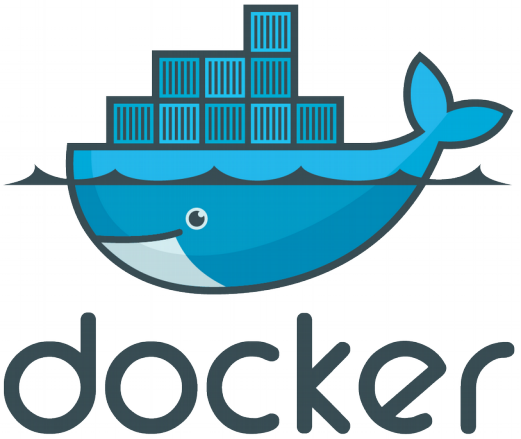
JDEV2017




# Déjouer les pièges du Dockerfile

**Nicolas CARPi** – UMR144 Institut Curie / CNRS
Team of Matthieu Piel

cnrs

INSTITUT
PIERRE-GILLES
DE GENNES
Pour la microfluidique

institutCurie
Ensemble, prenons le cancer de vitesse.

SYSTEMS BIOLOGY
OF CELL POLARITY
AND CELL DIVISION

# Attention !

Cette présentation est en franglais !

Pardon les puristes…

# Qui suis-je ?

- Ingénieur à l'Institut Curie
- Biochimiste de formation
- Biologiste cellulaire de métier
- Développeur d'eLabFTW : le cahier de labo électronique open-source pour la Recherche


https://www.elabftw.net

# La galaxie elab

elabftw

# La galaxie elab

elabimg
Dockerfile repo

elabctl
install/update/
backup script

elabftw

elabget
get.elabftw.net

elabdoc
Documentation

elabrls
To make a
new release

elabapy
Python lib for
API access

elabweb
www.elabftw.net

# La galaxie elab

**elabimg**
Dockerfile repo

**elabctl**
install/update/
backup script

**elabftw**

**elabget**
get.elabftw.net

**elabdoc**
Documentation

**elabrls**
To make a
new release

**elabapy**
Python lib for
API access

**elabweb**
www.elabftw.net

# On met quoi dans un container ?

- Une préoccupation par container
  - mais on peut tricher (php + nginx ça passe)
- Un seul process par container
  - mais on peut tricher (supervisord)

# Les principes de base

- Minimiser le nombre de couches
  - && est votre ami
- Enlever ce qui ne servira pas
  - rm -rf /var/lib/apt/lists
  - --no-install-recommends pour apt
- Utiliser les bonnes images de base

# Utilisez les bonnes images

- Il n'y a pas que Ubuntu dans la vie :
  - Alpine linux (1.9 Mo)
  - Busybox (1~5 Mo)
  - Scratch
- Les images de base pour certains languages :
  - Python, ruby, PHP, …
- Les images pour les services :
  - Mysql, nginx, redis, rabbitmq, …

# Mise en pratique

```
FROM ubuntu
COPY ./code /app
RUN apt-get update
RUN apt-get install -y python git ssh python-pip

CMD python /app/app.py
```

# Mise en pratique

```
FROM ubuntu:16.04
COPY ./code /app
RUN apt-get update
RUN apt-get install -y python git ssh python-pip

CMD python /app/app.py
```

Fix version in FROM

# Mise en pratique

```
FROM ubuntu:16.04
COPY ./code /app
RUN apt-get update
RUN apt-get install -y python git ssh python-pip
COPY ./code /app

CMD python /app/app.py
```

Be smart about caching

# Mise en pratique

```
FROM ubuntu:16.04
COPY ./code /app
RUN apt-get update && apt-get install -y python git ssh python-pip
COPY ./code /app

CMD python /app/app.py
```

Group related commands

# Mise en pratique

```
FROM python:3.4
COPY ./code /app
RUN apt-get update && apt-get install -y python git ssh python-pip
COPY ./code /app

CMD python /app/app.py
```

Use correct base image

# Mise en pratique

```
FROM python:3.4
COPY ./code /app
RUN apt-get update && apt-get install -y python git ssh python-pip
COPY ./code /app

CMD python /app/app.py
```

Use `docker exec` if you want to go inside a container

# Mise en pratique

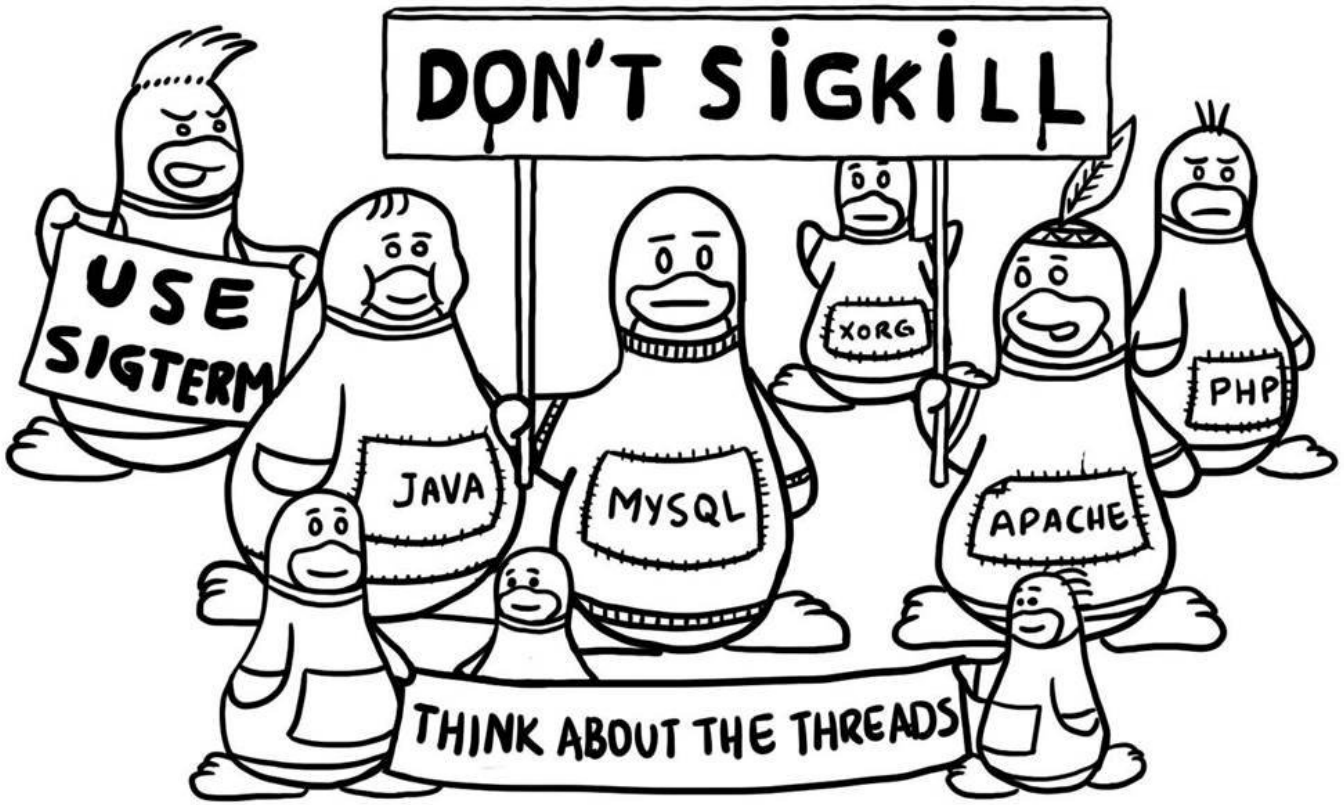
```
FROM python:3.4
COPY ./code /app
RUN apt-get update && apt-get install -y git \
&& rm -rf /var/lib/apt/lists*
COPY ./code /app

CMD python /app/app.py
```

Cleanup!

# Propagation du signal

- `docker stop =>` envoie un SIGTERM, puis 10 secondes plus tard un SIGKILL



# CMD, ENTRYPOINT and pid 1

- Your process should be pid 1
- Use ENTRYPOINT and CMD for binary like behavior (CMD is the default arg to ENTRYPOINT (--help))
- CMD will use `/bin/sh -c` <= pas de propagation du signal
  - better to use array syntax:

```
CMD [“/run.sh”]
```

# Good things to do

- Use Read Only filesystem
- Drop unused capabilities (audit_write, mknod, sys_chroot, setfcap, net_raw, …)
- Do not use the root user (use USER)
- Use labels

| Labels | | |
|---|---|---|
| | org.label-schema.description | Run nginx and php-fpm to serve elabftw |
| | org.label-schema.name | elabftw |
| | org.label-schema.schema-version | 1.0 |
| | org.label-schema.url | https://www.elabftw.net |
| | org.label-schema.vcs-url | https://github.com/elabftw/elabimg |
| | org.label-schema.version | 1.6.1 |

# The startup script

- /run.sh:
  - change configuration files based on ENV
  - start services with supervisord

- Note : les secrets dans ENV c'est pas top
  - ENV variables show up in `docker inspect`
  - Et aussi pour tous les autres process
  - Donc une fois que je les ai utilisées je les `unset`

# Git things

- Use .dockerignore (.git .gitignore)
- Use a separate repo from the main one
- Use hub.docker.com for automated builds
- Use online tools to inspect layers (microbadger.com)

# Les secrets

- On parle de quoi ?
  - API key
  - Database user/password

Une solution possible :

- https://www.vaultproject.io/
- https://square.github.io/keywhiz/

# Ayénafini


J'entends et j'oublie, Je vois et je me souviens,
Je fais et je comprends.
(Confucius)
QQcitations