

Lab: Troubleshooting Containers

Troubleshoot the quotes-api application.

The application consists of the following components:

- The quotes-api-v1 service with the v1 quotes API version.
- The quotes-api-v2 service with the v2 quotes API version.
- The quotes-ui service, which accepts the quotes API version as an environment variable.

The quotes-ui container starts an NGINX proxy that performs the following tasks:

- Serves the web application.
- Acts as a reverse proxy to route the UI requests to the version specified in the QUOTES_API_VERSION environment variable.

Outcomes

You should be able to:

- Check container logs.
- Run container commands.
- Troubleshoot container networking problems.
- Configure containers by using environment variables.
- Configure containers by overriding container configuration files with host files.

As the student user on the workstation machine, use the lab command to prepare your system for this exercise.

This command deploys the quotes application in a non-working state. You must troubleshoot and fix the application.

```
[student@workstation ~]$ lab start troubleshooting-lab
```

The lab script continuously evaluates the objectives of this lab. Keep the script running in a terminal window and complete the objectives of this lab from a new terminal window.

Instructions

1. Verify that the following containers are running with the following configuration:

quotes-api-v1 and quotes-api-v2

- Container image: registry.ocp4.example.com:8443/redhattraining/wiremock

quotes-ui

- Port mapping: 3000:8080
- Container image: registry.ocp4.example.com:8443/redhattraining/quotes-ui-versioning:1.0
- Environment variable: QUOTES_API_VERSION=v2

If any of the containers are missing, then read the container logs to determine what is preventing the container from starting. Then, fix the issue and start the container.

Use the following troubleshooting guide to identify and fix the problems:

```
Problem: unable to retrieve auth token: invalid username/password: unauthorized
Solution: You must log into the image registry.
```

```
Problem: requires at least 1 arg(s), only received 0
Solution: Correct your Podman command.
```

```
Problem: nginx: [emerg] host not found in upstream
Solution: Nginx reverse proxy cannot resolve hostname to IP address, probably due to a missing Podman network.
```

Run the podman ps command with the -a option to show stopped containers.

```
[student@workstation ~]$ podman ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
b72652504844 ...wiremock:latest 28 sec... Up... quotes-api-v1
a3a2f6809446 ...wiremock:latest 27 sec... Up... quotes-api-v2
867f6f559629 ...quotes-ui-ve... /bin... 21 sec... Exited... 0... quotes-ui
```

The quotes-ui container failed to start.

Read the quotes-ui logs with the podman logs command to look for container start-up errors.

```
[student@workstation ~]$ podman logs quotes-ui
nginx: [emerg] host not found in upstream "quotes-api-v1" in /etc/nginx/nginx.conf:45
```

The container fails to start because the NGINX proxy in the quotes-ui container cannot resolve the quotes-api-v1 hostname to an IP address.

Use the podman inspect command to determine the Podman networks that the containers use to communicate.

Inspect the quotes-api-v1 networks.

```
[student@workstation ~]$ podman inspect \
  quotes-api-v1 --format='{{.NetworkSettings.Networks}}'
map[ttroubleshooting-lab:0xc000a825a0]
```

Inspect the quotes-api-v2 networks.

```
[student@workstation ~]$ podman inspect \
  quotes-api-v2 --format='{{.NetworkSettings.Networks}}'
map[ttroubleshooting-lab:0xc000a825a0]
```

Repeat the command for the quotes-ui container.

```
[student@workstation ~]$ podman inspect \
  quotes-ui --format='{{.NetworkSettings.Networks}}'
map[pasta:0xc0006cd200]
```

The quotes-ui container is missing the troubleshooting-lab network and thus it cannot resolve the quotes-api-v1 and quotes-api-v2 hostnames.

Remove the current quotes-ui container.

```
[student@workstation ~]$ podman rm quotes-ui
quotes-ui
```

Recreate the quotes-ui container and attach it to the troubleshooting-lab network.

```
[student@workstation ~]$ podman run -d \
  --name quotes-ui -p 3000:8080 \
  -e QUOTES_API_VERSION=v2 \
  --net troubleshooting-lab \
  registry.ocp4.example.com:8443/redhat-training/quotes-ui-versioning:1.0
d838...7432
```

Verify that the quotes-ui container starts.

```
[student@workstation ~]$ podman ps
CONTAINER ID  IMAGE          COMMAND CREATED     STATUS    PORTS NAMES
b72652504844  ...wiremock:latest   28 sec...  Up...      quotes-api-v1
a3a2f6809446  ...wiremock:latest   27 sec...  Up...      quotes-api-v2
6e8581a9d73b  ...quotes-ui-versioning:1.0 1 sec...  Up...      quotes-ui
```

2. Ensure that the NGINX reverse proxy can route requests to both versions of the quotes API.

The NGINX reverse proxy in the quotes-ui container should route requests as indicated in the following table:

Table 6.1. NGINX mappings

From	To
http://localhost:8080/api/v1/quotes	http://quotes-api-v1:8080/api/v1/quotes
http://localhost:8080/api/v2/quotes	http://quotes-api-v2:8081/api/v2/quotes

The quotes-ui logs provide some network information that you might need.

Perform any configuration modifications in the quotes-ui container.

Use a bind mount to replace the /etc/nginx/nginx.conf file in the quotes-ui container with a modified /home/student/D0188/labs/troubleshooting-lab/nginx.conf file, if necessary.

Use the podman exec command to run the curl command from the quotes-ui container to test the v1 and v2 mappings. Use the localhost hostname to test the NGINX mappings. Use the silent option (-s) with the curl command to print the output only.

Test that the v1 mapping works.

```
[student@workstation ~]$ podman exec quotes-ui \
curl -s http://localhost:8080/api/v1/quotes
[
{
  "id": 1,
  "quote": "I live my daydreams in music.",
  "author": "Albert Einstein"
},
...output omitted...
```

Test that the v2 mapping does not work.

```
[student@workstation ~]$ podman exec quotes-ui \
curl -s http://localhost:8080/api/v2/quotes
<html>
<head><title>502 Bad Gateway</title></head>
<body>
<center><h1>502 Bad Gateway</h1></center>
<hr><center>nginx/1.20.1</center>
</body>
</html>
```

This error means that NGINX was able to map the request but did not get a response from the quotes-api-v2 service.

Investigate if the quotes-api-v2 service is using a different port. Run the podman logs command on the quotes-api-v2 container to find the port that the service is using.

```
[student@workstation ~]$ podman logs quotes-api-v2
...output omitted...
port:          8081
...output omitted...
```

The quotes-api-v2 service is using the 8081 port.

Read the quotes-ui container NGINX configuration file at the /etc/nginx/nginx.conf path.

```
[student@workstation ~]$ podman exec quotes-ui \
cat /etc/nginx/nginx.conf
...output omitted...
location /api/v2 {
  rewrite /api/v2/(.*) /$1 break;
  proxy_pass http://quotes-api-v2:8080;
}
...output omitted...
```

The mapping for v2 is redirecting requests to the http://quotes-api-v2:8080 address, which points to the wrong port.

Read the updated copy of the NGINX configuration at /home/student/D0188/labs/troubleshooting-lab/nginx.conf.

```
[student@workstation ~]$ cat ~/D0188/labs/troubleshooting-lab/nginx.conf
...output omitted...
location /api/v2 {
  rewrite /api/v2/(.*) /$1 break;
  proxy_pass http://quotes-api-v2:8081;
}
...output omitted...
```

This version points to the right 8081 port.

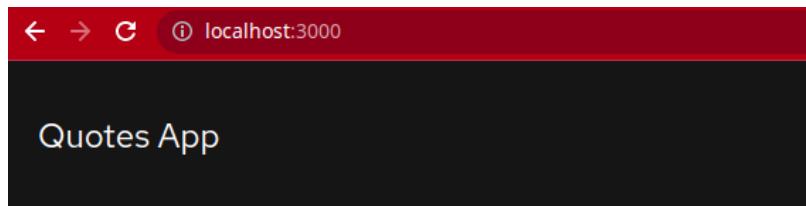
Remove the quotes-ui container by running podman rm using the -f option.

```
[student@workstation ~]$ podman rm -f quotes-ui
```

Recreate the quotes-ui container. Override the container's /etc/nginx/nginx.conf file with the /home/student/DO188/labs/troubleshooting-lab/nginx.conf file by using a bind mount with the :z option.

```
[student@workstation ~]$ podman run -d \
--name quotes-ui \
-p 3000:8080 \
-e QUOTES_API_VERSION=v2 \
--net troubleshooting-lab \
-v ~/DO188/labs/troubleshooting-lab/nginx.conf:/etc/nginx/nginx.conf:z \
registry.ocp4.example.com:8443/redhattraining/quotes-ui-versioning:1.0
a0d1...a73f
```

3. Confirm that the application is accessible by using a web browser to go to <http://localhost:3000>.



I live my daydreams in music.

- Albert Einstein

Finish

As the student user on the workstation machine, use the lab command to complete this exercise. This is important to ensure that resources from previous exercises do not impact upcoming exercises.

Press **y** when the lab start command prompts you to execute the finish function. Alternatively, execute the following command:

```
[student@workstation ~]$ lab finish troubleshooting-lab
```