



### Exercise 29.1 Install and test **Apache (httpd)** as a **Docker** application.

In this exercise, we will install, run and test the **docker** package, and follow with getting and deploying **httpd**, the **Apache** web server container.

**Docker** requires a **Linux** kernel of 3.10 or greater, which is not a problem for all recent **Linux** distributions. (**RedHat/CentOS 7** has a 3.10-based kernel, the others are all more recent.)

A 64-bit system is **required** for this exercise. Do not attempt on a 32-bit system.

1. Make sure **Docker** is installed. Pick the right command for your distribution from the below:

```
$ sudo yum install docker
$ sudo zypper install docker
$ sudo apt-get install docker.io
```

- If you get strange errors at later points in the exercise you might find it useful to **reinstall docker**. We have observed cases (for example, with **RHEL 7**) where **docker** configurations were broken, after a system upgrade,

2. Start the **docker** service.

```
$ sudo systemctl start docker
```

You may want to verify that it is running properly with `systemctl status docker`:

```
File Edit View Search Terminal Help
c7:/tmp>sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
   Active: active (running) since Fri 2017-10-06 07:30:24 CDT; 3h 14min ago
     Docs: http://docs.docker.com
    Main PID: 6185 (dockerd-current)
    CGroup: /system.slice/docker.service
            └─6185 /usr/bin/dockerd-current --add-runtime docker-runc=/usr/libexec/docker/docker-runc-current --default-runtime=docker-runc --authoriz...
              └─6212 /usr/bin/docker-containerd-current -l unix:///var/run/docker/libcontainerd/docker-containerd.sock --shim docker-containerd-shim --m...

Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.400523213-05:00" level=warning msg="Your kernel does not support cgroup cfs period"
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.400531993-05:00" level=warning msg="Your kernel does not support cgroup cfs quotas"
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.400778902-05:00" level=info msg="Loading containers: start."
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.406788047-05:00" level=info msg="Firewalld running: true"
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.740036679-05:00" level=info msg="Default bridge (docker0) is assigned with IP address 172.17.0.1"
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.873256177-05:00" level=info msg="Loading containers: done."
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.873424546-05:00" level=info msg="Daemon has completed initialization"
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.873435907-05:00" level=info msg="Docker daemon" commit="c4618fb/1.12.6" golang.version="1.12.6"
Oct 06 07:30:24 c7 dockerd-current[6185]: time="2017-10-06T07:30:24.876818269-05:00" level=info msg="API listen on /var/run/docker.sock"
Oct 06 07:49:02 c7 dockerd-current[6185]: time="2017-10-06T07:49:02.019628490-05:00" level=info msg="{Action=ping, Username=root, LoginUID=0...D=8273}"
Hint: Some lines were ellipsized, use -l to show in full.
c7:/tmp>
```

Figure 29.1: Checking docker status

If you see anything indicating failure you should inspect `/var/log/messages` or whatever other logging file you have on your system for clues. If you are running a standard distribution kernel you should be fine, but if you are running a custom **Linux** kernel, it is likely you have to select the proper configuration options, especially as regards to networking. This is too complicated to go into here, so please stay with a distribution supplied kernel unless you want a challenging exercise!

3. Search for the **httpd** container, with `docker search apache`:

File	Edit	View	Search	Terminal	Help
c7:/tmp>sudo docker search apache					
INDEX	NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
docker.io	docker.io/tomcat	Apache Tomcat is an open source implementa...	1510	[OK]	
docker.io	docker.io/httpd	The Apache HTTP Server Project	1274	[OK]	
docker.io	docker.io/cassandra	Apache Cassandra is an open-source distrib...	652	[OK]	
docker.io	docker.io/maven	Apache Maven is a software project managem...	474	[OK]	
docker.io	docker.io/solr	Solr is the popular, blazing-fast, open so...	438	[OK]	
docker.io	docker.io/eboraas/apache-php	PHP5 on Apache (with SSL support), built o...	132		[OK]
docker.io	docker.io/eboraas/apache	Apache (with SSL support), built on Debian	85		[OK]
docker.io	docker.io/webdevops/php-apache	Apache with PHP-FPM (based on webdevops/php)	43		[OK]
docker.io	docker.io/solr	Apache TomEE is an all-Apache Java EE cert...	40	[OK]	
docker.io	docker.io/nimmis/apache-php5	This is docker images of Ubuntu 14.04 LTS ...	36		[OK]
docker.io	docker.io/fedora/apache		33		[OK]
docker.io	docker.io/bitnami/apache	Bitnami Apache Docker Image	32		[OK]
docker.io	docker.io/webdevops/php-apache-dev	PHP with Apache for Development (eg. with ...	27		[OK]
docker.io	docker.io/apacheignite/ignite	Apache Ignite In-Memory docker image	23		[OK]
docker.io	docker.io/linuxserver/apache	An Apache container, brought to you by Lin...	14		
docker.io	docker.io/apache/nutch	Apache Nutch	13		[OK]
docker.io	docker.io/chriswayg/apache-php	Apache 2.4 web server with PHP 5.6 based o...	11		[OK]
docker.io	docker.io/coreos/apache	Basic install of Apache for use in example...	11		
docker.io	docker.io/webdevops/apache	Apache container	10		[OK]
docker.io	docker.io/antage/apache2-php5	Docker image for running Apache 2.x with P...	4		[OK]
docker.io	docker.io/lephare/apache	Apache container	3		[OK]
docker.io	docker.io/landinternet/ubuntu-16-apache	ubuntu-16-apache	2		[OK]
docker.io	docker.io/landinternet/ubuntu-16-apache-php-7.0	ubuntu-16-apache-php-7.0	1		[OK]
docker.io	docker.io/mastertinner/apache-directory-index-resource	A Concourse resource for the apache direct...	1		[OK]
docker.io	docker.io/landinternet/ubuntu-16-apache-php-5.6	ubuntu-16-apache-php-5.6	0		[OK]
redhat.com	registry.access.redhat.com/rhsccl/httpd-24-rhel7	Apache HTTP 2.4 Server	0		
redhat.com	registry.access.redhat.com/rhsccl/s2i-core-rhel7	The s2i core container image serves as a b...	0		

Figure 29.2: Using docker search

(You could have used **httpd** instead of **apache** in the above command with very similar results.)

From now on we will not show detailed output since if you have gotten this far, things should be fine.

- Retrieve the container:

```
$ sudo docker pull docker.io/httpd
```

This may take a couple of minutes while all the components download.

- List the installed containers:

```
$ sudo docker images
```

- List the components associated with the images.

```
$ sudo docker images --all
```

- Start the **httpd docker** container. The terminal will appear to hang as it is now connected to the **httpd** daemon.

```
c7:/tmp>sudo docker run httpd
```

```
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, \
using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
.....
```

- You can open a graphical web browser pointing to the IP address in the above output.

Or you can use a text-based browser if you are not in a graphical environment, by opening up a new terminal window (do not kill the one in which the **docker httpd** container is running) by doing one of the following commands:

```
$ lynx http://172.17.0.2
$ w3m http://172.17.0.2
$ elinks http://172.17.0.2
```

using whichever graphical browser is installed on your system.

- Stop the container and **docker** service and clean up.

```
c7:/tmp>sudo docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b936b0afeb23	httpd	"httpd-foreground"	41 seconds ago	Up 40 seconds	80/tcp	boring_turing

```
c7:/tmp>sudo docker stop b936b0afeb23
b936b0afeb23
```

```
c7:/tmp>sudo docker rmi -f docker.io/httpd
Untagged: docker.io/httpd:latest
Untagged: docker.io/httpd@sha256:cf774f082e92e582d02acdb76dc84e61dcf5394a90f99119d1ae39bcecbff075
Deleted: sha256:cf6b6d2e846326d2e49e12961ee0f63d8b5386980b5d3a11b8283151602fa756
```

```
c7:/tmp>sudo systemctl stop docker
```