IBM Cloud Pak for Business Automation Demos and Labs 2022

Bring-up Lab

Thomas Schulze
ThomasSchulze@de.ibm.com

Matthias Jung
matthias.jung@de.ibm.com

Jorge D. Rodriguez
jorgedr@us.ibm.com

Zhong Tao Gao
gaozt@cn.ibm.com

V 1.0

NOTICES

This information was developed for products and services offered in the USA.

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

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

> IBM Director of Licensing
> IBM Corporation
> North Castle Drive, MD-NC119
> Armonk, NY 10504-1785
> United States of America

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

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

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

TRADEMARKS

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

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

IT Infrastructure Library is a Registered Trade Mark of AXELOS Limited.

ITIL is a Registered Trade Mark of AXELOS Limited.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

© Copyright International Business Machines Corporation 2020.

This document may not be reproduced in whole or in part without the prior written permission of IBM.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Table of Contents

1 Introduction

1.1 IBM Cloud Pak for Business Automation

IBM Cloud Pak for Business Automation (CP4BA) assembles certified software from the IBM Automation Platform for Digital Business on multiple cloud infrastructures. It offers design, build, run, and automation services to rapidly scale your programs and fully execute and operationalize an automation strategy.

You can read more about CP4BA here: **https://www.ibm.com/docs/en/cloud-paks/cp-biz-automation/21.0.3?topic=overview-what-is-cloud-pak-business-automation**

1.2 Lab Overview

In this lab, you will learn how to **configure and install the CP4BA Production mode** on an OpenShift cluster using our **rapid deployment scripts**.

The rapid deployment scripts **simplify** the configuration and installation of CP4BA. They are available on **public github**, and therefore can be used by customers, business partners and IBMers:

https://github.com/IBM/cp4ba-rapid-deployment

They automate **parts** of the official **CP4BA planning and installation steps** that can be found here for CP4BA version 21.0.3:

Planning for a production deployment:
https://www.ibm.com/docs/en/cloud-paks/cp-biz-automation/21.0.3?topic=planning-production-deployment

Installing production deployments (on Red Hat OpenShift):
https://www.ibm.com/docs/en/cloud-paks/cp-biz-automation/21.0.3?topic=openshift-installing-production-deployments

The rapid deployment scripts are available for the following **CP4BA versions**:
- 21.0.1
- 21.0.2
- 21.0.3

As part of this lab, you will configure and install version **21.0.3**.

The github also contains **documentation** on how to bring up CP4BA **from scratch** on IBM Cloud using a ROKS cluster. In that case, you:

- First must decide what you want to deploy – multiple templates that provide you a predefined set of CP4BA capabilities are available – these define the sizing requirements for your environment (partly automates the planning step for you)
- Second need an IBM Cloud account
- Third create new Red Hat OpenShift cluster on IBM Cloud
- Fourth must create a VM on IBM Cloud that hosts the LDAP (SDS)

- Fifth deploy DB2 on your OpenShift cluster (automates part of the installation step for you)

  and finally

- Sixth deploy CP4BA on your OpenShift cluster (automates part of the installation step for you)

- In addition, you require a bastion host to access the OpenShift cluster by command line and execute the scripts.

Other optional steps, for example to deploy Process Mining, are also documented and scripted by this repository.

As part of this Lab, you will only **deploy DB2 and CP4BA**. All other steps are already taken care of for you so that you can concentrate on the most important part to learn: **Configure and Install IBM Cloud Pak for Business Automation version 21.0.3** on Red Hat OpenShift.

For this, you will reserve a TechZone environment preconfigured with a bastion host and a three-worker node Red Hat OpenShift cluster. In addition, an LDAP server will be installed and configured on the bastion host so that you can start the CP4BA deployment without worrying about the LDAP services required by the CP4BA installation. Given the size of the OpenShift cluster provided, we have also pre-selected the template that will be used by the rapid deployment scripts during the lab. This template will install foundational services required by CP4BA as well as Filenet Content Manager components. With that you can easily complete the lab in a self-paced manner without having to worry about the sizing requirements for the installation.

IMPORTANT: The primary use for those scripts and templates is for rapid set-up of CP4BA DEMO and ENABLEMENT environments, for example to host the Client Onboarding Demo. Those scripts and templates should not be used as is to set-up customer environments for development, test, pre-production or production as such installations might have for example stronger security requirements.[1] For such deployments use the official CP4BA documentation that can be found here: **https://www.ibm.com/docs/en/cloud-paks/cp-biz-automation**

During the first exercise you will reserve an environment on TechZone, access it and verify that your LDAP and RedHat OpenShift cluster are working properly.

As part of the second exercise, you will then deploy the DB2 operator and cluster on your OpenShift environment. In addition, you will create all needed databases.

The third exercise will guide you through configuring and installing CP4BA version 21.0.3.

Finally, a chapter with troubleshooting instructions is available for you to work through.

Approximate Duration: 8 hours

---

[1] Communication to the Database and to the LDAP server will for example might require encrypted communication in such environments, whereas the scripts and templates might not encrypt all internal communication.

2 Exercise: Prepare yourself for this Bring-Up Lab

2.1 Introduction

As part of this exercise, you will verify that all prerequisites are in place, and you will be instructed on how to reserve and access your lab environment.

The lab environment that you will reserve consists of a bastion host and a Red Hat OpenShift Container Platform (OCP) cluster.

A bastion host is the computer from which you access the OpenShift cluster through command line to administrate the OCP cluster. Administration of an OCP cluster also includes the configuration and installation of new software such as CP4BA . In addition, the rapid deployment scripts and some scripts from the product will get executed on that bastion host, therefore the bastion host must be one of RHEL, CentOS or macOS. Finally, all commands needed by these scripts must be available on the bastion host , for example the OpenShift CLI, Kubernetes CLI, and so on. To save you time and trouble with setting up your own bastion host, **a RHEL VM is made available for you** together with the OpenShift cluster.

The OpenShift cluster will host all DB2 and CP4BA containers needed. OCP version 4.8.41 is used. From the bastion host you can access your OCP cluster either by command line (oc command), or you can access the OpenShift Web Console by Browser.

2.2 Exercise Instructions

Before you can start this lab, you need the following prerequisites:
- An entitlement key, and
- An environment with a **bastion host** and **OpenShift cluster** from TechZone.

On the bastion host, some tools need to be available, among them the Openshift Command line interface, and podman for checking the entitlement key.

1. To be able to complete this lab you **must have an entitlement key already available** that includes a license for CP4BA to pull the container images from IBM's container library cp.icr.io. As a first step, please check that you have such an entitlement key with a license for CP4BA available. For this, please **open https://myibm.ibm.com/products-services/containerlibrary** and log in with your **IBM ID**. IBMers can use their **w3 ID**.

   Then, switch to the **View library** page. **IBMers** should see the following, what means that they **do have such an entitlement key available**:
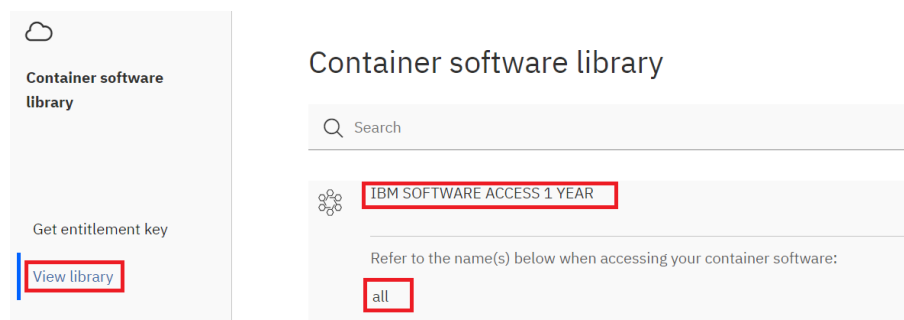
Business Partners before proceeding **must verify that in their list a license for CP4BA is included**. Without this entitlement / license, you can't perform this bring-up lab, which means you must stop here.

Switch back to **Get entitlement key** and leave this page open. You will require the key multiple times while this lab.

2. To get hold of an **OpenShift cluster** from TechZone, access IBM Technology Zone: **https://techzone.ibm.com/**

3. To sign-in, either **use your IBM ID or your company credentials** if SSO is set up between your company and IBM. For example, if you have an IBM W3 or IBM Partnerworld ID, you should use this ID.

    Note: Dependent of the IBM ID used to sign-in, you will be able to reserve a demo or not. If you can't reserve a demo with the current IBM ID, check if you have another IBM ID that is enabled for IBM Technology Zone reservations.

4. Once signed in, open the **lab resource page**:

    https://techzone.ibm.com/collection/ibm-cloud-pak-for-business-automation-demos-and-labs-bring-up-lab

5. Scroll down to the **Environments** section and click on the tile shown.

    Environments

    Jun 20, 2022
    IBM Cloud Pak for Business Automation - Demos and Labs - Bring-up Lab

    Skytap: EMEA, US-Central

    IBM Cloud Pak for Business Automation - Demos and Labs

    Visibility
    IBMers, Business Partners
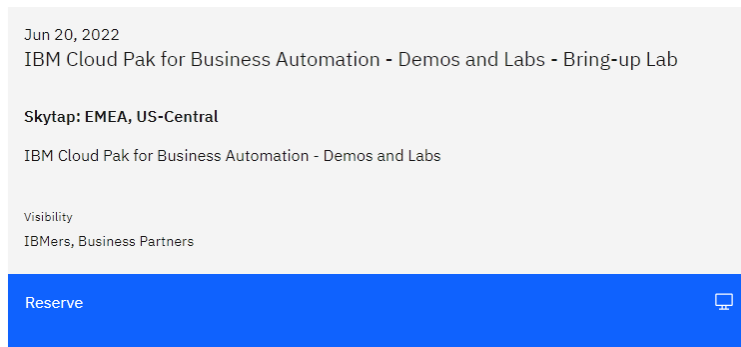
    Reserve

    Note: If you don't see the tile, go back to step 4 and sign-in with a different ID.

6. Select Reserve now.

Select your reservation type. Do you need this now or later?

Single environment reservation options:

◯ Reserve now
◯ Schedule for later

Request multiple environments:

Leverage workshop manager to request multiple environments for your upcoming hands-on workshop and manage them all in one place.
All workshop requests must be submitted 72-hours before the intended workshop start date/time. Any request before the 72-hour time frame will not be able to submit this request form.

◯ Schedule a workshop

| Cancel | Reset | Submit 🖫 |

7. On the next page provide the necessary information: Select your nearest GEO, provide the purpose, e.g. "Practice / Education" a description and the end time and date for the reservation. Plan for at least 8 hours, maybe more. Then click **Submit**.

8. After you click **Submit**, you'll get an Email, that provides you a **Desktop URL** and a **Desktop password** to access the environment.

   Once you get the Email with the link and password to access your **lab environment**, you can start your lab.

9. Open the link and enter your password to access your environment. Click **Submit**.

10. Your demo and lab environment opens. It will look like the following picture:

Note: If you are not going to be working with your environment for an extended period of time, you must **do a clean shut down of your environment**. If you don't do so, after a certain period of inactivity, your environment will be shut down automatically. Under some circumstances this can cause issues when you try to restart it. Therefore, always make sure to perform a clean shut down. You will get detailed instructions on this later in this document.

Now that you have your demo and lab environment available, learn how to work with your environment and the VMs before you start with the lab.

11. Your environment consists of the following VMs:

    VM 0 – Client: This Red Hat Enterprise Linux (RHEL) Server VM is the bastion host and provides access to the Red Hat OpenShift Container Platform (OCP) master and infrastructure servers. In addition, a DNS server, IBM SDS (LDAP) and other required services are also running there.

    VM 1 – master-0, VM 2 – master-1 and VM 3 – master-2: On this Red Hat Enterprise Linux CoreOS (RHCOS) Server VMs, the Red Hat OpenShift Container Platform (OCP) master and infrastructure services are running. In addition, these VMs function as the compute nodes. Here, any containers installed on top of OCP are running. These VMs do not have a desktop, they can't be accessed directly.

12. All VMs are initially in **Powered off** state. To start the VMs in your environment in the correct order and with the needed delays, click one of the two **Run VM(s)** buttons marked with green rectangles. Do **NOT** use the **Run this VM buttons**.
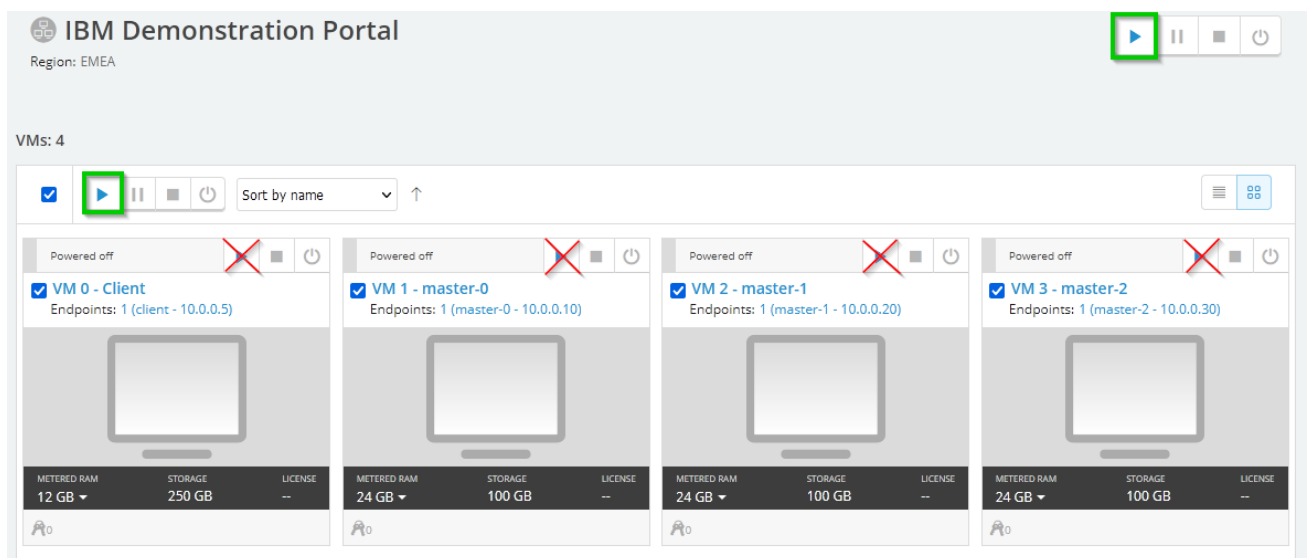
It will take a while until your environment fully starts, wait until the VMs are in Running state.
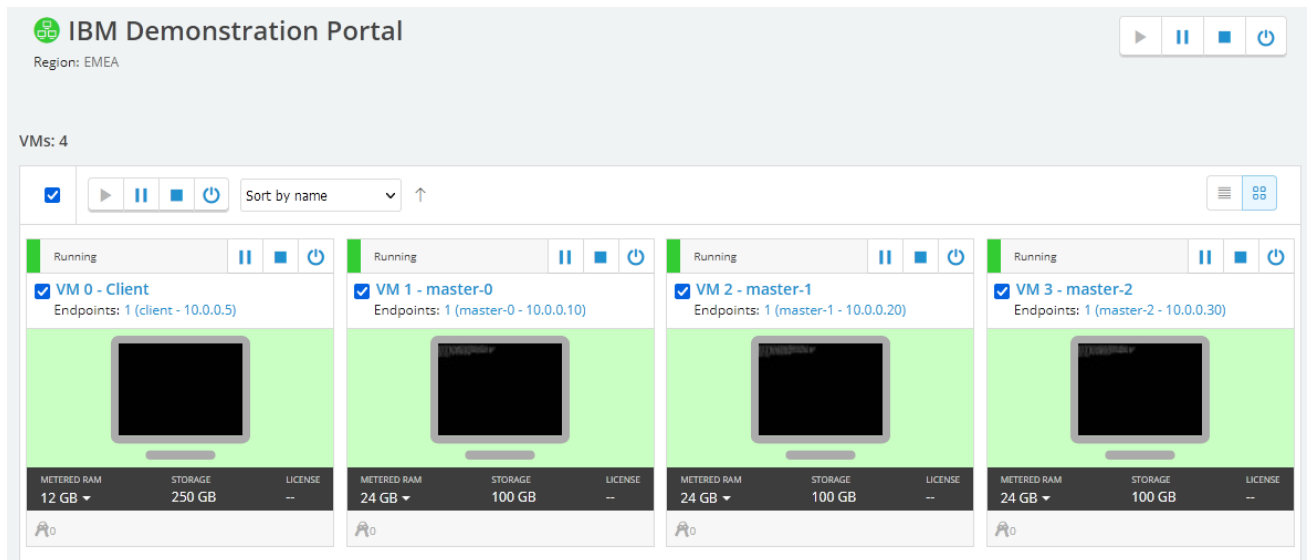
13. Access the desktop of VM 0 through the Browser by **clicking on the picture of the monitor** (Access this VM)
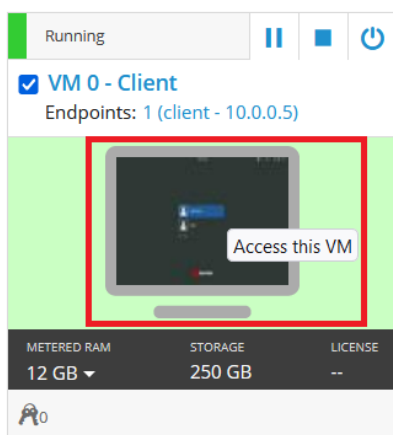
14. The desktop of VM 0 opens in a new Browser tab.

15. Next, you need to **log in**. In case you see the current time and date, **press the Space key** to get to the log-in screen. If the screen is blank, **move the mouse**.

16. On the log-in screen select **cp4badmin**.
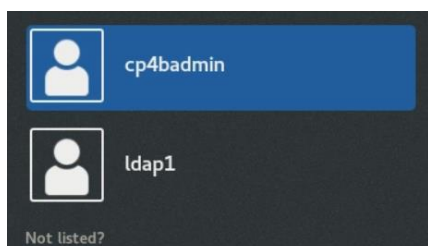
17. Enter password **passw0rd** (where the third-last character is a zero) and sign in.

18. You now see the Red Hat Enterprise Linux (RHEL) Server desktop of VM 0.

19. Next, make sure the VM is **correctly connected to the network** before you proceed using it. Check that the network icon shows connected. If the VM is not correctly connected to the network the environment is not usable. If not correctly connected, re-start the entire environment.

    Connected: (Not connected:)

20. Change the resolution of the desktop to your liking. Either click the **Fit to Window** icon or the **Change Resolution** icon on the top of the VM's desktop.

    or
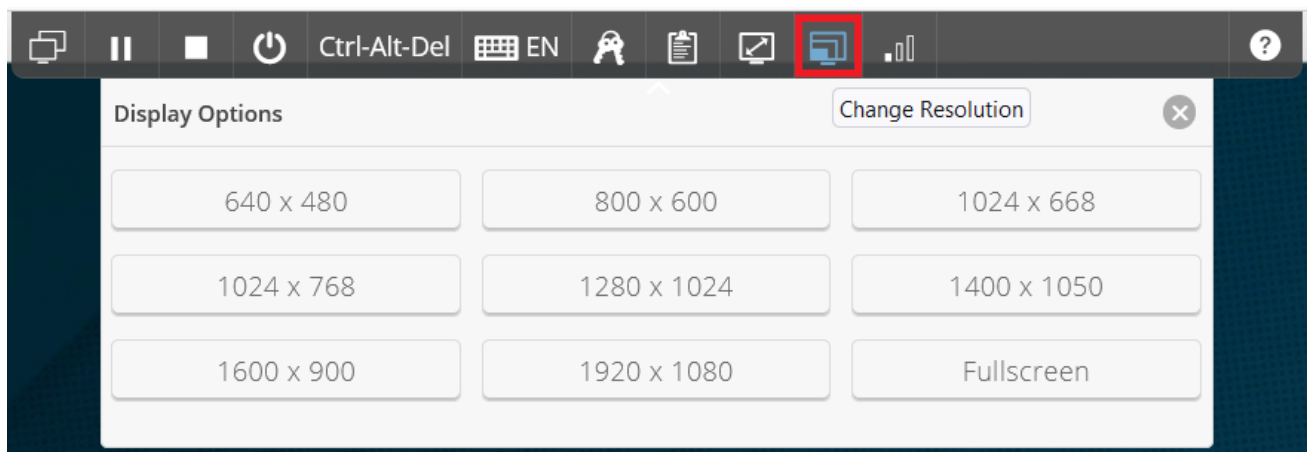
21. To change the keyboard to your liking you must first click the **en** icon **in the top right corner of the desktop** and select your preferred keyboard layout.
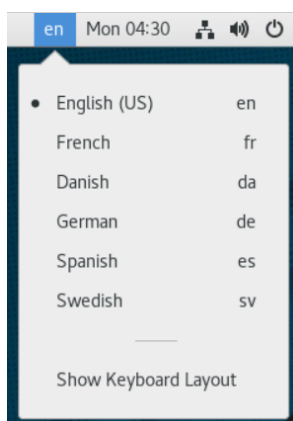
Second you must select the same keyboard layout in the top bar.
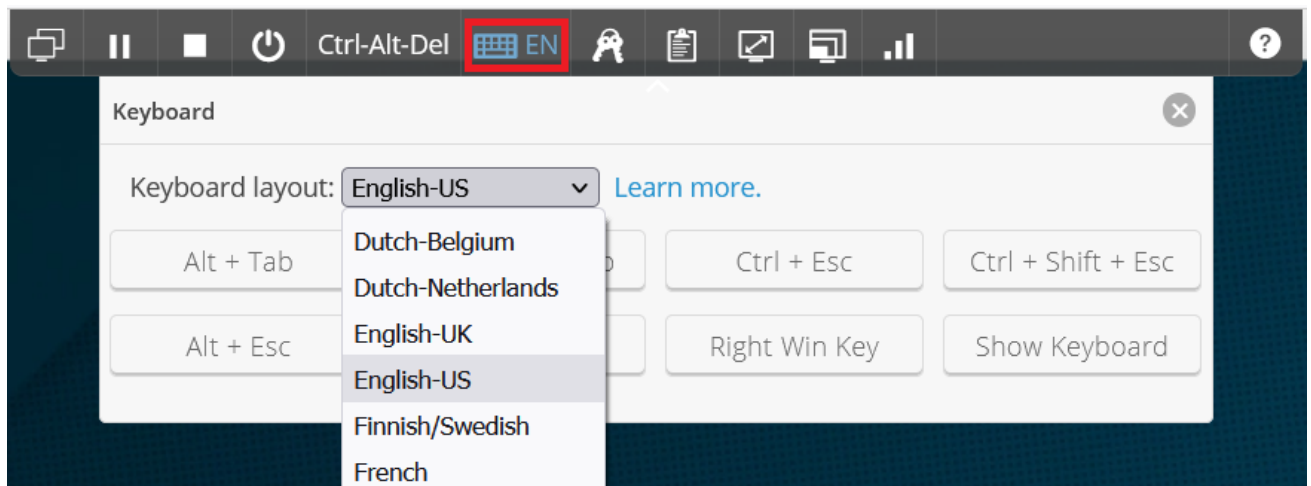
To check that your keyboard works as expected, open Firefox and enter in the URL field some special characters of your language.
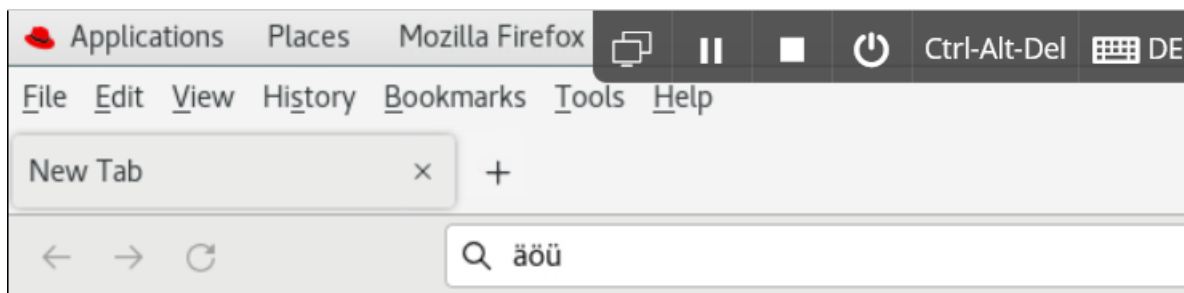
22. Finally check the connection from your local machine to the VM.
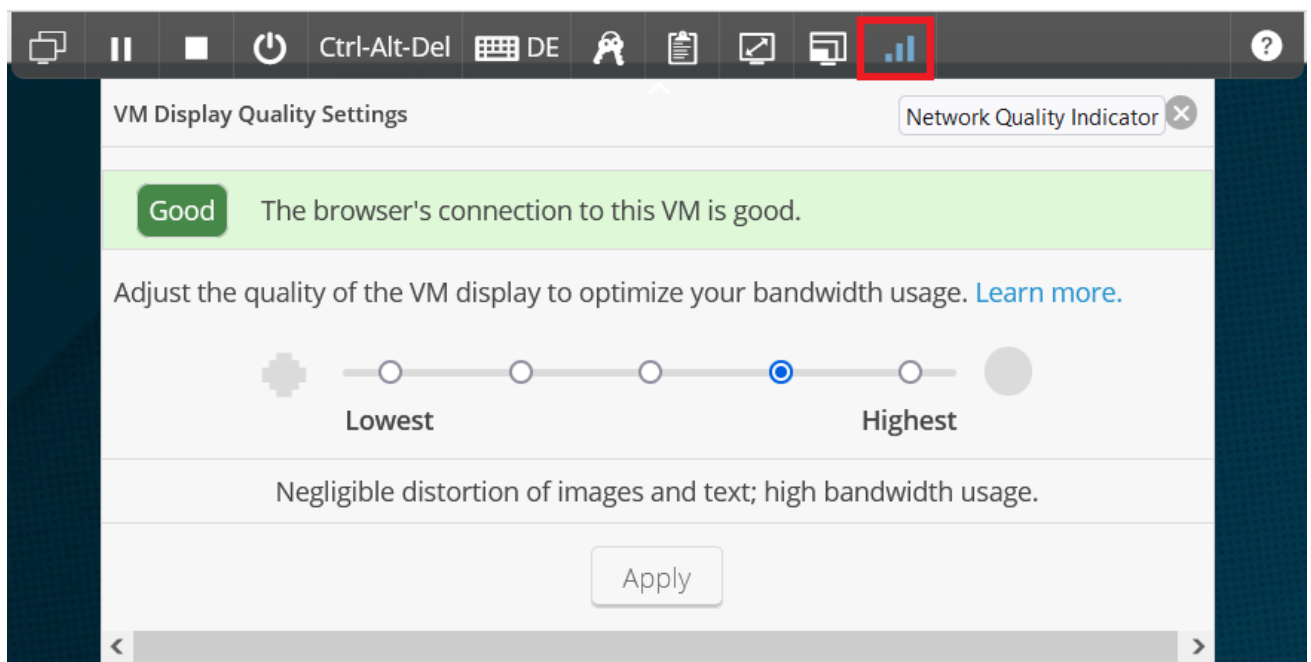
If the Network Quality Indicator shows yellow or red quality, try a different network with better

quality on your local machine.

23. If you are not working on your environment for a long period of time, you must **do a clean shut down of the environment**. If you don't do so, after a certain period of inactivity, your environment will be shut down automatically. Under some circumstances this can cause issues when you try to restart it. Therefore, **always make sure to perform a clean shut down**.

   To perform a clean shut down of your environment, first make sure **any running script got completed** and **all changes on the OpenShift cluster applied by that script are complete**. This means, although the script got completed, it might be that the operators running on OpenShift are still applying the changes. In this lab documentation you will find validation steps that indicate when this is the case.

   Second, save your work on VM 0 and close the Browser tab with the desktop of the VM.

   Third, on the Browser tab showing your entire environment, make sure all VMs are selected and hit the Shutdown VM(s) button. **However, don't do that now!**
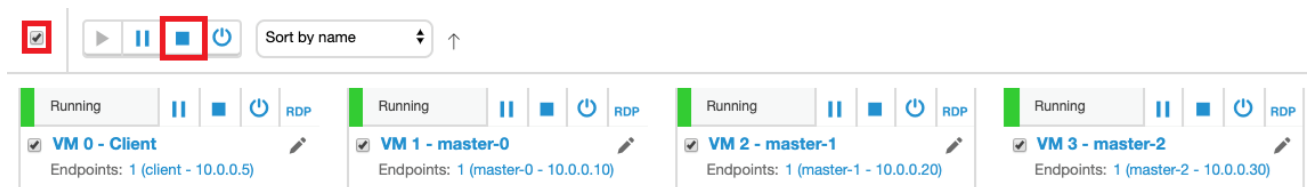
24. You can Copy&Paste text into the Virtual Machine by clicking the clipboard icon and pasting the text into the resulting box. After you do that, you can right-click on the terminal window and paste the text. (If it does not work the first time, just try again)
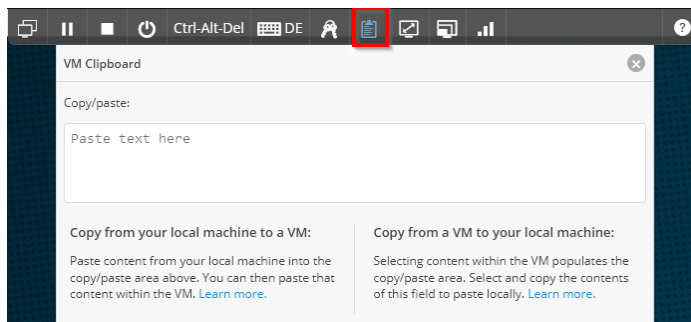

25. After **each start** of the environment, **including the very first time the environment is started**, you must manually **start SDS first**. On the desktop of VM 0 open the Terminal and run the following command:

```
sudo /root/bin/startSds.sh
```

   When you are asked for the password, enter **passw0rd** (where the third-last character is a zero).

26. To access your OpenShift cluster through a Browser, **open Firefox** first (shortcut on the desktop).

27. In Firefox, open bookmark **OpenShift Web Console**.

28. Log in with **admin / passw0rd** (where the third-last character is a zero).

29. Verify that the OpenShift web console opens and that you have **Administrator** access.
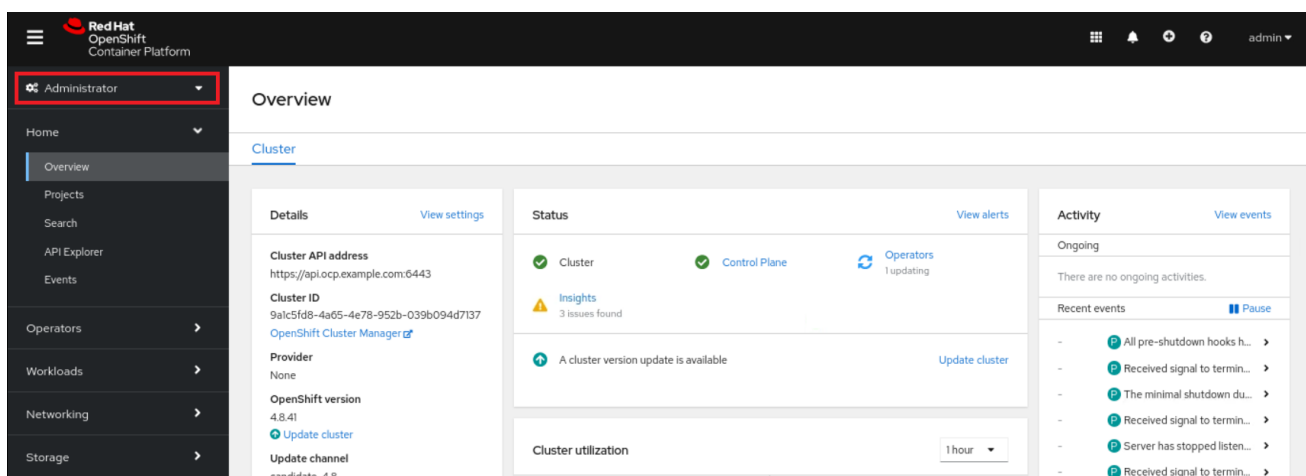
30. To allow for smooth start-up, updates have been paused. First you must **allow updates**, otherwise the OCP environment might become unusable for you. This is because the environment is shut down for a longer period where no updates are installed. A normal OCP cluster is running 24/7 and will not hit such issues, those are specific to this lab environment.

31. On the left-hand side, scroll down to **Compute** and select **MachineConfigPools**.
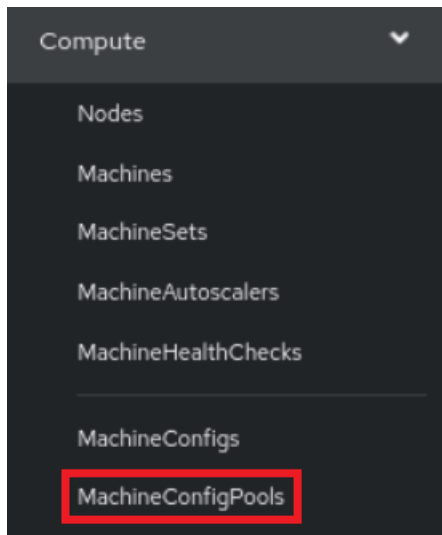
32. For both, **master** and **worker**, resume the updates.
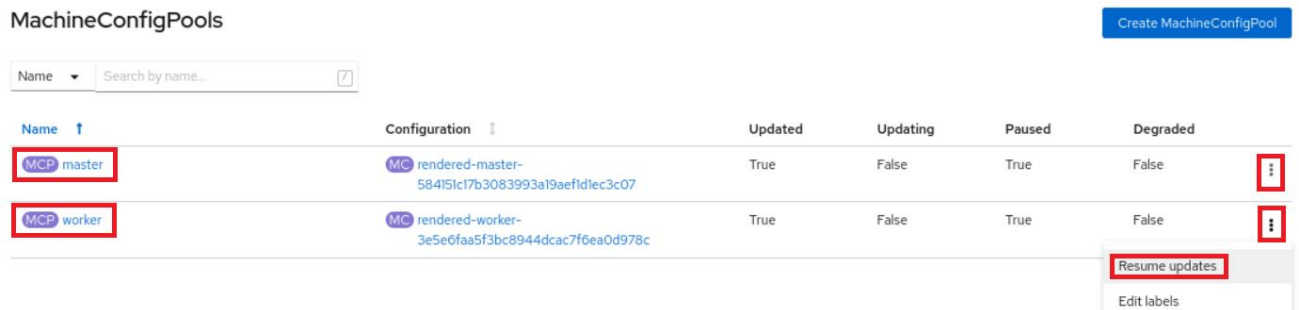
33. On the left-hand side, scroll up to **Home** and select **Overview**.

34. Wait until the status of **Cluster**, **Control Plane** and **Operators** is green.

35. On the Overview main page scroll down and verify that the **Cluster inventory** shows no errors.

Cluster inventory

3 Nodes

164 Pods

1 StorageClass

0 PersistentVolumeClaims

In case of **errors, warnings or pending indicators**, consult the Troubleshooting section to resolve those.

36. To log-in through the **oc command line interface**, expand **admin** in the top right corner and select **Copy login command**.

admin ▾

Copy login command ⬈

Language preference

Log out

37. A new tab opens, log-in again with **admin / passw0rd** (where the third-last character is a zero) and select **Display Token**.

38. Copy the entire **oc login** command to the clipboard.

Log in with this token

oc login --token=sXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX --server=https://api.ocp.example.com:6443

39. Switch to your **Terminal window** within VM 0, **paste the content of the clipboard** and hit **Enter**.

```
[cp4badmin@client ~]$ oc login --token=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX--server=https://api.ocp.example.com:644
3
Logged into "https://api.ocp.example.com:6443" as "admin" using the token provided.

You have access to 64 projects, the list has been suppressed. You can list all projects with 'oc projects'

Using project "default".
[cp4badmin@client ~]$
```

Note: After **each start** of the environment, **including the very first time the environment is started**, you must log-in through the **oc command line interface**.

40. Verify that you successfully logged in.

41. To verify the oc command line interface and OCP cluster **version** fit together run this command:

```
oc version
```

```
[cp4badmin@client ~]$ oc version
Client Version: 4.8.0-202205121606.p0.g41ff67e.assembly.stream-41ff67e
Server Version: 4.8.41
Kubernetes Version: v1.21.11+6b3cbdd
[cp4badmin@client ~]$
```

With that you have successfully accessed your environment and are ready to start deploying DB2.

3 Exercise: Deploy IBM DB2 & create needed databases

3.1 Introduction

A CP4BA production mode deployment requires you to provide the databases needed by the CP4BA components being installed. In this exercise you'll first **deploy the DB2 operator and containers** on the OpenShift cluster. Other databases are supported too, for example Oracle, SQL Server or PostgreSQL. All supported databases can be found in the **Detailed system requirements**:

https://www.ibm.com/docs/en/cloud-paks/cp-biz-automation/21.0.3?topic=ppd-system-requirements

The database can run on OpenShift, but on-premises installations are also supported by CP4BA.

Once the DB2 operator and containers are installed on your OpenShift cluster, you will also **create the needed databases**.

3.2 Exercise Instructions

1. On your bastion host **VM 0** use the **Terminal** to switch to a directory prepared for you to contain the rapid deployment and other scripts.

   ```
   cd /cp4ba
   ```

2. **Clone** the rapid deployment git repository onto your bastion host and copy the deployment scripts.

   Note: As part of this doc, it is assumed that you copy the deployment scripts into directory `mycluster` . If you select a different directory, you must change some of the commands provided in the steps below, otherwise you can simply copy/paste them.

   ```
   git clone https://github.com/IBM/cp4ba-rapid-deployment.git
   cd /cp4ba/cp4ba-rapid-deployment/cp4ba-21-0-3
   cp -r scripts mycluster
   cd mycluster/deployment-db2-cp4ba
   ```

   Sample output of the previous commands:

```
[cp4badmin@client ~]$ cd /cp4ba
[cp4badmin@client cp4ba]$ git clone https://github.com/IBM/cp4ba-rapid-deployment
Cloning into 'cp4ba-rapid-deployment'...
remote: Enumerating objects: 1022, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 1022 (delta 3), reused 3 (delta 1), pack-reused 1008
Receiving objects: 100% (1022/1022), 85.17 MiB | 30.08 MiB/s, done.
Resolving deltas: 100% (526/526), done.
[cp4badmin@client cp4ba]$ cd /cp4ba/cp4ba-rapid-deployment/cp4ba-21-0-3
[cp4badmin@client cp4ba-21-0-3]$ cp -r scripts mycluster
[cp4badmin@client cp4ba-21-0-3]$ cd mycluster/deployment-db2-cp4ba
[cp4badmin@client deployment-db2-cp4ba]$
```

3. **Open** the parameters file for DB2, `01-parametersForDb2OnOCP.sh`, with `gedit`.

   `gedit 01-parametersForDb2OnOCP.sh`

4. **Apply** the following changes:

   - `cp4baTemplateToUse` – Name of CP4BA deployment template that will be used

     set the value to `ibm_cp4a_cr_template.002.ent.FoundationContent.yaml`

     Note: The rapid deployment scripts provide multiple CP4BA templates from which you can choose from. This "FoundationContent" template for example deploys the foundation and the content patterns of CP4BA. Other templates deploy different patterns, and thus can be used when other components are needed.
     The ClientOnboardingDemo pattern, for example deploys additionally the Application, Decisions_ads and Workflow patterns and brings with it most components which are needed to run the Client Onboaring Demo.[2]
     Due to the size of the for you available OpenShift cluster, other templates can't be used for this bring-up lab. Of course, if you later deploy on your own OCP cluster and that provides more CPU and memory, you would also be able to deploy larger templates.
     The template name is important at this point already as it defines the needed databases and through that the CPU and memory needed for the DB2 container. It also contains a definition of the databases that need to be generated by the scripts.

   - `db2OnOcpProjectName` – Project/namespace where DB2 should be installed, for example `ibm-db2` (you could specify any project name here, if that does not exist yet, it will be created by the scripts)

     set the value to `ibm-db2`

   - `db2AdminUserPassword` – Password that will be assigned to the db2 instance user e.g., `passw0rd` (you could specify any password here)

---

[2] Newer versions of the Client Onboarding Demo also make use of the document-processing pattern, for example.

set the value to `passw0rd`

- `db2StandardLicenseKey` – A DB2 Standard License key to assign to DB2 to allow for larger use of CPU and memory

  remove the default value `REQUIRED`, leave this parameter empty

  Note: The CP4BA template you are deploying as part of this lab only requires three databases. DB2 community edition (installed when you don't specify a DB2 Standard License key) allows for enough CPU and memory for those three databases.
  Only when you would select a larger CP4BA template this key would be needed.

- `db2Cpu` – Number of CPUs for DB2 pod according to the selected CP4BA template

  set the value to `4`

- `db2Memory` – Amount of memory for DB2 pod according to the selected CP4BA template

  set the value to `16Gi`

```
# Selected CP4BA template to use for deployment, for example ibm_cp4a_cr_template.100.ent.ClientOnboardingDemo.yaml
#   Available templates:
#     ibm_cp4a_cr_template.001.ent.Foundation.yaml
#     ibm_cp4a_cr_template.002.ent.FoundationContent.yaml
#     ibm_cp4a_cr_template.200.ent.ClientOnboardingDemoWithADP.yaml
cp4baTemplateToUse=ibm_cp4a_cr_template.002.ent.FoundationContent.yaml

# OCP Project Name for DB2, for example ibm-db2
db2OnOcpProjectName=ibm-db2

# Password for DB2 Admin User (Admin User name see below), for example passw0rd
db2AdminUserPassword=passw0rd

# DB2 Standard license key base64 encoded
#   If this key is not available, leave empty (but remove the value 'REQUIRED') - then the Community edition is used that allows less CPU & RAM
#   In that case, also update parameters db2Cpu and db2Memory below (the defaults there assume you have a DB2 Standard license available)
db2StandardLicenseKey=

# CPUs to assign to DB2 pod (max with DB2 Standard license is 16, max with Community edition is 4)
#   If you selected CP4BA template    ibm_cp4a_cr_template.001.ent.Foundation.yaml                set it to 4
#   If you selected CP4BA template    ibm_cp4a_cr_template.002.ent.FoundationContent.yaml          set it to 4
#   If you selected CP4BA template    ibm_cp4a_cr_template.200.ent.ClientOnboardingDemoWithADP.yaml set it to 16
db2Cpu=4

# RAM to assign to DB2 pod (max with DB2 Standard license is 128Gi, max with Community edition is 16Gi)
#   If you selected CP4BA template    ibm_cp4a_cr_template.001.ent.Foundation.yaml                set it to 16Gi
#   If you selected CP4BA template    ibm_cp4a_cr_template.002.ent.FoundationContent.yaml          set it to 16Gi
#   If you selected CP4BA template    ibm_cp4a_cr_template.200.ent.ClientOnboardingDemoWithADP.yaml set it to 110Gi
db2Memory=16Gi
```

- Scroll further down to property named `cp4baDeploymentPlatform` – specifies the OpenShift deployment platform, `ROKS` and `OCP` are supported values

  change the value from `ROKS` to `OCP`

- Scroll further down to property named `db2OnOcpStorageClassName` – specifies the name of the storage class to be used for dynamic storage provisioning

  change the value from `cp4a-file-delete-gold-gid` to `cp4ba-storage`

  Note: CP4BA needs a storage class supporting storage in read/write many(RWX) mode. Storage in RWX mode can be mounted on different OCP Workers at the same time. For more details refer to https://www.ibm.com/docs/en/cloud-paks/cp-biz-automation/21.0.3?topic=deployment-storage-considerations

  The Storage class `cp4ba-storage` supporting also this mode has been created for you

based on the [NFS Subdir External Provisioner](). The NFS server providing the storage for this storage class is running on VM 0, your bastion host. There, directory `/nfs/cp4auto-storage` will contain the directories created for the containers to store their files, for example log files.

```
# Deployment platform, either ROKS or OCP
cp4baDeploymentPlatform=OCP

# Name of the storage class used for DB2's PVC
db2OnOcpStorageClassName=cp4ba-storage
```

- Leave all other properties at their default values

   Note: With the other parameters you can tune further settings, like the name of the administrator user, the name of the storage provisioner, and the size of the persistent storage volume to create. For the environment on ROKS, the storage size also determines the input/output performance of the filesystems, therefore we recommend setting the size to a larger value on ROKS environments.

- Finally save you changes and close the editor

5. **Run** script `02-createDb2OnOCP.sh` . This script will now install and configure DB2 for you based on the parameters previously specified in `01-parametersForDb2OnOCP.sh` .

```
./02-createDb2OnOCP.sh
```

   Notes:
- To successfully run the script, you need the `jq` tool and `podman` installed on your bastion host. For this lab these tools are already available on your bastion host.
- This script will prompt you for various input. Please have a look at the sample script output provided below to see the answers you must use, they are marked in red.
- You need your Entitlement Registry key handy, see also **https://myibm.ibm.com/products-services/containerlibrary**
- When you paste your Entitlement Registry key it will not be shown, therefore paste it just once and hit the Enter key.
- This script will exit if errors are hit during the installation.
- If you get the error message:

   ```
   error: unable to recognize "ibmOperatorCatalog.yaml": Unauthorized
   ```

   check weather you have successfully logged on to the OCP cluster on the command line.

   Sample script output:

```
./02-createDb2OnOCP.sh
```

```
Found 01-parametersForDb2OnOCP.sh.  Reading in variables from that script.

  Reading 01-parametersForDb2OnOCP.sh ...
Done!


Installing DB instance for CloudPak.


This script installs Db2u on OCP into project ibm-db2. For this, you need the jq
tool installed and your Entitlement Registry key handy.


Do you want to continue (Yes/No, default: No): Yes


Installing Db2U on OCP...


Installing the IBM Operator Catalog...

Warning: resource catalogsources/ibm-operator-catalog is missing the
kubectl.kubernetes.io/last-applied-configuration annotation which is required by
oc apply. oc apply should only be used on resources created declaratively by
either oc create --save-config or oc apply. The missing annotation will be patched
automatically.

catalogsource.operators.coreos.com/ibm-operator-catalog configured


Creating project ibm-db2...

namespace/ibm-db2 created

Now using project "ibm-db2" on server "https://api.ocp.example.com:6443".


Creating secret ibm-registry. For this, your Entitlement Registry key is needed.


You can get the Entitlement Registry key from here:
https://myibm.ibm.com/products-services/containerlibrary


Enter your Entitlement Registry key: <paste your Entitlement Registry key here
ONCE, it will not be shown, then hit the Enter key>

Verifying the Entitlement Registry key...

Login Succeeded!

Entitlement Registry key is valid.

secret/ibm-registry created


Modifying the OpenShift Global Pull Secret (you need jq tool for that):

secret/pull-secret data updated


Creating Operator Group object for DB2 Operator

operatorgroup.operators.coreos.com/ibm-db2-group created


Creating Subscription object for DB2 Operator
```

```
subscription.operators.coreos.com/db2u-operator created


Waiting up to 5 minutes for DB2 Operator install plan to be generated.
<your current date and time>


Approving DB2 Operator install plan.
installplan.operators.coreos.com/install-pbn59 patched


Waiting up to 15 minutes for DB2 Operator to install.
<your current date and time>


Deploying the Db2u cluster.
db2ucluster.db2u.databases.ibm.com/db2ucluster created


Waiting up to 15 minutes for c-db2ucluster-db2u statefulset to be created.
<your current date and time>


Patching c-db2ucluster-db2u statefulset.
statefulset.apps/c-db2ucluster-db2u patched (no change)


Waiting up to 20 minutes for c-db2ucluster-restore-morph job to complete
successfully.
<your current date and time>



Updating number of databases allowed by DB2 installation from 8 to 30.
configmap/c-db2ucluster-db2dbmconfig replaced


Updating database manager running configuration.
DB20000I  The UPDATE DATABASE MANAGER CONFIGURATION command completed
successfully.


Restarting DB2 instance.
Wolverine HA management state was disabled successfully.
<your current date and time>     0   0   SQL1064N  DB2STOP processing was
successful.
SQL1064N  DB2STOP processing was successful.
<your current date and time>     0   0   SQL1063N  DB2START processing was
successful.
SQL1063N  DB2START processing was successful.
Wolverine HA management state was enabled successfully.


********************************************************************************
```

```
********* Installation and configuration of DB2 completed successfully! *********
*****************************************************************************


Removing BLUDB from system.
DB20000I  The FORCE APPLICATION command completed successfully.
DB21024I  This command is asynchronous and may not be effective immediately.


DB20000I  The DEACTIVATE DATABASE command completed successfully.
DB20000I  The DROP DATABASE command completed successfully.


Existing databases are:

Use this hostname/IP to access the databases e.g. with IBM Data Studio.
  Hostname: router-default.apps.ocp.example.com
  Other possible addresses(If hostname not available above): 10.0.0.10,master-0


Use one of these NodePorts to access the databases e.g. with IBM Data Studio
(usually the first one is for legacy-server (Db2 port 50000), the second for ssl-
server (Db2 port 50001)).
                "nodePort": <your legacy-server node port>,
                "nodePort": <your ssl-server node port>,


Use "db2inst1" and password "passw0rd" to access the databases e.g. with IBM Data
Studio.


Db2u installation complete! Congratulations. Exiting...
```

6. **Review** the output of your script and make sure that there are no errors and that it completed as shown in the sample script output. Especially, make sure you see the banner

```
*****************************************************************************
********* Installation and configuration of DB2 completed successfully! *********
*****************************************************************************
```

If you don't see this banner, use the Troubleshooting section to learn how to troubleshoot DB2, delete the DB2 cluster, operator and the created namespace. Then, run script `02-createDb2OnOCP.sh` again.

After this banner is displayed it is possible for the script to fail on some post deployment cleanup work. Failing the cleanup work does not impact the successful deployment and configuration of DB2.

7. Look at the output of your script and **verify** that the deletion of **BLUDB** was successful. If this is not the case, use the Troubleshooting section to manually delete it.

Note: This database is created out of the box by DB2, but is not needed for the CP4BA deployment.

8. Run this command and verify that you see two completed pods and four pods Running and Ready (1/1).

```
oc get pods
```

```
[cp4badmin@client deployment-db2-cp4ba]$ oc get pods
NAME                                      READY   STATUS       RESTARTS   AGE
c-db2ucluster-db2u-0                      1/1     Running      0          30m
c-db2ucluster-etcd-0                      1/1     Running      0          31m
c-db2ucluster-instdb-nb7tf                0/1     Completed    0          31m
c-db2ucluster-ldap-58846f7c46-c544s       1/1     Running      0          31m
c-db2ucluster-restore-morph-4nf69         0/1     Completed    0          28m
db2u-operator-manager-5768b9bf79-97s5x    1/1     Running      0          33m
[cp4badmin@client deployment-db2-cp4ba]$
```

9. **Run** script `03-createCp4baDBs4Db2OnOCP.sh` to create the databases needed for the CP4BA template that you selected.

```
./03-createCp4baDBs4Db2OnOCP.sh
```

Note: You can ignore the following errors / warnings:

```
DB21034E The command was processed as an SQL statement because it was not
a valid Command Line Processor command.  During SQL processing it
returned: SQL0554N  An authorization ID cannot grant a privilege or
authority to itself. SQLSTATE=42502
```

and

```
SQL1363W  One or more of the parameters submitted for immediate
modification were not changed dynamically. For these configuration
parameters, the database must be shutdown and reactivated before the
configuration parameter changes become effective.
```

10. **Review** the output of the script and make sure there were no errors and that the following **three databases were created**:

```
Existing databases are:
  Database name                          = CLOS
  Database name                          = GCDDB
  Database name                          = ICNDB
```

Note: In case you got errors creating DBs, please use script `99-dropCp4baDBs4Db2OnOCP.sh` to drop all DBs – then re-run script `03-createCp4baDBs4Db2OnOCP.sh` .

Note: When selecting a different CP4BA template for deployment, a different number of databases will get created.

11. **Verify** that the script also has **activated** these databases:

```
Activating databases...

ICNDB...
DB20000I   The ACTIVATE DATABASE command completed successfully.

CLOS...
DB20000I   The ACTIVATE DATABASE command completed successfully.

GCDDB...
DB20000I   The ACTIVATE DATABASE command completed successfully.

Done. Exiting...
```

Note: In case you got errors **activating DBs**, please use script `04-activateDBs.sh` to try to activate them again – if your DB2 pod got enough memory assigned according to the template used, activation of all DBs must be successful.

With that you have successfully deployed the DB2 operator and a DB2 cluster on your OpenShift cluster. In addition, you have generated three databases needed for the CP4BA deployment. You are now ready to start deploying CP4BA.

This is also a possible point where you optionally could take a break working on this lab, means perform a clean shut-down of the environment as instructed earlier and continue the lab at a later point in time.

If you shut down your environment at this point, please make sure to perform the following steps after you re-started the environment again:

- Log in to VM 0 with **cp4badmin / passw0rd** (where the third-last character is a zero)

- Verify that VM 0 is properly connected to the network

- Change the screen resolution and keyboard layout as needed

- Verify that your connection to VM 0 is green

- On VM 0, start a Terminal and then start SDS using command

  ```
  sudo /root/bin/startSds.sh
  ```

- Access your OpenShift cluster through Browser, log in with **admin / passw0rd** (where the third-last character is a zero)

- Verify that under **Home** – **Overview** the status of **Cluster** and **Control Plane** is green. If not, wait until it becomes green, that should happen automatically. For **Operators** there now will be at least one upgrade available, this is for the previously installed IBM DB2 operator. This is as expected, no action need. If other Operators are upgrading automatically, wait until this activity has finished. This can happen as you have resumed updates before and Red Hat publishes new fixes for the used OCP version.

- Verify that the **Cluster inventory** shows no errors
- In the Terminal, log in to your OpenShift cluster through **command line** by **copying the entire oc login command** from the OCP Web Console
- In the Terminal, switch to the directory with the rapid deployment scripts

```
cd /cp4ba/cp4ba-rapid-deployment/cp4ba-21-0-3/mycluster/deployment-db2-cp4ba
```

- In the Terminal, run script `04-activateDBs.sh` to make sure the previously created DBs are all activated again

```
./04-activateDBs.sh
```

- Verify that activation of all three databases was successful

4 Exercise: Install IBM Cloud Pak for Business Automation Operator & deploy IBM Cloud Pak for Business Automation (Production)

4.1 Introduction

Now that **all prerequisites** for the CP4BA deployment **are in place** – SDS is up and running on VM 0, DB2 is deployed on the OpenShift cluster and the needed databases are created, the **deployment of CP4BA can be started**.

First step here is to get access to the product scripts needed for the deployment of the CP4BA operator and **deploy the operator**.

Second step is to provide some more CP4BA specific configuration parameters and then kick-off the CP4BA deployment.

Finally, there are some **post-deployment steps** to be completed to finish the deployment, and some **verification steps** to verify the deployment is healthy, all deployed components are accessible.

4.2 Exercise Instructions

1. On your bastion host, **download the Case package** by opening the following URL in the Browser:

   https://github.com/IBM/cloud-pak/raw/master/repo/case/ibm-cp-automation-3.2.5.tgz

   The archive is automatically saved into directory `/home/cp4badmin/Downloads` .

2. Using the **Terminal**, extract `cert-kubernetes` into directory /cp4ba .

   ```
   cd /home/cp4badmin/Downloads

   tar -xvzf ibm-cp-automation-3.2.5.tgz

   cd ibm-cp-automation/inventory/cp4aOperatorSdk/files/deploy/crs

   tar -xvzf cert-k8s-21.0.3.tar -C /cp4ba
   ```

3. To install the **CP4BA operator and other needed operators**, from `cert-kubernetes`, execute script `cp4a-clusteradmin-setup.sh`

   ```
   cd /cp4ba/cert-kubernetes/scripts/


   ./cp4a-clusteradmin-setup.sh
   ```

   Notes:

- This script will prompt you for various input. Please have a look at the sample script output provided below to see the answers you must provide, they are marked in red.[3]
- You need your Entitlement Registry key handy, see also **https://myibm.ibm.com/products-services/containerlibrary**
- When you paste your Entitlement Registry key it will not be shown, therefore paste it just once and hit the Enter key.

Sample script output

```
./cp4a-clusteradmin-setup.sh
creating temp folder


Select the cloud platform to deploy:
1) RedHat OpenShift Kubernetes Service (ROKS) - Public Cloud
2) Openshift Container Platform (OCP) - Private Cloud
3) Other ( Certified Kubernetes Cloud Platform / CNCF)
Enter a valid option [1 to 3]: 2


This script prepares the OLM for the deployment of some Cloud Pak for Business
Automation capabilities


What type of deployment is being performed?
1) Starter
2) Production
Enter a valid option [1 to 2]: 2


Do you want CP4BA Operator support 'All Namespaces'? (Yes/No, default: No) No


Where do you want to deploy Cloud Pak for Business Automation?
Enter the name for a new project or an existing project (namespace): ibm-cp4ba


The Cloud Pak for Business Automation Operator (Pod, CSV, Subscription) not found
in cluster
Continue....


Using project ibm-cp4ba...


Here are the existing users on this cluster:
```

---

[3] Even though the script is interactive in nature, it can be fully automated by setting environment variables, if needed. This is not done here, though.

1) admin

Enter an existing username in your cluster, valid option [1 to 1], non-admin is suggested: **1**

Follow the instructions on how to get your Entitlement Key:

https://www.ibm.com/support/knowledgecenter/en/SSYHZ8_21.0.x/com.ibm.dba.install/op_topics/tsk_images_enterp_entitled.html

Do you have a Cloud Pak for Business Automation Entitlement Registry key (Yes/No, default: No): **Yes**

Enter your Entitlement Registry key: <paste your Entitlement Registry key here ONCE, it will not be shown, then hit the Enter key>

Verifying the Entitlement Registry key...

Login Succeeded!

Entitlement Registry key is valid.

The existing storage classes in the cluster:
```
NAME             PROVISIONER                              RECLAIMPOLICY
VOLUMEBINDINGMODE    ALLOWVOLUMEEXPANSION    AGE

cp4ba-storage    k8s-sigs.io/nfs-subdir-external-provisioner    Delete
Immediate            false                   16d
```

To provision the persistent volumes and volume claims please enter the dynamic storage classname: **cp4ba-storage**

Creating docker-registry secret for Entitlement Registry key in project ibm-cp4ba...

secret/admin.registrykey created

Done

Creating ibm-entitlement-key secret for IAF in project ibm-cp4ba...

secret/ibm-entitlement-key created

Done

Applying the persistent volumes for the Cloud Pak operator by using the storage classname: cp4ba-storage...

persistentvolumeclaim/operator-shared-pvc created

persistentvolumeclaim/cp4a-shared-log-pvc created

Done

Waiting for the persistent volumes to be ready...

Done

ibm-operator-catalog   IBM Operator Catalog   grpc   IBM   450d

Found existing ibm operator catalog source, updating it

```
catalogsource.operators.coreos.com/ibm-operator-catalog configured

IBM Operator Catalog source updated!

Waiting for CP4A Operator Catalog pod initialization

Waiting for CP4A Operator Catalog pod initialization

CP4BA Operator Catalog is running ibm-operator-catalog-l9hkw
1/1    Running    0    30s

operatorgroup.operators.coreos.com/ibm-cp4a-operator-catalog-group created

CP4BA Operator Group Created!

subscription.operators.coreos.com/ibm-cp4a-operator-catalog-subscription created

CP4BA Operator Subscription Created!

Waiting for CP4BA operator pod initialization

No resources found in ibm-cp4ba namespace.

Waiting for CP4BA operator pod initialization

No resources found in ibm-cp4ba namespace.

Waiting for CP4BA operator pod initialization

No resources found in ibm-cp4ba namespace.

Waiting for CP4BA operator pod initialization

Waiting for CP4BA operator pod initialization

Waiting for CP4BA operator pod initialization

Waiting for CP4BA operator pod initialization

Waiting for CP4BA operator pod initialization

CP4A operator is running ibm-cp4a-operator-7995f9b66-bjbjl
1/1    Running    0    2m


Adding the user admin to the ibm-cp4a-operator role...Done!


Label the default namespace to allow network policies to open traffic to the
ingress controller using a namespaceSelector...namespace/default labeled

Done


Storage classes are needed to run the deployment script. For the Starter
deployment scenario, you may use one (1) storage class.  For an Production
deployment, the deployment script will ask for three (3) storage classes to meet
the slow, medium, and fast storage for the configuration of CP4A components.  If
you don't have three (3) storage classes, you can use the same one for slow,
medium, or fast.  Note that you can get the existing storage class(es) in the
environment by running the following command: oc get storageclass. Take note of
the storage classes that you want to use for deployment.
NAME            PROVISIONER                              RECLAIMPOLICY
VOLUMEBINDINGMODE    ALLOWVOLUMEEXPANSION    AGE

cp4ba-storage   k8s-sigs.io/nfs-subdir-external-provisioner   Delete
Immediate        false                   16d
```
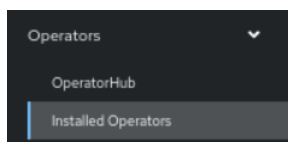
Note: Currently the rapid deployment scripts only support the deployment of CP4BA clusters in Production mode using images pulled from the IBM public container library. The Starter deployment mode, or Air-Gapped environments are not (yet) supported.

4. **Review** your script output and make sure there were no errors, it completed as shown in the sample script output.

5. **Wait** until all Operators are installed, this might take a while.

6. To **verify** the deployment of the operators, switch to your Browser tab with the OCP Web Console and select on the left-hand side within **Operators** the menu entry **Installed Operators**.

7. At the top left side of the main pain switch to project **ibm-common-services**. Verify that the following **five operators** show status **Succeeded**.

   Note: It might happen that going forward the number of operators does change. Important point here is that all show **Succeeded**.

8. Switch to project **ibm-cp4ba**. Verify that the following **six operators** show status **Succeeded**.

   Note: It might happen that going forward the number of operators does change. Important point here is that all show **Succeeded**.
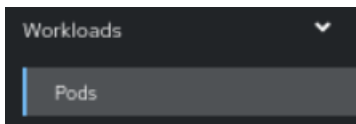
9. Select on the left-hand side within **Workloads** the menu entry **Pods**.

10. Verify that the following **9 pods** exist, all **Running** and **Ready 1/1**).

    Note: It might happen that going forward the number of pods does change. Important point here is that all show **Running** and **Ready 1/1**.

11. Switch to project **ibm-common-services**. Verify that the following **12 pods** exist, all **Running** and **Ready 1/1**).

    Note: It might happen that going forward the number of pods does change. Important point here is that all show **Running** and **Ready 1/1**.

Name ↑	Status	Ready	Restarts	Owner	Memory	CPU	Created
cert-manager-cainjector-75ccfc98c4-5xp8j	Running	1/1	0	RS cert-manager-cainjector-75ccfc98c4	141.5 MiB	0.002 cores	Jun 16, 2022, 1:51 AM
cert-manager-controller-7684dd8d97-fpp4v	Running	1/1	0	RS cert-manager-controller-7684dd8d97	115.1 MiB	0.001 cores	Jun 16, 2022, 1:51 AM
cert-manager-webhook-749998d7f8-rcs4w	Running	1/1	0	RS cert-manager-webhook-749998d7f8	18.9 MiB	0.001 cores	Jun 16, 2022, 1:51 AM
configmap-watcher-74b5b6b848-j4pg7	Running	1/1	0	RS configmap-watcher-74b5b6b848	9.7 MiB	0.001 cores	Jun 16, 2022, 1:51 AM
ibm-cert-manager-operator-6f8d8b66dd-dzq75	Running	1/1	0	RS ibm-cert-manager-operator-6f8d8b66dd	236.6 MiB	0.007 cores	Jun 16, 2022, 1:51 AM
ibm-common-service-operator-6f64486c9d-bgmqb	Running	1/1	0	RS ibm-common-service-operator-6f64486c9d	47.6 MiB	0.006 cores	Jun 16, 2022, 1:46 AM
ibm-common-service-webhook-7f56b4658b-2zjmd	Running	1/1	0	RS ibm-common-service-webhook-7f56b4658b	19.8 MiB	0.003 cores	Jun 16, 2022, 1:48 AM
ibm-licensing-operator-5cdb8cdf89-6npd7	Running	1/1	0	RS ibm-licensing-operator-5cdb8cdf89	34.8 MiB	0.004 cores	Jun 16, 2022, 1:51 AM
ibm-licensing-service-instance-865776b444-57csl	Running	1/1	0	RS ibm-licensing-service-instance-865776b444	232.5 MiB	0.003 cores	Jun 16, 2022, 1:52 AM
ibm-namespace-scope-operator-58b6c5c648-qbjqd	Running	1/1	0	RS ibm-namespace-scope-operator-58b6c5c648	24.4 MiB	0.001 cores	Jun 16, 2022, 1:48 AM
operand-deployment-lifecycle-manager-6b6fb98857-mvdfq	Running	1/1	0	RS operand-deployment-lifecycle-manager-6b6fb98857	29.4 MiB	0.001 cores	Jun 16, 2022, 1:50 AM
secretshare-6b9f4fd485-5rbkq	Running	1/1	0	RS secretshare-6b9f4fd485	22.3 MiB	0.028 cores	Jun 16, 2022, 1:48 AM

12. Now that the Operators are deployed, you can proceed installing CP4BA.

   In the **Terminal** switch to the rapid deployment scripts and **open** the parameters file for CP4BA, `05-parametersForCp4ba.sh`, with `gedit`.

   `cd /cp4ba/cp4ba-rapid-deployment/cp4ba-21-0-3/mycluster/deployment-db2-cp4ba`

   `gedit 05-parametersForCp4ba.sh`

13. **Apply** the following changes:

   - `cp4baProjectName` – Name of the project/namespace to use to deploy CP4BA, for example `ibm-cp4ba` – make sure to use the same value as used before when running script `cp4a-clusteradmin-setup.sh`[4]

     set the value to `ibm-cp4ba`

   - `cp4baTlsSecretName` – Parameter for ROKS deployments on IBM Cloud only – as you are not deploying on ROKS, leave empty

     remove the default value `REQUIRED`, leave this parameter empty

   - `cp4baAdminPassword` – Use the password for user `cp4badmin` – this user got created in the LDAP running on VM 0 that was already prepared for you.

---

[4] Please note that Kubernetes limits the name. So the name can only consist of lowercase characters, numbers and -, and it must not start or end with a -, so that it can also be used in URLs and hostnames.

set the value to `passw0rd`

- `ldapAdminPassword` – Use the password that was specified for `cn=root` when setting up LDAP

  set the value to `passw0rd`

- `ldapServer` – The hostname or IP of the used LDAP server

  set the value to `10.0.0.5`

```
# OCP project name for CP4BA, for example ibm-cp4ba - make sure to use the same value as used before when running script cp4a-clusteradmin-setup.sh
cp4baProjectName=ibm-cp4ba

# TLS secret name - see also secret name in project ibm-cert-store
#   If this secret is not available, leave empty (but remove the value 'REQUIRED') - then self-signed certificates will be used at the routes
cp4baTlsSecretName=

# Password for CP4BA Admin User (cp4baAdminName name see below), for example passw0rd - see ldif file you applied to LDAP
cp4baAdminPassword=passw0rd

# Password for LDAP Admin User (ldapAdminName name see below), for example passw0rd - use the password that you specified when setting up LDAP
ldapAdminPassword=passw0rd

# LDAP instance access information - hostname or IP
ldapServer="10.0.0.5"
```

- Scroll further down to the Storage Class Settings section and modify the storage classes to be used

  change all four values to `cp4ba-storage`

```
# --- Storage Class Settings ---
cp4baScSlow=cp4ba-storage
cp4baScMedium=cp4ba-storage
cp4baScFast=cp4ba-storage
cp4baBlockScFast=cp4ba-storage
```

- Leave all other properties at their default values

- Finally save you changes and close the editor

14. At this point, before starting the deployment, we recommend checking whether the database and the LDAP Server are reachable. This can avoid difficult to analyze problems while the deployment is running. To do that, please refer to section Troubleshooting CP4BA.

15. **Run** script `07-createCp4baDeployment.sh` . This script will now deploy CP4BA using the parameters you previously provided in `01-parametersForDb2OnOCP.sh` and `05-parametersForCp4ba.sh` .

```
./07-createCp4baDeployment.sh
```

Sample script output

```
./07-createCp4baDeployment.sh
```

```
Found 01-parametersForDb2OnOCP.sh.  Reading in variables from that script.
  Reading 01-parametersForDb2OnOCP.sh ...
Done!


Found 05-parametersForCp4ba.sh.  Reading in variables from that script.
  Reading 05-parametersForCp4ba.sh ...
Extracting OCP Hostname
OCPHostname set to apps.ocp.example.com
Done!


This script PREPARES and optionally CREATES the CP4BA deployment using template
ibm_cp4a_cr_template.002.ent.FoundationContent.yaml in project ibm-cp4ba.


Are 01-parametersForDb2OnOCP.sh and 05-parametersForCp4ba.sh up to date, and do
you want to continue? (Yes/No, default: No): Yes


Preparing the CP4BA deployment...


Switching to project ibm-cp4ba...
Already on project "ibm-cp4ba" on server "https://api.ocp.example.com:6443".


Collecting information for secret ibm-entitlement-key. For this, your Entitlement
Registry key is needed.


You can get the Entitlement Registry key from here:
https://myibm.ibm.com/products-services/containerlibrary


Enter your Entitlement Registry key: <paste your Entitlement Registry key here
ONCE, it will not be shown, then hit the Enter key>
Verifying the Entitlement Registry key...
Login Succeeded!
Entitlement Registry key is valid.


Copying jdbc for Db2 from Db2 container to local disk...
Now using project "ibm-db2" on server "https://api.ocp.example.com:6443".
tar: Removing leading `/' from member names
tar: Removing leading `/' from member names
Now using project "ibm-cp4ba" on server "https://api.ocp.example.com:6443".


Preparing the CP4BA secrets...


Preparing the CR YAML for deployment...
```

```
All artefacts for deployment are prepared.

Do you want to CREATE the CP4BA deployment in project ibm-cp4ba now? (Yes/No,
default: No): Yes

Creating the CP4BA deployment...

Creating secret ibm-entitlement-key in project ibm-common-services...
Now using project "ibm-common-services" on server
"https://api.ocp.example.com:6443".
secret/ibm-entitlement-key created
Now using project "ibm-cp4ba" on server "https://api.ocp.example.com:6443".
Done.

Copying the jdbc driver to ibm-cp4a-operator...
Done.

Creating CP4BA secrets...
secret/ldap-bind-secret created
secret/icp4a-shared-encryption-key created
secret/resource-registry-admin-secret created
secret/ibm-ban-secret created
secret/ibm-fncm-secret created
secret/icp4adeploy-bas-admin-secret created
secret/playback-server-admin-secret created
secret/icp4adeploy-workspace-aae-app-engine-admin-secret created
secret/ibm-adp-secret created
secret/ibm-bawaut-server-db-secret created
secret/ibm-pfs-admin-secret created
secret/ibm-bawaut-admin-secret created
Done.

Creating the CP4BA deployment...
icp4acluster.icp4a.ibm.com/icp4adeploy created
Done.

All changes got applied. Exiting...
```

16. The deployment of CP4BA might now take some time, **usually about one hour**. Monitor the progress of the deployment either through the OpenShift Web Console in the Browser, or by command line using the `oc get pods` command or monitor the logs of the Operator to spot any

potential issues, see the Troubleshooting section for more details about that.

Note: As part of the CP4BA deployment, additional operators are added to project `ibm-common-services`. Also, more pods are created there. It might be that you see these new pods in various states indicating issues, but normally these are resolved by the operators automatically.

17. The CP4BA deployment is completed when you see in your CP4BA project `ibm-cp4ba` about **31 Running and Ready pods**, and about **10 Completed pods**, but no Pending / CrashLoopBackOff pods, plus in project `ibm-common-services` about **32 Running and Ready pods**, and about **5 Completed pods**, but no Pending / CrashLoopBackOff pods.

Note: It might be that some pods are in Failed or Error state, for those make sure there is another instance of that pod in Completed state. If this is the case, you can delete the Failed or Error pods. If there are pods in Failed or Error state where there is no other instance of that pod in Completed state, the deployment is not healthy. Check the Troubleshooting chapter for help on resolving these issues.

Note: It might be that going forward the number of pods mentioned here does change, as with every new installation latest version of `ibm-common-services` and IBM Automation Foundation are installed and those latest versions might come with a different number of Running and / or Completed pods. The most important point here is that you don't see pods in any other state (Pending / CrashLoopBackOff / Failed / Error / ...).

18. Now that the CP4BA deployment is complete, you need to apply some **post-deployment steps**.

    First post-deployment step is to enable you to log in with the users from LDAP. For this, first get the user ID and password of the **zen admin user** by running those two commands:

    ```
    oc get secret ibm-iam-bindinfo-platform-auth-idp-credentials -o
    jsonpath='{.data.admin_username}' | base64 -d && echo
    ```

    ```
    oc get secret ibm-iam-bindinfo-platform-auth-idp-credentials -o
    jsonpath='{.data.admin_password}' | base64 -d && echo
    ```

    Note: By default, the user id is **admin**.

19. Open the `cpd` route. Switch to your Browser tab with the OCP Web Console and select on the left-hand side within **Networking** the menu entry **Routes**. If not on project `ibm-cp4ba`, switch to it.

Project: ibm-cp4ba ▾

Routes

Name ↑	Status	Location
RT cpd	✔ Accepted	https://cpd-ibm-cp4ba.apps.ocp.example.com ⧉

20. In column **Location** click on the URL to open the `cpd` route in a new tab.

21. Accept the self-signed certificates two times.

    Note: For the deployment of the CP4BA Cluster, also properly signed certificates can be created, which don't raise security exceptions in the browser. This requires to have SSL Certificates from a Certificate Authority, which would be available for example in an OCP environment on IBM Cloud.

22. On the Log in to IBM Cloud Pak screen, select IBM provided credentials (admin only).

23. Log in using the **zen admin user credentials** you gathered before.

24. Click Manage users.
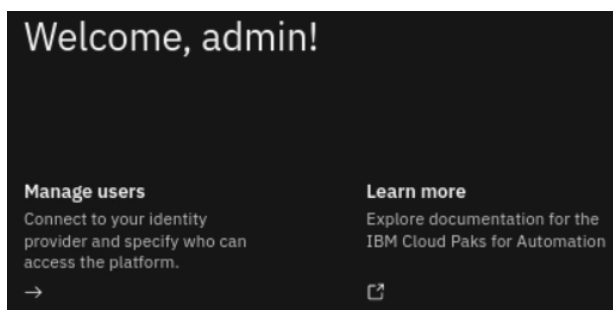
25. Select the **User groups** tab and click **New user group**.
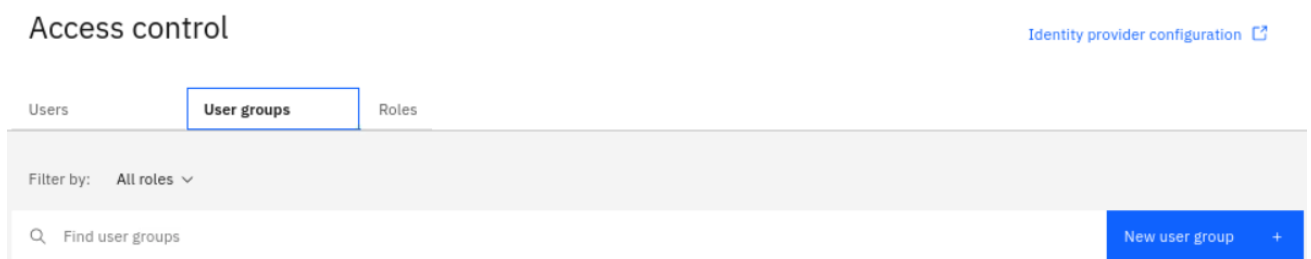
26. Enter name **cp4bausers** and click **Next**.

27. On the Users page select Identity provider groups, search for cp4bausers, select it and click Next.

28. On the **Roles** page select roles **Automation Developer** (needed for CP4BA, for example to access BAStudio) and **User**, then click **Next**.

29. On the **Summary** page review the selections and click **Create**.

30. Select the **Users** tab and click **cp4badmin**.

31. Click on **Assign roles**, select all roles and click **Assign**.

32. Log out with the zen admin user.
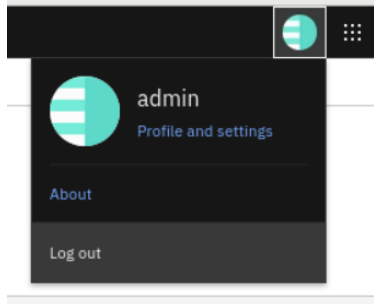
33. Second post-deployment step is to verify that users from LDAP can log-in.

    For this, back on the **Log in to IBM Cloud Pak** page, first select **Change your authentication method** and then **Enterprise LDAP**.

34. Log in with **cp4badmin** which is a user from LDAP, password is **passw0rd**.

35. Verify that cp4admin now has full administrative access to zen: **cp4badmin** should also see the **Manage users** option, and in the hamburger menu the entry **Administration**.
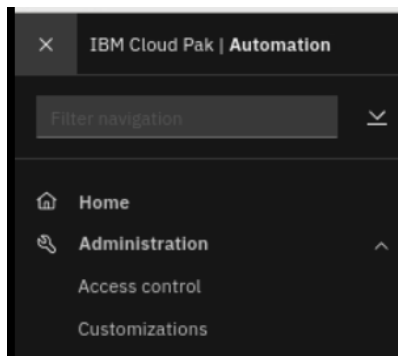
36. Last post-deployment step is to verify that the deployed CP4BA components do work fine.

    For this, gather the cluster's URLs from config map **icp4adeploy-cp4ba-access-info** and test that all URLs work. In the OpenShift Web Console under **Workloads** open **ConfigMaps**. Search for **icp4adeploy-cp4ba-access-info**.
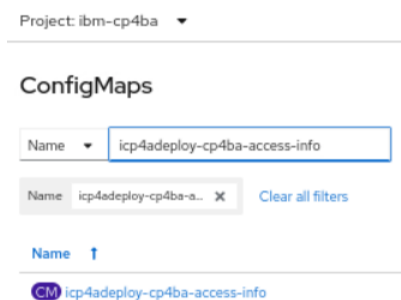
37. Open this config map. Scroll down to the **Data** section where the URLs can be found.

38. Under cpe-access-info open the URL for Content Platform Engine administration in a new Browser tab.

39. As you are already logged in as **cp4badmin**, the **ACCE** will show up. Expand **Object Stores** and verify **CLOS** is available.
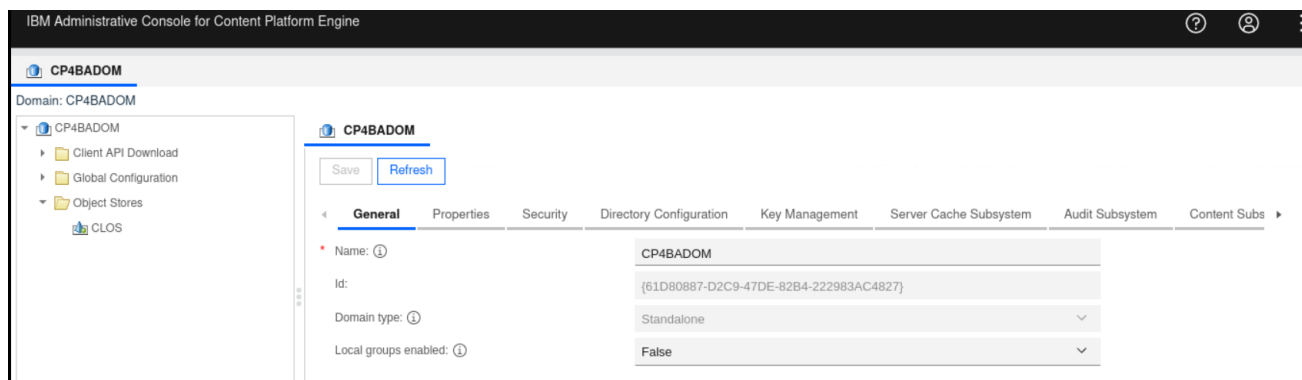
40. Similarly, open the following URLs from the config map to verify all deployed components are available:

   - Business Automation Navigator for CP4BA
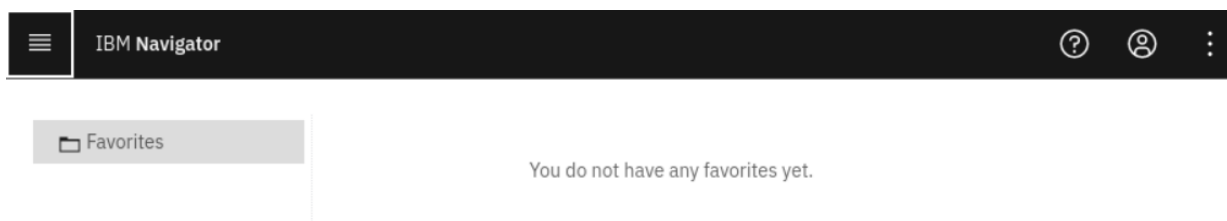
   Note: The hamburger menu provides you access to some other desktops.

   - Content Management Interoperability Services for CP4BA (log in with **cp4badmin** / **passw0rd**)
   - Content Services GraphQL

With that, you have successfully completed this exercise and learned how to leverage the CP4BA rapid deployment scripts to install CP4BA on an OCP cluster.

The CP4BA rapid deployment scripts (**https://github.com/IBM/cp4ba-rapid-deployment**) are accessible by customers, business partners and IBMers. They are continuously maintained by the IBM Automation SWAT team. Feel free to use them.

Going forward, if you like to deploy one of the larger templates on your own cluster, the differences to what you have just done are:

- Planning: Check out Step 0 (**https://github.com/IBM/cp4ba-rapid-deployment/blob/main/cp4ba-21-0-3/00selectTemplate.md**) to understand which templates are available and what their requirements in terms of number of workers and resources are. Larger templates will require a larger OpenShift cluster with more worker nodes, CPU and memory. Creating your own template is also possible.
- If you want to create a cluster on IBM Cloud, make sure you have or create an account there (**https://github.com/IBM/cp4ba-rapid-deployment/blob/main/cp4ba-21-0-3/01createIBMCloudAccount.md**).

- You need your own Red Hat OpenShift cluster, for example on IBM Cloud with funding for hosting the infrastructure. Check out how easy it is to create a new ROKS cluster on IBM Cloud here: **https://github.com/IBM/cp4ba-rapid-deployment/blob/main/cp4ba-21-0-3/02createRedHatOpenShiftCluster.md**.
- You need your own LDAP. If there is no existing, to set-up your own LDAP on IBM Cloud is documented here: **https://github.com/IBM/cp4ba-rapid-deployment/blob/main/cp4ba-21-0-3/03createVMForLDAP.md**.
- You optionally need to get for DB2 the DB2 Standard Edition license key (PPA or XL SW, search for part number CC36WML), and you need to assign more resources to your DB2 deployment (see additional parameters in **01-parametersForDb2OnOCP.sh**).
- After the CP4BA deployment there are more post-deployment steps to be applied. Also, you might want to install other components like Process Mining, the ADS ML Service, ... and scale up your deployment for High Availability. All this is documented in the rapid deployment github.
- Once the installation is complete and you selected the ClientOnboarding template, you might want to import the Client Onboarding Solution, see also **https://github.com/IBM/cp4ba-client-onboarding-scenario**

Congratulations on completing this lab!

Optionally, you can also complete the Troubleshooting CP4BA exercise.

This is also a possible point where you optionally could take a break working on this lab, means perform a clean shut-down of the environment as instructed earlier and continue the lab at a later point in time.

If you shut down your environment at this point, please make sure to perform the following steps after you re-started the environment again:

- Log in to VM 0 with **cp4badmin / passw0rd** (where the third-last character is a zero)
- Verify that VM 0 is properly connected to the network
- Change the screen resolution and keyboard layout as needed
- Verify that your connection to VM 0 is green

- On VM 0, start a Terminal and then start SDS using command

  ```
  sudo /root/bin/startSds.sh
  ```

- Access your OpenShift cluster through Browser, log in with **admin / passw0rd** (where the third-last character is a zero)

- Verify that under **Home** – **Overview** the status of **Cluster** and **Control Plane** is green. If not, wait until it becomes green, that should happen automatically. For **Operators** there now will be at least one upgrade available, this is for the previously installed IBM DB2 operator. This is as expected, no action need. If other Operators are upgrading automatically, wait until this activity has finished. This can happen as you have resumed updates before and Red Hat publishes new fixes for the used OCP version.

- Verify that the **Cluster inventory** shows no errors

With that, you are after a clean shut down and startup of your environment again ready to complete the Troubleshooting CP4BA exercise.

5 Troubleshooting CP4BA

5.1 Introduction

When you finished the previous exercise, you are ready with the deployment. However, you might have run into troubles, or might be interested how further troubleshooting can be made for the deployment of a Cloud Pak 4 Business Automation deployment. Also, at some points during the exercises above you might have been referred to one or more of the sections in this chapter, for troubleshooting problems, shutting down the environment properly, or checking availability of components.

5.2 Determining Deployment Status of the CP4BA Cluster

This section show how you can check the correct installation of the various parts of the CP4A deployment steps. Use it as a guide to check where you are with the deployment, and weather anything failed to install.

First, it should be avoided to perform any action while the Openshift cluster is being upgraded. Therefore, before running any other task, determine if the Openshift cluster is currently being upgraded, perform the following steps:

1. Open the browser and use the bookmark to login on the Openshift Web Console. Login using admin / passw0rd.

2. On the left navigation bar open the section "Compute" and select "Machine Config Pools".

3. Verify on the right side if you see "true" on the column "Updating".

To determine if the DB2 Operator is currently being installed, or the DB2 cluster is being configured to the following:

1. Open the browser and use the bookmark to login on the Openshift Web Console. Login using admin / passw0rd.

2. On the left navigation bar, open the section "Operators" and select "Installed Operators". Then select the project ibm-db2. The IBM DB2 Operator should be there and should show "Status

Succeeded". If that's not the case, it might still be installing.

Name ↑	Managed Namespaces	Status	Last updated	Provided APIs	
IBM Db2 1.1.9 provided by IBM	NS ibm-db2	✓ Succeeded ⬆ Upgrade available	🌐 3 minutes ago	Db2u Cluster Db2u Hadr Db2u Helm Migration	⋮

3. Then open the DB2 operator, and scroll to the very bottom of the "Details" tab. At the bottom of it you should see the following, if the operator is already fully installed:

Installing	True	🌐 6 minutes ago	InstallSucceeded	waiting for install components to report healthy
Installing	True	🌐 6 minutes ago	InstallWaiting	installing: waiting for deployment db2u-operator-manager to become ready: deployment "db2u-operator-manager" waiting for 1 outdated replica(s) to be terminated
Succeeded	True	🌐 4 minutes ago	InstallSucceeded	install strategy completed with no errors

4. When the DB2 Operator is completely installed, the next step would be to install a DB2 database server. To check the status of this, with the open DB2 Operator, go to the "All instances" tab and verify the status of the db2 cluster. When the DB2cluster is ready deployed the state should show as Ready, as here:

Project: ibm-db2 ▼

Installed Operators > Operator details

IBM Db2
1.1.9 provided by IBM

Actions ▼

Details   YAML   Subscription   Events   All instances   Db2u Cluster   Db2u Hadr   Db2u Helm Migration

Create new ▼

▼ Filter ▼   Name ▼   Search by name...   🔲

Name ↑	Kind	Status	Labels	Last updated	
DC db2ucluster	Db2uCluster	State: ✓ Ready	No labels	🌐 Jun 23, 2022, 9:25 AM	⋮

5. When the db2cluster is successfully deployed the DB2 installation script 02-createDb2OnOCP.sh would not terminate immediately, as its still configuring the database server, and removing the default BLUDB database. Before you suspend the cluster, wait for that to complete too.

6. The next step is the database creation script. That's not completed as long as the script is still running, though.

After setup of the DB2 Operator, the DB2 Cluster and the DB2 Databases comes the deployment of the CP4BA Operator using the clusteradmin-setup script. When the clusteradmin script terminates, the CP4BA Operators might not yet be fully installed. Please check:

1. Open the browser and use the bookmark to login on the Openshift Web Console. Login using admin / passw0rd

2. On the left navigation bar, open the section "Operators" and select "Installed Operators". Then select the project ibm-cp4ba.

3. Check that all of the operators show Status Succeeded as indicated below:

Project: ibm-cp4ba ▾

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	
IBM Automation Foundation Core 1.3.7 provided by IBM	NS ibm-cp4ba	✔ Succeeded Up to date	⊕ 8 minutes ago	Automation UIConfig Cartridge	⋮
IBM Automation Foundation Insights Engine 1.3.6 provided by IBM	NS ibm-cp4ba	✔ Succeeded Up to date	⊕ Jun 23, 2022, 10:08 AM	InsightsEngine	⋮
IBM Automation Foundation 1.3.7 provided by IBM	NS ibm-cp4ba	✔ Succeeded Up to date	⊕ 8 minutes ago	AutomationBase CartridgeRequirement	⋮
IBM Cloud Pak foundational services 3.19.0 provided by IBM	NS ibm-cp4ba	✔ Succeeded Up to date	⊕ Jun 23, 2022, 10:06 AM	CommonService	⋮
IBM Cloud Pak for Business Automation 21.3.9 provided by IBM	NS ibm-cp4ba	✔ Succeeded Up to date	⊕ Jun 23, 2022, 10:05 AM	CP4BA deployment	⋮

4. Then switch to the ibm-common-services project, and check there too, that all of the installed operators are completely deployed too.

After you started the deployment of the Openshift cluster by running the script 07-createCp4baDeployment.sh, the deployment of the openshift cluster will be performed by the Cloud Pak 4 Business Automation Operator. You can determine if the operator finished the deployment by checking as follows:

1. Open the browser and use the bookmark to login on the Openshift Web Console. Login using admin / passw0rd

2. On the left navigation bar, open the section "Home" and select "Search". Then select the project ibm-cp4ba. Click on the "Resources" to get a pull down menu for selection of the Kubernetes Resource to search for.

Project: ibm-cp4ba ▾

Search

Resources 0 ▾   ▼ Label ▾   app=frontend

Select Resource

☐ APIS APIServer
config.openshift.io/v1

☐ APIS APIServer
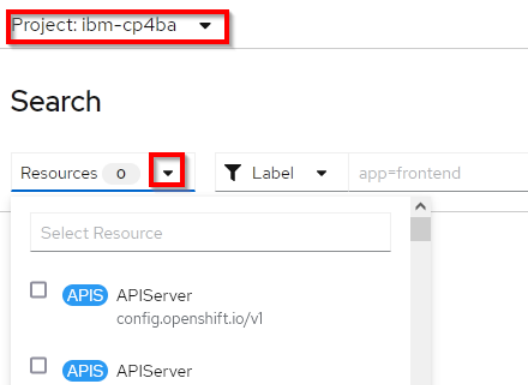
3. Type "ICP4ACluster" into the textbox, and click on the name once it appears. The search should show one result object named "icp4adeploy". This is the resource representing the installed CP4BA cluster. Click on the name, and scroll to the bottom of the "Details" tab. The CP4BA deployment is complete when it shows the following:

Type	Status	Updated	Reason	Message
Running	True	⊕ ▨▨▨ ▨▨▨▨	Running	Running reconciliation
ReconciliationSuccess	True	⊕ ▨▨▨ ▨▨▨ ▨▨	Successful	-

When you reached this state, you can continue with the post-deployment steps described starting step 18 in section 4.2, or shut down the cluster to do the post deployment later.

5.3 Check Health of CP4BA Installation

During the installation of CP4BA you can check the health of various resources created by the CP4BA operator to better understand the state of the installation and help you triage issues found along the way. These steps should be run from the bastion host.

1. Go to the OpenShift project where CP4BA is installed. This is the same project specified in the `cp4baProjectName` variable inside the `05-parametersForCp4ba.sh` script.

   ```
   # If you use a project name other than ibm-cp4ba you must use that instead
   of ibm-cp4ba
   oc project ibm-cp4ba
   ```

2. Check the status of the Automation Foundation Core services. These services must be deployed before any of the CP4BA components are deployed. If the deployment of these services fails, the overall CP4BA installation will fail. First, we can check the `AutomationUIConfig` resource. This resource can help you determine if Zen services, the services responsible to provide UI capabilities for CP4BA, have been deployed properly.

   ```
   # Check AutomationUIConfig status
   oc get AutomationUIConfig iaf-system
   ```

   Output:
   ```
   NAME          READY
   iaf-system    True
   ```

   If the output of the command shows `True` under the `READY` column, Zen services were properly deployed. If the output shows `False` under the `READY` column, either the services are still being deployed or there are issues with the deployment. You can gain additional insight on the deployment by getting the full set of details for the `AutomationUIConfig` resource.

   ```
   # Get details of AutomationUIConfig
   oc describe AutomationUIConfig iaf-system
   ```

   If the `READY` state for this resource gets stuck on `False`, the first thing to look at are the logs produced by the Zen operator located in the `ibm-common-services` project.

   ```
   # Get full logs for Zen operator
   oc logs -n ibm-common-services $(oc get pods -n ibm-common-services | grep
   zen-operator | awk '{print $1}')

   # If you rather you can follow the Zen operator logs
   oc logs -f -n ibm-common-services $(oc get pods -n ibm-common-services |
   grep zen-operator | awk '{print $1}')
   ```

3. Check the `Cartridge` resource status. This is another way to get the status of the Automation Foundation Core services. A cartridge is a module or extensions required by CP4BA. The status of the base `Cartridge` resource can be checked with the following commands:

```
# Check Cartridge status
oc get Cartridge icp4ba
```

Output:
```
NAME      READY
icp4ba    True
```

When the Automation Foundation Core services deploy properly, the `READY` status for this resource is set to `True`. You can gain additional insights for the Cartridge resource by running the following command:

```
# Check Cartridge status
oc describe Cartridge icp4ba
```

On successful deployments the Conditions section of this resource should show three different checks all with a `Status` set to `True`.

If the status of this resource remains set to `False` during the installation or if error messages are available, a good place to start looking for answers is the automation foundation core operator logs.

```
# Get automation foundation core logs
oc logs $(oc get pods | grep iaf-core-operator | awk '{print $1}')
```

4. Check the status of the of the `ICP4ACluster` resource. This custom resource, managed by the CP4BA Operator, contains the overall status of the CP4BA installation as well as the individual deployment status for each of the CP4BA components deployed. To get the details on the `ICP4ACluster` resource run the following command:

    ```
    # Get details of ICP4ACluster resource
    oc describe icp4acluster icp4adeploy
    ```

    The individual deployment status for CP4BA components can be found under the `Status` section of the command output. For example, to see the deployment status of the Content Platform Engine, which is part of the template installed in this lab, look for the `Cpe` stanza. Components that have been properly deployed shows as Ready or Successful as listed in the snippet below.

    ```
    Cpe:
      Cpe Deployment:       Ready
      Cpe JDBC Driver:      Ready
      Cpe Route:            Ready
      Cpe Service:          Ready
      Cpe Storage:          Ready
      Cpe Zen Integration:  Ready
    ```

    It is important to understand that the `Status` section is dynamically populated as the CP4BA operator makes progress with the installation, hence the `Status` section for specific components of CP4BA will not be populated until the operator tries to install the specific component.

    When failures occur during the deployment of CP4BA components, errors are reported via the component specific status section of the `ICP4ACluster` resource. In those cases, you should look at the pods and logs for the component that is failing to better understand the failure. You can also inspect the CP4BA Operator Logs. See the Inspect CP4BA Operator Logs for additional details.

5.4 Inspect CP4BA Operator Logs

The following steps show you how to inspect the CP4BA Operator logs. These logs are the first place to look when problems are found during the CP4BA Installation. The CP4BA Operator produces two types of logs. Logs produced by the standard out/error of the operator and logs produced by the operator framework and stored in the container's filesystem. In our experience the second type of logs are clearer and more useful, but we will show you how to look at both. These steps should be run from the bastion host.

To get the logs produced by the standard out/error of the CP4BA operator do the following:

1. Go to the OpenShift project where CP4BA is installed. This is the same project specified in the `cp4baProjectName` variable inside the `05-parametersForCp4ba.sh` script.

   ```
   # If you use a project name other than ibm-cp4ba you must use that instead
   of ibm-cp4ba
   oc project ibm-cp4ba
   ```

2. Use the `oc logs` command to get the standard out/error from the operator pod.

   ```
   # Get full set of log entries from CP4BA operator pod
   oc logs $(oc get pods | grep cp4a-operator | awk '{print $1}')

   # If you rather follow the logs
   oc logs -f $(oc get pods | grep cp4a-operator | awk '{print $1}')
   ```

To get logs produced by the operator framework:

1. Carefully inspect the date and time of each file, all called `stdout`, listed by the command below and copy the full path of the file that you want to inspect. Typically, you want to look at the current file being used by the operator, that is the file with the latest date and time in the list. This step can be a little cumbersome if the `stdout` file generated by the operator framework has rolled over multiple times.

   ```
   oc exec $(oc get pods | grep cp4a-operator | awk '{print $1}') -- sh -c
   "ls -ltrR /tmp/ansible-operator/runner/icp4a.ibm.com/v1/ICP4ACluster/ibm-
   cp4ba/icp4adeploy/artifacts/*/stdout"
   ```

2. Once you have the full path of the `stdout` file to be inspected, use it to copy or inspect the content of the file.

   ```
   # Set the value of the STDOUT_PATH variable to the path of the stdout file
   that you want to inspect
   STDOUT_PATH=<stdout path here>

   # Copy the file to the bastion host if you want the full log
   oc cp $(oc get pods | grep cp4a-operator | awk '{print
   $1}'):${STDOUT_PATH}/cp4a-operator.log

   # If you rather follow the file
   oc exec $(oc get pods | grep cp4a-operator | awk '{print $1}') -- tail -f
   $STDOUT_PATH
   ```

   When trailing the stdout file, you can see color coded messages. Look out for red as a sign of

trouble. You can also search for "error" or "fail" strings (make the search case insensitive) when looking for errors.

5.5 Inspecting and Restarting Jobs on Error

As previously stated in the lab instructions, all pods responsible to execute job tasks should go into Completed status. At times these pods might end up in `Error` state but eventually another instance of the pod is automatically created to retry the failed task. When a subsequent pod for the same job goes to `Completed` state the task required by the job is completed and the previous pod in `Error` state can be deleted. There are times when every automatic attempt to complete a job fails. In these situations, you can look at the logs for the failing pod, correct the issues when possible and manually restart the job so that the required installation task can be completed. The following steps shows you how to inspect the logs of a failed pod and restart the job. These steps must be run from the bastion host.

1. Get the logs from the failed pod. You will need the name of the pod to run this command.

    ```
    # Replace <pod-name> with the name of the failing pod.
    oc logs <pod-name>
    ```

2. Once you have corrected the issue, you can restart the job with the following commands:

    ```
    # Replace <pod-name> with the name of the failing pod.
    export POD_NAME=<pod-name>

    # Get name of the job that initialized the pod.
    export JOB_NAME=$(oc get pod $POD_NAME -
    o=jsonpath='{.metadata.ownerReferences[?(@.kind=="Job")].name}')

    # Restart the job
    oc get job $JOB_NAME -o json | jq 'del(.spec.selector)' | jq
    'del(.spec.template.metadata.labels)' | oc replace --force -f -
    ```

5.6 Inspecting Individual Pod Health

All pods, other than pods that run job tasks, deployed by the CP4BA operator should go into `Running` state and all containers running within should report as ready. When issues arise during installation you might find one or more pods suck on `Pending`, `Init`, `Error` or other states. These pods might require further inspection to determine issues. To determine the health of the pod and find the root cause of the problem you can run the commands listed below. These steps must be run from the bastion host.

1. Verify the current state of the pod

    ```
    # Replace <pod-name> with the name of the pod you are inspecting.
    oc get pod <pod-name>
    ```

    If the `STATUS` column shows `Running` and the `READY` column shows `1/1`, or `2/2`, or `3/3`, etc., then the pod is in a healthy sate. If the pod `STATUS` column is stuck in any other state than `Running` or if the `READY` column never goes to an equal number above and below the `/` character then the pod might be having trouble starting or staying up and you need to inspect the pod further.

2. Look at the details of the pod to see if there are any events that might be showing errors. This could happen for example when a pod is waiting for storage to be provisioned but OpenShift is unable to obtain it from the storage providers. You can see the events associated with a pod by inspecting

the `Events` section available in the full description of the pod.

```
# Replace <pod-name> with the name of the pod you are inspecting.
oc describe pod <pod-name>
```

3. If further analysis is needed you might have to look at the logs for each of the containers running as part of the pod.

```
# If the pod is running a single container, you can run the following
command.
# Replace <pod-name> with the name of the pod you are inspecting.
oc logs <pod-name>

# If the pod is running more than one container use the -c option to
specify the container.
# Replace <pod-name> with the name of the pod you are inspecting.
# Replace <container-name> with the name of the container to get the logs
from.
# To get the container names by first running the command without the -c
option
oc logs <pod-name> -c <container-name>
```

4. Sometimes restarting a pod fixes the problem without having to make any changes.

```
# Restart a pod. Replace <pod-name> with the name of the pod you want to
restart.
oc delete pod <pod-name>
```

After running this command the pod will restart automatically or in very rare cases will be eventually recreated by the CP4BA operator.

5.7 Clean CP4BA Failed Installation

Follow the steps below to clean up a failed CP4BA installation. These steps should be run from the bastion host. Depending on where the failure occurred, one or more of the following commands might fail because the resource being removed does not exist. Failures stating that resource does not exist are ok.

1. Go to the OpenShift project where CP4BA is installed. This is the same project specified in the `cp4baProjectName` variable inside the `05-parametersForCp4ba.sh` script.

```
# If you use a project name other than ibm-cp4ba you must use that instead
ibm-cp4ba
oc project ibm-cp4ba
```

2. Run the `98-deleteCp4baDeployment.sh` script to clean up the installation. This script will remove all the resources deployed by the CP4BA installation.

```
# Go to your rapid deployment scripts directory
cd /cp4ba/cp4ba-rapid-deployment/cp4ba-21-0-3/mycluster/deployment-db2-
cp4ba
```

```
# Run script
./98-deleteCp4baDeployment.sh
```

If you are planning to install the CP4BA again, make sure that you have dropped and recreated the Databases as well by running the `99-dropCp4baDBs4Db2OnOCP.sh` and `03-createCp4baDBs4Db2OnOCP.sh` found on the rapid deployment scripts directory.

5.8 Shutting down the OpenShift Cluster

When you want to shut down the Openshift Cluster, its recommended to avoid doing so while the cluster is upgrading, while the DB2 Operator is busy installing DB2, or while the CP4BA cluster is being deployed. Please refer to section 5.2 Determining Deployment Status of the CP4BA Cluster to check weather this is the case.

For a general discussion about shutting down Openshift clusters, please see the openshift installation on this page: https://docs.openshift.com/container-platform/4.8/backup_and_restore/graceful-cluster-shutdown.html Care need to be taken to verify that no certificates expire while the cluster is shut down, and it's also recommended to take a backup of the etcd databases of the Openshift cluster.

When the cluster is idle, apply the following steps for shutting it down:

1. In a terminal window, run the following command to shut down the LDAP server:

    ```
    sudo /root/bin/stopSds.sh
    ```

    The output should indicate that the Directory Server was successfully stopped, and that DB2 stop command was executed.

2. Close all windows and log out of the Bastion "VM0". Click at upper right corner, then click on "cp4badmin" to find the log out function, then click on "Log Out" and confirm the dialog:
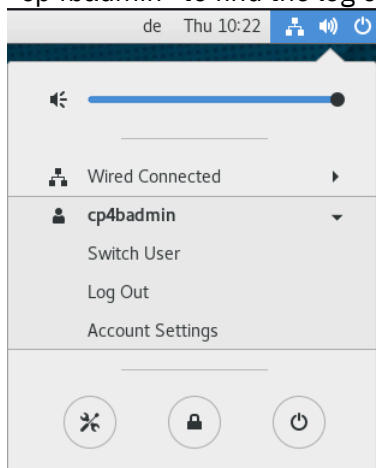
3. You should get the login screen when you logged out. On the login screen, click the indicated icon to bring back the list of servers of the environment. In the resulting drop-down dialog click on "View all VM(s)".

4. This should bring you back the "IBM Demonstration Portal" with the list of all Virtual Machines of the environment. Click on one of the stop icons with the green rectangle around it to shut down all

VMs in the right order. Avoid clicking the ones with the red x on it, that might result in the wrong order and might cause problems after starting up the environment again:
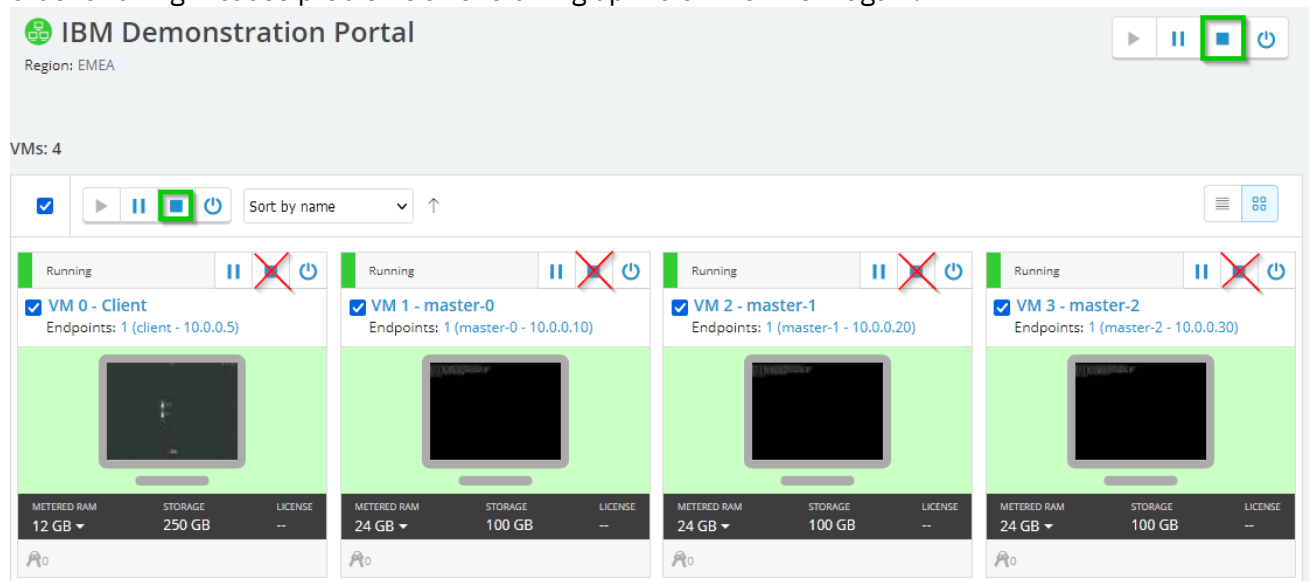
5.9 Clean Failed DB2 installation

Follow the steps below to clean up a failed DB2 installation. These steps should be run from the bastion host. Depending on where the failure occurred, one or more of the commands provided below might fail because the resource being removed does not exist. Failures stating that resource does not exist are ok.

1. Go to the OpenShift project where DB2 is installed. This is the same project specified in the `db2OnOcpProjectName` variable inside the `01-parametersForDb2OnOCP.sh` script.

    ```
    # If you use a project name other than ibm-db2 you must use that instead
    ibm-db2
    oc project ibm-db2
    ```

2. Delete the instance of the DB2 cluster created.

    ```
    oc delete Db2uCluster db2ucluster
    ```

    Expected output:
    ```
    db2ucluster.db2u.databases.ibm.com "db2ucluster" deleted
    ```

3. Delete the instance of the DB2 operator subscription created

    ```
    oc delete subscription db2u-operator
    ```

    Expected output:
    ```
    subscription.operators.coreos.com "db2u-operator" deleted
    ```

4. Delete the instance of the DB2 operator created

    ```
    oc delete csv $(oc get csv --no-headers | awk '{print $1}')
    ```

Expected output:
```
clusterserviceversion.operators.coreos.com "db2u-operator" deleted
```

5. Wait for all PODs to be removed from the DB2 project.

```
oc get pods
```

Expected output:
```
No resources found in ibm-db2 namespace.
```

6. Delete the project where DB2 was installed. This step could take a couple of minutes.

```
# If you use a project name other than ibm-db2 you must use that instead
ibm-db2
oc delete namespace ibm-db2
```

Expected output:
```
namespace "ibm-db2" deleted
```

Once these steps are completed, you can try the DB2 installation once again.

5.10 Delete Single Database Manually

The following steps shows how you can gain access to the container where DB2 is running and delete a database manually using the DB2 command line. These steps should be run from the bastion host. For this section, we will show you how to delete the sample database that comes with the DB2 installation called BLUDB.

1. Go to the OpenShift project where DB2 is installed. This is the same project specified in the db2OnOcpProjectName variable inside the 01-parametersForDb2OnOCP.sh script.

```
# If you use a project name other than ibm-db2 you must use that instead
ibm-db2
oc project ibm-db2
```

2. Use the oc command to access the container where DB2 is running.

```
# Access DB2 container
oc rsh -c db2u c-db2ucluster-db2u-0
```

You should see the following command prompt (or similar) once inside the container:
```
sh-4.2$
```

3. Change user to db2inst1 which is the default DB2 user created by the install scripts.

```
# Run the su command inside of the db2 container
su - db2inst1
```

4. Use the DB2 command line to delete the BLUDB

```
db2 drop database BLUDB
```

Expected output:

```
DB20000I  The DROP DATABASE command completed successfully.
```

5. Exit the session as `db2inst1`

```
# Run exit command to go back to DB2 container as default user
exit
```

6. Exit the DB2 container

```
# Run exit command to exit the DB2 container and go back to bastion
exit
```

## 5.11 Checking availability of DB2 and LDAP server

This step can be done to verify that the LDAP and the DB2 Server can be reached from the CP4BA cluster. You can apply the step after the cp4ba-clusteradmin-setup.sh script has completed, and the Cloud Pak 4 Business Automation operator is running. To prepare, use the following steps:

1. In a Terminal window, run this command to determine IP Address and Port number of the DB2 server, note down the values obtained for later:

```
cd /cp4ba/cp4ba-rapid-deployment/cp4ba-21-0-3/mycluster/deployment-db2-
cp4ba

egrep -i 'ldapServer|ldapPort' 05-parametersForCp4ba.sh
```

2. Open the browser and use the bookmark to login on the Openshift Web Console. Login using admin / passw0rd

3. In the navigation bar open "Workloads" and select "Pods". Set the Project name to **ibm-cp4ba** then write "ibm-cp4a-operator" into the filter box below the project name. That should narrow down the pods to just one or two. Click on the pod name, which is starting with "ibm-cp4a-operator".
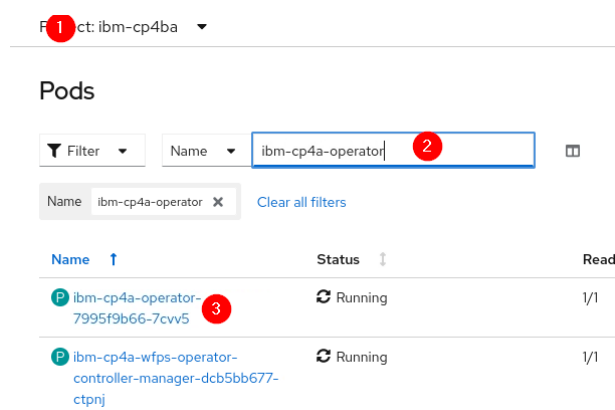
On the resulting window with information about the cp4a operator pod, click on the "Terminal" tab. To check for availability of the LDAP server, run this command. Please replace the ldap server name and port with the information obtained earlier:

```
curl -v telnet://<ldap server name>:<ldap server port>
```

```
sh-4.4$ curl -v telnet://10.0.0.5:389
* Rebuilt URL to: telnet://10.0.0.5:389/
*   Trying 10.0.0.5...
* TCP_NODELAY set
* Connected to 10.0.0.5 (10.0.0.5) port 389 (#0)
^C
sh-4.4$ █
```

4. When it shows "Connected" the server can be reached from inside the Openshift Cluster. If you get nothing, then no connection can be done. That might be the case if you for example forgot to start the LDAP Server. Abort the command by pressing CTRL-C. The CTRL key might have a different caption on your keyboard.

5. An alternative method would e.g. run a different command like `ldapsearch`. In the form stated below it would use anonymous access to the LDAP server. Take attention as it would list the complete content of the LDAP server, which might be very long in a Production environment:

```
ldapsearch -x -b dc=example,dc=com -H ldap://<ldap server ip address>
```

```
sh-4.4$ ldapsearch -x -b dc=example,dc=com -H ldap://10.0.0.5 | tail
objectclass: groupOfNames
objectclass: top
member: cn=cp4badmin,dc=example,dc=com

# search result
search: 2
result: 0 Success

# numResponses: 205
# numEntries: 204
```

6. Testing connection to the DB2 server can be made similar and is very useful, as connection might be inhibited between the pods in the OCP cluster by network policies. For addressing the DB2 server, the OCP service name of DB2 would be used, it is `c-db2ucluster-db2u-engn-svc`. After that comes the OCP project name, and the keyword svc. The name is resolved to an IP address by a name server which is part of the OCP cluster.

```
curl -v telnet:// c-db2ucluster-db2u-engn-svc.ibm-db2.svc:50000
```

```
sh-4.4$ curl -v telnet:// c-db2ucluster-db2u-engn-svc.ibm-db2.svc:50000
* Bad URL
* Closing connection -1
curl: (3) Bad URL
* Rebuilt URL to: c-db2ucluster-db2u-engn-svc.ibm-db2.svc:50000/
*   Trying 172.30.201.118...
* TCP_NODELAY set
* Connected to c-db2ucluster-db2u-engn-svc.ibm-db2.svc (172.30.201.118) port 50000 (#0)
> GET / HTTP/1.1
> Host: c-db2ucluster-db2u-engn-svc.ibm-db2.svc:50000
> User-Agent: curl/7.61.1
> Accept: */*
```

Again when you see "Connected" the connection to the server can be made. Press CTRL+C to stop the command.