

Twitter San Francisco, CA

2013-11-05



docker

Dockerizing your applications

Daniel Mizyrycki
daniel@docker.com

Introduction

Dockerizing a typical desktop application



The image shows a screenshot of the Firefox homepage. At the top right, there is a "mozilla" logo with a dropdown arrow. Below it, the Firefox logo (a stylized orange and blue fox) is positioned to the left of the text "Welcome to Firefox!". Underneath this, the text "See how Firefox keeps the power of the web in your hands:" is displayed. A large, central image shows a close-up of a hand-drawn circle on a textured, light-colored surface. At the bottom of the page, there is a light gray box containing the text "Get Firefox news" followed by a text input field with the placeholder "YOUR EMAIL HERE" and a blue button labeled "Sign me up »". Below this box, there are several links: "Release Notes »", "Firefox Features »", "Firefox Help »", "Privacy Policy", and "Legal Notices".

Downloads

* Download Docker

```
wget -O docker http://get.docker.io/builds/Linux/x86_64/docker-latest
```

* Download data and firefox Dockerfiles from contrib/desktop-integration

```
wget http://raw.githubusercontent.com/dotcloud/docker/master/  
contrib/desktop-integration/data/Dockerfile
```

```
wget http://raw.githubusercontent.com/dotcloud/docker/master/  
contrib/desktop-integration/firefox/Dockerfile
```

Building images with a Dockerfile

- * base image
- * load new packages
- * optional customization
- * optional addition of host files
- * optional run command

Data Dockerfile (Example of customization)

```
# Smallest base image, just to launch a container  
from busybox  
maintainer Daniel Mizyrycki <daniel@docker.com>  
  
# Create a regular user  
run echo 'sysadmin:x:1000:1000:./data:/bin/sh' >> /etc/passwd  
run echo 'sysadmin:x:1000:' >> /etc/group  
  
# Create directory for that user  
run mkdir /data  
run chown sysadmin.sysadmin /data  
  
# Add content to /data. This will keep sysadmin ownership  
run touch /data/init_volume  
  
# Create /data volume  
VOLUME /data
```

Docker and the desktop:

How to communicate with the desktop?

Video:

X11 unix socket. `/tmp/.X11-unix/X0`

Audio:

Alsa. `/dev/snd` plus access to the physical devices

Launch Docker

```
docker -d &
```

Building containers

```
cd data; docker build -t data -rm .  
cd firefox; docker build -t firefox -rm .
```

Running ephemeral Firefox container

```
docker run -e DISPLAY=unix$DISPLAY \  
-v /tmp/.X11-unix:/tmp/.X11-unix \  
-v /dev/snd:/dev/snd \  
-lxc-conf='lxc.cgroup.devices.allow = c 116:* rwm' \  
firefox
```


Running stateful data-on-host Firefox container

```
docker run -e DISPLAY=unix$DISPLAY \  
-v /data/firefox:/data \  
-v /tmp/.X11-unix:/tmp/.X11-unix \  
-v /dev/snd:/dev/snd \  
-lxc-conf='lxc.cgroup.devices.allow = c 116:* rwm' \  
firefox
```

Create dockerized data container

```
docker run -name firefox-data data true
```

Monitor status in the data container

```
while true; do  
  docker run -volumes-from firefox-data busybox ls -al /data  
  sleep 3; clear; done
```

Running stateful firefox with dockerized data container

```
docker run -e DISPLAY=unix$DISPLAY \  
  -volumes-from firefox-data \  
  -v /tmp/.X11-unix:/tmp/.X11-unix \  
  -v /dev/snd:/dev/snd \  
  -lxc-conf='lxc.cgroup.devices.allow = c 116:* rwm' \  
  firefox
```

