Exercise 1. Accessing lab environment

*(*with hints)

Estimated time

01:00h

Overview

This exercise introduces and familiarizes you with the class lab environment. It includes discovering key documentation that supports the objectives of this course.

Objectives

At the end of this exercise, you should be able to:

- Access lab systems
- Configure Command Line Interface autocompletion
- Change the Logo of your OpenShift cluster to be fit for a Power 10 Cloud!

Introduction

This is an exploratory lab exercise to verify the lab environment, access OpenShift documentation.

Requirements

- You must be comfortable using a command line, note that any cloud will require that.
- The TechZone Project document.
- This workbook.
- A computer with a network connection to the lab environment.
- Access to a POWER system that is made available by TechZone - either P8 or P9 are available at TechZone and there is not much difference in performance for most of the labs we will be running.
- Username and password of all systems assigned to your team. Those are provided by your instructor.

1. Lab operations

This part of the exercise verifies the configuration of the lab systems. If you find that the configuration does not match the examples in this guide, notify your instructor.

Section 1. IBM TechZone OpenShift on Power

In the future you may use what you learned here in this training with your customer and colleagues spreading the knowledge you acquired. You are using the TechZone environment built on Purpose to be a complete OpenShift solution on a single server. It has 16 vCPU and 64 GB of RAM available and will suffice for all the labs we will go through.

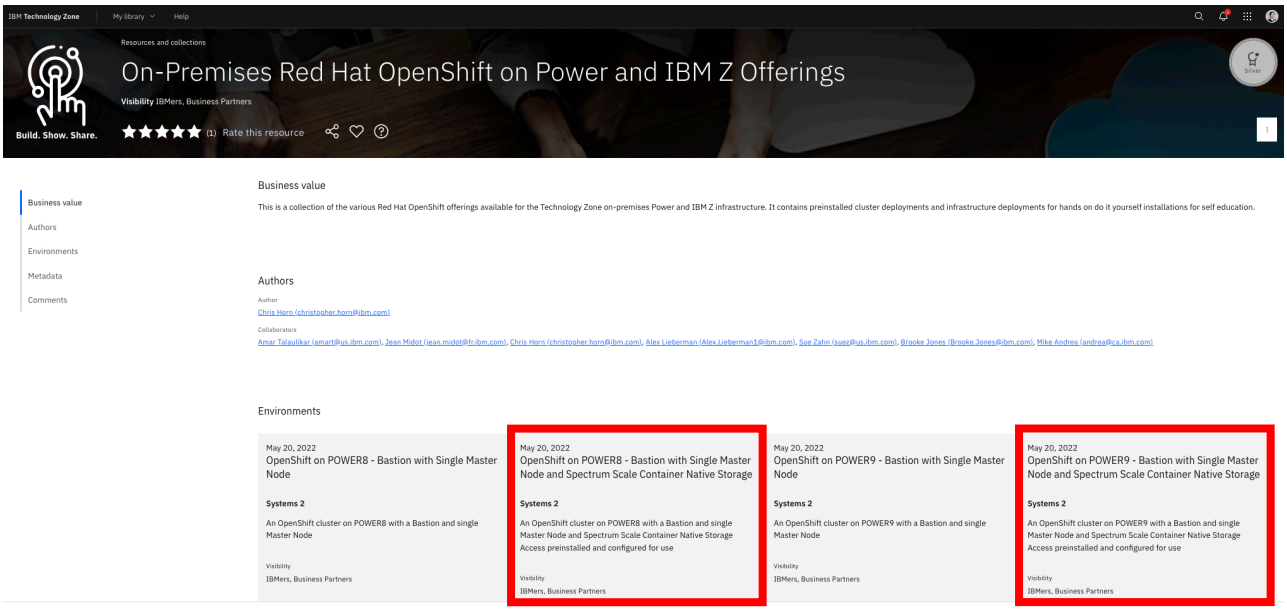
This offering was created to show all the possibilities for a Cloud Environment.

Important: For a quick start, the Lab Environment for this workshop was already created for you, and you do not need to open any request for this workshop. In the future, if you want to Demo the cloud capabilities to your customer you can always order an environment just like the one we are using at:

https://techzone.ibm.com/collection/on-premises-redhat-openshift-on-power-and-ibm-z-offerings

Choose either:

- OpenShift on POWER8 - Bastion with Single Master Node and Spectrum Scale Container Native Storage
- OpenShift on POWER9 - Bastion with Single Master Node and Spectrum Scale Container Native Storage

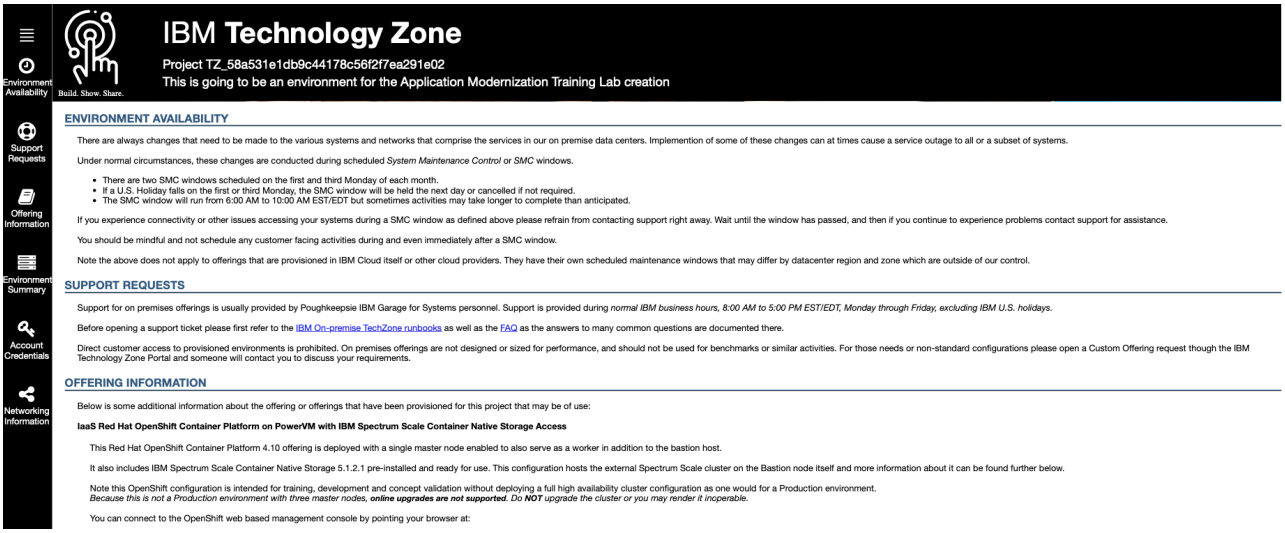


Note that this is available for both IBMers and Business Partners so you can reach out to your customers and show the capabilities of OpenShift on Power. Pick the latest OpenShift Version Available.

Section 2. First connection to your environment

As you get your Environment you will be given a Project Kit. This will contain the IP address for your environment.

It looks like this:



__ 1. You will need this information to work on your lab.

Enter your Power server system assignments in the table below and save it:

Default console	IP Address	Login	Password

Important:

We will need both terminal and web access to the environment. You may use your preferred terminal to connect via SSH to the IP address. If you don't know how to do this, take the time to get started and ask help from one of the instructors.

Most of the commands will be done with the normal cecuser user.

All Command Line Interface are expected to be done on the Bastion node terminal with cecuser unless explicitly mentioned on the Example.

Optional: If you need tips for connecting to the Bastion node terminal

If you are a Windows user Putty is a great terminal to use, I used it for many years.

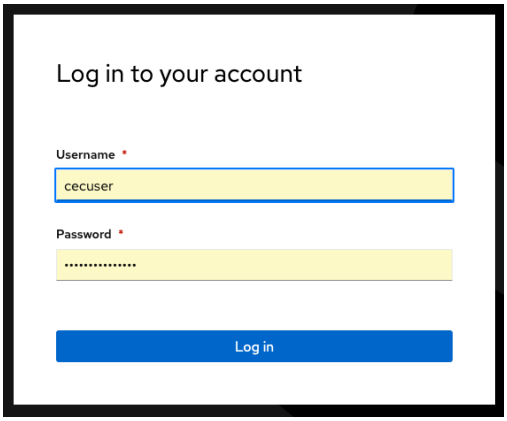
For Mac users iterm2 is the terminal I use nowadays.

Section 3. Connecting to the web console

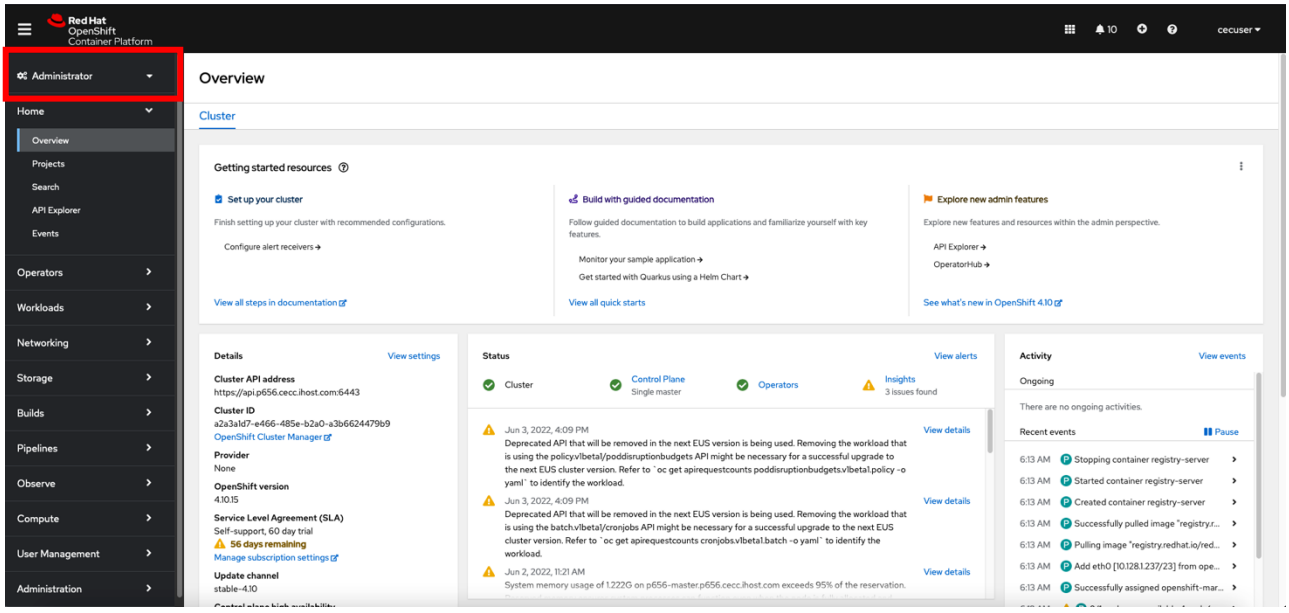
__ 2. Point your browser to your web console (found in the project kit and pasted on step number 1)

__ 3. Click on the htpasswd option:

__ 4. Add your user and password contained on the step 1 and Click login.



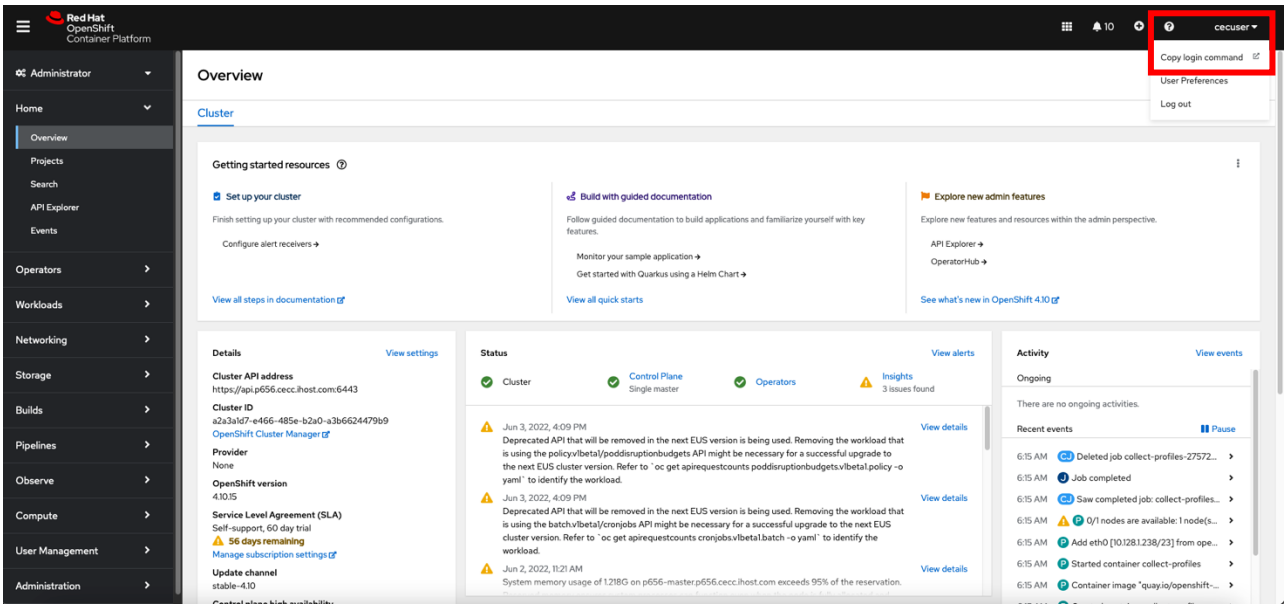
__ 5. Take about 10 minutes to get familiar with the navigation of it if you never done it, you can change between Developer and Administrator view on the menu on the top left.



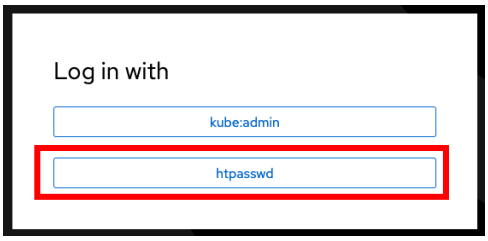
Section 4. Copy login command for CLI use.

If you need to login again to the CLI, for any reason, you can find the login command on main OpenShift web console page.

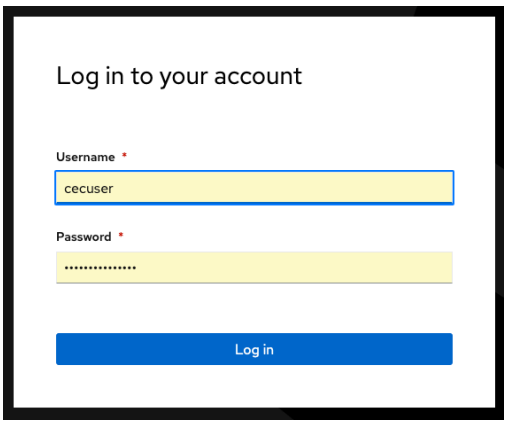
__ 6. On the top right side, you will see the cecuser drop down click on it and then on “Copy login command”



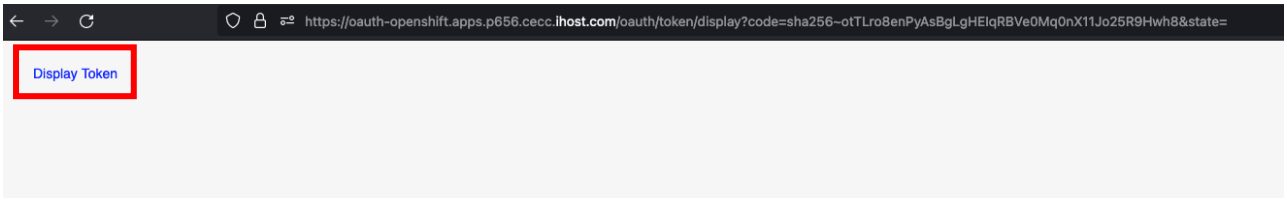
__ 7. Once again click on the htpasswd option:



__ 8. Add your user and password contained on the step 1 and Click login.



__ 9. Click on Display Token on the top right



__ 10. You can use the oc login command whenever your Authorization is expired. You may need to use the API token for login in into the registry.

Important: You may need these steps more than one time during this training.

2. Setting up the environment

Some basics settings will be done in this lab you can familiarize yourself with the lab environment Command Line Interface and to the Web Interface.

Section 1. Configure Command Line Interface auto-completion

This is a great first step to get to know your environment and will help you on the lab making sure you can use the Tab key to autocomplete the Command Line Interface “oc” client.

Important:

During the whole course the CLI steps will be followed by examples that can be used to complete the step. The command will be in Bold. Copy the entire line and paste it on the shell. Expect an output like the one you see in the example. Commands may be long. Make sure you copied the full command that is in bold preceded by the pound Signal #.

__ 11. Run the command “oc completion bash” that generates the full completion list in text to the standard output, direct it to oc_bash_completion.

Example:

```
# oc completion bash > oc_bash_completion
#
```

Note: If you get the error bellow:

```
# oc completion bash > oc_bash_completion
error: You must be logged in to the server (Unauthorized)
```

Look at the Section 3 to make sure you log in into your console

---

__ 12. Copy the oc_bash_completion to the correct location to be used on future sessions.

Example:

```
# sudo cp oc_bash_completion /etc/bash_completion.d/
#
```

__ 13. Check that the file is correctly in place.

Example:

```
# ls -la /etc/bash_completion.d/oc_bash_completion
-rw-r--r--. 1 root root 712992 Jun  4 05:22 /etc/bash_completion.d/oc_bash_completion
#
```

__ 14. Load bash again to make sure you can use the bash completion

Example:

```
# bash
#
```

This will get your auto completion set.

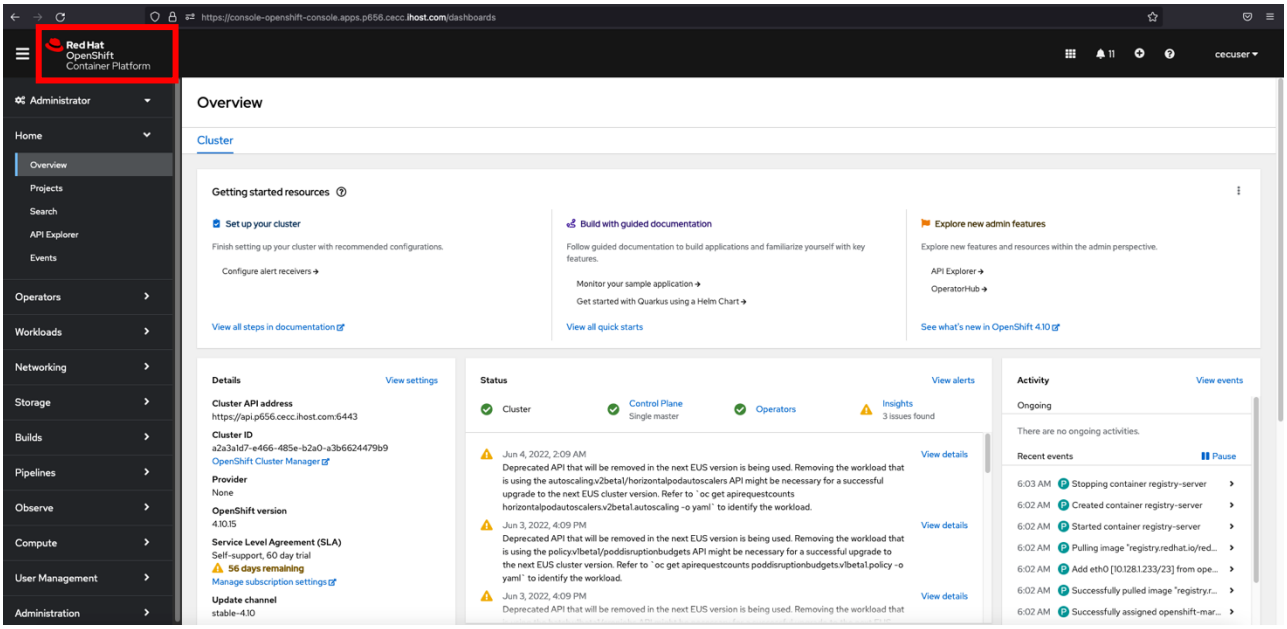
Optional:

You may test it worked by typing “oc g” and press tab it should automatically complete as “oc get”

3. Customize your Cloud Console.

The idea is to provide customers with a Cloud look and feel and some customizations can be done by taking the logo from the company and adding it to the console.

__ 15. Log in to your console. If you don't remember how to do it, no worries, go back to Section 3 of the "1. Lab Operations" part of this Lab on page 4. Note that the OpenShift logo will be on the left Corner.



We have a Logo done specially for this by a graphical artist that fits perfectly for the console. The Graphical artist left it on the /home/cecuser/assets directory. If you want to create one with the logo of your customer to create a demo you can also do that, best size is 200x50 pixels.

__ 16. Create a configmap that will contain the binary data for the file so it can be read by the console on the openshift-config namespace. Configmaps are used on Kubernetes by customers and applications to hold information, files and environment variables. A special type of configmap, called secret, is used for passwords, certificates and other configuration that need extra security we will use it on next labs.

Example:

```
# oc create configmap icon-power --from-file ./assets/Power10Cloud.png -n openshift-config
configmap/icon-power created
```

Now that the image is available to be used on OpenShift you can add it to the console. As we know OpenShift is all automated with Operators and we can configure the operator to do all the work for us. We just state the specifications of is needed to do on the console and the operator will do everything to configure it. The whole Kubernetes configuration (and OpenShift inherited) is

done mainly in JSON or YAML format. JSON is preferred when doing command lines because the configuration statement is clearly defined and that is what we will use since we are learning. Using command lines is called imperative. Using YAML files that can be created and applied is called declarative method and used by many automation pipelines. A new trend is called GitOps that automates the command line work and the preferred method for automation for applications running on Kubernetes.

Note: Ansible has no real place in Application Modernization on OpenShift as an external tool, should be treated as Infrastructure Modernization. On the server side, Ansible is meant to connect to Operating Systems that needs configuration and modules are available for many of them.

Since OpenShift is fully automated by Operators, the external automation Ansible provides is redundant. Although you could create Ansible operators to be used inside OpenShift, golang is the preferred method by RedHat.

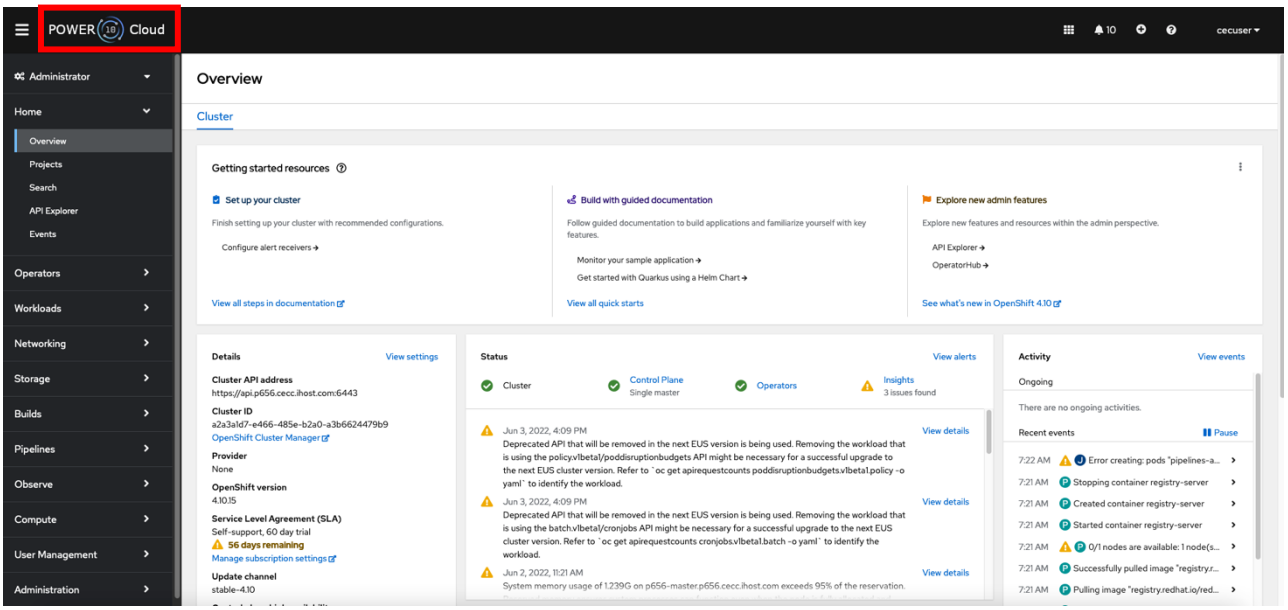
Use Ansible to automate tasks on RHEL, AIX and IBMi and modernize the infrastructure for traditional applications.

__ 17. Now merge with the existing cluster console operator (using the --type=merge patch command) and specification that contain the customization with the Logo File and a Different Product Name.

Example:

```
# oc patch consoles.operator cluster --patch '{"spec": {"customization":
{"customLogoFile": {"key": "Power10Cloud.png","name": "icon-
power"},"customProductName": "PowerCloud"}}}' --type=merge
console.operator.openshift.io/cluster patched
```

This will reset your console and after it comes back (wait a minute or two) You will see that the Icon is changed, and the page now is “PowerCloud” on your Browser.



4. Access relevant documentation

Optional: This part of the exercise presents the key documentation that supports the objectives of this course.

__ 1. Go to *https://docs.openshift.com/* in a web browser to view the OpenShift Container Platform (**OCP**) documentation. Spend a few minutes navigating to any links of interest.

__ 2. Go to *https://kubernetes.io/* in a web browser to view the Kubernetes (**K8s**) website. Spend a few minutes navigating to any links of interest.

__ 3. Go to *https://cri-o.io/* in a web browser to view the CRI-O container runtime website. Spend a few minutes navigating to any links of interest.

__ 4. Go to *https://podman.io/* in a web browser to view the Pod Manager Tool (**`podman`**) website. Spend a few minutes navigating to any links of interest.

__ 5. Tell your instructor that you have completed the lab exercise.

End of exercise

Exercise review and wrap-up

The first part of the exercise verified lab access and was meant to make you comfortable with the Command Line Interface and the Web Graphical Interface.