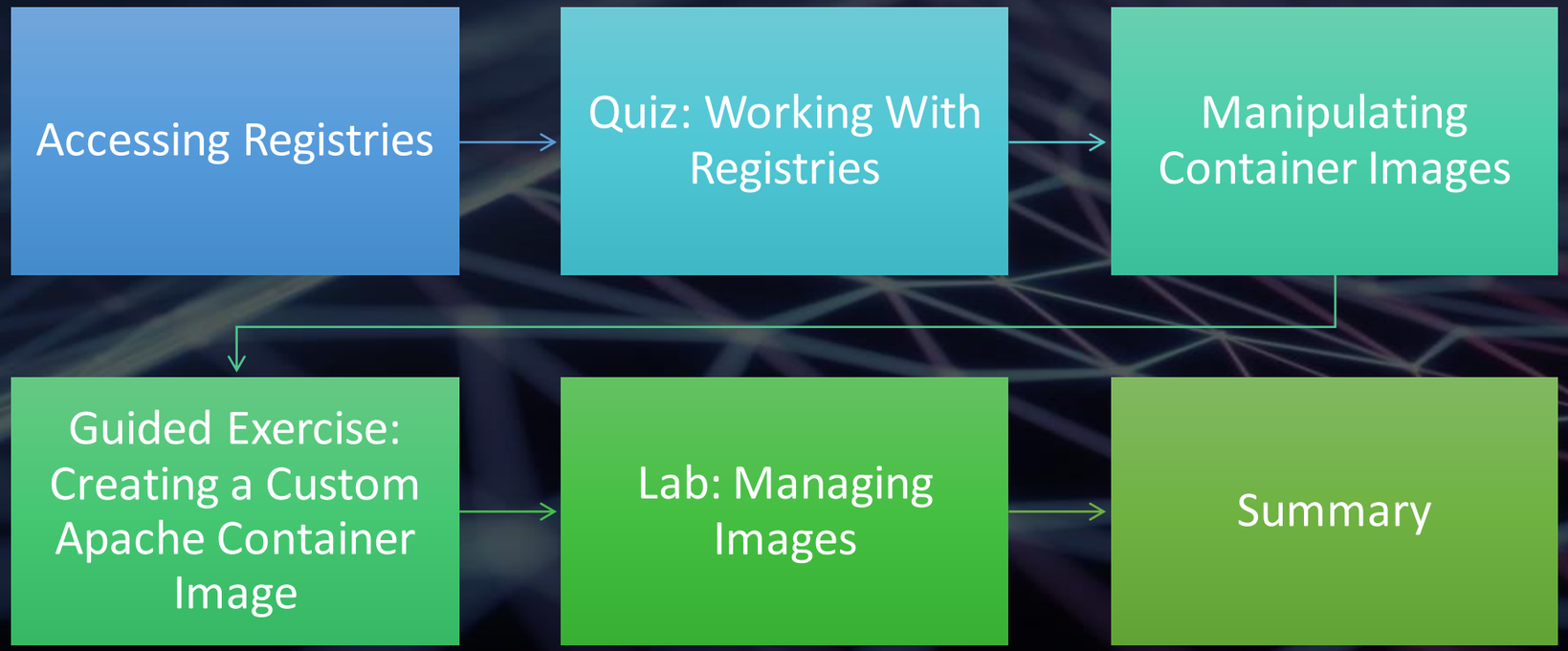
Managing Container Images

Chapter objectives



Accessing Registries

After completing this section, you will be able to:

Search and pull
- images from remote registries

List advantages of
- using certified public registry

Customize the configuration to
- access alternative container image registries.

List downloaded images
- from a registry to the local file system.

Manage
- tags to pull tagged images.

Public Registries

- Containing container images to download
- Store and distribute images
- Podman can search images from public or private registries
- Example of public registries:
  - Red Hat Container Catalog
  - Quay.io
  - Docker Hub
  - Google Container Registry
  - and etc

Red Hat Container Catalog

- Trusted Source: All container images comprise sources known and trusted by Red Hat.
- Original dependencies: None of the container packages have been tampered with, and only include known libraries.
- Vulnerability-free: Container images are free of known vulnerabilities in the platform components or layers.
- Runtime protection: All applications in container images run as non-root users, minimizing the exposure surface to malicious or faulty applications.
- Red Hat Enterprise Linux (RHEL) compatible: Container images are compatible with all RHEL platforms, from bare metal to cloud.
- Red Hat support: Red Hat commercially supports the complete stack.

QUAY.io

- Public registry managed by Red Hat team
- Server-side image building
- Automatic scanning for known vulnerabilities
- Offers live upload by Image creator
- Fine-grained access controls with namespace

Private Registries

- Company privacy and secret protection.
- Legal restrictions and laws.
- Avoidance of publishing images in development.
- Total control about their image's placement, distribution and usage.

Configuring Registries in Podman

- Update /etc/containers/registries.conf

```
$vi /etc/containers/registries.conf

…output omitted…

[registries.search]

registries = [“registry.access.redhat.com”, “quay.io”, “docker.io”]


[registries.insecure]

registries = [“localhost:5000”]
```

Accessing Registries

- Searching for Images in Registries

```
$ sudo podman search [OPTIONS] <term>
```

Option	Description
--limit <number>	Limits the number of listed images per registry.
--filter <filter=value>	Filter output based on conditions provided. Supported filters are: • **stars=<number>**: Show only images with at least this number of stars. • **is-automated=<true|false>**: Show only images automatically built. • **is-official=<true|false>**: Show only images flagged as official.
--tls-verify <true|false>	Enables or disables HTTPS certificate validation for all used registries. `true`

- TERM refers to name of image, application, software and etc
  - Example: mysql, sql, apache, apache:2.4

Search for images using --limit

- Search for mysql images but limit to first 2 entries for each registries

```
[student@workstation ~]$ sudo podman search --limit 2 mysql
INDEX        NAME                                                DESCRIPTION
                                 STARS   OFFICIAL   AUTOMATED
redhat.com   registry.access.redhat.com/rhscl/mysql-57-rhel7   Docker image for
running MySQL 5.7 server. T...   0
redhat.com   registry.access.redhat.com/rhscl/mysql-56-rhel7   MySQL 5.6 SQL dat
abase server                     0
redhat.io    registry.redhat.io/rhscl/mysql-57-rhel7           Docker image for
running MySQL 5.7 server. T...   0
redhat.io    registry.redhat.io/rhscl/mysql-56-rhel7           MySQL 5.6 SQL dat
abase server                     0
docker.io    docker.io/library/mysql                           MySQL is a widely
 used, open-source relation...   11020   [OK]
docker.io    docker.io/mysql/mysql-server                      Optimized MySQL S
erver Docker images. Create...   819                 [OK]
```

Search for images using --filter

- Search for apache images with 1000 and above stars

```
[student@workstation ~]$ sudo podman search --filter stars=1000 apache
INDEX       NAME                                DESCRIPTION
            STARS   OFFICIAL   AUTOMATED
docker.io   docker.io/library/httpd             The Apache HTTP Server Project
            3544    [OK]
docker.io   docker.io/library/maven             Apache Maven is a software project man
agemen...   1216    [OK]
docker.io   docker.io/library/tomcat            Apache Tomcat is an open source implem
entati...   3050    [OK]
docker.io   docker.io/library/cassandra         Apache Cassandra is an open-source dis
tribut...   1280    [OK]
```

Search for images using multiple options

- Search for apache images with 1000 and above stars and also its must be TLS verified

```
[student@workstation ~]$ sudo podman search --filter stars=1000 --tls-verify=tru
e apache
INDEX       NAME                                DESCRIPTION
            STARS   OFFICIAL   AUTOMATED
docker.io   docker.io/library/httpd             The Apache HTTP Server Project
            3544    [OK]
docker.io   docker.io/library/maven             Apache Maven is a software project man
agemen...   1216    [OK]
docker.io   docker.io/library/tomcat            Apache Tomcat is an open source implem
entati...   3050    [OK]
docker.io   docker.io/library/cassandra         Apache Cassandra is an open-source dis
tribut...   1280    [OK]
```

Registry HTTP API

- Docker Registry HTTP API v2 specification – docker.io / Quay.io
- Interface to access or interact with remote registries
- Example: To list all repositories available in a registry

```
$ curl -Ls https://myserver/v2/_catalog?

{"repositories":["centos/httpd","do180/custom-httpd","hello-openshift"]}
```

- Example: To list all versions of specific image in a quay.io registry

```
$ curl -Ls https://quay.io/v2/jason.wong76/webserver/tags/list

{“name”:”jason.wong76/webserver”,”tags”:[“1.0”,”2.0”,”3.0”,”latest”,”4.0”]}


$ curl -Ls  https://quay.io/v2/redhattraining/httpd-parent/tags/list

…output omitted…
```

Registry HTTP API

- Make it more readable

```
$ curl -Ls https://quay.io/v2/redhattraining/httpd-parent/tags/list  | python3 -m json.tool
{
    "name": "redhattraining/httpd-parent",
    "tags": [
        "latest",
        "2.4"
    ]
}


$ curl -Ls https://quay.io/v2/jason.wong76/webserver/tags/list | python3 -m json.tool

…output omitted…
```

Registry Authentication

- Some registries require access authentication
- Use podman login -u <username> -p <password> <hostname to registry>

```
$ sudo podman login –u username –p password registry.access.redhat.com
```

- Or via API

```
$ curl -u username:password –Ls \
“https://sso.redhat.com/auth/realms/rhcc/protocol/redhat-docker-v2/auth?service=docker-registry”
```

NOTE

Other registries may require different steps to provide credentials. If a registry adheres to the `Docker Registry HTTP V2 API`, authentication conforms to the RFC7235 scheme.

Pulling Images

- The podman pull command syntax

```
$ sudo podman pull [OPTIONS] [REGISTRY[:PORT/]]NAME[:TAG]
```

- Pull an NGINX container from quay.io registry

```
$ sudo podman pull quay.io/bitnami/nginx
```

NOTE

If the image name does not include a registry name, Podman searches for a matching container image using the registries listed in the `/etc/containers/registries.conf` configuration file. Podman search for images in registries in the same order they appear in the configuration file.

Pulling Images with –creds option

- Pull an image from a private repository

```
$read -s password

<enter password to private repo>


$ sudo podman pull --creds jason.wong76:$password quay.io/jason.wong76/webserver:2.0
```

Listing Local Copies of Images

- Downloaded image is stored locally
- Avoids repeating download
- Minimize deployment time
- Custom images built
- Default location:
  - /var/lib/containers/storage/overlay-images
- List all images

```
$ sudo podman images

Or

$ sudo podman image list
```

Image Tags

- Support multiple versions or releases of same image
- Old and new version
- Evaluation and production copy
- Community copy
- Default to latest tag

- Example: Pull an image with specific version

```
$ sudo podman pull httpd:2.4
```

- Example: Pull an image with latest version from specific registry

```
$ sudo podman pull quay.io/redhattraining/httpd-parent:latest
```

Quiz 1

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

Which two commands display mysql images available for download from registry.redhat.io? (Choose two.)

a) podman search registry.redhat.io/mysql
b) podman images
c) podman pull mysql
d) podman search mysql

Quiz 1

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

Which two commands display mysql images available for download from registry.redhat.io? (Choose two.)

a) podman search registry.redhat.io/mysql
b) podman images
c) podman pull mysql
d) podman search mysql

Quiz 2

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

Which command is used to list all available image tags for the httpd container image?

a) podman search httpd
b) podman images httpd
c) podman pull --all-tags=true httpd
d) There is no podman command available to search for tags.

Quiz 2

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

Which command is used to list all available image tags for the httpd container image?

a) podman search httpd
b) podman images httpd
c) podman pull --all-tags=true httpd
d) There is no podman command available to search for tags.

Quiz 3

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

Which two commands pull the httpd image with the 2.2 tag? (Choose two.)

a) podman pull httpd:2.2
b) podman pull httpd:latest
c) podman pull quay.io/httpd
d) podman pull registry.redhat.io/httpd:2.2

Quiz 3

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

Which two commands pull the httpd image with the 2.2 tag? (Choose two.)

a) podman pull httpd:2.2
b) podman pull httpd:latest
c) podman pull quay.io/httpd
d) podman pull registry.redhat.io/httpd:2.2

Quiz 4

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

When running the following commands, which container images will be downloaded?

$ podman pull registry.redhat.io/httpd:2.2

$ podman pull quay.io/mysql:5.6

a) quay.io/httpd:2.2
registry.redhat.io/mysql:5.6

b) registry.redhat.io/httpd:2.2
registry.redhat.io/mysql:5.6

c) registry.redhat.io/httpd:2.2
No image will be downloaded for mysql.

d) quay.io/httpd:2.2
No image will be downloaded for mysql.

Quiz 4

Podman is available on a RHEL host with the following entry in `/etc/containers/registries.conf` file:

```
[registries.search]
registries = ["registry.redhat.io","quay.io"]
```

The `registry.redhat.io` and `quay.io` hosts have a registry running, both have valid certificates, and use the version 1 registry. The following images are available for each host:

Table 4.1. Image names/tags per registry

Regisitry	Image
registry.redhat.io	• nginx/1.0 • mysql/5.6 • httpd/2.2
quay.io	• mysql/5.5 • httpd/2.4

No images are locally available.

When running the following commands, which container images will be downloaded?

$ podman pull registry.redhat.io/httpd:2.2

$ podman pull quay.io/mysql:5.6

a) quay.io/httpd:2.2
registry.redhat.io/mysql:5.6

b) registry.redhat.io/httpd:2.2
registry.redhat.io/mysql:5.6

c) registry.redhat.io/httpd:2.2
No image will be downloaded for mysql.

d) quay.io/httpd:2.2
No image will be downloaded for mysql.

Manipulating Container Images

After completing this section, you will be:

Save and load
- container images to local files.

Delete images
- from the local storage.

Create new container images
- from containers and update image metadata.

Manage image tags
- for distribution purposes.

Saving and Loading Images

- Use podman save command. The syntax:

```
$ sudo podman save [-o FILE_NAME] IMAGE_NAME[:tag]
```

- Example: Backup following mysql image to mysql.tar

```
$ sudo podman save  -o mysql.tar registry.access.redhat.com/rhscl/mysql-57-rhel7:5.7
```

- Example: Restore the image

```
$  sudo podman load  -i mysql.tar
```

NOTE

To save disk space, compress the file generated by the **save** subcommand with Gzip using the `--compress` parameter. The **load** subcommand uses the **gunzip** command before importing the file to the local storage.

Delete image

- Use podman rmi command. The syntax:

```
$ sudo podman rmi [OPTIONS] IMAGE [IMAGE…]
```

- Delete operation fails if the image is being used
- Use --force or -f option to force
- Use --all or -a option to remove all locally stored images

```
$ sudo podman rmi -af
```

- Recommended workflow

```
$ sudo podman kill -s 9 my-container or sudo podman stop -f my-container
$ sudo podman rm -f my-container
$ sudo podman rmi -f container-image:1.0
```

Saving containers

- Latest configuration
- Use *podman commit* command: The syntax:

```
$ sudo podman commit [OPTIONS] container  [REGISTRY[:PORT]/IMAGE_NAME[:TAG]
```

- REGISTRY: Public or private registry url
- PORT: Registry's port. Default to 5000
- IMAGE_NAME: Container image name
- TAG: Version of image. Default to latest

Option	Description
`--author ""`	Identifies who created the container image.
`--message ""`	Includes a commit message to the registry.
`--format`	Selects the format of the image. Valid options are `oci` and `docker`.

Saving containers - EXAMPLE

To find the ID of a running container in Podman, run the **podman ps** command:

```
[student@workstation ~]$ sudo podman ps
CONTAINER ID IMAGE ...                  NAMES
87bdfcc7c656 mysql ...output omitted... mysql-basic
```

Eventually, administrators might customize the image and set the container to the desired state. To identify which files were changed, created, or deleted since the container was started, use the **diff** subcommand. This subcommand only requires the container name or container ID:

```
[student@workstation ~]$ sudo podman diff mysql-basic
C /run
C /run/mysqld
A /run/mysqld/mysqld.pid
A /run/mysqld/mysqld.sock
A /run/mysqld/mysqld.sock.lock
A /run/secrets
```

The **diff** subcommand tags any added file with an A, any changed ones with a C, and any deleted file with a D.

Saving containers - EXAMPLE

NOTE

The **diff** command only reports added, changed, or deleted files to the container file system. Files that are mounted to a running container are not considered part of the container file system.

To retrieve the list of mounted files and directories for a running container, use the **podman inspect** command:

```
[student@workstation ~]$ sudo podman inspect \
> -f "{{range .Mounts}}{{println .Destination}}{{end}}" CONTAINER_NAME/ID
```

Any file in this list, or file under a directory in this list, is not shown in the output of the **podman diff** command.

To commit the changes to another image, run the following command:

```
[student@workstation ~]$ sudo podman commit mysql-basic mysql-custom
```

Tagging Images

- A project with multiple different version of the same image
- Example: a container image to run with either MySQL or PostgreSQL database
- Use podman tag command. The syntax:

```
$ sudo podman tag [OPTIONS] IMAGE[:TAG] [REGISTRYHOST/][USERNAME/]IMAGE_NAME[:TAG]
```

- Example: Tag a local image named mysql-custom as devops/mysql

```
$ sudo podman tag mysql-custom quay.io/devops/mysql
```

- Example: Similar as above but use different tag name

```
$ sudo podman tag mysql-custom quay.io/devops/mysql:snapshot
```

List and Remove Tagged Images

- List tagged images

```
$ sudo podman images
```

- Remove the tagged image

```
$ sudo podman rmi devops/mysql:snapshot
```

- Best Practice for Tagging Images
  - Latest tag is used by default
  - Specify a version or functional purpose of the tagged image

Publishing Images to Registry

- Images must reside in local storage
- Tagged for identification purposes
- Use *podman push* command. The syntax:

```
$ sudo podman push [OPTIONS] IMAGE [DESTINATION]
```

- Example: Tag and push a local image to remote registry

```
$ sudo podman tag  httpd-parent:latest  my-httpd:snapshot

$ sudo podman login -u jason.wong76 -p $password quay.io

$ sudo podman push my-httpd:snapshot  quay.io/jason.wong76/my-httpd:snapshot
```

Guided Exercise: Creating a Custom Apache Container Image

You should be able to:

- Create a custom Apache container image
- Tag an image
- Push the tagged image to Quay.io registry

Lab: Managing Images

You should be able to:

- Download image
- Log into container and customize it
- Commit to new container image
- Tag and push to quay.io registry
- Start new container from the pushed image

Chapter Summary

In this chapter, you learned:

The Red Hat Container Catalog provides tested and certified images at registry.redhat.io.

Podman provides commands to manage container images both in local storage and as compressed files.

Podman can interact with remote container registries to search, pull, and push container images.

Use the podman commit to create an image from a container.

Image tags are a mechanism to support multiple releases of a container image.