**INSTITUTO INFNET**

**ESCOLA SUPERIOR DE TECNOLOGIA DA**

**INFORMAÇÃO**

**GRADUAÇÃO EM GESTÃO DE TI**



**PROJETO DE BLOCO – ARQUITETURA DE**

**INFRAESTRUTURA DE APLICAÇÕES**

**TESTE DE PERFORMANCE DA ETAPA 4**

**ALUNO: ALAN AGUINAGA**

**E-MAIL: alansenairj@gmail.com**

**TURMA: GRLRDC01C1-M1-L1 - Classe 1 - Manhã**



RDC - GRADUAÇÃO EM REDE DE COMPUTADORES

ALAN FERREIRA AGUINAGA

**Projeto de Bloco Arquitetura e Infraestrutura de Aplicações**

Rio de Janeiro

2018

**Sumário**

[1. Introdução 3](#_Toc526890777)

[2. Objetivo. 3](#_Toc526890778)

[3. Justificativa 3](#_Toc526890779)

[4. Desenvolvimento 3](#_Toc526890780)

[4.1 Instalação do Docker 3](#_Toc526890781)

[4.2 Explicação da aplicação. 9](#_Toc526890782)

[5. Bibliografia 10](#_Toc526890783)

# 1. Introdução

Documente a execução de uma aplicação distribuída, composta por dois ou mais containers Docker. Seu trabalho deve incluir um texto de pelo menos uma página explicando que aplicação vai ser executada e quais containers serão usados para isto. O restante da entrega deve incluir uma documentação completa da execução dos containers, e uma captura de tela mostrando o resultado alcançado.

# 2. Objetivo.

Trabalhar com o Docker e as ferramentas que ele disponibiliza para controlar execução de contêiner e execução de uma aplicação distribuída.

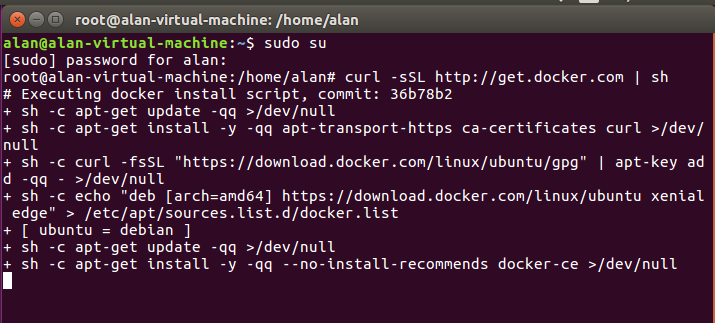
# 3. Justificativa

Demonstrar a instalação de contêineres e aplicações distribuídas para fixar os conhecimentos sobre a ferramenta.

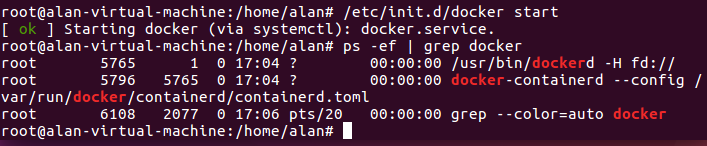
# 4. Desenvolvimento

# 4.1 Instalação do Docker

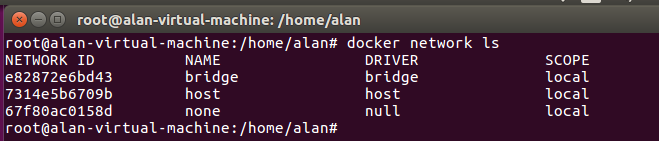
Foi instalado o Docker através de um script de instalação.



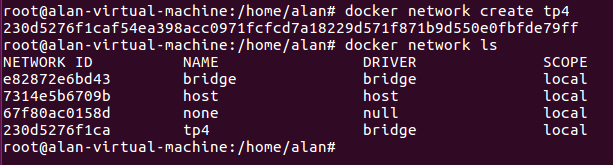
Foi realizada a inicialização para a verificação do processo no sistema.



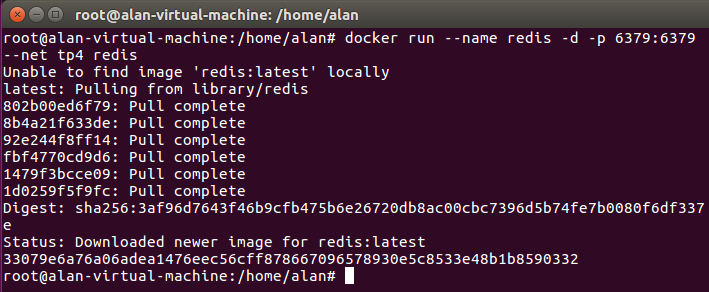
Foi criada uma rede padrão pelo Docker.



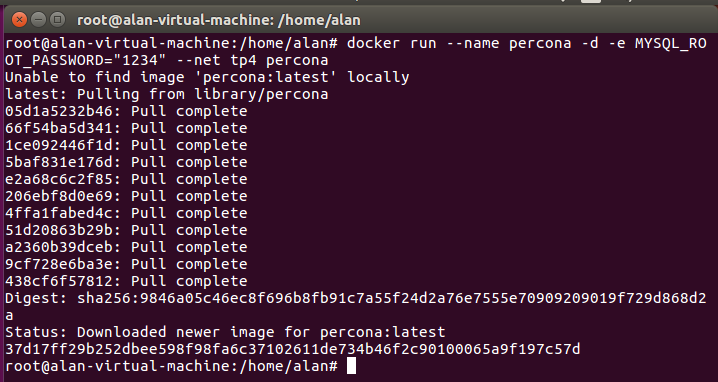
Foi criada uma rede com o nome “tp4”.



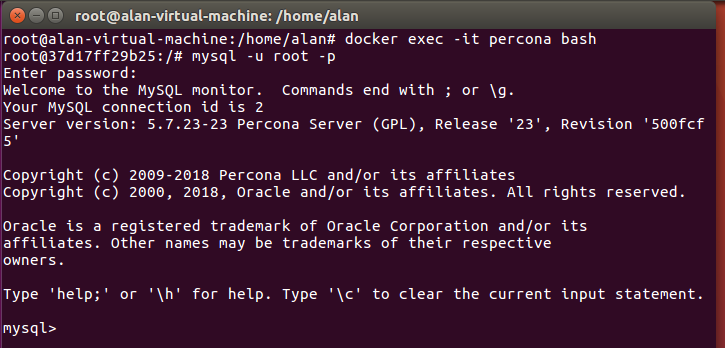
Foi instalado uma máquina Ubuntu em um container que contém o Wordpress instalado



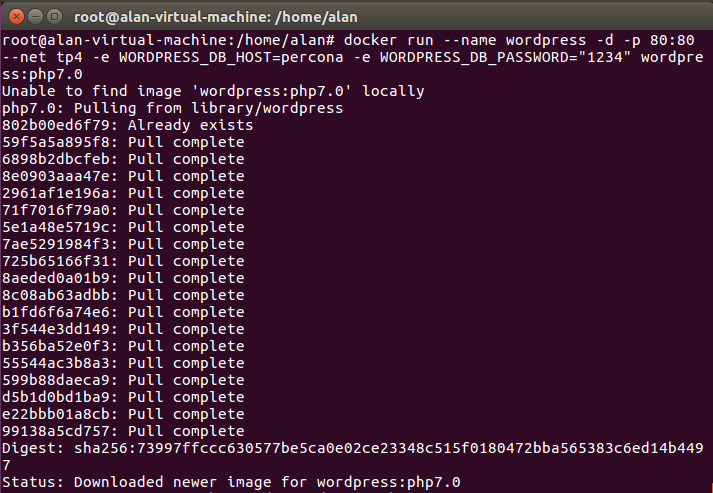
Foi instalada a aplicação com os seus respectivos bancos de dados.



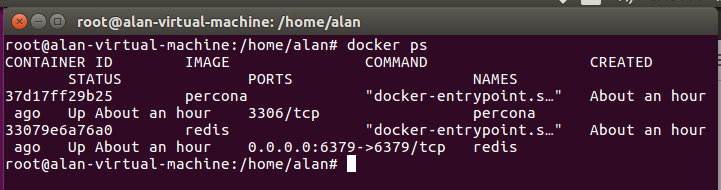
Foi executado o bash da aplicação do banco de dados para definição de senha do usuário root.



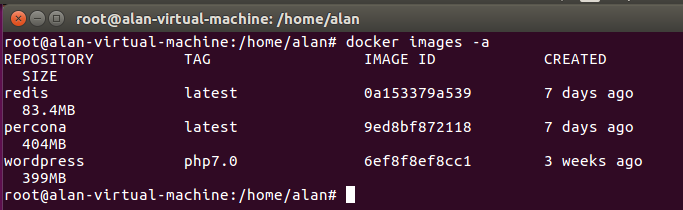
Foi inicializada a instalação da aplicação.



Com o comando abaixo, foram listados os 2 containers percona e redis.

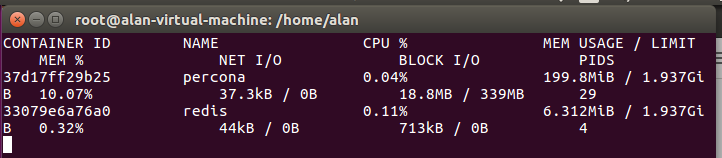


Essas são as imagens baixadas e hospedadas pelo Docker.

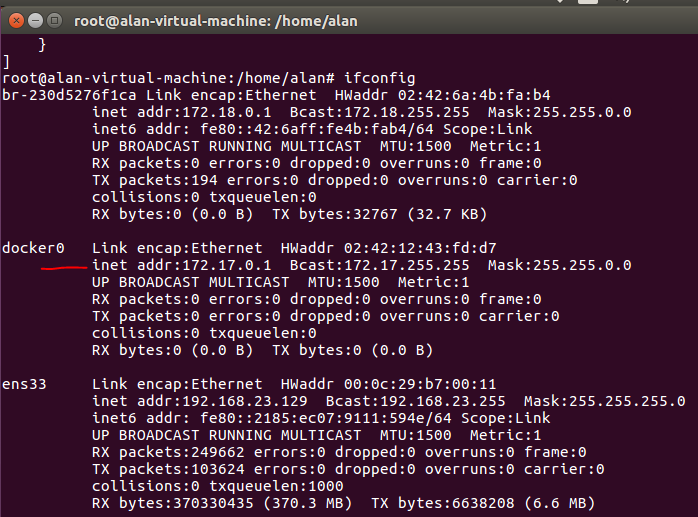


Com o comando abaixo, as estatísticas de uso dos recursos da máquina foram exibidas.

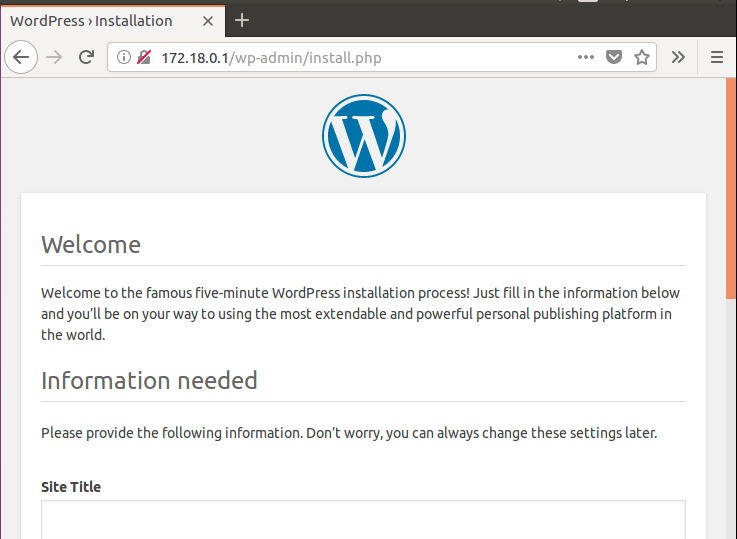




A interface de rede do Docker exibe o IP para acessar a aplicação.



A aplicação se encontra hospedada e funcional.



# 4.2 Explicação da aplicação.

Em se tratando de Docker, existe uma grande facilidade em se instalar a funcionalidade em um sistema Linux. Esse sistema tem grande suporte para a utilização de contêineres de aplicação.

Docker é uma plataforma de código aberto feito pela Google que usa uma linguagem chamada Go. O Docker necessita de um kernel Linux atualizado para operar. O Docker usa um recurso que já existe no Linux, porem conta com um site que hospeda, de maneira similar aos repositórios de aplicação, imagens de sistemas com aplicações já instaladas e configuradas.

Um contêiner pode ser até 7 vezes mais rápido que uma máquina virtual. Ele usa recursos do sistema que trata os processos (chroot) de maneira modular, permitindo o isolamento de um ambiente. Cada imagem hospedada serve como base de um sistema operacional. Os contêineres s são os locais que as aplicações são modificadas.

O Docker apresenta uma maneira muito segura para executar, pro exemplo, a implementação e teste de uma aplicação, usando uma máquina física, e disponibilizando diversos ambientes de teste para o desenvolvimento de software. É possível prover diversos contêineres diferentes para uma mesma imagem, de forma a isolar cada um deles.

O Docker utiliza uma rede paralela própria que conecta as imagens e seus respectivos contêineres em uma rede própria e paralela a rede do sistema. A classe da rede padrão do Docker é a classe B. é importante criar uma rede que não se misture com a rede física real existente, por isso o Docker cria por padrão uma interface de rede e a insere dentro de sua própria rede.

No caso da instalação e disponibilização do Wordpress através do Docker, foi escolhido a instalação de uma imagem Ubuntu Linux versão 16.04 para evitar incompatibilidades.

Um container pode ser comparado a um processo. O contêiner só fica exposto em usa utilização. Ao sair do contêiner o processo é encerrado.

No caso da hospedagem do Wordpress foram usadas 3 imagens diferentes. A imagem redis é um servidor de dicionário para armazenamento persistente, disponibilizado no repositório Docker em: <https://hub.docker.com/_/redis/>

A outra imagem percona tem a finalidade de armazenar o banco de dados da aplicação, nesse caso o banco de dados mysql. <https://hub.docker.com/_/percona/>

A ultima imagem é Wordpress que contém a aplicação que servirá de ambiente de desenvolvimento do site. Essa aplicação roda nessa imagem através do servidos apache.

Com essas 3 imagens é possível gerar um container de aplicação Wordpress operacional.

# 5. Bibliografia

https://www.redhat.com/pt-br/topics/containers/what-is-docker

https://www.opservices.com.br/o-que-e-docker/

http://sinestec.com.br/blog/o-que-e-container-docker/

https://devops.profitbricks.com/tutorials/install-wordpress-with-docker-on-ubuntu-1604/

https://www.howtoforge.com/tutorial/how-to-install-wordpress-with-docker-on-ubuntu/

https://www.youtube.com/watch?v=0cDj7citEjE