RED HAT®
CONSULTING

redhat.

OCP Tips and Resources

Travis Michette

Version 1.0

Table of Contents

1. Openshift Tips and Lab Guides

1.1. Placing OCP Variables in .bashrc

The initial lab setup command will take the values for the OpenShift environment and store in the file **/usr/local/etc/ocp4.config**. This allows the file to be sourced for the labs and the user to provide the variables in the OC command for things like **login**. These variable names are lengthy and this step requires remembering knowing where the file is located and sourcing it each time.

Modifying the **.bashrc** *File for OCP Variables*

It is possible to modify the **.bashrc** with your own variables preventing the need to source the **/usr/local/etc/ocp4.config** file each time you wish to complete a guided exercise or a lab.

1. Place Comment in **.bashrc** for your custom variables.

Listing 1. Creating Comment Section of **.bashrc**

```
[student@workstation ~]$ echo "#" >> .bashrc
[student@workstation ~]$ echo "#" >> .bashrc
[student@workstation ~]$ echo "#=== OpenShift Custom Varibables for Lab Environment===#" >> .bashrc
```

2. Copy file contents from the **/usr/local/etc/ocp4.config** file into the **.bashrc** file.

Listing 2. Appending Contents from **/usr/local/etc/ocp4.config** *to* **.bashrc**

```
[student@workstation ~]$ cat /usr/local/etc/ocp4.config >> .bashrc
```

3. Edit the **.bashrc** file to change variable names

Listing 3. Opening **.bashrc**

```
[student@workstation ~]$ vim .bashrc
```

Listing 4. Locating Variables from **/usr/local/etc/ocp4.config** *File*

```
... Output Omitted ...

#=== OpenShift Custom Varibables for Lab Environment===#
RHT_OCP4_MASTER_API=https://api.ocp-qmkgvgdonzljvvp200406.do280.rht-na.nextcle.com:6443
RHT_OCP4_CLUSTER_ID=3ae5f2bb-dd1c-49b3-88ec-2824d6db8074
RHT_OCP4_WILDCARD_DOMAIN=apps.ocp-qmkgvgdonzljvvp200406.do280.rht-na.nextcle.com
RHT_OCP4_KUBEADM_PASSWD=BwYfi-VM3Kt-a34FS-uoKqC
RHT_OCP4_USER_PASSWD=1d48c9fe757fd761df6b591306f1887e

... Output Omitted ...
```

Listing 5. Modified Variables from **/usr/local/etc/ocp4.config** *File*

```
... Output Omitted ...

#=== OpenShift Custom Varibables for Lab Environment===#
API=https://api.ocp-qmkgvgdonzljvvp200406.do280.rht-na.nextcle.com:6443
OCP_ID=3ae5f2bb-dd1c-49b3-88ec-2824d6db8074
OCP_DOMAIN=apps.ocp-qmkgvgdonzljvvp200406.do280.rht-na.nextcle.com
OCP_ADMINPW=BwYfi-VM3Kt-a34FS-uoKqC
OCP_USERPW=1d48c9fe757fd761df6b591306f1887e


... Output Omitted ...
```

The modifications performed above will automatically load variables as environmental variables for your BASH shell. There is no need to source the files from the lab and you can utilize your own variable names.

Utilizing Variables from **.bashrc**

Listing 6. Source Description

```
[student@workstation ~]$ oc login -u developer -p $OCP_USERPW $API
Login successful.

You don't have any projects. You can try to create a new project, by running

    oc new-project <projectname>
```

1.2. Setting up Command-Line Completion for the oc Command

The command line completion must be setup using the OC command to first extract all the commands possible and placed into a CLI completion file. The file is then copied to the **/etc/bash_completion.d/** where it can then be source from the user's **.bashrc** file.

1. Create the listing of commands for command line completion.

Listing 7. Create OCP Bash CLI Completion File

```
[student@workstation ~]$ oc completion bash > oc_bash_completion
```

2. Copy OCP Bash CLI Completion File to **/etc/bash_completion.d/**

Listing 8. Copy OCP Bash CLI Completion File

```
[student@workstation ~]$ sudo cp oc_bash_completion /etc/bash_completion.d/
```

3. Edit the **.bashrc** file for the **student** user

Listing 9. Edit **.bashrc** *for the User*

```
[student@workstation ~]$ vim .bashrc

... output omitted ...

# OCP Bash Completion
source /etc/bash_completion.d/oc_bash_completion

... output omitted ...
```

4. Source the **.bashrc** file for the user

Listing 10. Source the **.bashrc** *file*

```
[student@workstation ~]$ source .bashrc
```