



Nome: Matheus Avelino

Turma: RDC

CONTEXTO

Uma empresa no ramo de imobiliária de pequeno porte que conta com cerca de 80 empregados apresenta na área de administração (40 pessoas) necessidade de armazenamento para suas informações e seus documentos, e um visualizador e editor de documento para aumentar a praticidade e velocidade de armazenamento e edição de seus documentos e informações.

SOLUÇÃO

A solução ideal para todas as necessidades desta empresa é a aplicação Owncloud.

OwnCloud é um software open-source, similar ao Dropbox e Google Driver, amplamente utilizado, com a principal diferença de que este software é gratuito, permitindo a instalação na sua empresa, um servidor privado, sem custos de licenciamento, podendo alocar o espaço de armazenamento que a empresa necessite.

CARACTERÍSTICAS DA APLICAÇÃO

- Banco de dados Postgres;
- Pode ter um servidor web como front-end;
- Oferece uma interface simples e fácil ao usuário para pesquisa, favoritos e acesso de arquivos;
- Criptografia de arquivos do usuário;
- Suporte a Calendário, Lista de Tarefas, Livro de Endereços, streaming de música e etc.;
- Administração de usuários e grupos (via OpenID ou LDAP);
- Compartilhamento de conteúdo através de grupos ou URLs públicas;
- Editor de texto online com sintaxe;
- Galeria de Fotos;
- Visualizador de vídeo;
- Visualizador de arquivos Open Office e PDF;
- Suporte a storages externos (Se você tiver contas no Dropbox, GoogleDrive ou Amazon S3, você pode montar esses armazenamentos em sua instalação ownCloud).
- Versionamento de arquivos: Quando habilitado permite a recuperação de antigas versões de documentos (interessante).

RECURSOS DA APLICAÇÃO

- Multi-plataforma: Aplicativos para Desktop (Windows, Linux e Mac) e plataforma móveis Android, iPhone;
- Antivírus interno: Os arquivos são verificados com ClamAV, impedindo que o potencial de distribuição automatizada de arquivos infectados, detectando Trojans, vírus, malware e outras ameaças maliciosas;
- Versionamento de arquivos: Quando habilitado permite a recuperação de antigas versões de documentos (interessante); [?](#)
- Servidor de banco de dados;
- Servidor de aplicação;

DESENVOLVIMENTO

- Open source;
- PHP e JavaScript;
- Colaborativo;
- Código no GitHub;

PLANO DE IMPLANTAÇÃO

1. Instalar banco de dados;
2. Clonar código do GitHub;
3. Verifique a integridade do pacote;
4. Verifique a autenticidade através do PGP;
5. Siga o capítulo de instalação do ownCloud Admin Manuals. Se você já executa o ownCloud, consulte a documentação de atualização para lançamentos menores e o manual de atualização para mover para novas versões principais do ownCloud; (<https://owncloud.org/install/#edition>)
6. Fazer configuração para conexão de banco;
7. Testes com os usuários;

AUTOMATIZAÇÃO COM ANSIBLE E GITHUB.

É uma ferramenta de automatização de tarefas semelhante a Puppet e Chef, porém muito mais poderosa e a queridinha do pessoal DevOps hoje. Com ela é possível fazer o deploy de aplicações, provisionando de servidores, automatizar tarefas, e outras funções.

O Ansible trabalha com arquivos YAML, o principal arquivo de configuração é chamado de PLAYBOOK onde você coloca todas as tarefas que serão executadas “yum”, “mkdir”, “useradd” e no arquivo .ini você adiciona os servidores onde o Ansible irá executar esse PLAYBOOK. Cada Playbook possui roles que são as informações de provisionamento, após definir as roles você definirá em quais hosts essas roles serão executadas.

Dentro das roles existem as tasks, handlers, variaveis e templates:

- Tasks: Tarefas de provisionamento que serão executadas.
- Handlers: Tarefas para manipular serviços e arquivos.
- Templates: Arquivos para serem transformados em configurações dentro das máquinas.
- Variaveis: Valores que são definidos para serem usados dentro das tasks, handlers e templates

matheus84 playbook Owncloudaaadecc 11 minutes ago

1 contributor

46 lines (36 sloc) | 1.22 KB

RawBlameHistory

```
1 ---
2 - hosts: exemplo
3   remote_user: matheus
4   become: yes
5   tasks:
6
7   - name: Instala DB's e app necessários
8     apt: name={{ item }} state=latest update_cache=yes
9     with_items:
10      - apache2
11      - python2.7
12      - python-mysqldb
13      - mariadb-client
14      - mariadb-server
15      - php7.0
16      - libapache2-mod-php7.0
17      - php7.0-gd
18      - php7.0-mysql
19      - php7.0-curl
20      - php7.0-intl
21      - php7.0-mcrypt
22      - php-imagick
23      - php7.0-zip
24      - php7.0-xml
25      - php7.0-mbstring
26
27   - name: Garante que o Apache esta rodando
28     service: name=apache2 state=started enabled=yes
29
30   - name: Cria DB Owncloud
31     mysql_db: name=dbOwncloud state=present
32
33   - name: Instalar Repositorio Owncloud
34     raw: "sudo wget -nv https://download.owncloud.org/download/repositories/9.1/Ubuntu_16.10/Release.key -O Release.key && sudo apt-key a
35
36   - name: Instalar repositorio pt2
37     raw: "sudo sh -c 'echo deb http://download.owncloud.org/download/repositories/9.1/Ubuntu_16.10/ / > /etc/apt/sources.list.d/owncloud.
38
39   - name: apt-get update
40     raw: "sudo apt-get update"
41
42
43   - name: Instalar Owncloud
44     apt: name=owncloud state=latest
45
```

(Captura de tela do Playbook)

```
Terminal File Edit View Search Terminal Help
matheus@ubuntu:~/playbook$ ansible-playbook playbook.yml

PLAY *****
TASK [setup] *****
ok: [127.0.0.1]

TASK [Instala DB's e app necessários] *****
ok: [127.0.0.1] => (item=[u'apache2', u'python2.7', u'python-mysqldb', u'mariadb-client', u'mariadb-server', u'php7.0', u'libapache2-mod-php7.0', u'php7.0-gd', u'php7.0-mysql', u'php7.0-curl', u'php7.0-intl', u'php7.0-mcrypt', u'php-imagick', u'php7.0-zip', u'php7.0-xml', u'php7.0-mbstring'])

TASK [Garante que o Apache esta rodando] *****
ok: [127.0.0.1]

TASK [Cria DB Owncloud] *****
ok: [127.0.0.1]

TASK [Instalar Repositorio Owncloud] *****
ok: [127.0.0.1]

TASK [Instalar repositorio pt2] *****
ok: [127.0.0.1]

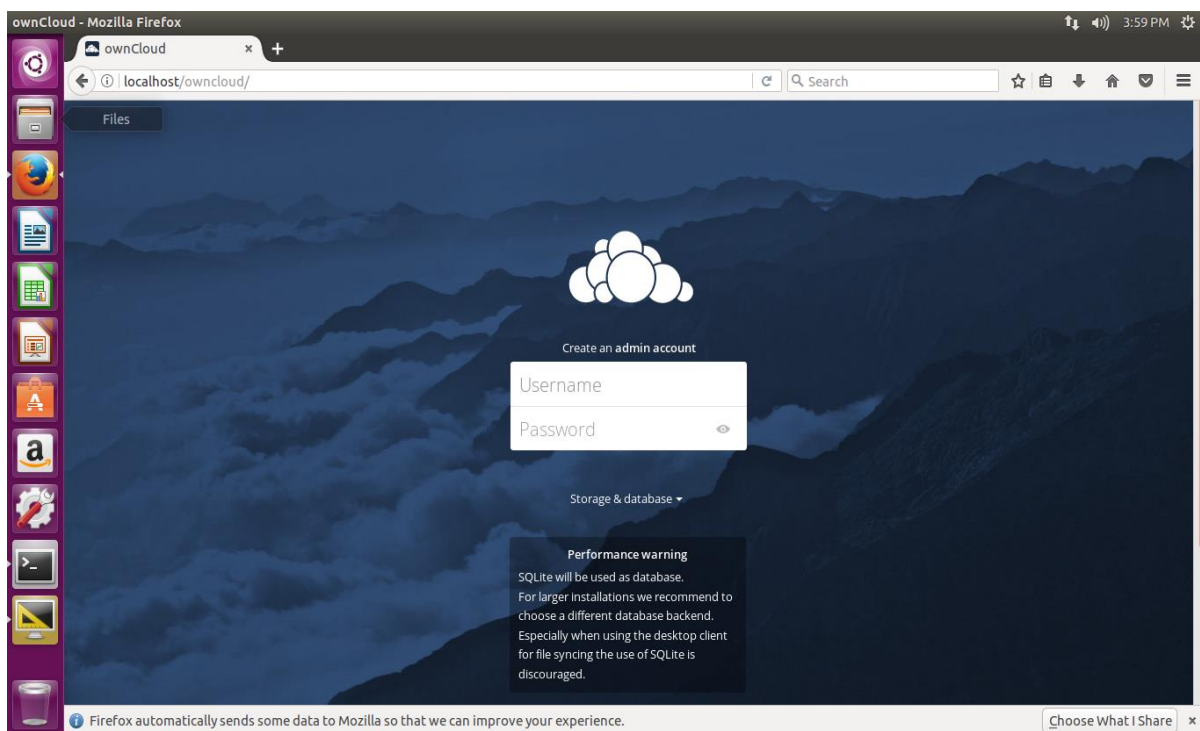
TASK [apt-get update] *****
ok: [127.0.0.1]

TASK [Instalar Owncloud] *****
ok: [127.0.0.1]

PLAY RECAP *****
127.0.0.1 : ok=8 changed=0 unreachable=0 failed=0

matheus@ubuntu:~/playbook$
```

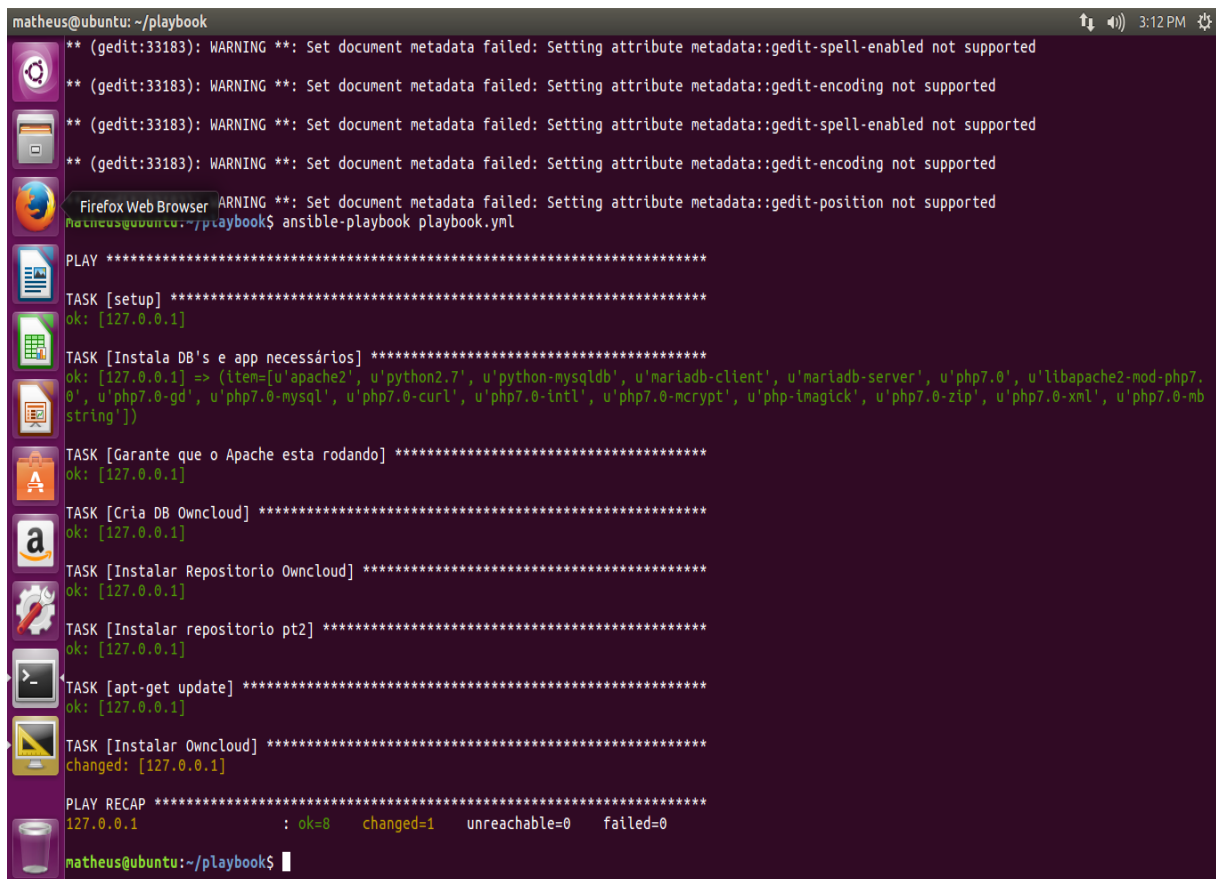
(Captura de tela do Playbook que automatiza instalação do Owncloud e seus pré requisitos.)



(<https://github.com/matheus84/Projeto>)

TELAS CAPTURADAS DO PLAYBOOK E PUSH/COMMIT PRO GITHUB

(<https://github.com/matheus84/Projeto>)



```
matheus@ubuntu: ~/playbook
** (gedit:33183): WARNING **: Set document metadata failed: Setting attribute metadata::gedit-spell-enabled not supported
** (gedit:33183): WARNING **: Set document metadata failed: Setting attribute metadata::gedit-encoding not supported
** (gedit:33183): WARNING **: Set document metadata failed: Setting attribute metadata::gedit-spell-enabled not supported
** (gedit:33183): WARNING **: Set document metadata failed: Setting attribute metadata::gedit-encoding not supported
** (gedit:33183): WARNING **: Set document metadata failed: Setting attribute metadata::gedit-position not supported
matheus@ubuntu: ~/playbook$ ansible-playbook playbook.yml

PLAY *****
TASK [setup] *****
ok: [127.0.0.1]

TASK [Instala DB's e app necessários] *****
ok: [127.0.0.1] => (item=[u'apache2', u'python2.7', u'python-mysqldb', u'mariadb-client', u'mariadb-server', u'php7.0', u'libapache2-mod-php7.0', u'php7.0-gd', u'php7.0-mysql', u'php7.0-curl', u'php7.0-intl', u'php7.0-mcrypt', u'php-tmagick', u'php7.0-zip', u'php7.0-xml', u'php7.0-mbstring'])

TASK [Garante que o Apache esta rodando] *****
ok: [127.0.0.1]

TASK [Cria DB Owncloud] *****
ok: [127.0.0.1]

TASK [Instalar Repositorio Owncloud] *****
ok: [127.0.0.1]

TASK [Instalar repositorio pt2] *****
ok: [127.0.0.1]

TASK [apt-get update] *****
ok: [127.0.0.1]

TASK [Instalar Owncloud] *****
changed: [127.0.0.1]

PLAY RECAP *****
127.0.0.1 : ok=8  changed=1  unreachable=0  failed=0

matheus@ubuntu: ~/playbook$
```

(Captura de tela do playbook funcionando sem erros)

```
Terminal File Edit View Search Terminal Help
Password for 'https://matheus84@github.com':
To https://github.com/matheus84/Projeto.git
! [rejected]        master -> master (fetch first)
error: failed to push some refs to 'https://github.com/matheus84/Projeto.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
matheus@ubuntu:~/playbook$ git push origin master
Username for 'https://github.com': matheus84
Password for 'https://github.com':
To https://github.com/matheus84/Projeto.git
! [rejected]        master -> master (fetch first)
error: failed to push some refs to 'https://github.com/matheus84/Projeto.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
matheus@ubuntu:~/playbook$ ^C
matheus@ubuntu:~/playbook$ git push -f origin master
Username for 'https://github.com': matheus84
Password for 'https://github.com':
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/matheus84/Projeto.git/'
matheus@ubuntu:~/playbook$ git push -f origin master
Username for 'https://github.com': matheus84
Password for 'https://github.com':
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 738 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/matheus84/Projeto.git
+ 17f88ad...aaadecc master -> master (forced update)
error: update_ref failed for ref 'refs/remotes/origin/master': cannot lock ref 'refs/remotes/origin/master': unable to create directory for .g
it/refs/remotes/origin/master
matheus@ubuntu:~/playbook$ sudo git push -f origin master
Username for 'https://github.com': matheus84
Password for 'https://github.com':
Everything up-to-date
matheus@ubuntu:~/playbook$
```

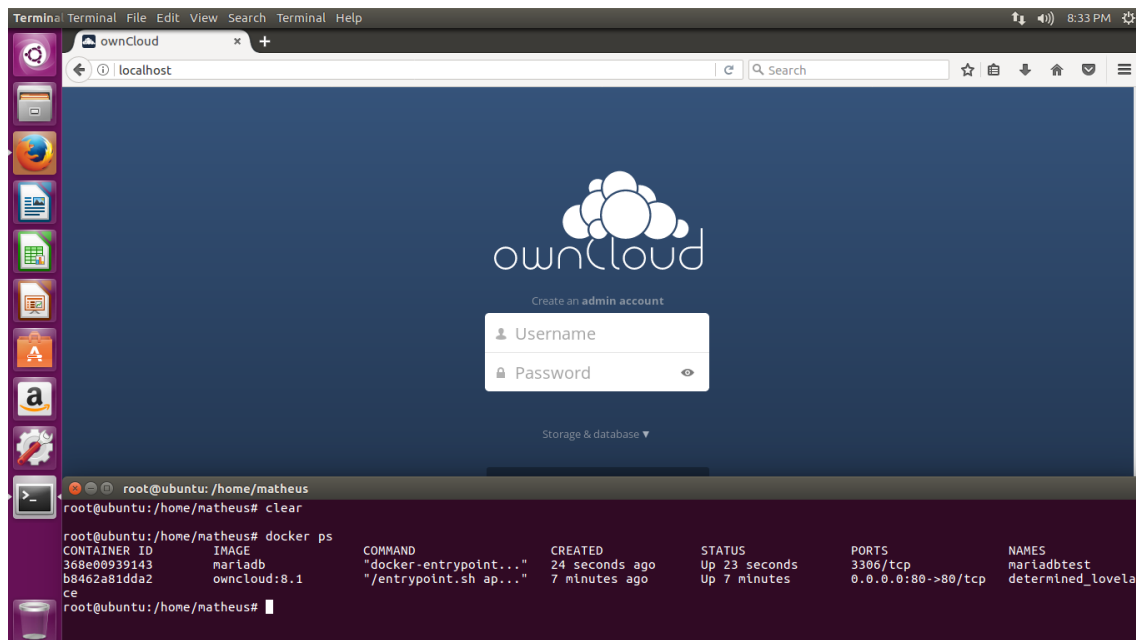
(Captura de tela do push para o repositório do GitHub)

DOCKER

Docker é uma plataforma Open Source escrito em Go, que é uma linguagem de programação de alto desempenho desenvolvida dentro do Google, que facilita a criação e administração de ambientes isolados.

O Docker possibilita o empacotamento de uma aplicação ou ambiente inteiro dentro de um container, e a partir desse momento o ambiente inteiro torna-se portátil para qualquer outro Host que contenha o Docker instalado.

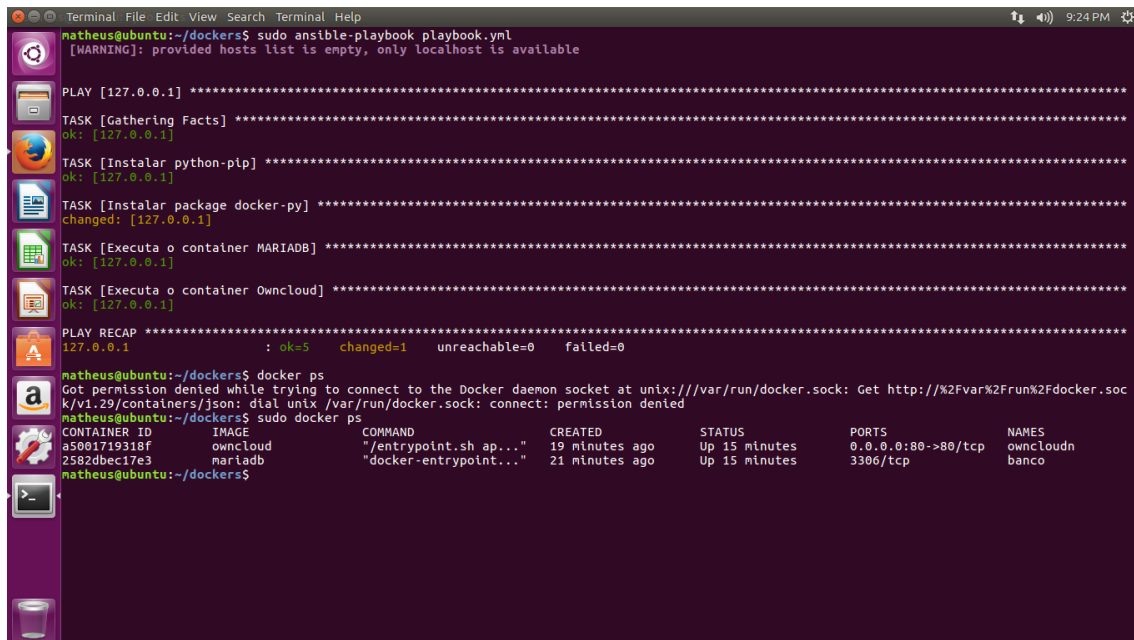
Isso reduz drasticamente o tempo de deploy de alguma infraestrutura ou até mesmo aplicação, pois não há necessidade de ajustes de ambiente para o correto funcionamento do serviço, o ambiente é sempre o mesmo, configure-o uma vez e replique-o quantas vezes quiser.



(A tela capturada apresenta os dois containers sendo executados, sendo um da própria aplicação owncloud e outra mariadb-server que é a database utilizada pelo owncloud. E no fundo da tela, o front-end da aplicação.)

CRIAÇÃO DO DOCKER AUTOMATIZADO COM ANSIBLE

(<https://github.com/matheus84/DockerandAnsible>)



The terminal window shows the execution of an Ansible playbook named 'playbook.yml' on a host named '127.0.0.1'. The playbook performs several tasks: gathering facts, installing python-pip, installing the docker-py package, and executing two Docker containers: 'mariadb' and 'owncloud'. A 'PLAY RECAP' section shows that all tasks were successful (ok=5, changed=1, unreachable=0, failed=0). Below the recap, the user runs 'docker ps' to list the running containers. The output shows two containers: 'a5001719318f' (owncloud) and '2582dbec17e3' (mariadb). The 'owncloud' container is using the 'owncloud' image and has its entrypoint set to '/entrypoint.sh ap...'. The 'mariadb' container is using the 'mariadb' image and has its entrypoint set to 'docker-entrypoint...'. The terminal also shows a warning about the provided hosts list being empty and a permission denied error when trying to connect to the Docker daemon socket.

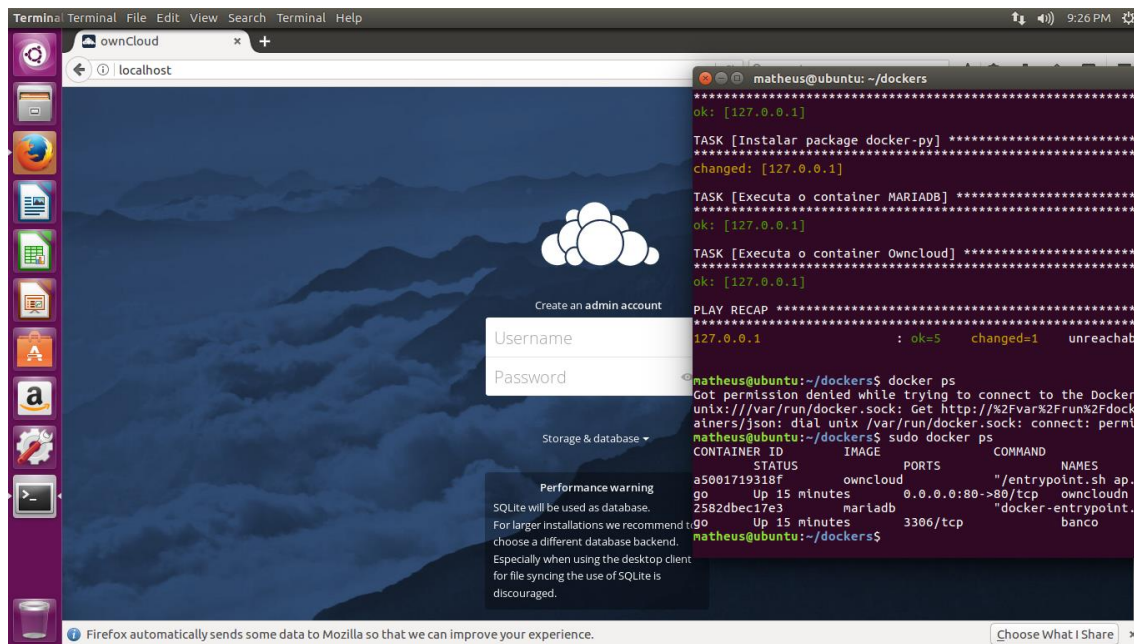
```
matheus@ubuntu:~/dockers$ sudo ansible-playbook playbook.yml
[WARNING]: provided hosts list is empty, only localhost is available

PLAY [127.0.0.1] *****
TASK [Gathering Facts] *****
ok: [127.0.0.1]
TASK [Instalar python-pip] *****
ok: [127.0.0.1]
TASK [Instalar package docker-py] *****
changed: [127.0.0.1]
TASK [Executa o container MARIADB] *****
ok: [127.0.0.1]
TASK [Executa o container Owncloud] *****
ok: [127.0.0.1]
PLAY RECAP *****
127.0.0.1 : ok=5 changed=1 unreachable=0 failed=0

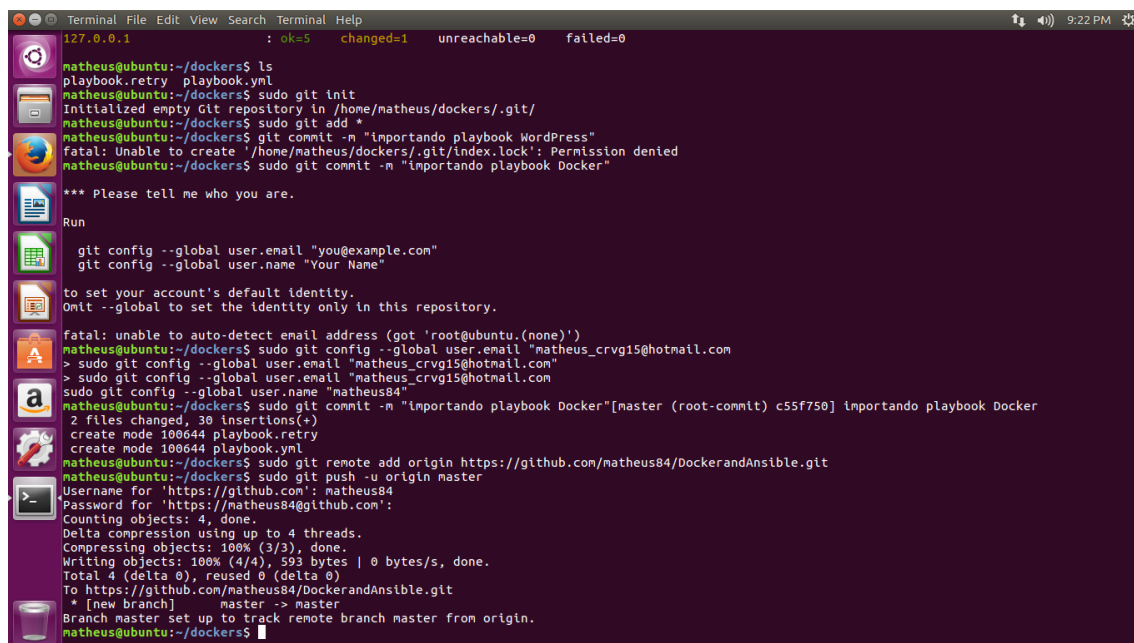
matheus@ubuntu:~/dockers$ docker ps
Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get http://%2Fvar%2Frun%2Fdocker.sock/v1.29/containers/json: dial unix /var/run/docker.sock: connect: permission denied
matheus@ubuntu:~/dockers$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED            STATUS              PORTS               NAMES
a5001719318f        owncloud            "/entrypoint.sh ap..." 19 minutes ago     Up 15 minutes      0.0.0.0:80->80/tcp   owncloudn
2582dbec17e3        mariadb             "docker-entrypoint..." 21 minutes ago     Up 15 minutes      3306/tcp            banco

matheus@ubuntu:~/dockers$
```

(Tela capturada mostrando a execução do Playbook que automatiza a criação dos containers owncloud e mariadb)



(Captura de tela mostrando que a aplicação está funcionando a partir do container criado pelo playbook anterior)



(Captura de tela do envio do playbook com as automatizações para o GitHub)

ARQUITETURA E CUSTOS

— Arquitetura:

A aplicação owncloud precisa de apenas uma maquina para seu servidor. O servidor tem como recomendo para tal tipo de empresa, no mínimo dois núcleos de CPU, 16 GB de RAM e armazenamento local conforme o necessário. No caso, seria necessário a criação de uma maquina virtual com esses requisitos em um servidor físico. A maquina virtual seria criada com o sistema operacional Linux, pois o servidor da aplicação só é compatível com Linux e de preferencia a versão mais atualizada.

— Custos:

- Um servidor DELL com 16gb de RAM, Intel® Xeon® E5-2620 v3 e 2TB de armazenamento custa em torno de R\$ 15.000,00.
- O Sistema operacional é gratuito.
- Virtualizador Vmware custa em torno de 250 U\$.

PLANEJAMENTO PASSO A PASSO/CRONOGRAMA

1. Instalação da Máquina Servidor. (1-2 dias)

2.1 Criação de uma máquina virtual através de um Hipervisor (Vmware) atendendo os requisitos do server da aplicação. (1-2 horas)

2.2 Requisitos:

- Tamanho do armazenamento: 100 GB a 10 TB.
- Uma máquina executando o servidor de aplicativos, servidor Web, servidor de banco de dados e armazenamento local.
- Autenticação através de um servidor LDAP ou Active Directory existente.

3. Instalar o sistema operacional na máquina. (1-2 horas)

4. Instalar banco de dados. (1-2 horas)

5. Clonar código do GitHub. (1-2 hora)

6. Verifique a integridade do pacote; (1/2 hora)

7. Verifique a autenticidade através do PGP; (1/2 hora)

8. Siga o capítulo de instalação do ownCloud Admin Manuals. Se você já executa o ownCloud, consulte a documentação de atualização para lançamentos menores e o manual de atualização para mover para novas versões principais do ownCloud; (<https://owncloud.org/install/#edition>) (1-2 dias)

9. Fazer configuração para conexão de banco; (1 dia)

10. Testes com os usuários; (3-7 dias)