The variety of environments and use cases in which Docker can be used means that there are a variety of storage needs. Container storage can be implemented in multiple ways through the use of various storage drivers, and those provide a pluggable framework for using different kinds of container storage. In this lesson, we will discuss what storage drivers are and identify the most widely-used ones. We will also demonstrate how to explicitly set the storage driver to use for a system.

Relevant Documentation

- https://docs.docker.com/storage/storagedriver/select-storage-driver/

Lesson Reference

This lesson was performed on a CentOS 7 server running Docker CE

Get the current storage driver:

```
docker info
```

Set the storage driver explicitly by providing a flag to the Docker daemon:

```
sudo vi /usr/lib/systemd/system/docker.service
```

Edit the `ExecStart` line, adding the `--storage-driver devicemapper` flag:

```
ExecStart=/usr/bin/dockerd --storage-driver devicemapper ...
```

After any edits to the unit file, reload Systemd and restart Docker:

```
sudo systemctl daemon-reload
sudo systemctl restart docker
```

We can also set the storage driver explicitly using the daemon configuration file. This is the method that Docker recommends. Note that we cannot do this and pass the `--storage-driver` flag to the daemon at the same time:

```
sudo vi /etc/docker/daemon.json
```

Set the storage driver in the daemon configuration file:

```
{
  "storage-driver": "devicemapper"
}
```

Restart Docker after editing the file. It is also a good idea to make sure Docker is running properly after changing the configuration file:

```
sudo systemctl restart docker
sudo systemctl status docker
```