the drop-down menu.

f. In the Certificate object field, select the name of the certificate object that you created from the drop-down menu.



g. Select Apply to commit your changes.

The screenshot shows the DataPower Gateway configuration interface. On the left, there's a navigation sidebar with various options like Control Panel, Blueprint Console (deprecated), Status, Services, Network, Administration, Objects, and Crypto Configuration. The main content area is titled "Configure Crypto Identification Credentials" for "Dashen_Certificates". It includes fields for "Administrative state" (set to enabled), "Crypto Key" (Dashen_KEY), "Certificate" (Dashen_Cert), and "Intermediate CA certificates" (Dashen_CA). A message box at the top right says "The running configuration of the device contains unsaved changes. [Review changes](#)". At the bottom, there are "Apply", "Cancel", "Delete", and "Undo" buttons, along with links for "Export", "View Log", "View Status", and "Help".

9. Create your TLS Client profile.

a. Search for TLS Client profile in the navigation search bar, and select it.

b. Select Add to create a client profile.

c. Create a unique name for the profile in the Name field.

d. Select your Identification credential from the drop-down list.

e. Ensure that the value of the Validate server certificate is set to off.

f. Ensure that the value of Use SNI is set to on.

g. Select Apply to save the changes.



10. Create your TLS Server profile.

- a. Search for TLS Server Profile in the navigation search bar, and select it.
- b. Select Add to create a server profile.
- c. Create a unique name for the profile in the Name field.
- d. Select your Identification credential from the drop-down list.
- e. Ensure that the value of Request client authentication is set to off.
- f. Disabling the request for client authentication for the TLS client profile and the validation of server certificates for the TLS server profile disables security for the ease of configuring the gateway. To enable the secure communication using mutual TLS between IBM API Connect and Gateway, see [Binding a TLS server profile to a gateway service](#).

- g. Select Apply to save the changes.

The screenshot shows the DataPower Gateway Control Panel interface. The top navigation bar includes links for Activities, Firefox, Log In, IBM Cloud, ibm da, Config, How t, The Q, Error, Using, Error, and dashenba. The main content area has tabs for Control Panel, Blueprint Console (deprecated), and a search bar. A message box at the top right says "The running configuration of the device contains unsaved changes. [Review changes](#)". Below this is a "Configure TLS Server Profile" section. The "Main" tab is selected. It shows a "TLS Server Profile: Dashen_Server_Profile [up]" section with "Apply", "Cancel", "Delete", and "Undo" buttons. To the right are links for Export, View Log, View Status, and Help. The "General" section includes fields for "Administrative state" (radio buttons for enabled and disabled, with enabled selected) and "Comments". The "Protocols" section lists checkboxes for SSL version 3, TLS version 1.0, TLS version 1.1 (checked), TLS version 1.2 (checked), and TLS version 1.3 (checked). The "Ciphers" section lists several cipher suites: AES_256_GCM_SHA384 (TLSv1.3), CHACHA20_POLY1305_SHA256 (TLSv1.3), AES_128_GCM_SHA256 (TLSv1.3), ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, and ECDHE_RSA_WITH_AES_256_GCM_SHA384. Each cipher suite has up and down arrows for reordering and a delete icon. On the left sidebar, there's a tree view of objects under "Objects": Network Settings, Protocol Handlers, API Processing Action, API Assembly, Service Configuration, Parsing, XML Processing, JSON Processing, Web Services, Policy Configuration, Web Applications, Monitoring, and Crypto Configuration, with further sub-options like Cookie Attribute Policy, CRL Retrieval, etc.



11. Configure your gateway peering object for the API Connect Gateway Service. This step is required when you set up a peer group of gateways, even if there is only a single gateway server in the gateway service.

- a. Search for Gateway peering in the navigation search bar, and select it.
- b. Select Add.
- c. Enter a unique name for your gateway peering object.
- d. Ensure that the Administrative state has a value of enabled.
- e. Select a local address for the communications among the members of the peer group. f.

Select a local port for the communication. You can use the default value of 16380.

- g. Select a monitor port for the communication. You can use the default value of 26380. h.

Because this procedure uses only one gateway, ensure that Peer group mode is not selected. i.

Clear the Enable TLS check box. TLS is not needed for a single peer.

- j. Set the Persistence location value to Memory for either physical DataPower appliance or virtual DataPower appliance.



k. Select Apply to commit your changes.

A screenshot of a Firefox browser window showing the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/GatewayPeering/peering. The page title is "Configure Gateway Peering". On the left, there is a navigation sidebar with a search bar and a tree view of service configurations. The main form has tabs for "Main" and "Advanced". The "Main" tab shows fields for "Administrative state" (radio buttons for "enabled" and "disabled", with "enabled" selected), "Comments" (a text input field), "Password alias" (a dropdown menu with "(none)" selected), "Local address" (a text input field with "Select Alias" button), "Local port" (a text input field with "16380" and an asterisk), "Primary count" (a dropdown menu with "1" and an asterisk), "Monitor port" (a text input field with "26380" and an asterisk), "Peer group mode" (a checkbox), "Enable TLS" (a checkbox), and "Dominance location" (a dropdown menu). At the top right, there are links for "Export", "View Log", "View Status", "Help", "Switch primary", "Remove stale peers", "Transfer different primary", and "Remove stale node". At the bottom right, there are buttons for "Apply", "Cancel", "Delete", and "Undo". The status bar at the bottom of the browser window shows "Mar 1 10:55".

12. Configure your gateway peering object for rate limit information.

- Search for Gateway peering in the navigation search bar, and select it.
- Select Add.
- Enter a unique name for your gateway peering object.
- Ensure that the Administrative state has a value of enabled.
- Select a local address for the communications among the members of the peer group.
- Select a local port for the communication. Use a unique port, different than the ports used for communication by other gateway peering objects.
- Select a monitor port for the communication. Use a unique port, different than the ports used for monitoring by other gateway peering objects.



h. Because this procedure uses only one gateway, ensure that Peer group mode is not selected. i.

Clear the Enable TLS check box. TLS is not needed for a single peer.

j. Set the Persistence location value to Memory for either physical DataPower appliance or virtual DataPower appliance.

k. Select Apply to commit your changes.

A screenshot of a web browser displaying the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/GatewayPeering/rate_limit. The page title is "Configure Gateway Peering". On the left, there is a navigation sidebar with various options like Control Panel, Blueprint Console (deprecated), and a search bar. The main content area shows fields for "Administrative state" (set to "enabled"), "Comments", "Password alias", "Local address", "Local port" (set to 16381), "Primary count" (set to 1), "Monitor port" (set to 26381), "Peer group mode" (unchecked), and "Enable TLS" (unchecked). There are also links for "Switch primary", "Remove stale peers", "Transfer different primary", and "Remove stale node". The top of the screen shows the browser's header with tabs and status information.

13. Configure your gateway peering object for script rate limit information.

a. Search for Gateway peering in the navigation search bar, and select it.

b. Select Add.

c. Enter a unique name for your gateway peering object.

d. Ensure that the Administrative state has a value of enabled.

e. Select a local address for the communications among the members of the peer group.

f. Select a local port for the communication. Use a unique port, different from the ports used for



communication by other gateway peering objects.

g. Select a monitor port for the communication. Use a unique port, different from the ports used for monitoring by other gateway peering objects.

h. Because this procedure uses only one gateway, ensure that Peer group mode is not selected. i.

Clear the Enable TLS check box. TLS is not needed for a single peer.

j. Set the Persistence location value to Memory for either physical DataPower appliance or virtual DataPower appliance.

k. Select Apply to commit your changes.

A screenshot of a Firefox browser window showing the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/GatewayPeering/script. The page title is "Configure Gateway Peering". On the left, there is a navigation sidebar with a search bar and links for Control Panel, Blueprint Console (deprecated), Status, Services, Network, Administration, Objects, API Processing Action, API Assembly, Service Configuration, API Connect Gateway Service, API Gateway, Cloud Gateway Service, Gateway Peering, HTTP Service, Multi-Protocol Gateway, TCP Proxy Service, TLS Proxy Service, UDDI Subscription (deprecated), Web Application Firewall, Web Service Proxy, Web Token Service, WSRR Saved Search Subscription, WSRR Subscription, XML Firewall, and XSL Coprocessor. The main content area shows fields for "Main": "Gateway Peering: script [up]", "Apply", "Cancel", "Delete", "Undo", "Administrative state" (radio buttons for enabled and disabled, with enabled selected), "Comments" (text input field), "Password alias" (dropdown with "(none)" and a plus sign button), "Local address" (text input field with "Select Alias" link), "Local port" (text input field with "16382" and an asterisk), "Primary count" (dropdown with "1" and an asterisk), "Monitor port" (text input field with "26382" and an asterisk), "Peer group mode" (checkbox), and "Enable TLS" (checkbox). There are also "Export", "View Log", "View Status", "Help", "Switch primary", "Remove stale peers", "Transfer different primary", and "Remove stale node" links at the bottom of the form.

14. Configure your gateway peering object for subscription information.

a. Search for Gateway peering in the navigation search bar, and select it.

b. Select Add.

c. Enter a unique name for your gateway peering object.



- d. Ensure that the Administrative state has a value of enabled.
 - e. Select a local address for the communications among the members of the peer group.
 - f. Select a local port for the communication. Use a unique port, different than the ports used for communication by other gateway peering objects.
 - g. Select a monitor port for the communication. Use a unique port, different than the ports used for monitoring by other gateway peering objects.
 - h. Because this procedure uses only one gateway, ensure that Peer group mode is not selected. i.
- Clear the Enable TLS check box. TLS is not needed for a single peer.
- j. Set the Persistence location value to Memory for either physical DataPower appliance or virtual DataPower appliance.
 - k. Select Apply to commit your changes.

A screenshot of a Firefox browser window showing the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/GatewayPeering/subscription. The page title is "Configure Gateway Peering".

The left sidebar shows a navigation tree with categories like Control Panel, Blueprint Console (deprecated), Status, Services, Network, Administration, Objects, Network Settings, Protocol Handlers, API Processing Action, API Assembly, Service Configuration, API Connect Gateway Service, API Gateway, Cloud Gateway Service, Gateway Peering, HTTP Service, Multi-Protocol Gateway, TCP Proxy Service, TLS Proxy Service, UDDI Subscription (deprecated), Web Application Firewall, Web Service Proxy, Web Token Service, WSRR Saved Search Subscription, WSRR Subscription, XML Firewall, and XSL Coprocessor.

The main content area displays the "Main" tab for "Gateway Peering: subscription [up]". It includes fields for "Administrative state" (radio buttons for "enabled" and "disabled", with "enabled" selected), "Comments" (text input field), "Password alias" (dropdown menu with "(none)" selected), "Local address" (text input field with "Select Alias" button), "Local port" (text input field with value "16383" and an asterisk), "Primary count" (dropdown menu with value "1" and an asterisk), "Monitor port" (text input field with value "26383" and an asterisk), "Peer group mode" (checkbox), and "Enable TLS" (checkbox). There are also buttons for "Apply", "Cancel", "Delete", and "Undo". At the bottom, there are links for "Export", "View Log", "View Status", "Help", "Switch primary", "Remove stale peers", "Transfer different primary", and "Remove stale node".

15. Configure the gateway peering object for the API probe. In order for API Connect to receive trace data in the Test tab's debugger, the DataPower API Gateway must be configured to support the API probe.



- a. Search for Gateway peering in the navigation search bar, and select it.
- b. Select Add.
- c. Enter a unique name for your gateway peering object.
- d. Ensure that the Administrative state has a value of enabled.
- e. Select a local address for the communications among the members of the peer group.
- f. Select a local port for the communication. Use a unique port, different from the ports used for communication by other gateway peering objects.
- g. Select a monitor port for the communication. Use a unique port, different from the ports used for monitoring by other gateway peering objects.
- h. Because this procedure uses only one gateway, ensure that Peer group mode is not selected. i.

Clear the Enable TLS check box. TLS is not needed for a single peer.

- j. Set the Persistence location value to Memory for either physical DataPower appliance or virtual DataPower appliance.



k. Select Apply to commit your changes.

A screenshot of a Firefox browser window showing the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/GatewayPeering/probe. The page title is "Configure Gateway Peering". On the left, there is a navigation sidebar with a search bar and links to various service configurations like Status, Services, Network, Administration, and Objects. The main content area shows fields for "Administrative state" (set to "enabled"), "Comments", "Password alias", "Local address", "Local port" (set to 16384), "Primary count" (set to 1), "Monitor port" (set to 26384), "Peer group mode", "Enable TLS", and "Dominance location". At the top right, there are links for "Export", "View Log", "View Status", "Help", "Switch primary", "Remove stale peers", "Transfer different primary", and "Remove stale node".

16. Configure the gateway peering manager.

- Search for Gateway Peering Manager in the navigation search bar, and select it.
- Set the Administrative state to enabled.
- In the pull-down menu next to API Connect Gateway Service, select the gateway peering object configured for the API Connect Gateway Service.
- In the pull-down menu next to Rate Limit, select the gateway peering object configured for rate limit information.
- In the pull-down menu next to Subscription, select the gateway peering object configured for subscription.
- In the pull-down menu next to API Probe, select the gateway peering object that you configured for the API probe.



g. Select Apply to commit your changes.

A screenshot of a Firefox browser window showing the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/GatewayPeeringManager. The page title is "Configure Gateway Peering Manager". On the left, there is a navigation sidebar with a search bar and a tree view of configuration objects. The main panel shows fields for "Administrative state" (set to "enabled"), "Comments", "API Connect Gateway Service" (set to "peering"), "API rate limiting" (set to "rate_limit"), "API subscription" (set to "subscription"), "API probe" (set to "probe"), and "GatewayScript rate limiting" (set to "script"). Buttons for "Apply", "Cancel", and "Undo" are at the top right, along with links for "Export", "View Log", "View Status", and "Help".

Activities Firefox Mar 1 10:57

IBM Data IBM D Log In IBM Cloud IBM Cloud ibm da Config The se K How t The Q Error Using Error dashenba +

Centos Wiki Documentation Forums Other Bookmarks

DataPower Gateway admin @ 10.0.20.10:9090 3/1/2022, 5:56:53 AM (EST) Domain: dashenbank_prod_01 Save Configuration Logout IBM

Control Panel Blueprint Console (deprecated)

Search

Main

Gateway Peering Manager [up]

Apply Cancel Undo Export | View Log | View Status | Help

Administrative state: enabled

Comments:

API Connect Gateway Service: peering

API rate limiting: rate_limit

API subscription: subscription

API probe: probe

GatewayScript rate limiting: script

17. Configure the API probe settings object.

- Search for API probe settings in the navigation search bar, and select it.
- Set the Administrative state to enabled.
- Set the maximum number of records to 1000.
- Set the expiration to 60 minutes.
- In the pull-down menu next to Gateway Peering, select the gateway peering object that you configured for the API probe.



f. Select Apply to commit your changes.

A screenshot of a Firefox browser window showing the DataPower Gateway configuration interface. The URL is https://10.0.20.10:9090/configure/APIDebugProbe. The page title is "Configure API Probe Settings". On the left, there's a navigation sidebar with a tree view of configuration categories like Status, Services, Network, Administration, Objects, and API Processing Action. The main content area shows fields for "Administrative state" (radio buttons for enabled or disabled, currently enabled), "Comments" (text input field), "Maximum records" (text input field set to 1000), "Expiration" (text input field set to 60 Minutes), and "Gateway peering" (dropdown menu with probe selected). At the bottom of the form are "Apply", "Cancel", and "Undo" buttons, along with links for "Export", "View Log", "View Status", and "Help". The top of the browser window shows various tabs and the date/time Mar 1 10:58.

18. Set the API Connect Gateway service to define the communication interface with the API Connect

Management server and for API transactions.

- a. Search for API Connect Gateway service in the navigation search bar, and select it.
- b. Ensure that the Administrative state is set to enabled.
- c. In the Local address field, enter the IP address of the DataPower gateway to which you want the traffic from the API Connect Management server to be sent.
- d. Specify a value for Local port. You can use the default value of 3000, or specify a different port value.
- e. In the TLS client field drop-down list, select the name of the TLS client profile that you created.



- f. In the TLS server field drop-down list, select the name of the TLS server profile that you created.
- g. In the API gateway address field, enter the IP address for the DataPower gateway to which you want the API traffic sent.
- h. Use the default port value of 9443 for the API gateway port. If the port is not being used by another service, you can also change it to port 443 if you want API transactions to be sent to the default port for HTTPS.
- i. For DataPower API Gateway, set the Gateway Peering to (none). When no gateway peering object is configured for the DataPower API Gateway, the peering configuration defined in the Gateway Peering Manager configuration is used.

A screenshot of a Firefox browser window showing the DataPower Gateway Control Panel. The URL in the address bar is https://10.0.20.10:9090/configure/APIConnectGatewayService. The page title is "Configure API Connect Gateway Service".

The left sidebar shows a navigation tree with categories like Control Panel, Blueprint Console (deprecated), and various service configurations. The main content area displays the configuration for the API Connect Gateway Service. It includes fields for Local address (0.0.0.0), Local port (3000), TLS client (selected alias dashenbank_prod_01), TLS server (selected alias dashenbank_prod_01), API gateway address (0.0.0.0), API gateway port (9443), V5 compatibility mode (unchecked), and Gateway Peering (set to "default-gateway-peering").



16.2. Register the gateway service in the API Connect Cloud Manager console:

- a. Open the API Connect Cloud Manager console.
- b. Navigate to Configure Topology.
- c Select Register Service.
- d Select DataPower API Gateway for the DataPower API Gateway.
- e Add a title, name, and summary for the gateway connection.
- f Enter the following values in the API Invocation Endpoint field:

IP address or host name of one of the gateways

<https://10.0.20.18:9443/>

- e. Enter the one of the following values in the Management Endpoint field:

IP address or hostname of one of the gateways

<https://10.0.20.18:3000/>

- f. Select the default TLS Client Profile

Click the Save button.

Note : Configure Server Name Indication (SNI) profiles.SNI profiles allow different TLS certificates to be used for API transaction requests from different host names.