

Docker tips and tricks for developers

Piotr Zalewski

6-01-2020

Docker tips and tricks for developers

What is this presentation about ?

Not about production, not about deploying

We will talk about using developer tools packed in docker images.

Why use docker on your workstation ?

Isolation



Why use docker on your workstation ?

- ▶ Isolation
- ▶ Experience with docker

Using ready images of tools

1 - 25 of 2,974,639 available images.

Most Popular ▾

Tool name	count
mysql	18.163
postgresql	4.137
java	21.419
consul	2.989

Postgresql

```
docker run --name pg9.6 -e POSTGRES_PASSWORD=mysecret  
-d postgres:9.6  
docker run --name pg11 -e POSTGRES_PASSWORD=mysecret  
-d postgres:11
```

```
zalewski@pl1lx1-108704 ~ docker run -it --rm --network container:pg11 postgres:11 psql -h localhost -U postgres  
psql (11.6 (Debian 11.6-1.pgdg90+1))  
Type "help" for help.  
  
postgres=# select 1;  
?column?  
-----  
1  
(1 row)
```

VSCode

```
docker run -it -p 8080:8080
-v "$(pwd):/home/coder/project"
codercom/code-server --auth none
```

The screenshot shows the Visual Studio Code interface with a Python test file open in the editor and a browser window displaying the test results.

VSCode Explorer:

- OPEN EDITORS: test_env_syntax.py
- PROJECT:
 - .aws
 - .az
 - .azure
 - .terraform.d
 - component
 - env
 - env-definitions
 - infra-config
 - service-config
 - tests
 - test_env_syntax.py
 - test_tf_syntax.py
 - .gitignore
 - Jenkinsfile
 - Jenkinsfile_run
 - Jenkinsfile_run_azure
- OUTLINE

Code Editor:

```
tests > test_env_syntax.py
1 import os
2 import unittest
3
4 import yaml
5
6 ENV_DIR = 'env'
7
8 class EnvironmentSyntaxTest(unittest.TestCase):
9     def test_unique(self):
10         env_names = set()
11         for root, dirs, files in os.walk(ENV_DIR):
12             for file_name in files:
13                 if file_name.endswith('.yaml'):
14                     if file_name in env_names:
15                         self.fail('Environment name {} is not unique'.format(file_name))
16                     env_names.add(file_name)
17
18     def test_dependencies(self):
19         for root, dirs, files in os.walk(ENV_DIR):
20             for file_name in files:
21                 if file_name.endswith('.yaml'):
22                     print('Validating', file_name)
23                     with open(os.path.join(root, file_name)) as file:
24                         self.check(yaml.safe_load(file))
25
```

Browser Tab:

- Address bar: 127.0.0.1:8888/?folder=/home/coder/project
- Bookmarks: Apps, HAD, frogfactory, Bookmarks, Free Learning..., AWS, Azure, Jenkins, Other bookmarks

Passing secrets to apps

```
docker run -it --rm  
  -e AWS_SECRET_KEY = XXXSSDSXSXXXX  
  -e AWS_ACCESS_KEY = XSewedfwewee  
  pzalews/awscli  
  aws rds describe-db-instances --output table  
  
docker run -it --rm -v ~/.aws:/aws/.aws:ro  
  pzalews/awscli  
  aws rds describe-db-instances --output table  
  
alias awsd='docker run -it --rm -v "$(pwd)"/aws"  
  -v ~/.aws:/aws/.aws:ro -e AWS_PROFILE=had-dev  
  pzalews/awscli aws'
```

```
zalewski@plilx1-108704 ~ alias awsd='docker run -it --rm -v "$(pwd)"/aws" -v ~/.aws:/aws/.aws:ro -e AWS_PROFILE=had-dev pzalews/awscli aws'  
zalewski@plilx1-108704 ~ awsd s3 ls  
2018-05-09 06:59:24 367040365964-audit-logs  
2019-12-13 12:46:25 ad-installation  
2019-10-08 07:05:23 addo-test-logs  
2019-10-16 06:30:00 aggregate-scope-coverage  
2019-10-16 06:30:16 aggregate-scope-coverage-tableau
```

Network mode host

```
alias consul='docker run -d --rm  
-e CONSUL_BIND_INTERFACE=enxa44cc8d6bc80  
--net=host  
consul'
```

Debug network interfaces

```
docker run --name ubuntu -it ubuntu /bin/bash
docker run --net=container:ubuntu
-v $(pwd)/temp:/dump
corfr/tcpdump -s 65535 -w /dump/test.dump
```

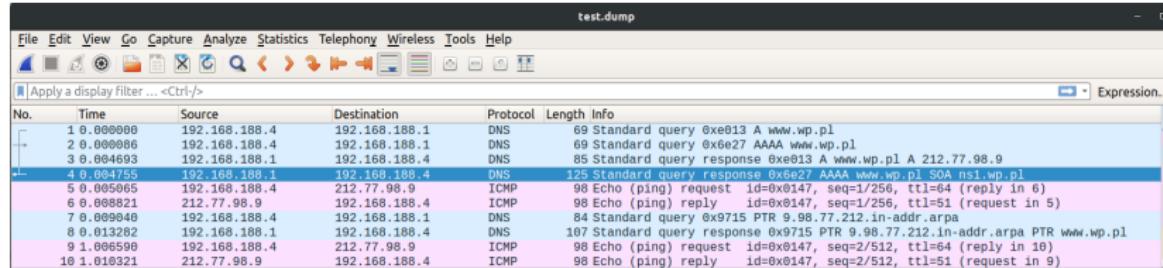


Figure 1: wireshark

Kali Linux

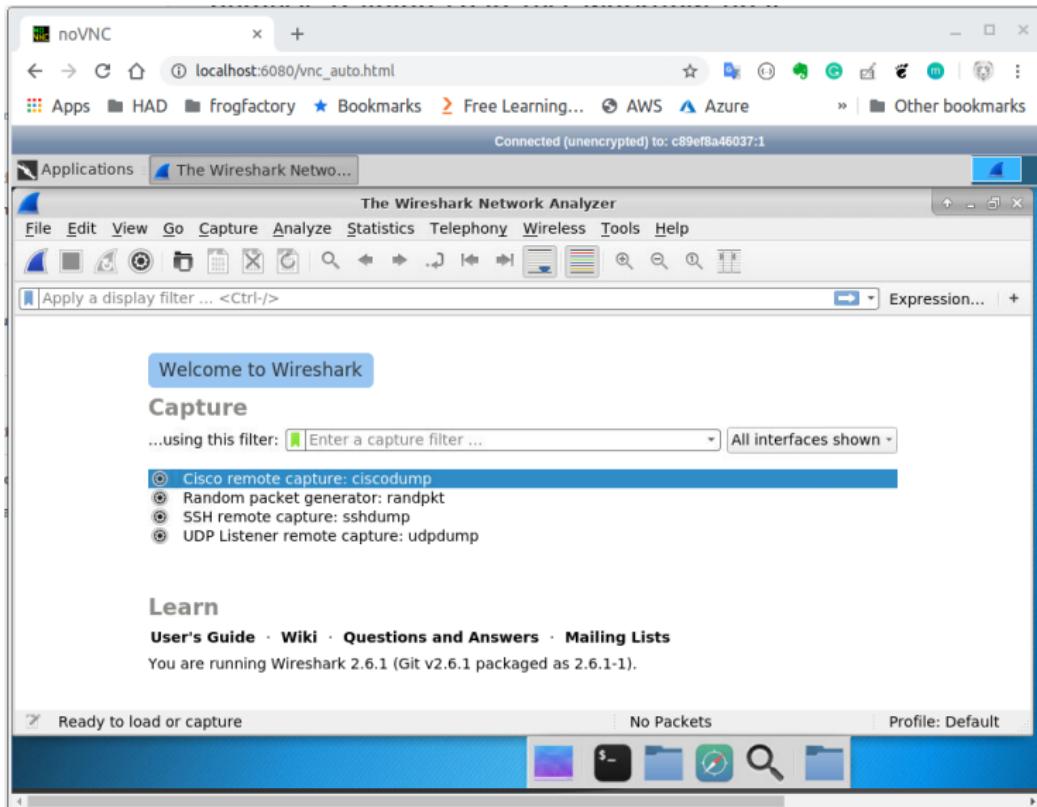


Figure 2: kalilinux

Dockerizing your tools

Dockerfile

```
zalewski@pl1lx1-108704:~/PIOTRZ/aztools$ pcat Dockerfile
FROM mcr.microsoft.com/azure-cli:2.0.76

ENV YQ_BIN_VERSION 2.2.0
ENV TERRAFORM_VERSION 0.12.13
ENV KUBECTL_VERSION 1.13.10

ENTRYPOINT ["/entrypoint.sh"]
COPY entrypoint.sh entrypoint.sh
COPY commands.sh /commands.sh

VOLUME /work
WORKDIR /work
ENV HOME /work

RUN curl --silent --location --output /usr/local/bin/yq https://github.com/mikefarah/yq/releases/download/${YQ_BIN_VERSION}/yq_linux_amd64 && \
    chmod 755 /usr/local/bin/yq && \
    chmod +x /entrypoint.sh && \
    chmod +x /commands.sh

RUN curl --silent -o terraform.zip https://releases.hashicorp.com/terraform/${TERRAFORM_VERSION}/terraform_${TERRAFORM_VERSION}_linux_amd64.zip && \
    unzip terraform.zip && \
    chmod +x terraform && \
    mv terraform /usr/bin && \
    rm terraform.zip

RUN curl --silent -LO https://storage.googleapis.com/kubernetes-release/release/v${KUBECTL_VERSION}/bin/linux/amd64/kubectl && \
    mv kubectl /usr/bin/kubectl && \
    chmod +x /usr/bin/kubectl
zalewski@pl1lx1-108704:~/PIOTRZ/aztools$ 
```

CMD and ENTRYPOINT

ADD vs COPY

Docker buildkit

`export DOCKER_BUILDKIT=1` (starting from Docker 18.09)

- ▶ new tty output
- ▶ better caching
- ▶ mounting some folders for build time
- ▶ adding secret informations for build time

Docker buildkit

```
Sending build context to Docker daemon 81.92kB
Step 1/13 : FROM mcr.microsoft.com/azure-cli:2.0.76
--> e1bb0df63df8
Step 2/13 : ENV YQ_BIN_VERSION 2.2.0
--> Running in daa48ba1199e
Removing intermediate container daa48ba1199e
--> fdef7d5d73b4
Step 3/13 : ENV TERRAFORM_VERSION 0.12.14
--> Running in bb31218f61c4
Removing intermediate container bb31218f61c4
--> 5342f3212306
Step 4/13 : ENV KUBECTL_VERSION 1.13.10
--> Running in c1fce1466ec2
Removing intermediate container c1fce1466ec2
--> 0e304ebc3703
Step 5/13 : ENTRYPOINT ["/entrypoint.sh"]
--> Running in 594f8d56c608
Removing intermediate container 594f8d56c608
--> 91aa980e1ec1
Step 6/13 : COPY entrypoint.sh entrypoint.sh
--> 2220b77248d2
Step 7/13 : COPY commands.sh /commands.sh
--> b0be0ea2b5ae
Step 8/13 : VOLUME /work
--> Running in 0e0516146ab8
Removing intermediate container 0e0516146ab8
--> eb1fc0b078f1
Step 9/13 : WORKDIR /work
--> Running in e3b070b66e35
Removing intermediate container e3b070b66e35
--> 0213665db226
Step 10/13 : ENV HOME /work
--> Running in 50a939058f06
Removing intermediate container 50a939058f06
--> a2d390bb483f
Step 11/13 : RUN curl --silent --location --output /usr/local/bin/yq \
  chmod +x /entrypoint.sh &&     chmod +x /commands.sh
--> Running in 8c5ac5b2e492
Removing intermediate container 8c5ac5b2e492
--> 39eb58dbea42
Step 12/13 : RUN curl --silent -o terraform.zip https://releases.hashicorp.com/terraform/0.14.0/terraform_0.14.0_linux_amd64.zip \
  chmod +x terraform &&     mv terraform /usr/bin &&     rm terraform.zip
```

Docker buildkit

```
zalewski@pl1lxl-108704 ~/PIOTRZ/aztools master ➜ docker build .
[+] Building 0.2s (12/12) FINISHED
=> [internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 38B                          0.0s
=> [internal] load .dockerignore                           0.0s
=> => transferring context: 28                            0.0s
=> [internal] load metadata for mcr.microsoft.com/azure-cli:2.0.76 0.0s
=> [1/7] FROM mcr.microsoft.com/azure-cli:2.0.76           0.0s
=> [internal] load build context                         0.0s
=> => transferring context: 648                          0.0s
=> CACHED [2/7] COPY entrypoint.sh entrypoint.sh        0.0s
=> CACHED [3/7] COPY commands.sh /commands.sh          0.0s
=> CACHED [4/7] WORKDIR /work                           0.0s
=> CACHED [5/7] RUN curl --silent --output /usr/local/bin/yq https://github.com/mikefarah/yq/releases/latest/download/yq      0.0s
=> CACHED [6/7] RUN curl --silent -o terraform.zip https://releases.hashicorp.com/terraform/0.12.13/terraform_0.12.13_linux_amd64.zip 0.0s
=> CACHED [7/7] RUN curl --silent -L https://storage.googleapis.com/kubernetes-release/release/v1.13.10/bin/linux/amd64/kubectl      0.0s
=> exporting to image                                     0.0s
=> => exporting layers                                    0.0s
=> => writing image sha256:cb879425691ea1ca7a142202b03c36fab09ff5c1ef01f5fc732f1c5ee5e07915 0.0s
zalewski@pl1lxl-108704 ~/PIOTRZ/aztools master ➜
```

Dockerfile - layers

```
zalewski@plilxl-108704:~/PIOTRZ/aztools$ pcat Dockerfile
FROM mcr.microsoft.com/azure-cli:2.0.76

ENV YQ_BIN_VERSION 2.2.0
ENV TERRAFORM_VERSION 0.12.13
ENV KUBECTL_VERSION 1.13.10

ENTRYPOINT ["/entrypoint.sh"]
COPY entrypoint.sh entrypoint.sh
COPY commands.sh /commands.sh

VOLUME /work
WORKDIR /work
ENV HOME /work

RUN curl --silent --location --output /usr/local/bin/yq https://github.com/mikefarah/yq/releases/download/${YQ_BIN_VERSION}/yq_linux_amd64 && \
    chmod 755 /usr/local/bin/yq && \
    chmod +x /entrypoint.sh && \
    chmod +x /commands.sh

RUN curl --silent -o terraform.zip https://releases.hashicorp.com/terraform/${TERRAFORM_VERSION}/terraform_${TERRAFORM_VERSION}_linux_amd64.zip && \
    unzip terraform.zip && \
    chmod +x terraform && \
    mv terraform /usr/bin && \
    rm terraform.zip

RUN curl --silent -L https://storage.googleapis.com/kubernetes-release/release/v${KUBECTL_VERSION}/bin/linux/amd64/kubectl && \
    mv kubectl /usr/bin/kubectl && \
    chmod +x /usr/bin/kubectl
zalewski@plilxl-108704:~/PIOTRZ/aztools$ []
```

Docker image history

IMAGE	CREATED	CREATED BY	SIZE	COMMENT
cb879425691e	2 minutes ago	RUN /bin/sh -c curl --silent -L0 https://sto...	39.3MB	buildkit.dockerfile.v0
<missing>	2 minutes ago	RUN /bin/sh -c curl --silent -o terraform.zi...	56.6MB	buildkit.dockerfile.v0
<missing>	2 minutes ago	RUN /bin/sh -c curl --silent --location --ou...	5.4MB	buildkit.dockerfile.v0
<missing>	2 minutes ago	ENV HOME=/work	0B	buildkit.dockerfile.v0
<missing>	2 minutes ago	WORKDIR /work	0B	buildkit.dockerfile.v0
<missing>	2 minutes ago	VOLUME [/work]	0B	buildkit.dockerfile.v0
<missing>	2 minutes ago	COPY commands.sh /commands.sh # buildkit	67B	buildkit.dockerfile.v0
<missing>	2 minutes ago	COPY entrypoint.sh entrypoint.sh # buildkit	73B	buildkit.dockerfile.v0
<missing>	2 minutes ago	ENTRYPOINT ["/entrypoint.sh"]	0B	buildkit.dockerfile.v0
<missing>	2 minutes ago	ENV KUBECTL_VERSION=1.13.10	0B	buildkit.dockerfile.v0
<missing>	2 minutes ago	ENV TERRAFORM_VERSION=0.12.13	0B	buildkit.dockerfile.v0
<missing>	2 minutes ago	ENV YQ_BIN_VERSION=2.2.0	0B	buildkit.dockerfile.v0
<missing>	2 months ago	/bin/sh -c #(nop) CMD ["/bin/sh" " -c" "bash_...	0B	buildkit.dockerfile.v0
<missing>	2 months ago	3 BUILD_DATE=2019-10-30T06:26:39Z CLI_VERSI...	1.25kB	buildkit.dockerfile.v0
<missing>	2 months ago	/bin/sh -c #(nop) WORKDIR /	0B	buildkit.dockerfile.v0
<missing>	2 months ago	3 BUILD_DATE=2019-10-30T06:26:39Z CLI_VERSI...	282MB	buildkit.dockerfile.v0
<missing>	2 months ago	/bin/sh -c #(nop) COPY dtr:7a3763f48c2fc6db3...	406MB	buildkit.dockerfile.v0
<missing>	2 months ago	/bin/sh -c #(nop) WORKDIR /azure-cli	0B	buildkit.dockerfile.v0

Layers cache

```
zalewski@pl1lxl-108704 ~/PIOTRZ/aztools master docker build .
[+] Building 0.2s (12/12) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 308B
=> [internal] load .dockerignore
=> => transferring context: 28
=> [internal] load metadata for mcr.microsoft.com/azure-cli:2.0.76
=> [1/7] FROM mcr.microsoft.com/azure-cli:2.0.76
=> [internal] load build context
=> => transferring context: 64B
=> CACHED [2/7] COPY entrypoint.sh entrypoint.sh
=> CACHED [3/7] COPY commands.sh /commands.sh
=> CACHED [4/7] WORKDIR /work
=> CACHED [5/7] RUN curl --silent --location --output /usr/local/bin/yq https://github.com/mikefarah/yq/releases/download/v4.13.1/yq_linux_amd64
=> CACHED [6/7] RUN curl --silent -o terraform.zip https://releases.hashicorp.com/terraform/0.12.13/terraform_0.12.13_linux_amd64.zip
=> CACHED [7/7] RUN curl --silent -L https://storage.googleapis.com/kubernetes-release/release/v1.13.10/bin/linux/amd64/kubectl
=> exporting to image
=> => exporting layers
=> => writing image sha256:cb879425691ea1ca7a142202b03c36fab09ff5c1ef01f5fc732f1c5ee5e07915
zalewski@pl1lxl-108704 ~/PIOTRZ/aztools master
```

```
zalewski@pl1lxl-108704 ~/PIOTRZ/aztools master docker build .
[+] Building 19.2s (12/12) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 1.00kB
=> [internal] load .dockerignore
=> => transferring context: 28
=> [internal] load metadata for mcr.microsoft.com/azure-cli:2.0.76
=> [1/7] FROM mcr.microsoft.com/azure-cli:2.0.76
=> [internal] load build context
=> => transferring context: 64B
=> CACHED [2/7] COPY entrypoint.sh entrypoint.sh
=> CACHED [3/7] COPY commands.sh /commands.sh
=> CACHED [4/7] WORKDIR /work
=> [5/7] RUN curl --silent --location --output /usr/local/bin/yq https://github.com/mikefarah/yq/releases/download/v4.13.1/yq_linux_amd64
=> [6/7] RUN curl --silent -o terraform.zip https://releases.hashicorp.com/terraform/0.12.14/terraform_0.12.14_linux_amd64.zip
=> [7/7] RUN curl --silent -L https://storage.googleapis.com/kubernetes-release/release/v1.13.10/bin/linux/amd64/kubectl
=> exporting to image
=> => exporting layers
=> => writing image sha256:4d974ba765bbdb106a149ccbe5014076d014910dadfac40c0e819d4b4732cd50
zalewski@pl1lxl-108704 ~/PIOTRZ/aztools master
```

Docker build context

One of first steps in build process is sending build context to docker daemon.

Default build context is whole directory with all subfolders.

Tip: The rule of thumb is not adding files to the build context, if you do not need them in your Docker image. The Docker context should be minimal and secret-free.

Use wisely .dockerignore

Docker build context

```
docker build -f- out/ < Dockerfile
```

Scanners - hadolint

Best practices for Dockerfile

```
docker run --rm -i hadolint/hadolint < Dockerfile
```

```
zalewski@pl1lx1-108704 ~/HAD/env-deployer 》 master ① docker run --rm -i hadolint/hadolint < Dockerfile
/dev/stdin:17 DL3018 Pin versions in apk add. Instead of 'apk add <package>' use 'apk add <package>=<version>'
/dev/stdin:17 DL3019 Use the '-no-cache' switch to avoid the need to use '--update' and remove '/var/cache/apk/*' when done installing packages
/dev/stdin:17 DL3013 Pin versions in pip. Instead of 'pip install <package>' use 'pip install <package>==<version>'.
X zalewski@pl1lx1-108704 ~/HAD/env-deployer 》 master ②
```

Fast use docker apps (Azure Container Instances)

Cleaninig after work

logs

add "log-opts": { "max-size": "10m", "max-file": "3" } to
daemon.logs

other logs drivers

you could use “syslog” for redirect all logs to syslog

`docker system prune`

`docker system prune`

This doesn't clean by default:

- * Running containers
- * tagged images
- * volumes

restart option

Flag	Description
<code>no</code>	Do not automatically restart the container. (the default)
<code>on-failure</code>	Restart the container if it exits due to an error, which manifests as a non-zero exit code.
<code>always</code>	Always restart the container if it stops. If it is manually stopped, it is restarted only when Docker daemon restarts or the container itself is manually restarted. (See the second bullet listed in restart policy details)
<code>unless-stopped</code>	Similar to <code>always</code> , except that when the container is stopped (manually or otherwise), it is not restarted even after Docker daemon restarts.

docker system prune

```
docker run -d --restart=unless-stopped --name cleanup -v /var/run/docker.sock:/tmp/docker.sock /bin/sh -c "while true; do docker system prune -f & sleep 5; done"
```

tips for java

java - cpu and memery , heapsize
https://youtu.be/RwtrLjh8_SU?t=2609