

# Linux Contêineres na Unha



THE  
DEVELOPER'S  
CONFERENCE



CONFERENCE

# Wellington F. Silva

**contato:**  
@\_wsilva

**nicks:**  
wsilva, boina, tom, fisi\*

**roles:**  
pai, tec. telecom,  
programador, sysadmin,  
instrutor, escritor, php zce,  
docker community leader,  
docker certified associate

\* deprecation in favor of Well



# Agenda

- Disclaimer
- O que são contêineres Linux
- O que são **Kernel Namespaces**
- O que são **CGroups**
- O que são **Rootfs**
- Demo

# Disclaimer

Palestra inspirada nas talks da  
**Liz Rice** (@lizrice) e no artigo\* de  
**Julian Friedman** (@doctor\_julz)

<http://bit.ly/container-golang> \*

# Julian Friedman

- Engenheiro líder do **Garden**  
(tecnologia de contêineres da  
**Cloud Foundry's**)
- Trabalha para **IBM**
- Defensor do **Open Source**



# Liz Rice



- Especialista em segurança de contêineres na [Aqua Security](#)
- Contribuidora Kubernetes
- Evangelista Go Lang

O que é **DOCKER**?

O que é **Container**?



“Ferramenta para  
virtualização no nível do  
**Sistema Operacional.**”

“Ferramenta para  
virtualização no nível do  
**Sistema Operacional.**”

–Wellington F. Silva

# Tipos de Virtualização

# Tipos de Virtualização

- Full Virtualization

# Tipos de Virtualização

- Full Virtualization
- Partial Virtualization

# Tipos de Virtualização

- Full Virtualization
- Partial Virtualization
- Paravirtualization

# Tipos de Virtualização

- Full Virtualization
- Partial Virtualization
- Paravirtualization
- OS Level Virtualization

# Tipos de Virtualização

- **Full Virtualization**
- **Partial Virtualization**
- **Paravirtualization**
- **OS Level Virtualization**  
(Docker, rkt, Garden, Guardian, containerD, cri-O)



Container **NÃO** é VM

**Evolução**



**Como Funciona?**

PROCESSOS

+

KERNEL NAMESPACES

+

CGROUPS

+

ROOTFS

PROCESSOS

# PROCESSOS

Aqueles que matamos com kill -9

# Processos

- **Processo Linux** com **ID** atribuído (**PID**)



# Processos

- **Processo Linux** com **ID** atribuído (**PID**)
- Podem ser listados (com **ps**) **dentro do contêiner** ou **fora (no host)**

# Processos

- **Processo Linux** com **ID** atribuído (**PID**)
- Podem ser listados (com **ps**) **dentro do contêiner** ou fora, **no host**
- Dentro do contêiner o **PID=1**, fora (no host) vai ter valor alto

# KERNEL NAMESPACES

# KERNEL NAMESPACES

Tudo aquilo que o processo pode "ver"

# Kernel Namespaces

- **IPC** - Interprocess communication

# Kernel Namespaces

- **IPC** - Interprocess communication
- **PID** - Processos

# Kernel Namespaces

- **IPC** - Interprocess communication
- **PID** - Processos
- **NET**- Rede

# Kernel Namespaces

- **IPC** - Interprocess communication
- **PID** - Processos
- **NET**- Rede
- **UTS** - Unix timesharing system



# Kernel Namespaces

- **IPC** - interprocess communication
- **PID** - Processos
- **NET**- rede
- **UTS** - Unix timesharing system
- **USER** - Usuários e grupos

# Kernel Namespaces

- **IPC** - interprocess communication
- **PID** - Processos
- **NET**- rede
- **UTS** - Unix timesharing system
- **USER** - Usuários e grupos
- **MNT** - Pontos de montagem

**CGROUPS**

# CGROUPS

Tudo aquilo que o processo pode "usar"

# CGroups

- CPU

# CGroups

- CPU
- Memória

# CGroups

- CPU
- Memória
- I/O em disco

# CGroups

- CPU
- Memória
- I/O em disco
- Interface de rede



# CGroups

- CPU
- Memória
- I/O em disco
- Interface de rede
- Hardwares periféricos  
(pen drives, cameras, etc)

# ROOTFS

Onde o processo vai rodar

# Rootfs

- **Filesystem é uma hierarquia de diretórios**

# Rootfs

- Filesystem é uma hierarquia de diretórios
- **Rootfs** é o filesystem base onde os demais sistemas de **arquivos** são montados

# Rootfs

- Filesystem é uma hierarquia de diretórios
- **Rootfs** é o filesystem base onde os demais sistemas de **arquivos** são montados
- Em linhas gerais é o / (barra) de uma distribuição **Linux**

# Resumo

- **Processo**  
(é o que roda a aplicação)
- **CGroups e Namespaces**  
(como o processo vai rodar)
- **Rootfs**  
(onde o processo vai rodar)

“Talk is cheap. Show me  
the code.”

–Linus Torvalds

# Demo

<https://github.com/wsilva/container-from-scratch-demo>



Valeu  
=)

Slides em  
<https://speakerdeck.com/wsilva/> \*