class-vmware-docker-2018-03-12

vmware advanced docker class page

What plugins are running

Run `docker info` to see what plugins are running.

Events and API

Open up two ssh connections to a lab machine.

In one window run:

```
docker events
```

Now in the other window run these commands against the docker engine API:

Pull an image

```
curl --unix-socket /var/run/docker.sock \
  -X POST "http:/v1.24/images/create?fromImage=alpine" | jq
```

Run a container

Note that unlike docker run, this api command does not automatically pull if the image is not found locally.

```
curl --unix-socket /var/run/docker.sock -H "Content-Type: application/json" \
  -d '{"Image": "alpine", "Cmd": ["echo", "hello world"]}' \
  -X POST http:/v1.24/containers/create | jq
```

Run a container in the background

```
curl --unix-socket /var/run/docker.sock -H "Content-Type: application/json" \
  -d '{"Image": "bfirsh/reticulate-splines"}' \
  -X POST http:/v1.24/containers/create | jq
```

List the containers

```
curl --unix-socket /var/run/docker.sock http:/v1.24/containers/json | jq
```

Stop a running containers

Be sure to use a container id from a previous step:

```
curl --unix-socket /var/run/docker.sock \
  -X POST http:/v1.24/containers/d3b83a42bb4f/stop
```

Print the container logs

```
#Run a hello world
docker run alpine echo hello world

#Store that container's ID in a variable:
CONTAINER_ID="$(docker ps -lq)"

#Now read that container's logs with the container id:
curl --unix-socket /var/run/docker.sock "http:/v1.24/containers/$CONTAINER_ID/logs?s
```

List all images

```
curl --unix-socket /var/run/docker.sock http:/v1.24/images/json | jq
```

Commit a container

```
docker run -d alpine touch /helloworld

#Store that container's ID in a variable:
CONTAINER_ID="$(docker ps -lq)"
```

```
curl --unix-socket /var/run/docker.sock\
    -X POST "http:/v1.24/commit?container=$CONTAINER_ID&repo=helloworld"
```

class-vmware-docker-2018-03-12 is maintained by **chrishiestand**.

This page was generated by GitHub Pages.