Creating Containerized Services

Chapter objectives

1. Search for and fetch container images with Podman.
2. Run and configure containers locally.
3. Use the Red Hat Container Catalog.

Fetching Container Images with Podman

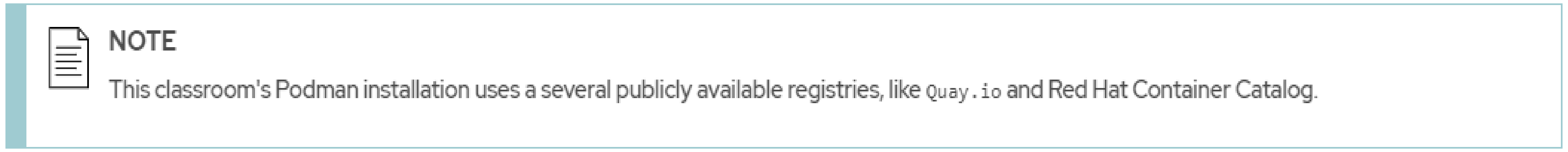
- Use podman search subcommand to find available images from remote or local registries

```
[student@workstation ~]$ sudo podman search rhel
```

- Use pull subcommand to download it to local storage

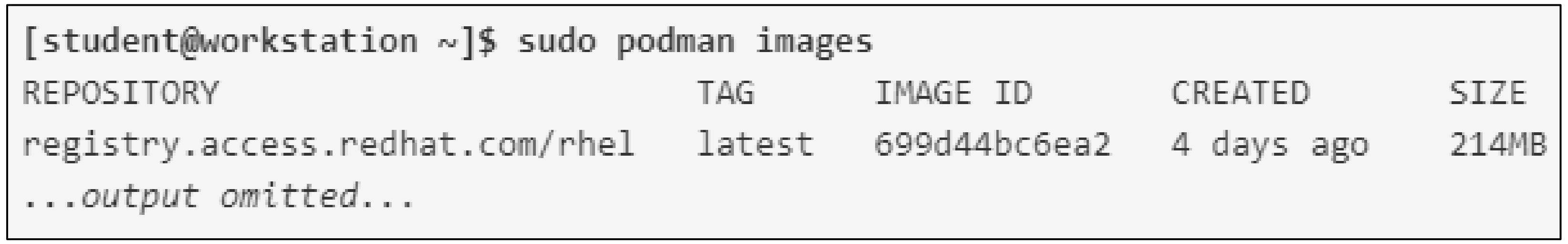
```
[student@workstation ~]$ sudo podman pull rhel

[student@workstation ~]$ sudo podman pull docker.io/library/httpd
```



NOTE

This classroom's Podman installation uses a several publicly available registries, like `Quay.io` and Red Hat Container Catalog.



```
[student@workstation ~]$ sudo podman images
REPOSITORY                        TAG       IMAGE ID       CREATED       SIZE
registry.access.redhat.com/rhel   latest    699d44bc6ea2   4 days ago    214MB
...output omitted...
```

Image : Naming convention

- registry_name/user_name/image_name:tag

- registry_name: FQDN of registry storing the image.
- user_name: user or organization the image belongs to
- image_name: unique name in user namespace
- tag: image version. If image name does not provide image tag, latest tag is assumed

Running Containers

- Use podman run subcommand to starts a container.

```
[student@workstation ~]$ sudo podman run ubi7/ubi:7.7
```

- Use podman run subcommand to starts a container with a command

```
[student@workstation ~]$ sudo podman run ubi7/ubi:7.7 echo “hello world”
```

- Image will be downloaded if it is missing

```
[student@workstation ~]$ sudo podman run ubi8/ubi sleep 100 &
```

- List running containers

```
[student@workstation ~]$ sudo podman ps
```

Podman flags

Flags	Description
--name	give a meaningful name to container
-l or --latest	execute any subcommand to the latest used container
-t or --terminal	equivalent to --tty, meaning a pseudo-tty is to be allocated for the container
-i or --interactive	equivalent to –interactive. When used, standard input is kept open into the
-d or --detach	equivalent to --detach. Means to start container and continue to run in the background.
-e or --env	Supply parameters to container

Example: -l flag

- Lets start a web server container in the background.

```
[student@workstation ~]$ sudo podman run --name mywww1  –d docker.io/library/httpd
```

- Verify the container is running

```
[student@workstation ~]$ sudo podman ps
```

- Let’s quickly stop it

```
[student@workstation ~]$ sudo podman stop -l
```

Example: -i and -t flag

- Starts a BASH terminal inside container, and interactively runs some commands in it

```
[student@workstation ~]$ sudo podman run --name mywww2 –it –d docker.io/library/httpd
bash-4.2# ls
…output omitted…
bash-4.2# whoami
root
bash-4.2# cat /etc/*release
…output omitted…
exit
[student@workstation ~]$
```

Example: -e flag

- Some containers need external parameters provided at startup.
- Use -e to inject environment variables into containers at startup.

```
[student@workstation ~]$ sudo podman run -e GREET=Hello -e NAME=Redhat \
Rhel7:7.5 printenv GREET NAME
Hello
Redhat
[student@workstation ~]$
```

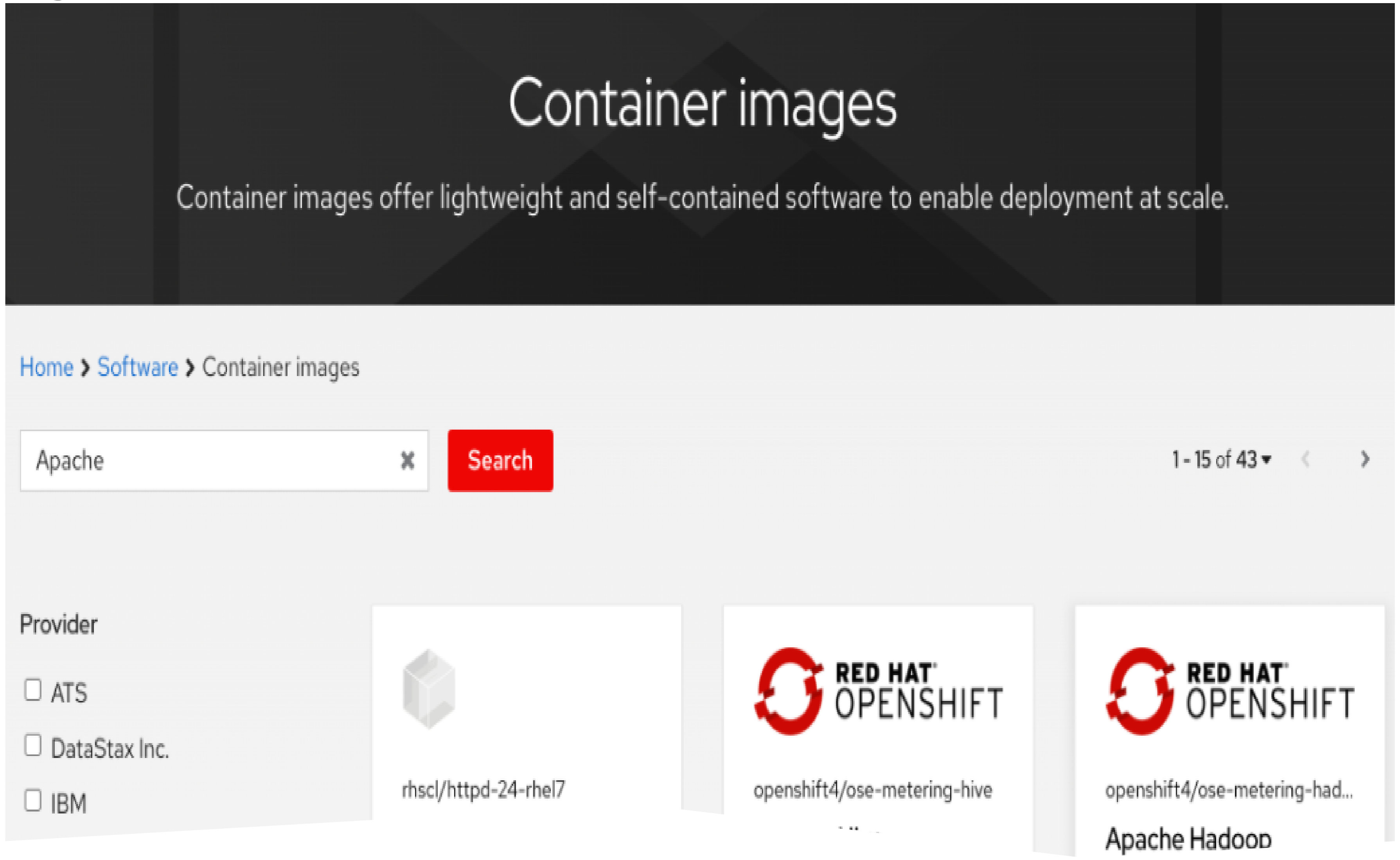
Example: -e flag

- Start MySQL container with needed environment variables

```
[student@workstation ~]$ sudo podman run ---name mydb \
> -e MYSQL_USER=redhat –e MYSQL_PASSWORD=r3dh4t \
> -d rhmap47/mysql:5.5
```

Using the Red Hat Container Catalog

- Finely tuned container images
- Maintained by Red Hat
- User-friendly interface
- Health index
- Errata documentation
- Get this image tab



Container images

Container images offer lightweight and self-contained software to enable deployment at scale.

Home > Software > Container images

Apache ✖ Search

1 - 15 of 43 ▾

Provider

☐ ATS

☐ DataStax Inc.

☐ IBM

rhscl/httpd-24-rhel7

openshift4/ose-metering-hive

openshift4/ose-metering-had...

Apache Hadoop

Guided Exercise: Creating a MySQL Database Instance

You should be able to:

- Start a database from a container image and store information inside the database.

Using Rootless Containers

After completing this section, you should be able to:

- Explain the differences between running root and rootless containers.
- Describe the advantages and disadvantages of each case.
- Run as root and rootless containers with Podman.

Evolution of Container Usage

- Privilege user
  - Needed to create resources such network interfaces, or mount file system
- Bad practice to use privilege user
- Security bugs, exposure to all possible attacks and CVE
- Better practice: Do not require privileged user
- Many community images: still require root to run
- Podman and OpenShift: Default start rootless container.
- Docker announced rootless mode

Pros and Cons : Rootless Containers

- Do not require root privileges to run
- Pros:
  - ✓ Root privilege within container, non-root privilege on host
  - ✓ New security layer
  - ✓ Multiple unprivileged users
  - ✓ Allows isolation inside nested containers
- Cons:
  - × Dropped capabilities
  - × Binding to port less than 1024
  - × Volume mounting

Understanding Rootless Containers

User Namespaces

- Containers isolate from host using namespace
- Rootless containers
- Container runs as root, appear under different ID to host
- Success attack on container, will only have capabilities of unprivileged user outside of container

Networking

- Rootfull container have full access to SDN
- Rootless container have limited access to SDN

Storage

- Rootfull container uses overlay2 (layers)
- Rootless container barred from mounting overlay file systems

Guided Exercise: Exploring root and rootless containers

You should be able to :

- Start a rootfull and rootless container
- Differentiate between them

Lab: Creating Containerized Services

You should be able to :

- Start a web application from container image
- Customize the container image.

Chapter Summary

In this chapter, you learned:

- Podman allows users to search for and download images from local or remote registries.
- The podman run command creates and starts a container from a container image.
- Containers are executed in the background by using the -d flag, or interactively by using the -it flag.
- Some container images require environment variables that are set using the -e option from the podman run command.
- Red Hat Container Catalog assists in searching, exploring, and analyzing container images from Red Hat's official container image repository.