**Deployment Document**

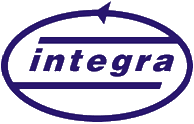
**Project**

**OCP 4.15 DR Cluster Deployment**

**v1.1**

**Client: Bank-ABC**

****



**Integra Micro Systems Pvt. Ltd.**

No. 4, 1st Floor, Bellary Road,

12th KM, Jakkur,

Bangalore 560 064, INDIA

Tel: 080-46632400

[**www.integramicro.com**](http://www.integramicro.com/)

**Disclaimer**

Copyright © 2013:  Integra Micro Systems Private Limited, Registered Office #4, Bellary Rd, 12, KM, Yashoda Nagar, Jakkur, Bangalore, Karnataka India.

All rights about this document are reserved and  shall not be , in whole or in part, copied, photocopied, reproduced, translated, or reduced to any manner including but not limited to electronic, mechanical, machine readable ,photographic, optic recording or otherwise without prior consent, in writing, of Integra Micro Systems Private Limited (the Company).

The information in this document is subject to changes without notice. This describes only the product defined in the introduction of this documentation. This document is intended for the use of prospective customers of the Company Products Solutions and or Services for the sole purpose of the transaction for which the document is submitted. No part of it may be reproduced or transmitted in any form or manner whatsoever without the prior written permission of the company. The Customer, who/which assumes full responsibility for using the document appropriately. The Company welcomes customer comments as part of the process of continuous development and improvement.

The Company, has made all reasonable efforts to ensure that the information contained in the document are adequate, sufficient and free of material errors and omissions. The Company will, if necessary, explain issues, which may not be covered by the document. However, the Company does not assume any liability of whatsoever nature, for any errors in the document except the responsibility to provide correct information when any such error is brought to company’s knowledge. The Company will not be responsible, in any event, for errors in this document or for any damages, incidental or consequential, including monetary losses that might arise from the use of this document or of the information contained in it.

This document and the Products, Solutions and Services it describes are intellectual property of the Company and/or of the respective owners thereof, whether such IPR is registered, pending for registration, applied for registration or not.

The only warranties for the Company Products, Solutions and Services are set forth in the express warranty statements accompanying its products and services. Nothing here in should be construed as constituting an additional warranty. The Company shall not be liable for technical or editorial errors or omissions contained herein.

The Company logo is a trademark of the Company. Other products, names, logos mentioned in this document, if any, may be trademarks of their respective owners.

Copyright © 2013:  Integra Micro Systems Private Limited. All rights reserved.

**Release History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. **Release Version** | 1. **Release Date** | 1. **Description of the release** | 1. **Created By** | 1. **Approved By** |
| **0.1** | 1. 10-09-2024 | 1. First Draft | 1. Vageesh/Sowjanya/Kiran | 1. Murali.K.Muddada |
| **0.2** | 12-09-2024 | Updated ELK details | 1. Vageesh/Sowjanya/Kiran | 1. Murali.K.Muddada |
| **0.3** | 13-09-2024 | Updated prometheous and Graphana details | 1. Vageesh/Sowjanya/Kiran | 1. Murali.K.Muddada |
| **0.4** | 16-09-2024 | Added Cronjob for ETCD backup | 1. Vageesh/Sowjanya/Kiran | 1. Murali.K.Muddada |

1. **Confidentiality Statement**

Copyright © 2011-2012, Integra Micro Systems Private Limited. All rights reserved. This product or document may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form, by any means electronic, mechanical, photographic, optic recording or otherwise without prior consent, in writing, of the copyright owner. Statutory declaration under section 52A of the Copyright Act 1957.

**Table of Contents**

[1 About the Guide 6](#__RefHeading___Toc7141_399447981)

[1.1 Target Audience 6](#__RefHeading___Toc7143_399447981)

[1.2 Acronyms and Abbreviations 6](#__RefHeading___Toc7145_399447981)

[2 Introduction 8](#__RefHeading___Toc7147_399447981)

[2.1 Topics Covered 8](#__RefHeading___Toc7149_399447981)

[2.2 Scope 8](#__RefHeading___Toc7151_399447981)

[2.3 Out of Scope 8](#__RefHeading___Toc7153_399447981)

[3 Implementation Details 9](#__RefHeading___Toc7155_399447981)

[3.1 Environment Setup 9](#__RefHeading___Toc7157_399447981)

[3.1.1 Network Information 9](#__RefHeading___Toc7159_399447981)

[3.1.2 Network Services 9](#__RefHeading___Toc7161_399447981)

[3.1.3 Node Information 9](#__RefHeading___Toc7163_399447981)

[3.1.4 Storage Information 9](#__RefHeading___Toc7165_399447981)

[3.1.5 Certificates 9](#__RefHeading___Toc7167_399447981)

[4 Technical Implementation 10](#__RefHeading___Toc7169_399447981)

[4.1 Check Bastion Node for all the pre-Installation requirements 10](#__RefHeading___Toc7171_399447981)

[4.1.1 Operating System Details 10](#__RefHeading___Toc7173_399447981)

[4.1.2 Block Device (Hard Disk) Details 10](#__RefHeading___Toc7175_399447981)

[4.1.3 Network Configuration 10](#__RefHeading___Toc7177_399447981)

[4.1.4 Route and Default Gateway details 10](#__RefHeading___Toc7179_399447981)

[4.1.5 Enabling Repositories 10](#__RefHeading___Toc7181_399447981)

[4.2 Installing a Connected Cluster 11](#__RefHeading___Toc7183_399447981)

[4.2.1 Configure Environmental Services in Bastion. 11](#__RefHeading___Toc7185_399447981)

[4.2.2 Generating an SSH private key and adding it to the agent 13](#__RefHeading___Toc7187_399447981)

[4.2.3 Create install-config.yaml file 13](#__RefHeading___Toc7189_399447981)

[4.2.4 Creating the Openshift/Kubernetes manifest 15](#__RefHeading___Toc7191_399447981)

[4.2.5 Create the Ignition config files 15](#__RefHeading___Toc7193_399447981)

[4.3 Deploying Openshift 16](#__RefHeading___Toc7195_399447981)

[4.3.1 Monitoring the Bootstrap Process 16](#__RefHeading___Toc7197_399447981)

[4.3.2 Remove the Bootstrap Node 16](#__RefHeading___Toc7199_399447981)

[4.3.3 Join Nodes 16](#__RefHeading___Toc7201_399447981)

[4.4 Accessing the Cluster 17](#__RefHeading___Toc7203_399447981)

[4.4.1 Accessing the Openshift Console 17](#__RefHeading___Toc7205_399447981)

[4.4.2 Accessing the Cluster Nodes. 17](#__RefHeading___Toc7207_399447981)

[4.5 Deploying Openshift Internal Registry with Persistent Storage 19](#__RefHeading___Toc7209_399447981)

[4.6 Configure System Clock Sync with NTP Server 19](#__RefHeading___Toc7211_399447981)

[4.6.1 Configure master nodes system clock sync with NTP server 20](#__RefHeading___Toc7213_399447981)

[4.7 Configuring an HTPasswd identity provider 21](#__RefHeading___Toc7215_399447981)

[4.8 Registering the Cluster 22](#__RefHeading___Toc7217_399447981)

[4.9 Encrypting the ETCD data 24](#__RefHeading___Toc7219_399447981)

[4.10 Backing Up ETCD data 24](#__RefHeading___Toc7221_399447981)

[4.11 Deploying and Configuring Openshift Logging 25](#__RefHeading___Toc7223_399447981)

[4.11.1 Install Cluster Logging Operators 25](#__RefHeading___Toc7225_399447981)

[4.11.2 Create Cluster Logging Instance 26](#__RefHeading___Toc7227_399447981)

[4.11.3 Viewing cluster logs from the Kibana dashboard 28](#__RefHeading___Toc7229_399447981)

[4.12 Applying a custom Alertmanager configuration with the SMTP details 29](#__RefHeading___Toc7231_399447981)

[4.13 Removing the kubeadmin user 30](#__RefHeading___Toc7233_399447981)

[4.14 Moving monitoring components from Worker to Infra labeled nodes 31](#__RefHeading___Toc7235_399447981)

[4.15 Configuring the default scheduler to control pod placement 33](#__RefHeading___Toc7237_399447981)

[4.16 Etcd backup Cronjob 33](#__RefHeading___Toc7398_399447981)

# About the Guide

This design document currently includes a Prod environment which will be hosted on Virtual Machines.

## Target Audience

This document is intended for Client technical staff responsible for the environment.

## Acronyms and Abbreviations

The table below provides a glossary of the terms and acronyms used within this document.

|  |  |
| --- | --- |
| **Acronym** | **Description** |
| RH | Red Hat, Inc |
| OKE | Openshift Kubernetes Engine |
| OCP | Openshift Container Platform |
| AD | Active Directory |
| API | Application Programming Interface |
| AZ | Availability Zone |
| CA | Certificate Authority |
| CR | Custom Resource |
| DC | Data Centre |
| DNS | Domain Name System |
| DHCP | Dynamic Host Configuration Protocol |
| DVR | Distributed Virtual Routers, a networking configuration for routing using Compute and Network nodes for separate networking functions |
| FQDN | Fully Qualified Domain Name |
| Guest | Also see “VM”. This is virtual machine running on a Host. |
| HA | High-Availability or Highly-Available |
| Host | The physical hardware or the logical OS which runs virtualisation technology allowing one or more Guest OS’s to run on the hardware owned by the Host |
| KVM | Kernel-based Virtual Machine |
| L2 | Layer 2, part of the TCP/IP Network Stack |
| L3 | Layer 3, part of the TCP/IP Network Stack |
| LBaaS | Load Balancing as a Service |
| NAT | Network Address Translation |
| NIC | Network Interface Card. References a virtual or a physical port allowing network access and interface to a Host or Guest VM. |
| NTP | Network Time Protocol |
| OCP | OpenShift Container Platform |
| OS | Operating System |
| OVS | Open vSwitch, a Linux-based virtual switch appliance used in OpenStack |
| PXE | Preboot Execution Environment |
| QA | Quality Assurance |
| SAML/SAML2 | A protocol allowing authentication and federation of multiple domains |
| SAN | Storage Area Network |
| SSL | Secure Sockets Layer |
| TLS | Transport Layer Security |
| VIP | Virtual IP address |
| VLAN | Virtual LAN is a networking virtualisation technology |
| VM | Virtual machine, in OSP terms, synonymous with “Workload” or “Guest” |
| VXLAN | Virtual Extensible LAN (VXLAN) is a network virtualisation technology |
| Workload | Synonym for “Guest” or “VM” |
| K8s | Kubernetes |
| RHCOS | Red Hat CoreOS |
| DIY | Do It Yourself |
| VM | Virtual Machine |
| OCR | OpenShift Container Registry (Integrated Registry) |

# Introduction

## Topics Covered

Openshift topology - Discussion about OCP high level architecture, subsystems as well as high-level overview of process, technology and operations.

Design guidelines and architecture - Interactive discussion covering minimal HA architectures for development environment and constraints.

## Scope

The scope of this document is to describe the architecture that Red Hat can offer for Openshift. This document is the result of the design workshop and should cover all requirements that Bank ABC put forward during the workshop. As a part of this project engagement with Bank ABC, Red Hat team will:

● Design and validate Openshift Kubernetes Engine (OKE) 4.15 in prod Datacenters.

## Out of Scope

This document limits the discussions and recommendations only to Red Hat Components involved for Bank ABC.

# Implementation Details

## Environment Setup

Steps to prepare or create the environment for the proposed architecture

### Network Information

*Table 1. Network Data*

|  |  |  |
| --- | --- | --- |
| **Name** | **IP Address Pool** | **Comments** |
| Cluster Network | 10.128.0.0/14 | Non Routable |
| Service Network | 172.30.0.0/16 | Non Routable |

### Network Services

*Table 2. Network Service Data*

|  |  |  |
| --- | --- | --- |
| **Services** | **Host** | **Comments** |
| DNS Server | 10.151.20.201 |  |
| DNS Server | 10.151.20.202 |  |

### Node Information

*Table 3. Nodes Data*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Server FQDN** | **IP** | **Role** | **Subs Used** | **OS** |
| drlonvsacmdrcm1.acmdrc.arabbanking.local | 10.151.24.78 | Master/Control-Plane | N/A | RHCOS |
| drlonvsacmdrcm2.acmdrc.arabbanking.local | 10.151.24.79 | Master/Control-Plane | N/A | RHCOS |
| drlonvsacmdrcm3.acmdrc.arabbanking.local | 10.151.24.81 | Master/Control-Plane | N/A | RHCOS |
| drlonvsacmdrcw1.acmdrc.arabbanking.local | 10.151.24.82 | Worker | N/A | RHCOS |
| drlonvsacmdrcw2.acmdrc.arabbanking.local | 10.151.24.83 | Worker | N/A | RHCOS |
| drlonvsacmdrcw3.acmdrc.arabbanking.local | 10.151.24.84 | Infra | N/A | RHCOS |
| drlonvsacmdrcw4.acmdrc.arabbanking.local | 10.151.24.85 | Infra | N/A | RHCOS |

### Storage Information

*Table 4. Storage Data*

|  |  |  |  |
| --- | --- | --- | --- |
| **Application** | **Storage Type** | **Storage Details** | **Size** |
| Registry | NFS | NFS\_Volume | 500GB |
| ELK Application Deployment | NFS | NFS\_Volume | 3\*300GB |
| Prometheous and Graphana | NFS | NFS\_Volume | 2\*300GB, 2\*5GB |

### Certificates

Bank ABC has planned to use certificates signed by their internal CA for Wild Card (\*.apps) Domain Ingress Controller.

# Technical Implementation

## Check Bastion Node for all the pre-Installation requirements

### Operating System Details

• Bastion Node

|  |
| --- |
| [core@drlonvsacmdrcmg ]$ cat /etc/redhat-release  Red Hat Enterprise Linux release 8.10(Ootpa) |

### Block Device (Hard Disk) Details

• Bastion Node

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ df -hT  Filesystem Type Size Used Avail Use% Mounted on  devtmpfs devtmpfs 7.7G 0 7.7G 0% /dev  tmpfs tmpfs 7.7G 252K 7.7G 1% /dev/shm  tmpfs tmpfs 7.7G 1.1M 7.7G 1% /run  tmpfs tmpfs 7.7G 0 7.7G 0% /sys/fs/cgroup  /dev/mapper/rhel\_rhel8-root xfs 15G 5.6G 9.5G 38% /  /dev/mapper/rhel\_rhel8-tmp xfs 10G 104M 9.9G 2% /tmp  /dev/mapper/rhel\_rhel8-var xfs 10G 1.2G 8.9G 12% /var  /dev/mapper/rhel\_rhel8-var\_log xfs 10G 1.4G 8.7G 14% /var/log  /dev/mapper/rhel\_rhel8-var\_tmp xfs 10G 104M 9.9G 2% /var/tmp  /dev/mapper/software--vg-software\_vol xfs 100G 1.9G 99G 2% /software  /dev/sda1 xfs 1014M 360M 655M 36% /boot  /dev/mapper/rhel\_rhel8-var\_log\_audit xfs 15G 176M 15G 2% /var/log/audit  /dev/mapper/rhel\_rhel8-home xfs 10G 8.2G 1.9G 82% /home  tmpfs tmpfs 1.6G 24K 1.6G 1% /run/user/4029  tmpfs tmpfs 1.6G 48K 1.6G 1% /run/user/4030  10.151.24.41:/ACM\_OC\_DRC nfs4 100G 289M 100G 1% /acmdata  tmpfs tmpfs 1.6G 0 1.6G 0% /run/user/4031  tmpfs tmpfs 1.6G 0 1.6G 0% /run/user/0  tmpfs tmpfs 1.6G 0 1.6G 0% /run/user/4028 |

### Network Configuration

* Bastion Node

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ cat /etc/resolv.conf  # Generated by NetworkManager  search arabbanking.local  nameserver 10.151.20.201  nameserver 10.151.20.202 |

### Route and Default Gateway details

* Bastion Node

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ route -n  Kernel IP routing table  Destination Gateway Genmask Flags Metric Ref Use Iface  0.0.0.0 10.151.27.254 0.0.0.0 UG 100 0 0 ens192  10.151.24.0 0.0.0.0 255.255.252.0 U 100 0 0 ens192 |

### Enabling Repositories

Bank ABC Team created a VM with RHEL 8.10 with minimal install which is registered with their internal Red Hat Satellite Server and the following repositories has been enabled by them.

• Bastion Node

|  |
| --- |
| [core@ drlonvsacmdrcmg ~]$ yum repolist  Not root, Subscription Management repositories not updated  repo id repo name  rhel-8-for-x86\_64-appstream-rpms Red Hat Enterprise Linux 8 for x86\_64 - AppStream (RPMs)  rhel-8-for-x86\_64-baseos-rpms Red Hat Enterprise Linux 8 for x86\_64 - BaseOS (RPMs) |

## Installing a Connected Cluster

### Configure Environmental Services in Bastion.

1. Install & configure HAProxy

|  |
| --- |
| dnf install haproxy -y |

[root@drlonvsacmlb1 ~]$ cat /etc/haproxy/haproxy.cfg

|  |
| --- |
| # Global settings  #---------------------------------------------------------------------  global  maxconn 20000  log /dev/log local0 info  chroot /var/lib/haproxy  pidfile /var/run/haproxy.pid  user haproxy  group haproxy  daemon  # turn on stats unix socket  stats socket /var/lib/haproxy/stats  #---------------------------------------------------------------------  # common defaults that all the 'listen' and 'backend' sections will  # use if not designated in their block  #---------------------------------------------------------------------  defaults  log global  mode http  option httplog  option dontlognull  option http-server-close  option redispatch  option forwardfor except 127.0.0.0/8  retries 3  maxconn 20000  timeout http-request 10000ms  timeout http-keep-alive 10000ms  timeout check 10000ms  timeout connect 40000ms  timeout client 300000ms  timeout server 300000ms  timeout queue 50000ms  # Enable HAProxy stats  listen stats  bind :9000  stats uri /stats  stats refresh 10000ms  # Kube API Server  frontend k8s\_api\_frontend  bind :6443  default\_backend k8s\_api\_backend  mode tcp  backend k8s\_api\_backend  mode tcp  balance source  # server drlonvsacmbst.acmdrc.arabbanking.local 10.151.24.91:6443 check  server drlonvsacmdrcm1.acmdrc.arabbanking.local 10.151.24.78:6443 check  server drlonvsacmdrcm2.acmdrc.arabbanking.local 10.151.24.79:6443 check  server drlonvsacmdrcm3.acmdrc.arabbanking.local 10.151.24.81:6443 check  # OCP Machine Config Server  frontend ocp\_machine\_config\_server\_frontend  mode tcp  bind :22623  default\_backend ocp\_machine\_config\_server\_backend  backend ocp\_machine\_config\_server\_backend  mode tcp  balance source  # server drlonvsacmbst.acmdrc.arabbanking.local 10.151.24.91:22623 check  server drlonvsacmdrcm1.acmdrc.arabbanking.local 10.151.24.78:22623 check  server drlonvsacmdrcm2.acmdrc.arabbanking.local 10.151.24.79:22623 check  server drlonvsacmdrcm3.acmdrc.arabbanking.local 10.151.24.81:22623 check  # OCP Ingress - layer 4 tcp mode for each. Ingress Controller will handle layer 7.  frontend ocp\_http\_ingress\_frontend  bind :80  default\_backend ocp\_http\_ingress\_backend  mode tcp  backend ocp\_http\_ingress\_backend  balance source  mode tcp  server drlonvsacmdrcw1.acmdrc.arabbanking.local 10.151.24.82:80 check  server drlonvsacmdrcw2.acmdrc.arabbanking.local 10.151.24.83:80 check  frontend ocp\_https\_ingress\_frontend  bind \*:443  default\_backend ocp\_https\_ingress\_backend  mode tcp  backend ocp\_https\_ingress\_backend  mode tcp  balance source  server drlonvsacmdrcw1.acmdrc.arabbanking.local 10.151.24.82:443 check  server drlonvsacmdrcw2.acmdrc.arabbanking.local 10.151.24.83:443 check |

|  |
| --- |
| Apply HAProxy configuration |

2. Enable and start the service

|  |
| --- |
| setsebool -P haproxy\_connect\_any 1 # SELinux name\_bind access systemctl enable haproxy systemctl start haproxy systemctl status haproxy |

1. Install & configure Apache Web Server

|  |
| --- |
| dnf install httpd -y |

1. Change default listen port to 8080 in httpd.conf

|  |
| --- |
| sed -i 's/Listen 80/Listen 0.0.0.0:8080/' /etc/httpd/conf/httpd.conf |

1. Enable and start the service

|  |
| --- |
| systemctl enable httpd systemctl start httpd systemctl status httpd |

1. Making a GET request to localhost on port 8080 should now return the default Apache webpage

|  |
| --- |
| curl localhost:8080 |

### Generating an SSH private key and adding it to the agent

* Generate SSH Key
* Start the ssh-agent process as a background task

|  |
| --- |
| [core@drlonvsacmdrcmg~]$ ssh-keygen  [core@drlonvsacmdrcmg~]$eval "$(ssh-agent -s)" |

* Add your SSH private key to the ssh-agent

|  |
| --- |
| **[[core@drlonvsacmdrcmg ~]**$ ssh-add /home/core/.ssh/id\_rsa  Identity added: /home/core/.ssh/id\_rsa **(**/home/core/.ssh/id\_rsa**)** |

### Create install-config.yaml file

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ cat install-config.yaml  apiVersion: v1  baseDomain: arabbanking.local  compute:  - hyperthreading: Enabled  name: worker  replicas: 0  controlPlane:  hyperthreading: Enabled  name: master  replicas: 3  metadata:  name: acmdrc  networking:  clusterNetwork:  - cidr: 10.128.0.0/14  hostPrefix: 23  networkType: OVNKubernetes  serviceNetwork:  - 172.30.0.0/16  platform:  none: {}  fips: false  pullSecret: '{"auths":{"cloud.openshift.com":{"auth":"b3BlbnNoaWZ0LXJlbGVhc2UtZGV2K29jbV9hY2Nlc3NfN2YwMjY5YmMzMTVkNGQ0MGEyZWVlMWY0MjRiYjFiMjc6TU5XRERFWjk2WDlDOUE5TEg5TEZHTlNJT0lFMkwxQko5QkxOTE5XOU9KVFExQjJLSVVGRUM4TUNTT01TUDlSMg==","email":"prem.ram@bank-abc.com"},"quay.io":{"auth":"b3BlbnNoaWZ0LXJlbGVhc2UtZGV2K29jbV9hY2Nlc3NfN2YwMjY5YmMzMTVkNGQ0MGEyZWVlMWY0MjRiYjFiMjc6TU5XRERFWjk2WDlDOUE5TEg5TEZHTlNJT0lFMkwxQko5QkxOTE5XOU9KVFExQjJLSVVGRUM4TUNTT01TUDlSMg==","email":"prem.ram@bank-abc.com"},"registry.connect.redhat.com":{"auth":"","email":"prem.ram@bank-abc.com"},"registry.redhat.io":{"auth":"","email":"prem.ram@bank-abc.com"}}}'  sshKey: 'ssh  ed25519AAAAC3NzaC1lZDI1NTE5AAAAINiuKrt0X3C3+3sreFbLXZ9kRUUEnZWS8VPQwGpyKG6T core@drlonvsacmdrcmg' |

### Creating the Openshift/Kubernetes manifest

|  |
| --- |
| [core@drlonvsacmdrcmg]$ openshift-install create manifests  INFO Consuming Install Config from target directory  WARNING Mak ing control-plane schedulable by setting MastersSchedulable to true for Scheduler cluster settings INFO Manifests created in: /home/core/clusterconfig/manifests and /home/core/clusterconfig/openshift |

### Create the Ignition config files

|  |
| --- |
| [core@drlonvsacmdrcmg]$ openshift-install create ignition-configs INFO Consuming Openshift Manifests from target directory INFO Consuming Master Machines from target directory INFO Consuming OpenShift Install (Manifests) from target directory INFO Consuming Worker Machines from target directory INFO Consuming Common Manifests from target directory INFO Ignition-Configs created in: . and auth |

The following files will be generated in the directory.

|  |
| --- |
| [core@drlonvsacmdrcmg]$ tree . ├── auth │ ├── kubeadmin-password │ └── kubeconfig ├── bootstrap.ign ├── master.ign ├── metadata.json └── worker.ign |

• Install HTTPD Package • Start and Enable httpd Service

|  |
| --- |
| [core@drlonvsacmdrcmg~]# yum install httpd –y  [core@drlonvsacmdrcmg~]# systemctl start httpd  [core@drlonvsacmdrcmg~]# systemctl enable httpd  [core@drlonvsacmdrcmg~]# systemctl status httpd ● httpd.service - The Apache HTTP Server Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled) Active: active (running) Docs: man:httpd(8) man:apachectl(8) Main PID: 25568 (httpd) • Verify the firewalld.service status [core@bastion ~]# systemctl status firewalld ● firewalld.service - firewalld - dynamic firewall daemon Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled) Active: inactive (dead) Docs: man:firewalld(1) |

• Create a directory in /var/www/html/ and copy the ignition files

|  |
| --- |
| [core@drlonvsacmdrcmg~]# mkdir -p /var/www/html/ocp4/ |

• Copy bootstrap.ign from /home/clusterconfig/ to /var/www/html/ocp4/  
directory

|  |
| --- |
| [core@drlonvsacmdrcmg~]# cp /home/clusterconfig//bootstrap.ign /var/www/html/ocp4/ |

• Make sure bootstrap.ign file has 644 permissions

|  |
| --- |
| [core@drlonvsacmdrcmg~]# chmod 644 /var/www/html/ocp4/bootstrap.ign |

## Deploying Openshift

Provisioning bootstrap node

|  |
| --- |
| sudo coreos-installer install /dev/sda --insecure-url [http://drlonvsacmdrcmg.acmdrc.arabbanking.local:8080/ocp/bootstrap.ign](http://10.192.242.171:8080/ocp/bootstrap.ign) --insecure-ignition --copy-network |

Run the above command to provision the master/worker node with respect to their ignition files.

### Monitoring the Bootstrap Process

Monitoring the bootstrap process from the Bastion host at different log levels (debug, error, info)

|  |
| --- |
| ~/openshift-install --dir ~/ocp-install wait-for bootstrap-complete --log-level=debug |

### [Remove the Bootstrap Node](https://github.com/ryanhay/ocp4-metal-install" \l "remove-the-bootstrap-node)

Wait for installation to complete

1. Collect the OpenShift Console address and kubeadmin credentials from the output of the install-complete event

|  |
| --- |
| [dbpuser@ drlonvsacmdrcmg ~]$ ./openshift-install --dir ~/clusterconfig wait-for install-complete  INFO Waiting up to 40m0s (until 11:23AM) for the cluster at https://api.acmdrc.arabbanking.local:6443 to initialize...  INFO Checking to see if there is a route at openshift-console/console...  INFO Install complete!  INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/home/dbpuser/clusterconfig/auth/kubeconfig'  INFO Access the OpenShift web-console here: https://console-openshift-console.apps.acmprod.arabbanking  INFO Login to the console with user: "kubeadmin", and password: "\*\*\*\*\*\*"  INFO Time elapsed: 1s |

1. Continue to join the mater or worker nodes to the cluster in a new tab whilst waiting for the above command to complete.

### Join Nodes

1. Setup 'oc' and 'kubectl' clients on the bastion machine.

|  |
| --- |
| export KUBECONFIG=~/ocp-install/auth/kubeconfig # Test auth by viewing cluster nodes oc get nodes |

1. View and approve pending CSRs.

Note: Once you approve the first set of CSRs additional 'kubelet-serving' CSRs will be created. These must be approved too. If you do not see pending requests wait until you do.

|  |
| --- |
| # View CSRs oc get csr  # Approve all pending CSRs oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}{{end}}{{end}}' | xargs oc adm certificate approve # Wait for kubelet-serving CSRs and approve them too with the same command oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}{{end}}{{end}}' | xargs oc adm certificate approve |

1. Watch and wait for the Nodes to join the cluster and enter a 'Ready' status.

|  |
| --- |
| watch oc get nodes |

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ oc get nodes  NAME STATUS ROLES AGE VERSION  drlonvsacmdrcm1.acmdrc.arabbanking.local Ready control-plane,master 7d v1.28.11+add48d0  drlonvsacmdrcm2.acmdrc.arabbanking.local Ready control-plane,master 7d v1.28.11+add48d0  drlonvsacmdrcm3.acmdrc.arabbanking.local Ready control-plane,master 7d v1.28.11+add48d0  drlonvsacmdrcw1.acmdrc.arabbanking.local Ready worker 6d21h v1.28.11+add48d0  drlonvsacmdrcw2.acmdrc.arabbanking.local Ready worker 6d21h v1.28.11+add48d0  drlonvsacmdrcw3.acmdrc.arabbanking.local Ready infra 6d21h v1.28.11+add48d0  drlonvsacmdrcw4.acmdrc.arabbanking.local Ready infra 6d3h v1.28.11+add48d0 |

## Accessing the Cluster

### Accessing the Openshift Console

1. Wait for the 'console' Cluster Operator to become available

|  |
| --- |
| oc get co |

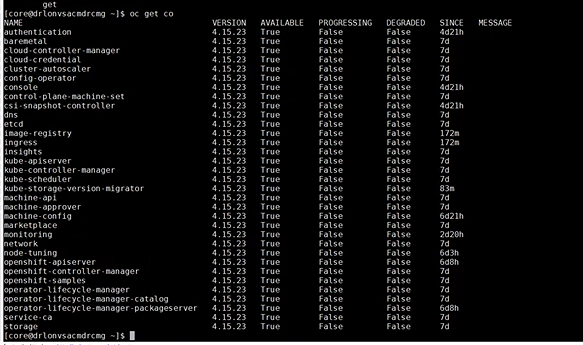
1. Navigate to the [OpenShift Console URL](https://console-openshift-console.apps.ocp-spoke.lab.local/) and log in as the 'admin' user.

You will get self-signed certificate warnings that you can ignore if you need to login as kubeadmin and need to the password again you can retrieve it with: **cat ~/clusterconfig/auth/kubeadmin-password.**

### Accessing the Cluster Nodes.

1. Setup 'oc' and 'kubectl' clients on the bastion machine.

|  |
| --- |
| export KUBECONFIG=~/clusterconfig/auth/kubeconfig # Test auth by viewing cluster nodes oc get nodes |

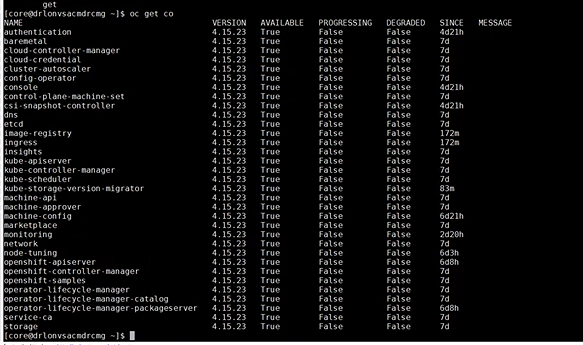
1. SSH to Master / Worker Node by the following command:

|  |
| --- |
| ssh core@<hostname or ip> |

* Verify the status of nodes.

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ oc get nodes  NAME STATUS ROLES AGE VERSION  drlonvsacmdrcm1.acmdrc.arabbanking.local Ready control-plane,master 7d v1.28.11+add48d0  drlonvsacmdrcm2.acmdrc.arabbanking.local Ready control-plane,master 7d v1.28.11+add48d0  drlonvsacmdrcm3.acmdrc.arabbanking.local Ready control-plane,master 7d v1.28.11+add48d0  drlonvsacmdrcw1.acmdrc.arabbanking.local Ready worker 6d21h v1.28.11+add48d0  drlonvsacmdrcw2.acmdrc.arabbanking.local Ready worker 6d21h v1.28.11+add48d0  drlonvsacmdrcw3.acmdrc.arabbanking.local Ready infra 6d21h v1.28.11+add48d0  drlonvsacmdrcw4.acmdrc.arabbanking.local Ready infra 6d3h v1.28.11+add48d0 |

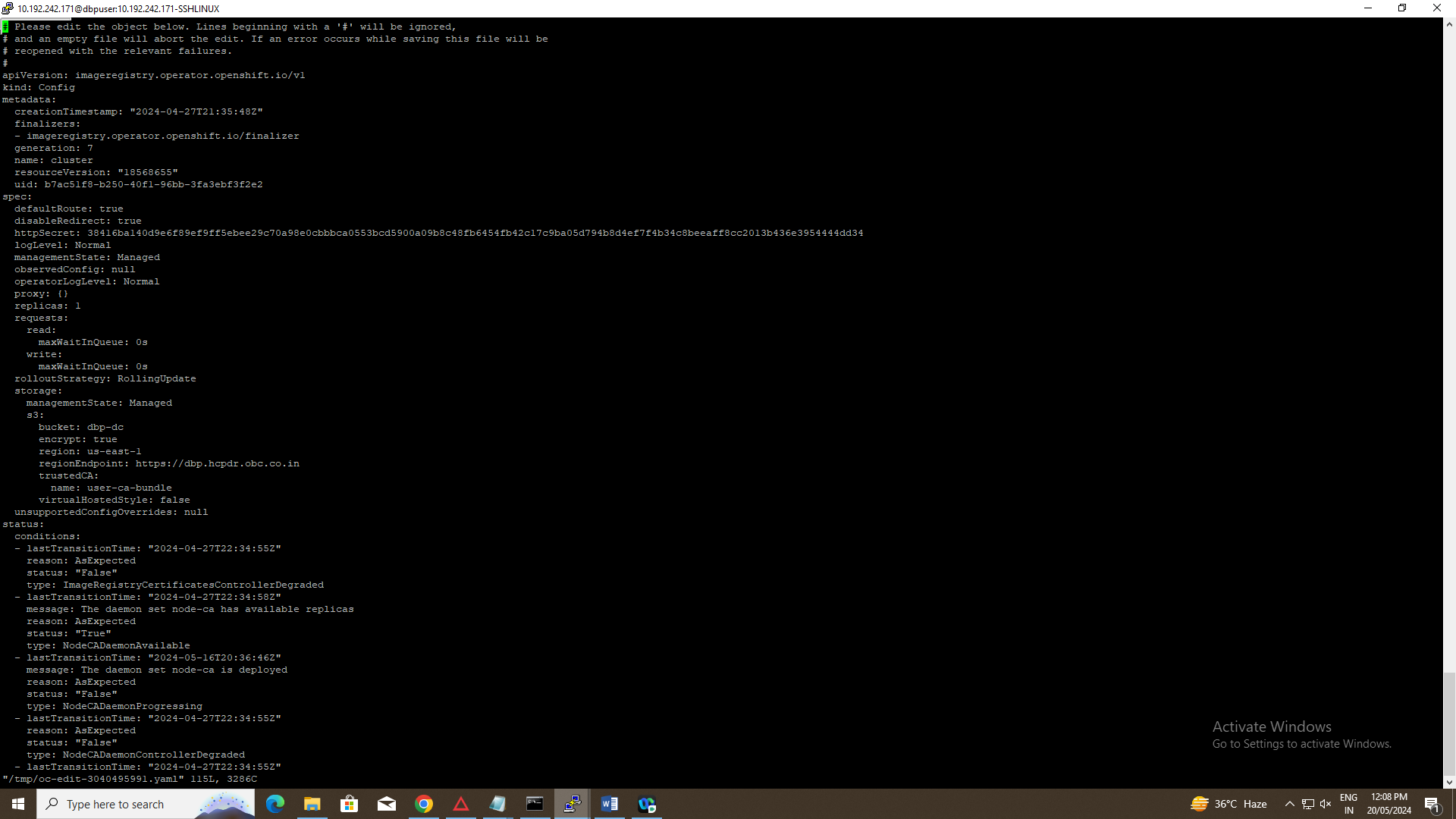
* Check the status of cluster operators



## Deploying Openshift Internal Registry with Persistent Storage

* By default, there will be no registry pod running as the registry operator’s "managementState" will be set as "Removed". Configure image registry to "Managed" state and move the registry pods to the dedicated infra nodes as shown below. Right now an object storage volume is being used inside the registry pods.

|  |
| --- |
| oc edit configs.imageregistry.operator.openshift.io/cluster |



* Verify if the registry pod is running on one of the dedicated infra aux nodes as shown below.

|  |
| --- |
| [core@drlonvsacmdrcmg ~ ~]$ oc get pods -o wide | grep "image-registry-88b86cdb4-mzbqk"  image-registry-88b86cdb4-mzbqk 1/1 Running 1 6d15h 10.139.0.4 drlonvsacmdrcm1.acmdrc.arabbanking.local <none> <none>  [dbpuser@bastion ~]$ |

## Configure System Clock Sync with NTP Server

### Configure master nodes system clock sync with NTP server

* Create a Butane config including the contents of the chrony.conf file, to configure chrony on worker nodes, create a 99-worker-chrony.bu file and use Butane to generate a MachineConfig object file, 99- worker-chrony.yaml, containing the configuration to be delivered to the nodes. Repeat the same steps for other machine-config-pool like master. .

|  |
| --- |
| [core@drlonvsacmdrcmg NTP]$ cat worker-chrony.bu  variant: openshift  version: 4.15.0  metadata:  name: 99-worker-chrony  labels:  machineconfiguration.openshift.io/role: worker  storage:  files:  - path: /etc/chrony.conf  mode: 0644  overwrite: true  contents:  inline: |  server 10.151.217.1 iburst  driftfile /var/lib/chrony/drift  makestep 1.0 3  rtcsync  logdir /var/log/chrony |

|  |
| --- |
| [core@drlonvsacmdrcmg NTP]$ ../butane 99-worker-chrony.bu -o 99-worker-chrony.yaml |

* Apply the MachineConfig YAML file

|  |
| --- |
| [core@drlonvsacmdrcmg NTP]$oc create -f ./99-worker-chrony.yaml  machineconfig.machineconfiguration.openshift.io/99-worker-chrony created  [core@drlonvsacmdrcmg NTP]$ pwd  /home/T910013790/NTP  [core@drlonvsacmdrcmg NTP]$  [core@drlonvsacmdrcmg NTP]$  [core@drlonvsacmdrcmg NTP]$  [core@drlonvsacmdrcmg NTP]$ ls -ltr  total 16  -rw-r--r-- 1 root root 489 Sep 1312:30 99-worker-chrony.bu  -rw-r--r-- 1 root root 489 Sep1 3 12:30 99-master-chrony.bu  -rw-r--r-- 1 root root 608 Sep 13 12:35 99-worker-chrony.yaml  -rw-r--r-- 1 root root 608 Sep 13 12:53 99-master-chrony.yaml |

* Once NTP config are applied, the worker and infra nodes will get rebooted and come back online.

|  |
| --- |
| [[core@drlonvsacmdrcmg](mailto:core@prbahvsacmprodmg) NTP]$ ssh -i ~/.ssh/ocp.9 [core@drlonvsacmdrcm1.acmdrc.arabbanking.local](mailto:core@prbahvsacmprodm1.acmprod.arabbanking.local)  [core@drlonvsacmdrcmg NTP]$ chronyc sources  MS Name/IP address Stratum Poll Reach LastRx Last sample  ===============================================================================  ^\* 10.151.217.1 2 7 377 97 -563us[ -564us] +/- 105ms[ |

|  |
| --- |
| [core@drlonvsacmdrcmg NTP]$ chronyc tracking  Reference ID : 0A97D901 (10.151.217.1)  Stratum : 3  Ref time (UTC) : Mon Sep 16 10:39:54 2024  System time : 0.000008051 seconds slow of NTP time  Last offset : -0.000000964 seconds  RMS offset : 0.000097927 seconds  Frequency : 13.198 ppm slow  Residual freq : -0.000 ppm  Skew : 0.095 ppm  Root delay : 0.130586922 seconds  Root dispersion : 0.020342302 seconds  Update interval : 257.9 seconds  Leap status : Normal |

Repeat the above commands on rest of master nodes to verify the system clock with NTP and make sure "System clock synchronized: yes"

## Configuring an HTPasswd identity provider

* Create a flat file with a user name and hashed password

|  |
| --- |
| [core@drlonvsacmdrcm1 ~] mkdir auth\_providers  [core@drlonvsacmdrcm1 ~] cd auth\_providers  [core@ drlonvsacmdrcm1 ~] mkdir htpasswd\_ auth\_providers  [core@ drlonvsacmdrcm1 ~] cd htpasswd\_ auth\_providers  [core@ drlonvsacmdrcm1 ~]$ htpasswd -cB users.htpasswd ocpadmin New password:  Re-type new password:  Adding password for user ocpadmin |

The password used for the ocpadmin account hasn’t been disclosed here but the same has been shared with the customer.

* Continue to add or update credentials to the file as required as given in the example below.

[

core@drlonvsacmdrcm1 ~]$ htpasswd -B users.htpasswd developer

New password:

Re-type new password:

Adding password for user developer

* Create an OpenShift Container Platform Secret object that contains the HTPasswd users file

[

core@drlonvsacmdrcm1 htpasswd]$ oc create secret generic htpass-secret --from-file=htpasswd=users.htpasswd -n openshift-config

secret/htpass-secret created

The secret key containing the users file for the --from-file argument must be named

**htpasswd**

, as shown in the above command.

* Create HTPasswd identity provider YAML manifests as below

[

core@drlonvsacmdrcm1 htpasswd

]

$ vi htpasswd-auth-secret.yaml

apiVersion

:

config.openshift.io/v1

kind

:

OAuth

metadata

:

name

:

cluster

spec

:

identityProviders

:

-

name

:

htpasswd\_provider

mappingMethod

:

claim

type

:

HTPasswd

htpasswd

:

fileData

:

name

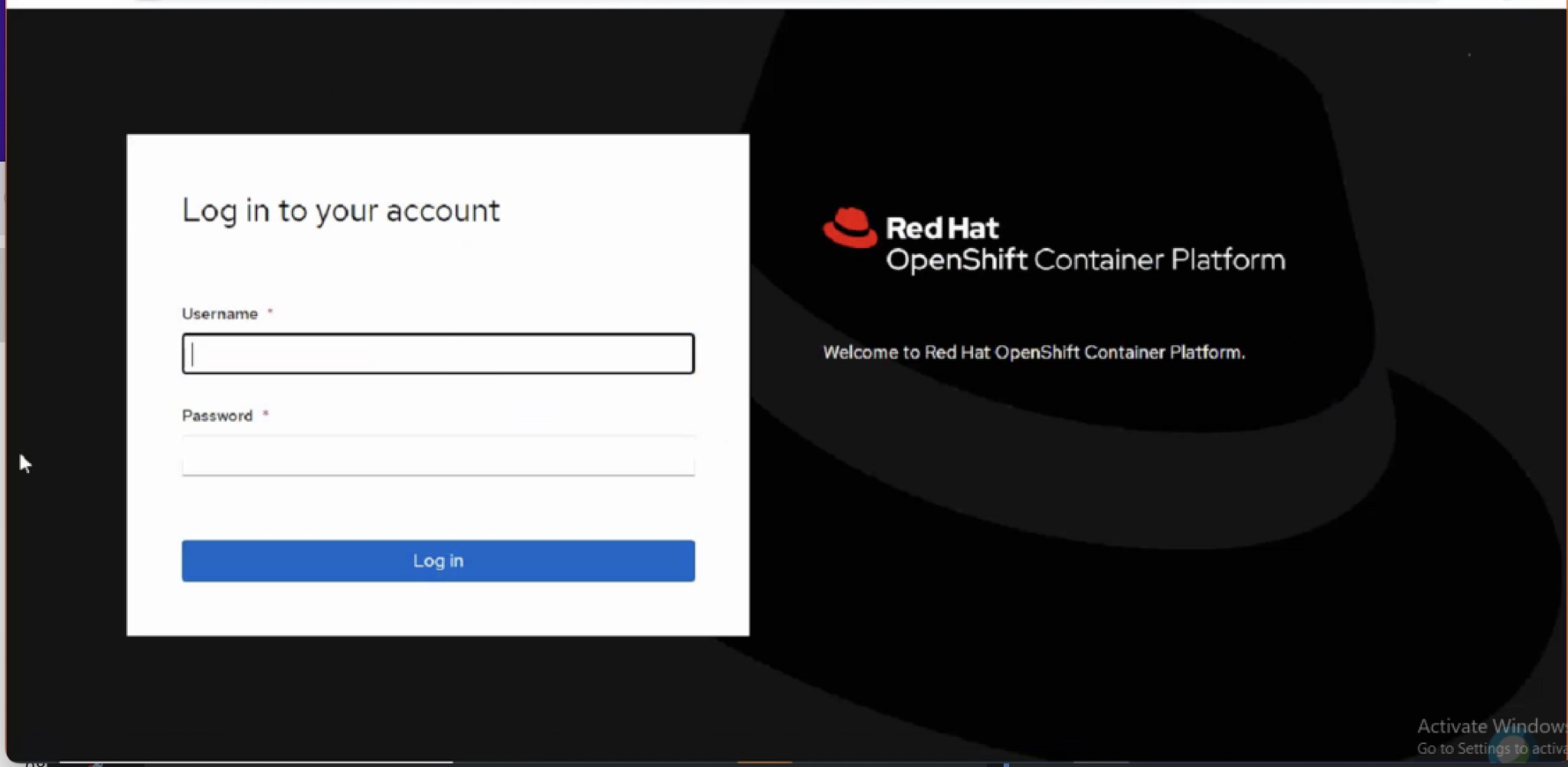
:

htpass-secret

Adding HTPasswd identity provider to the cluster

|  |
| --- |
| [core@drlonvsacmdrcm1 htpasswd]$ oc apply -f htpasswd-auth-secret.yaml  Warning: oc apply should be used on resource created by either oc create --save-config or oc apply oauth.config.openshift.io/cluster configured |

* Access the OCP Web Console and click on the HTPasswd identity provider & login using ocpadmin as the user with the respective password set earlier while creating the user account



* Grant the **ocpadmin** user account with **cluster-admin** access as shown below.

|  |
| --- |
| [core@drlonvsacmdrcm1 htpasswd]$ oc adm policy add-cluster-role-to-user cluster-admin ocpadmin clusterrole.rbac.authorization.k8s.io/cluster-admin added: "ocpadmin"  [core@rlonvsacmdrcm1 htpasswd]$ cd .. |

Please store the 'ocpadmin' password in a secured place as it has been granted with cluster-admin privilege to manage the entire OCP cluster.

## Registering the Cluster

* Subscription and support

The cluster will be automatically registered with a 60 day evaluation subscription which does not include support. In order to receive support for your cluster, you will need to register OpenShift Container Platform 4 cluster on cloud.redhat.com.

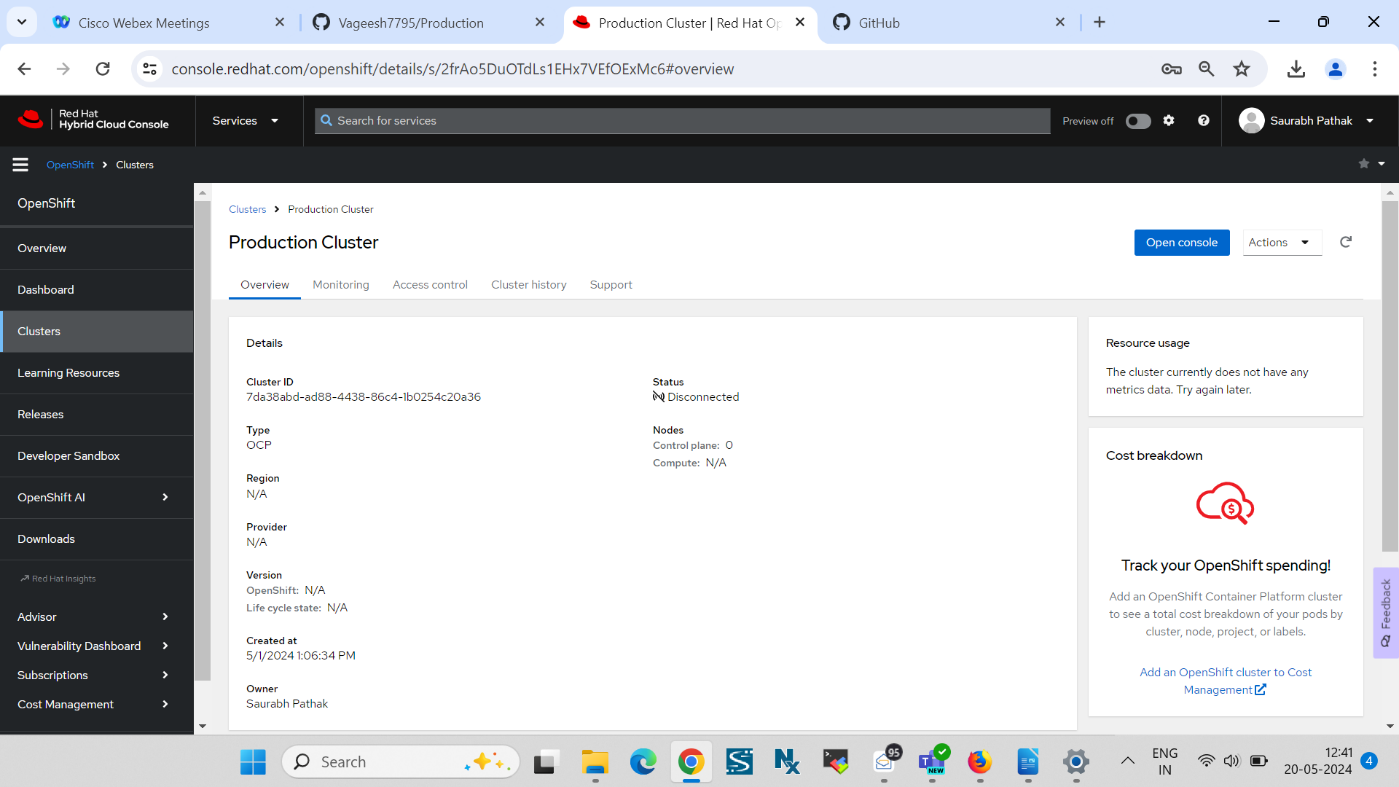
* To register a disconnected OCP 4 cluster on cloud.redhat.com the cluster profile needs to be created manually as outlined below:

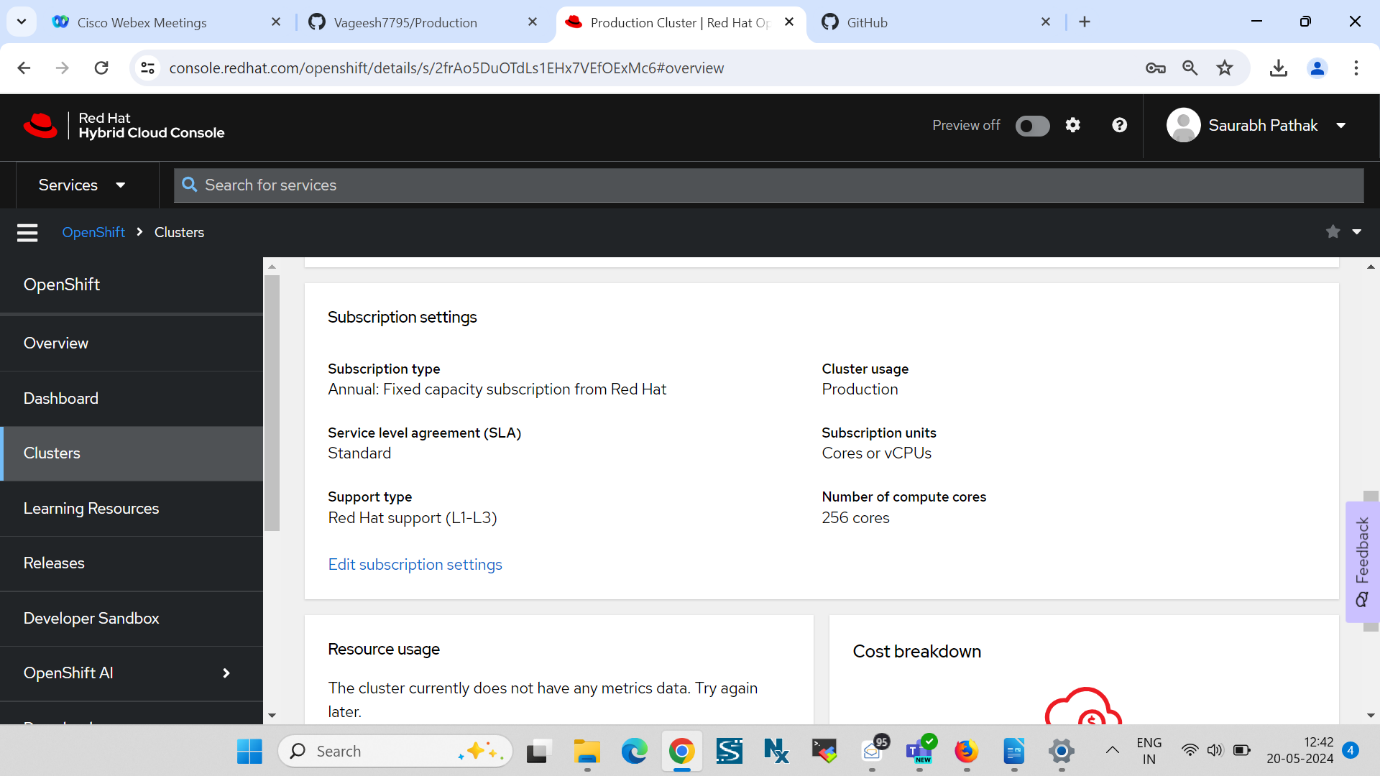
◦ Go to cloud.redhat.com and login with org\_admin login credentials.

◦ Go to the Clusters tab and select the three dots near Create cluster and click on the Register cluster button. Alternatively, use [**https://cloud.redhat.com/openshift/register**](https://cloud.redhat.com/openshift/register) ◦ ◦ Enter all the required details like cluster-id, number of sockets, memory, etc.

◦ After filling all the required details click on the Register cluster. This will create a cluster profile in disconnected mode.

◦ After that, click on the respective cluster name in the 'Clusters' tab. Edit the Subscription Settings and attach the subscription according to the cluster configuration and environment by scrolling down to 'Subscription settings'.





To get more details, Please visit the below URL:

<https://access.redhat.com/solutions/4930131>

## Encrypting the ETCD data

* Edit the apiserver object & make an entry under spec as shown below

[

core@drlonvsacmdrcm1]# oc edit apiserver

...

...

spec:

encryption:

type: aescbc

audit:

profile: Default

* Verify if all the resource under **openshiftapiserver** and **kubeapiserver** are encrypted as shown below.

[

core@prbahvsacmprodm1]# oc get openshiftapiserver -o=jsonpath='{range

.items[0].status.conditions[?(@.type=="Encrypted")]}{.reason}{"\n"}{.message}{"\n"}'

EncryptionCompleted

All resources encrypted: routes.route.openshift.io, oauthaccesstokens.oauth.openshift.io, oauthauthorizetokens.oauth.openshift.io

[

core@prbahvsacmprodm1]# oc get kubeapiserver -o=jsonpath='{range

.items[0].status.conditions[?(@.type=="Encrypted")]}{.reason}{"\n"}{.message}{"\n"}'

EncryptionCompleted

All resources encrypted: secrets, configmaps

## Backing Up ETCD data

* Login to one of the master node and perform the backup of ETCD data as shown below

[core@master1 ~]$ sudo /usr/local/bin/cluster-backup.sh /home/core/ETCD-Backup-16-09-2024/assets/backup c2d027a039b1fbd493d1689ed36f5b99a37daf764c207e904f5ba3308f2ea879 etcdctl version: 3.4.9 API version: 3.4 found latest kube-apiserver-pod: /etc/kubernetes/static-pod-resources/kube-apiserver-pod-38 found latest kube-controller-manager-pod: /etc/kubernetes/static-pod-resources/kube-controller-manager-pod-10 found latest kube-scheduler-pod: /etc/kubernetes/static-pod-resources/kube-scheduler-pod-9 found latest etcd-pod: /etc/kubernetes/static-pod-resources/etcd-pod-3 {"level":"info","ts":1630482862.6628346,"caller":"snapshot/v3\_snapshot.go:119","msg":"created temporary db file","path":"/home/dbpuser/ETCD-Backup-01-05-2024/assets/backup/snapshot\_2024-09-16\_145106.db.part"}

{"level":"info","ts":"2024-09-16T07:54:22.684Z","caller":"clientv3/maintenance.go:200","msg":"opened snapshot stream; downloading"}

{"level":"info","ts":1630482862.6849284,"caller":"snapshot/v3\_snapshot.go:127","msg":"fetching snapshot","endpoint":"https://10.1.108.136:2379"}

{"level":"info","ts":"2024-09-16T07:54:25.207Z","caller":"clientv3/maintenance.go:208","msg":"completed snapshot read; closing"}

{"level":"info","ts":1630482865.365249,"caller":"snapshot/v3\_snapshot.go:142","msg":"fetched snapshot","endpoint":"https://10.1.108.136:2379","size":"123 MB","took":2.70228545}

{"level":"info","ts":1630482865.3654559,"caller":"snapshot/v3\_snapshot.go:152","msg":"saved","path":"/home/core/assets/backup/snapshot\_2024-09-01\_075421.db"} Snapshot saved at /home/core/assets/backup/snapshot\_2024-09-16\_075421.db snapshot db and kube resources are successfully saved to /home/core/assets/backup

* Create a directory to store the ETCD Backup in Bastion Node

[core@master1 ~]$ sudo chown core -R assets/

* Create a directory to store the ETCD Backup in Bastion Node

[core@drlonvsacmdrcm1~]# mkdir ~/ETCD-Backup-16-09-2024/

* Copy the backed up files from the master node to the Bastion node as shown below

[core@bastian]$ scp -r -i ~/.ssh/ocp4 core@master1.prod.dbp.pnb:~/assets/

|  |
| --- |
| [core@drlonvsacmdrcm1 ETCD-Backup]$ ls assets/backup/ snapshot\_2024-09-16\_145106.db static\_kuberesources\_2024-09-16\_075421.tar.gz |

## Deploying and Configuring Openshift Logging

### Install Cluster Logging Operators

|  |
| --- |
| [core@drlvsonacmdrcm1]$ mkdir logging [core@drlvsonacmdrcm1]$ cd logging |

* Create a Namespace for the Elasticsearch Operator.

|  |
| --- |
| [core@drlonvsacmdrcmglogging]$ vi eo-namespace.yaml apiVersion: v1 kind: Namespace metadata:  name: openshift-operators-redhat  annotations:  openshift.io/node-selector: ""  labels:  openshift.io/cluster-monitoring: "true" |

* Creating a namespace for the Cluster Logging operator.

|  |
| --- |
| [core@drlonvsacmdrcmglogging]$ vi clo-namespace.yaml apiVersion: v1 kind: Namespace metadata: name: openshift-logging annotations: openshift.io/node-selector: "" labels: openshift.io/cluster-monitoring: "true" [core@drlonvsacmdrcmglogging]$ oc create -f clo-namespace.yaml namespace/openshift-logging created |

* Install the Elasticsearch Operator
* In the OpenShift Container Platform web console, click Operators → OperatorHub.
* Choose Elasticsearch Operator from the list of available Operators, and click Install.
* Ensure that the all namespaces on the cluster is selected under Installation Mode.
* Ensure that openshift-operators-redhat is selected under Installed Namespace.
* Select Enable operator recommended cluster monitoring on this namespace.
* Select stable as the Update Channel.
* Select an Approval Strategy to Automatic
* Click Install.
* Verify that the Elasticsearch Operator installed by switching to the Operators → Installed Operators page.
* Ensure that Elasticsearch Operator is listed in all projects with a Status of Succeeded.
* Install the Cluster Logging Operator.
* In the OpenShift Container Platform web console, click Operators → OperatorHub.
* Choose Cluster Logging from the list of available Operators, and click Install.
* Ensure that the A specific namespace on the cluster is selected under Installation Mode.
* Ensure that Operator recommended namespace is openshift-logging under Installed Namespace.
* Select Enable operator recommended cluster monitoring on this namespace.
* Select stable as the Update Channel.
* Select an Approval Strategy to Automatic. Click Install.
* Verify that the Cluster Logging Operator installed by switching to the Operators → Installed Operators page.
* Ensure that Cluster Logging is listed in the openshift-logging project with a Status of Succeeded.

### Create Cluster Logging Instance

* Create a cluster logging instance YAML manifest
* Create a Cluster Logging instance

|  |
| --- |
| [core@drlonvsacmdrcmglogging]$ vi clo-instance.yaml  apiVersion: "logging.openshift.io/v1" kind: "ClusterLogging" metadata:   name: "instance"   namespace: "openshift-logging" spec:   collection:     logs:       fluentd:         resources:           limits:             cpu: 500m             memory: 1Gi           requests:             cpu: 500m             memory: 1Gi       type: fluentd   curation:     curator:       nodeSelector:         node-role.kubernetes.io/infra: ""       resources:         limits:           memory: 1Gi         requests:           cpu: 200m           memory: 200Mi       schedule: 30 3 \* \* \*     type: curator   logStore:     elasticsearch:       nodeCount: 3       nodeSelector:         node-role.kubernetes.io/infra: ""       redundancyPolicy: SingleRedundancy       resources:         limits:           cpu: 2000m           memory: 16Gi         requests:           cpu: 1500m           memory: 16Gi       storage:         size: 300Gi         storageClassName: nfs01     retentionPolicy:       application:         maxAge: 3d       audit:         maxAge: 3d       infra:         maxAge: 3d     type: elasticsearch   managementState: Managed   visualization:     kibana:       nodeSelector:         node-role.kubernetes.io/infra: ""       replicas: 1       resources:         limits:           memory: 1Gi         requests:           cpu: 500m           memory: 1Gi     type: kibana |

* Create the ClusterLogging instance object as shown below

|  |
| --- |
| [core@drlonvsacmdrcmglogging]$ oc create -f clo-instance.yaml clusterlogging.logging.openshift.io/instance created |

* Verify the cluster logging pods deployed in the openshift-logging project

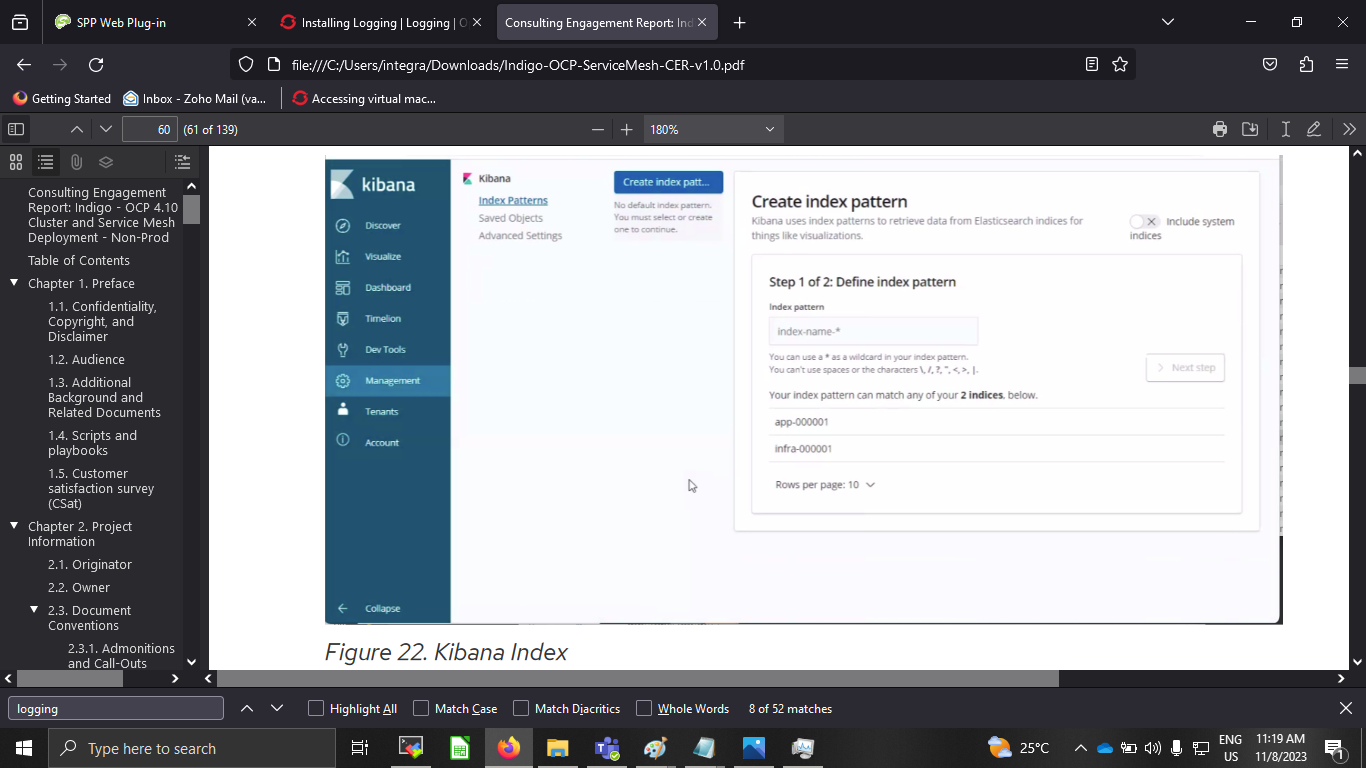
|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ oc get po -o wide -n openshift-logging  NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES  cluster-logging-operator-64654fbd86-7s454 1/1 Running 0 177m 10.129.2.7 drlonvsacmdrcw1.acmdrc.arabbanking.local <none> <none>  collector-6hpjm 1/1 Running 2 4d21h 10.128.1.151 drlonvsacmdrcm1.acmdrc.arabbanking.local <none> <none>  collector-cc898 1/1 Running 2 4d21h 10.131.0.37 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  collector-m565l 1/1 Running 2 4d21h 10.129.2.68 drlonvsacmdrcw1.acmdrc.arabbanking.local <none> <none>  collector-n2h29 1/1 Running 2 4d21h 10.130.1.160 drlonvsacmdrcm3.acmdrc.arabbanking.local <none> <none>  collector-n6prl 1/1 Running 2 4d21h 10.130.2.137 drlonvsacmdrcw2.acmdrc.arabbanking.local <none> <none>  collector-xbrkq 1/1 Running 2 4d21h 10.129.0.97 drlonvsacmdrcm2.acmdrc.arabbanking.local <none> <none>  collector-xp9cf 1/1 Running 2 4d21h 10.128.2.248 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  elasticsearch-cdm-j6bbys15-1-f47d4cc5b-s7v9x 2/2 Running 0 177m 10.128.2.19 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  elasticsearch-cdm-j6bbys15-2-b75f6d77-b2fjz 2/2 Running 0 177m 10.128.2.16 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  elasticsearch-cdm-j6bbys15-3-754df85bdb-v8nbc 2/2 Running 0 177m 10.128.2.15 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  elasticsearch-im-app-28774710-rpn2d 0/1 Completed 0 13m 10.131.0.65 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  elasticsearch-im-audit-28774710-nxgp5 0/1 Completed 0 13m 10.131.0.64 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  elasticsearch-im-infra-28774710-pl5wd 0/1 Completed 0 13m 10.131.0.63 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  kibana-86dc6b855c-mm7vf 2/2 Running 0 177m 10.128.2.18 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none> |

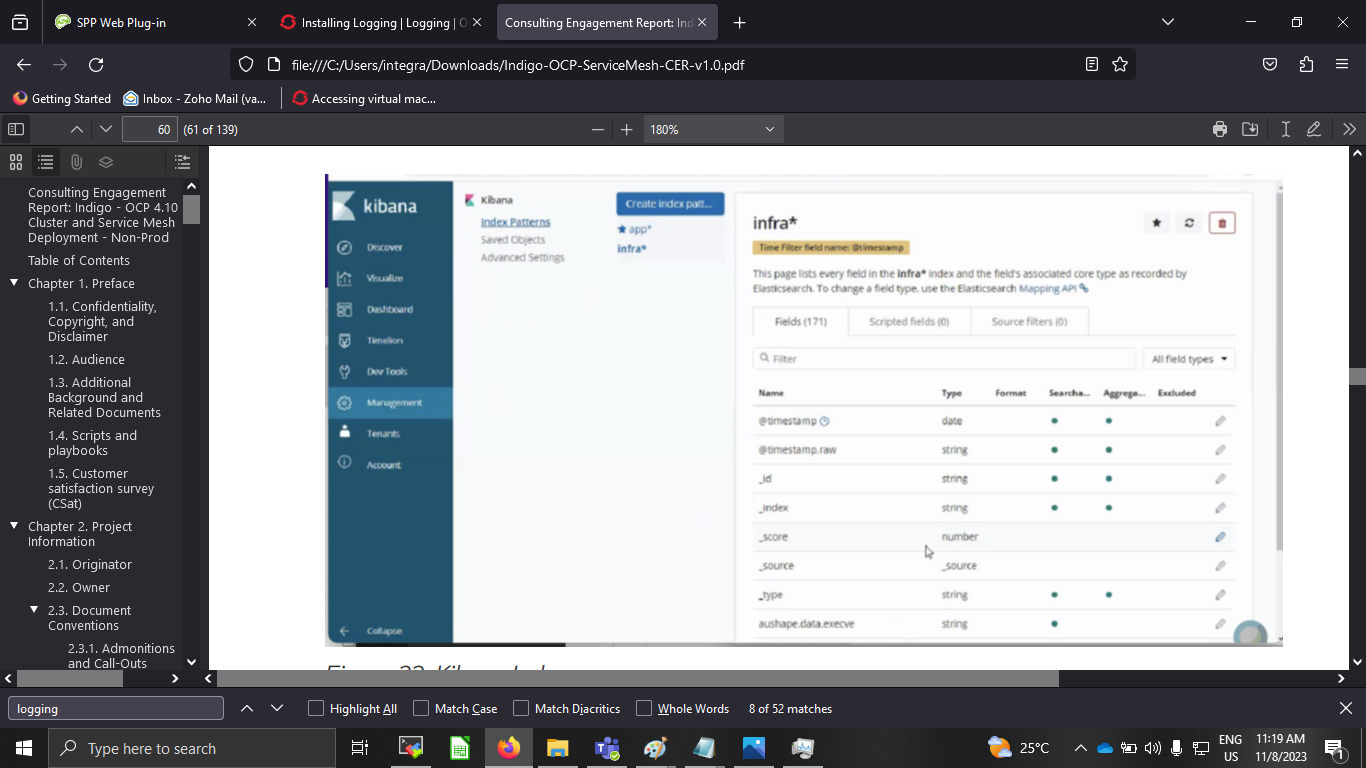
* Verify PVCs created for Elasticsearch components

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ oc get pvc -n openshift-logging  NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE  elasticsearch-elasticsearch-cdm-j6bbys15-1 Bound acmdrcelk1 300Gi RWO nfs01 5d3h  elasticsearch-elasticsearch-cdm-j6bbys15-2 Bound acmdrcelk2 300Gi RWO nfs01 5d3h  elasticsearch-elasticsearch-cdm-j6bbys15-3 Bound acmdrcelk3 300Gi RWO nfs01 5d3h |

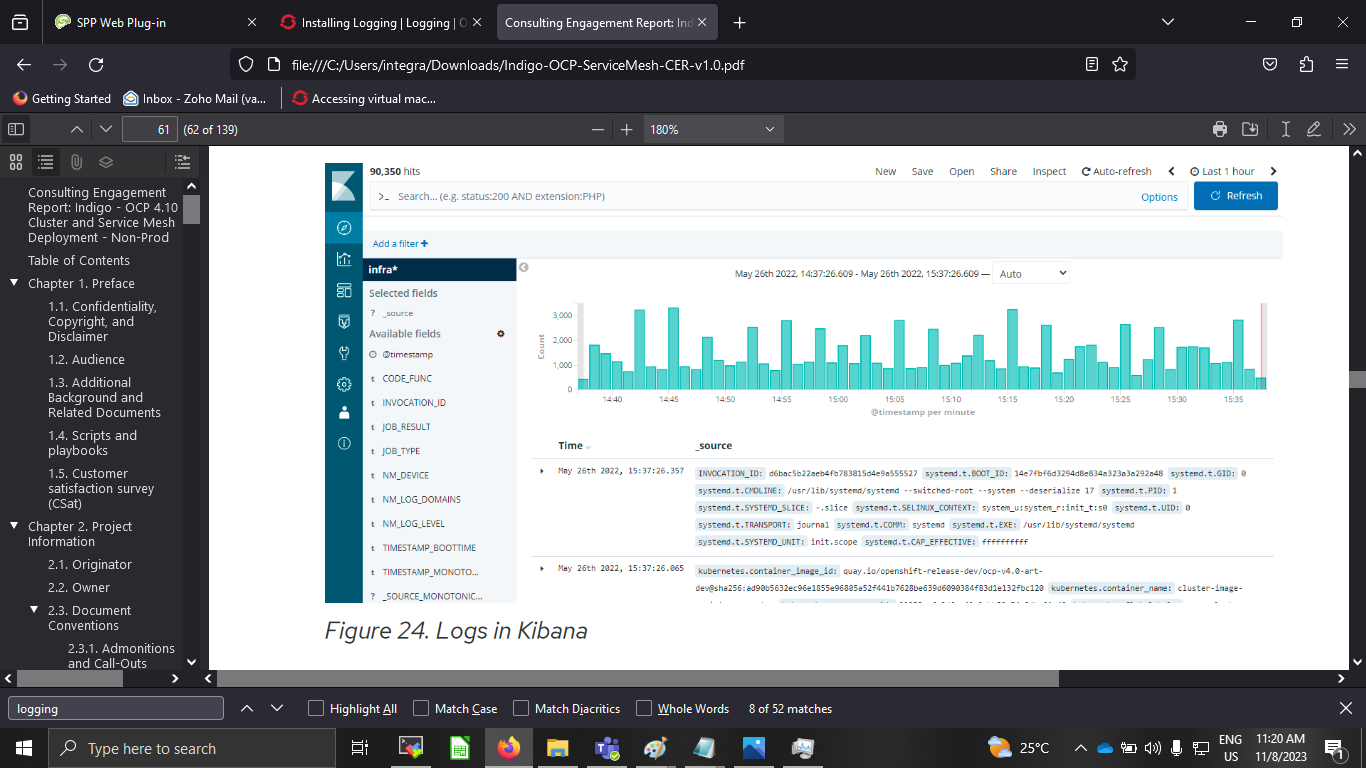
### Viewing cluster logs from the Kibana dashboard

* To define index patterns and create visualizations in Kibana
* In the OpenShift Container Platform console, click the Application Launcher and select Logging.
* Create your Kibana index patterns by clicking Management → Index Patterns → Create index pattern.
  + Users must manually create index patterns to see logs for their projects. Users should create a new index pattern named app and use the @timestamp time field to view their container logs.
  + Admin users must create index patterns for the app, infra, and audit indices using the @timestamp time field.
* Create Kibana Visualizations from the new index patterns.





* To view logs in Kibana
* In the OpenShift Container Platform console, click the Application Launcher and select  
  Logging.
* Log in using the same credentials you use to log in to the OpenShift Container Platform console.
* In Kibana, click Discover.
* Select the index pattern you created from the drop-down menu in the top-left corner: app, audit, or infra. The log data displays as time-stamped documents.
* Expand one of the time-stamped documents.



## Applying a custom Alertmanager configuration with the SMTP details

• Print the currently active Alertmanager configuration into a file **alertmanager.yaml**

|  |
| --- |
| [core@drlvsonacmdrcm1]$ mkdir SMTP  [core@drlvsonacmdrcm1 SMTP]$ cd SMTP  [core@drlvsonacmdrcm1 SMTP]$ oc -n openshift-monitoring get secret alertmanager-main --template='{{ index .data "alertmanager.yaml" }}' | base64 -d > alertmanager.yaml |

• Edit the configuration in the exported alertmanager.yaml file

|  |
| --- |
| "global":  "resolve\_timeout": "5m"  "smtp\_from": "CMS.Monitoring@bank-abc.com"  "smtp\_smarthost": "10.150.163.236:25"  "smtp\_require\_tls": false  "inhibit\_rules":  - "equal":  - "namespace"  - "alertname"  "source\_match":  "severity": "critical"  "target\_match\_re":  "severity": "warning"  - "equal":  - "namespace"  - "alertname"  "source\_match":  "severity": "warning"  "receivers":  - "name": "Default"  "email\_configs":  - "to": "GPH.Monitoring@bank-abc.com"  "headers":  "subject": 'CMS-DRC Alert {{ template "email.default.subject" . }}'  "route":  "group\_by":  - "namespace"  "group\_interval": "5m"  "group\_wait": "30s"  "receiver": "Default"  "repeat\_interval": "8h"  "routes":  - "match":  "alertname": "Watchdog"  "receiver": "Default"  - "match":  "severity": "critical"  "receiver": "Default” |

Apply the new edited configuration as shown below.

|  |
| --- |
| $ oc -n openshift-monitoring create secret generic alertmanager-main --from-file=alertmanager.yaml --dry-run -o=yaml | oc -n openshift-monitoring  replace secret --filename=-  secret/alertmanager-main replaced |

## Removing the kubeadmin user

After you define an identity provider and create a new cluster-admin user, you can remove the kubeadmin to improve cluster security.

If you follow this procedure before another user is a cluster-admin, then OpenShift

Container Platform must be reinstalled. It is not possible to undo this command.

**Prerequisites**

* You must have configured at least one identity provider.
* You must have added the cluster-admin role to a user.
* You must be logged in as an administrator.
* Retrieve the kubeadmin secret details as shown below

|  |
| --- |
| [[core@basian](mailto:core@basian)-node ~]$ oc get secrets kubeadmin -n kube-system  NAME TYPE DATA AGE  kubeadmin Opaque 1 97d |

Remove the kubeadmin secret as shown below

|  |
| --- |
| [core@drlvsonacmdrcm1e ~]$ oc delete secrets kubeadmin -n kube-system  secret "kubeadmin" deleted |

Verify if the kubeadmin secret no more exists as shown below

|  |
| --- |
| [core@drlvsonacmdrcm1e ~]$ oc get secrets kubeadmin -n kube-system  Error from server (NotFound): secrets "kubeadmin" not found |

## Moving monitoring components from Worker to Infra labeled nodes

Create a configmap as shown below

|  |
| --- |
| [core@drlvsonacmdrcm1e ~]$vi cluster-monitoring-config.yaml  apiVersion: v1  kind: ConfigMap  metadata:  name: cluster-monitoring-config  namespace: openshift-monitoring  data:  config.yaml: |+  alertmanagerMain:  volumeClaimTemplate:  metadata:  name: alertmanager  spec:  storageClassName: nfs01  volumeMode: Filesystem  resources:  requests:  storage: 5Gi  nodeSelector:  node-role.kubernetes.io/infra: ""  prometheusK8s:  volumeClaimTemplate:  metadata:  name: prometheus  spec:  storageClassName: nfs01  volumeMode: Filesystem  resources:  requests:  storage: 300Gi  retention: 30d  nodeSelector:  node-role.kubernetes.io/infra: ""  prometheusOperator:  nodeSelector:  node-role.kubernetes.io/infra: ""  grafana:  nodeSelector:  node-role.kubernetes.io/infra: ""  k8sPrometheusAdapter:  nodeSelector:  node-role.kubernetes.io/infra: ""  kubeStateMetrics:  nodeSelector:  node-role.kubernetes.io/infra: ""  telemeterClient:  nodeSelector:  node-role.kubernetes.io/infra: ""  openshiftStateMetrics:  nodeSelector:  node-role.kubernetes.io/infra: ""  thanosQuerier:  nodeSelector:  node-role.kubernetes.io/infra: "" |

|  |
| --- |
| [core@drlvsonacmdrcm1e ~]$ oc create -f cluster-monitoring-config.yaml |

Verify if all the core monitoring components have been moved to the dedicated nodes (infra) as shown below

|  |
| --- |
| [core@drlonvsacmdrcmg ~]$ oc get po -o wide  NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES  alertmanager-main-0 6/6 Running 0 3h 10.131.0.10 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  alertmanager-main-1 6/6 Running 0 3h2m 10.128.2.8 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  cluster-monitoring-operator-f5974dd98-2glfr 1/1 Running 0 84m 10.128.0.32 drlonvsacmdrcm1.acmdrc.arabbanking.local <none> <none>  kube-state-metrics-54bbbb5488-t82mx 3/3 Running 0 3h 10.128.2.14 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  monitoring-plugin-6c5cfd47bf-dtgwz 1/1 Running 0 3h 10.129.2.8 drlonvsacmdrcw1.acmdrc.arabbanking.local <none> <none>  monitoring-plugin-6c5cfd47bf-pk855 1/1 Running 0 3h 10.128.2.20 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  node-exporter-2x2ck 2/2 Running 6 6d21h 10.151.24.84 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  node-exporter-4zntd 2/2 Running 6 6d21h 10.151.24.79 drlonvsacmdrcm2.acmdrc.arabbanking.local <none> <none>  node-exporter-6bfgc 2/2 Running 6 6d21h 10.151.24.83 drlonvsacmdrcw2.acmdrc.arabbanking.local <none> <none>  node-exporter-b4tdc 2/2 Running 6 6d21h 10.151.24.82 drlonvsacmdrcw1.acmdrc.arabbanking.local <none> <none>  node-exporter-h2wbh 2/2 Running 6 6d21h 10.151.24.81 drlonvsacmdrcm3.acmdrc.arabbanking.local <none> <none>  node-exporter-hrzwf 2/2 Running 4 6d3h 10.151.24.85 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  node-exporter-j7q54 2/2 Running 6 6d21h 10.151.24.78 drlonvsacmdrcm1.acmdrc.arabbanking.local <none> <none>  openshift-state-metrics-649d97fd7b-xw9vz 3/3 Running 0 3h 10.128.2.12 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  prometheus-adapter-85696d9dfb-4jjgf 1/1 Running 0 3h 10.131.0.7 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  prometheus-adapter-85696d9dfb-nzrwg 1/1 Running 0 3h2m 10.128.2.11 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  prometheus-k8s-0 6/6 Running 0 3h2m 10.128.2.6 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  prometheus-k8s-1 6/6 Running 0 3h 10.131.0.9 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  prometheus-operator-84494bf4c9-2fvr7 2/2 Running 0 3h 10.128.2.13 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  prometheus-operator-admission-webhook-5cf6d4f94d-b27zq 1/1 Running 0 3h2m 10.128.2.5 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  prometheus-operator-admission-webhook-5cf6d4f94d-ll22g 1/1 Running 0 3h 10.131.0.11 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  telemeter-client-6877bfb864-qrxvb 3/3 Running 0 3h 10.128.2.17 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> <none>  thanos-querier-7f844b45bf-f9lnz 6/6 Running 0 3h 10.131.0.8 drlonvsacmdrcw4.acmdrc.arabbanking.local <none> <none>  thanos-querier-7f844b45bf-lg7br 6/6 Running 0 3h2m 10.128.2.10 drlonvsacmdrcw3.acmdrc.arabbanking.local <none> |

## Configuring the default scheduler to control pod placement

Edit the scheduler configuration & add the below stated line under spec section and save the configuration.

|  |
| --- |
| [core@drlvsonacmdrcm1e ~]$ oc edit scheduler cluster  ...  spec:  defaultNodeSelector: node-role.kubernetes.io/worker=  ... |

## Etcd backup Cronjob

Create Backup Cronjob using below yaml and apply the yaml.

|  |
| --- |
| kind: CronJob  apiVersion: batch/v1  metadata:  name: openshift-backup  namespace: ocp-etcd-backup  labels:  app: openshift-backup  spec:  schedule: "\* \* \* \* \*"  concurrencyPolicy: Forbid  successfulJobsHistoryLimit: 5  failedJobsHistoryLimit: 5  jobTemplate:  metadata:  labels:  app: openshift-backup  spec:  backoffLimit: 0  template:  metadata:  labels:  app: openshift-backup  spec:  containers:  - name: backup  image: "registry.redhat.io/openshift4/ose-cli"  command:  - "/bin/bash"  - "-c"  - oc get no -l node-role.kubernetes.io/master --no-headers -o name | head -n 1 |xargs -I {} -- oc debug {} --to-namespace=ocp-etcd-backup -- bash -c 'chroot /host rm -rf /home/core/backup && chroot /host mkdir /home/core/backup && chroot /host sudo -E mount -t nfs <nfs-server-IP>:<shared-path> /home/core/backup && chroot /host sudo -E /usr/local/bin/cluster-backup.sh /home/core/backup && chroot /host sudo -E find /home/core/backup/ -type f -mmin +"1" -delete'  restartPolicy: "Never"  terminationGracePeriodSeconds: 30  activeDeadlineSeconds: 500  dnsPolicy: "ClusterFirst"  serviceAccountName: "openshift-backup"  serviceAccount: "openshift-backup" |

**After creating CronJob, you can force the execution for validation with the command:**

|  |
| --- |
| oc create job bckup --from=cronjob/openshift-backup |

**Thank You**

1. Visit us at www.integramicro.com