

Provisionando ambientes com a ferramenta Docker

Docker é uma ferramenta que permite um conjunto completo do ambiente, desde configurações de máquinas, aplicações e programas necessários, permitindo ser distribuído de maneira fácil e rápida para outras máquinas. Para utilização da ferramenta é necessário seguir as seguintes etapas, este tutorial foi aplicado em um ambiente Linux (Ubuntu) para o provisionamento dos ambientes do CRS e TI Unochapecó:

1 – Adicionar chave publica Docker

```
sudo apt-key adv --keyserver hkp://pgp.mit.edu:80 --recv-keys  
58118E89F3A912897C070ADBF76221572C52609D
```

2 – Adicionar no arquivo /etc/apt/sources.list.d/docker.list

```
nano /etc/apt/sources.list.d/docker.list  
# Ubuntu Precise  
deb https://apt.dockerproject.org/repo ubuntu-precise main  
# Ubuntu Trusty  
deb https://apt.dockerproject.org/repo ubuntu-trusty main  
# Ubuntu Vivid  
deb https://apt.dockerproject.org/repo ubuntu-vivid main  
# Ubuntu Wily  
deb https://apt.dockerproject.org/repo ubuntu-wily main
```

3 – Atualizar pacotes

```
sudo apt-get update
```

4 – Instalação do docker

```
sudo apt-get install docker.io
```

5 – Atualizar sistemas

```
sudo apt-get update
```

6 – Reiniciar sistema

```
sudo reboot
```

7 – Configuração docker para inicialização

```
sudo systemctl enable docker
```

8 – Reiniciar os serviços NetworkManager e Docker

```
sudo restart network-manager
```

```
sudo restart docker
```

9 – Verificar se o Docker esta rodando

```
sudo docker run hello-world
```

10 – Depois de tudo instalado, o que falta é o container onde mantemos a maquina com suas configurações e programas necessários para o ambientes de desenvolvimento. O Docker mantém um local pra armazenar repositórios¹. E para este caso foi criado um com o nome de crs-uno-server² e também no github³.

11 – Para baixar o repositório crs-uno-server

```
docker pull alefevariani/crs-uno-server
```

12 – Listando os container

```
docker ps
```

13 – Rodando e acessando o shell do container, informando o container ID

```
docker run -it CONTEINER ID bash
```

Assim pode-se utilizar este container para várias máquinas, no caso, para o servidor de desenvolvimento e produção, evitando assim diferenças de versões em sistemas operacional, linguagem de programação, banco de dados e pacotes extras.

1 <https://hub.docker.com>

2 <https://hub.docker.com/r/alefevariani/crs-uno-server/>

3 <https://github.com/AlefeVariani/crs-uno-server>