UNIVERSIDAD MODELO

ESCUELA DE INGENIERIA

ING. EN DESARROLLO DE TECNOLOGÍA Y SOFTWARE

Fundamentos de la Nube

"Reporte Deploy"

Ana Cristina Castillo Escobar

ÍNDICE

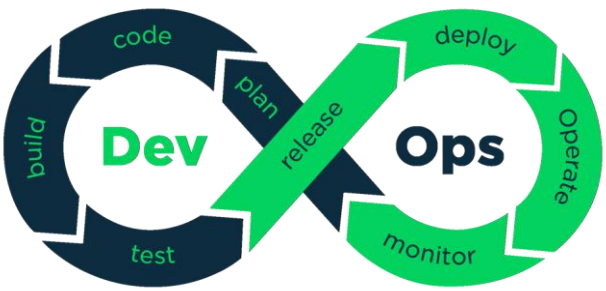
DEVOPS

INTRODUCCIÓN

DevOps es toda una cultura que nos habla de cómo las distintas etapas del desarrollo del software, aunque pueden dividirse en dos partes distintas (development y operation), deben ir de la mano, complementándose mutuamente, en un ciclo evolutivo sin fin.

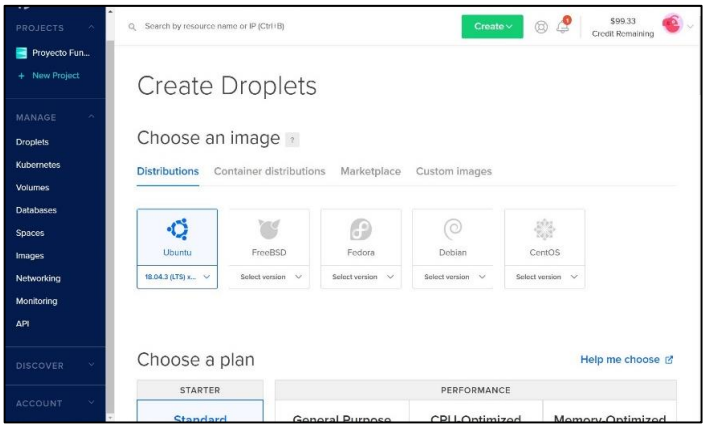
Este documento no es más que la demostración, paso por paso, de cómo funciona la etapa de deploy. Para ello utilicé herramientas como DigitalOcean, Fork, Docker, entre otras, que resultan intuitivas y son gratuitas (a menos que se requiera darles un uso mayor), y todas fueron vistas en clase.

Así, por medio de lo que dentro del entorno de DigitalOcean se llaman droplets, y conexiones SSH, cualquiera desde casa puede realizar prácticas para entender lo que es un deploy.

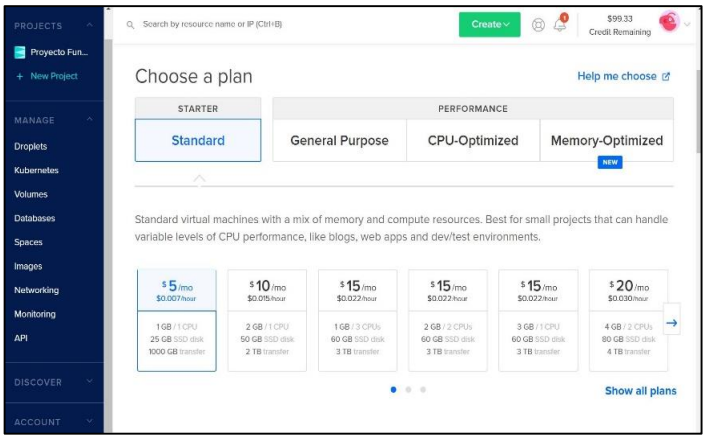


DigitalOcean
Container Registry

Creación del droplet

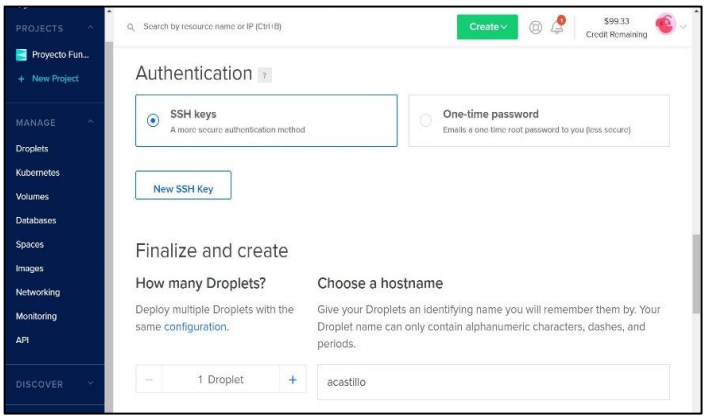


Para este paso utilicé el servicio de DigitalOcean. Una vez creada la cuenta, seguí los pasos para la creación del droplet.



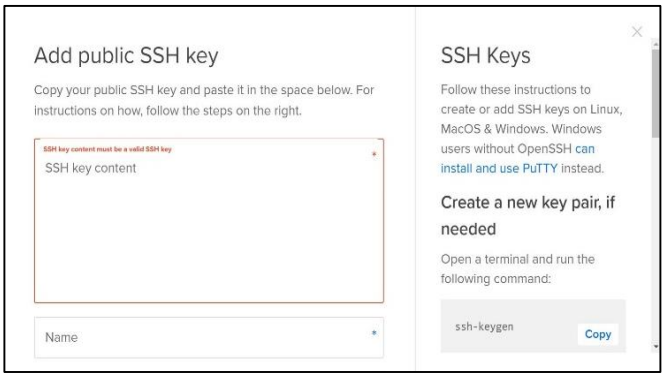
En esta ocasión seleccioné el sistema operativo de Ubuntu 18.04.3.

Las opciones mostradas en pantalla las dejé a como aparecían por default, pues al acabar la actividad tenía que destruir el droplet, por lo que no era necesaria una configuración o plan de pagos específico.

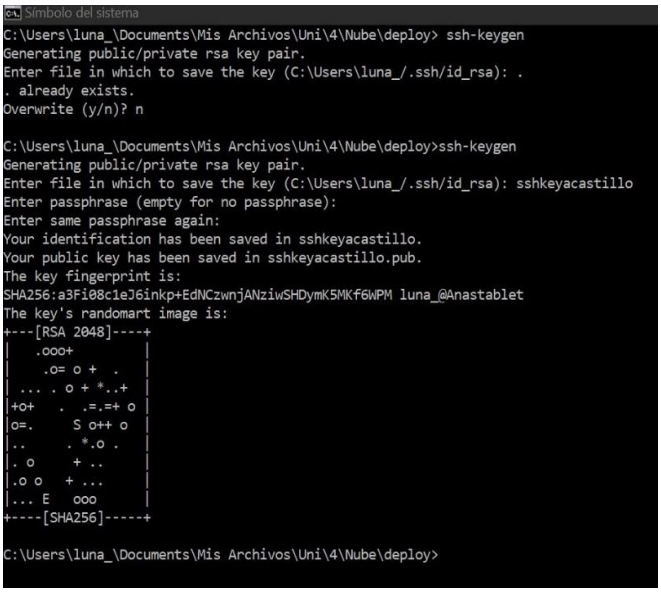


Para la autenticación escogí las llaves SSH. Cuando se selecciona esta opción se pide agregar una llave SSH. Al no contar con una, la creé ahí mismo.

Al hacer clic en el botón “New SSH Key”, aparece la pantalla emergente para agregar la llave que se quiera agregar al droplet.

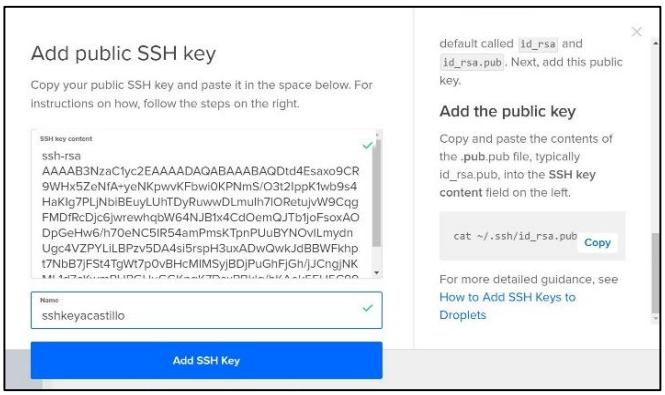


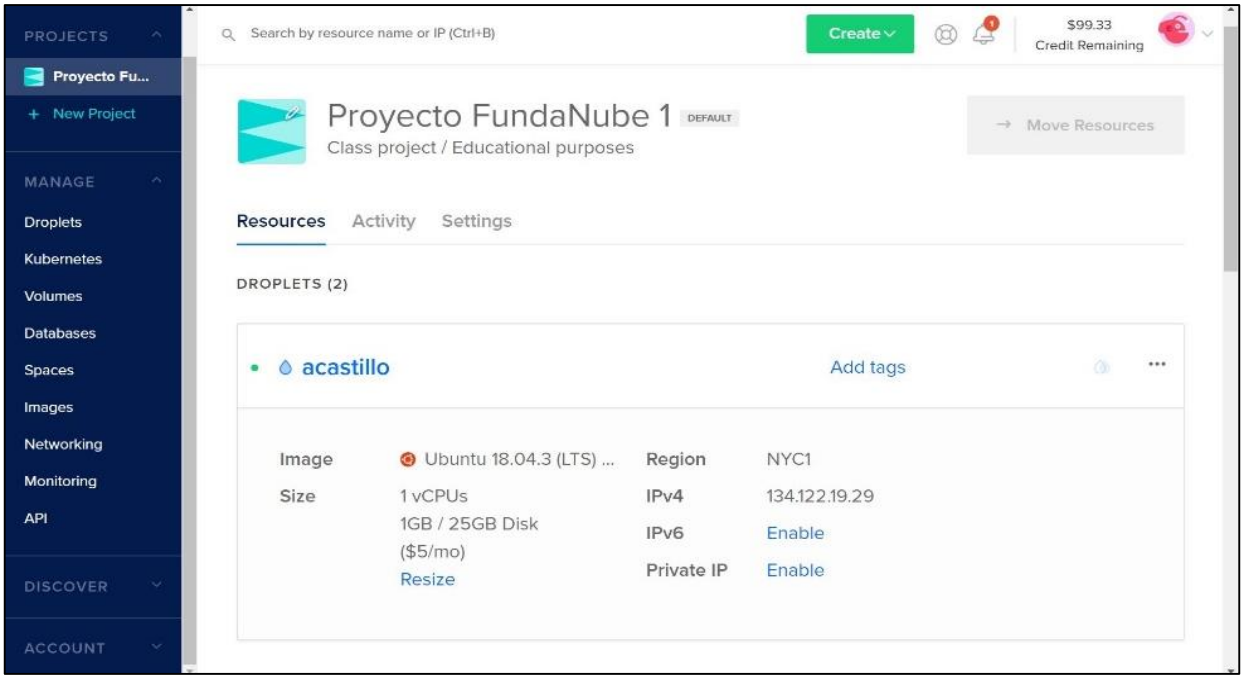
En la parte derecha de la misma, se muestran los pasos a seguir para crear tu propia llave, por medio del Símbolo del Sistema y el comando *ssh-keygen*.



Todos los pasos deben llevarse acabo dentro de la carpeta donde queremos guardar la llave, y desde la cual se realizará la conexión remota por ssh.

Una vez creada la llave, sólo falta copiar y pegar su contenido dentro de la caja de texto, y ponerle un nombre, para finalmente agregarla.



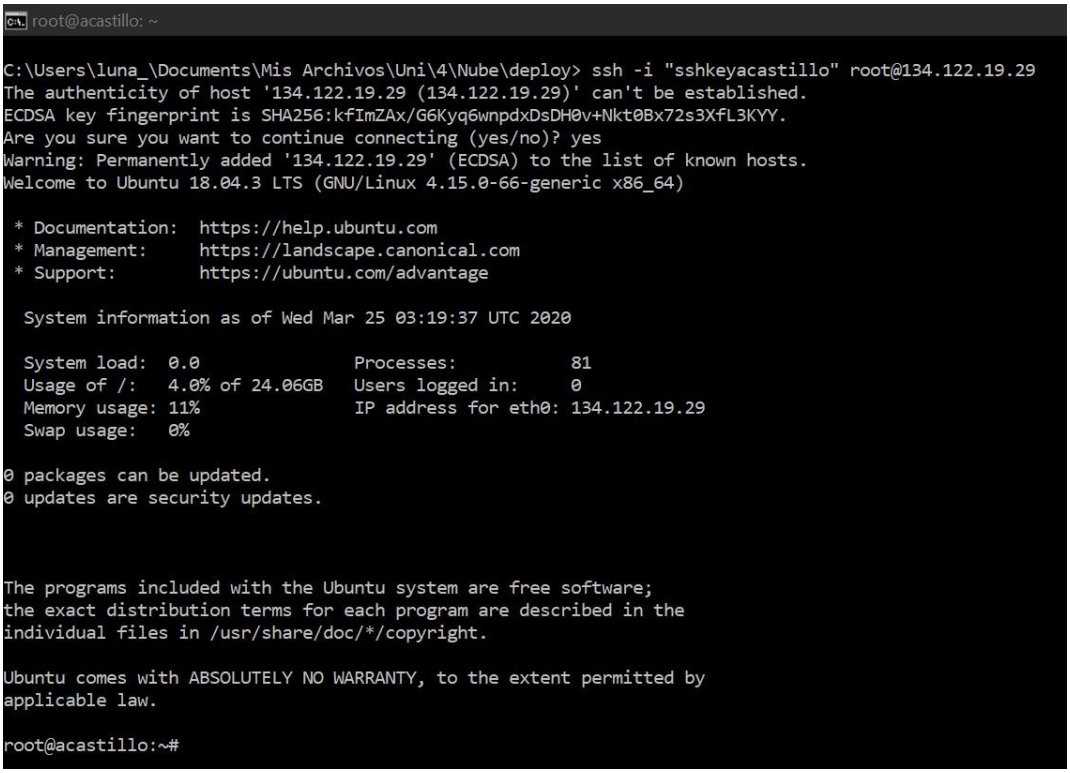


El resto de opciones para la configuración los dejé en default o sin seleccionar.

Al acabar el proceso de creación del droplet, se puede consultar sus características cómo se observa en la imagen superior. De todos estos, el **necesario para conectarse via ssh por medio de llaves** es la IP. En este caso **134.122.19.29.**

Cuando se escoge autenticación por contraseña, al usuario le llega un correo con las credenciales para la conexión. Entre estos datos se incluye un usuario. Como no seleccioné esa opción, no me llegó ningún correo, por lo que el usuario que utilicé fue el default (root).

Conexión vía SSH



El acceso via ssh se hace por medio del **cmd** (en el caso de Windows). Desde el directorio donde guardé la llave creada en el paso anterior, introduje el comando:

ssh –i sshkeyacastillo root@134.122.19.29

Donde **sshkeyacastillo** corresponde al nombre con el que guardé mi llave, **root** es el usuario y **134.122.19.29** la IP de mi droplet.

Luego seleccioné la opción “yes” y listo, la conexión fue exitosa.

Instalación de docker

En esta parte sólo hizo falta entrar a la página oficial de Docker con los comandos para instalar Docker Engine Community en Ubuntu y copiarlos en el cmd.

Get Docker Engine - Community for Ubuntu

Estimated reading time: 12 minutes

To get started with Docker Engine - Community on Ubuntu, make sure you meet the prerequisites, then install Docker.

Prerequisites

Docker EE customers

To install Docker Enterprise Edition (Docker EE), go to Get Docker EE for Ubuntu **instead of this topic.**

To learn more about Docker EE, see Docker Enterprise Edition.

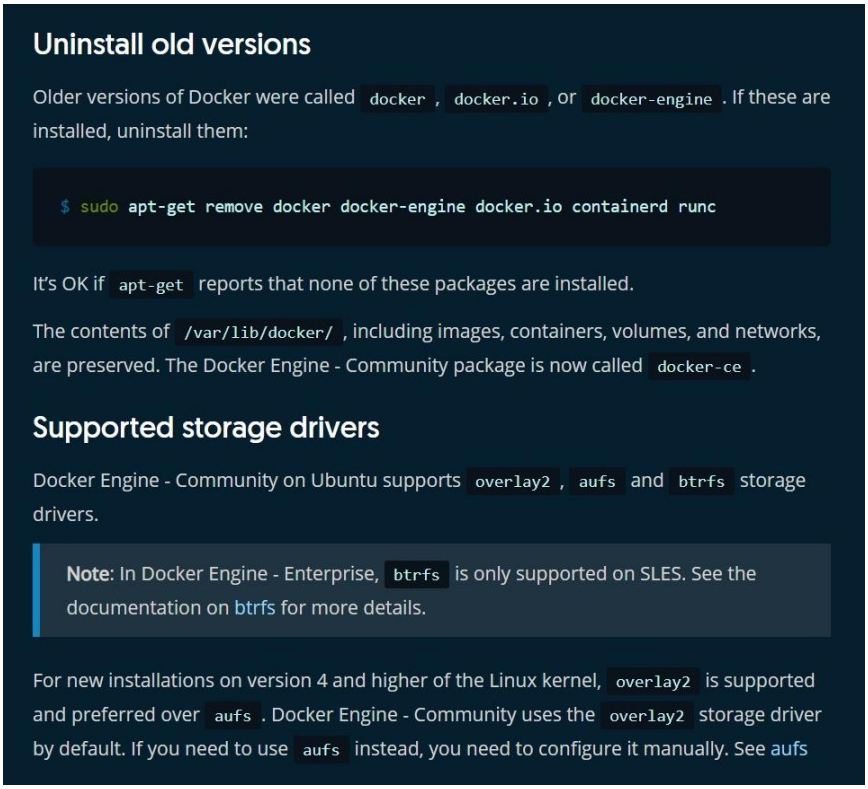
OS requirements

To install Docker Engine - Community, you need the 64-bit version of one of these Ubuntu versions:

- Eoan 19.10
- Bionic 18.04 (LTS)
- Xenial 16.04 (LTS)

Docker Engine - Community is supported on `x86_64` (or `amd64`), `armhf`, `arm64`, `s390x` (IBM Z), and `ppc64le` (IBM Power) architectures.

Antes que nada, desinstalar viejas versiones (aunque, evidentemente, el droplet no las tenga).



Uninstall old versions

Older versions of Docker were called `docker`, `docker.io`, or `docker-engine`. If these are installed, uninstall them:

```
$ sudo apt-get remove docker docker-engine docker.io containerd runc
```

It's OK if `apt-get` reports that none of these packages are installed.

The contents of `/var/lib/docker/`, including images, containers, volumes, and networks, are preserved. The Docker Engine - Community package is now called `docker-ce`.

Supported storage drivers

Docker Engine - Community on Ubuntu supports `overlay2`, `aufs` and `btrfs` storage drivers.

> **Note**: In Docker Engine - Enterprise, `btrfs` is only supported on SLES. See the documentation on btrfs for more details.

For new installations on version 4 and higher of the Linux kernel, `overlay2` is supported and preferred over `aufs`. Docker Engine - Community uses the `overlay2` storage driver by default. If you need to use `aufs` instead, you need to configure it manually. See aufs

```
root@acastillo:~#  sudo apt-get remove docker docker-engine docker.io containerd runc
Reading package lists... Done
Building dependency tree
Reading state information... Done

No apt package "docker", but there is a snap with that name.
Try "snap install docker"

E: Unable to locate package docker
E: Unable to locate package docker-engine
E: Unable to locate package docker.io
E: Couldn't find any package by glob 'docker.io'
E: Couldn't find any package by regex 'docker.io'
E: Unable to locate package containerd
E: Unable to locate package runc
root@acastillo:~#
```

Install using the repository

Before you install Docker Engine - Community for the first time on a new host machine, you need to set up the Docker repository. Afterward, you can install and update Docker from the repository.

SET UP THE REPOSITORY

1. Update the `apt` package index:

```
$ sudo apt-get update
```

2. Install packages to allow `apt` to use a repository over HTTPS:

```
$ sudo apt-get install \
    apt-transport-https \
    ca-certificates \
    curl \
    gnupg-agent \
    software-properties-common
```

3. Add Docker's official GPG key:

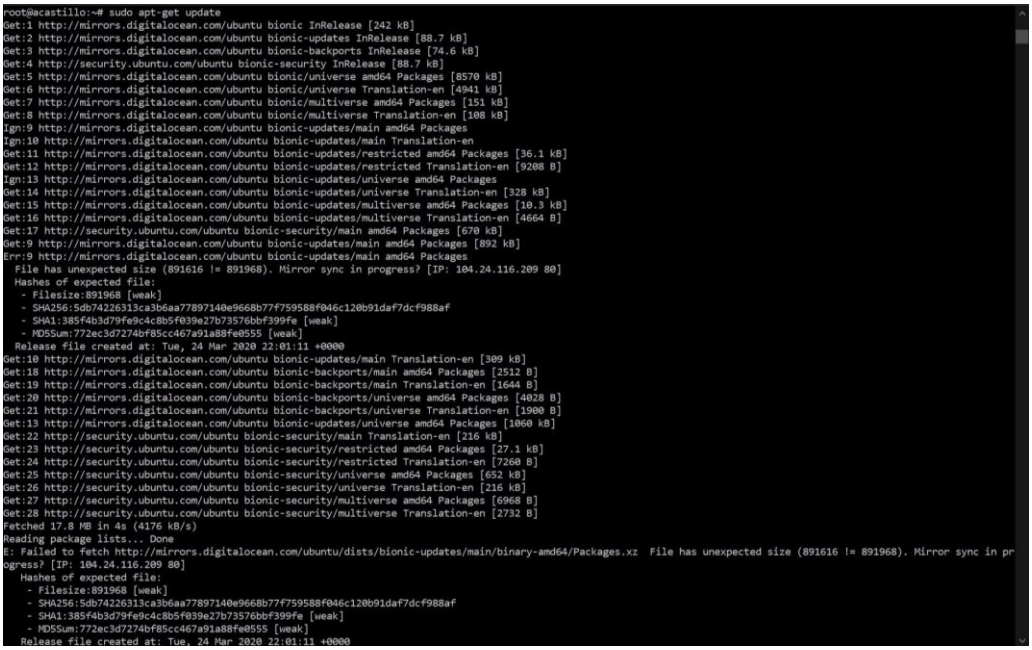
```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key a
```

Verify that you now have the key with the fingerprint

`9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88` , by searching for the last 8

Entre las opciones de instalación, tomé la instalación por medio del repositorio de Docker. Una vez que está listo, ya se puede instalar y actualizar Docker desde el repositorio.

En las siguientes páginas se observa la terminal después de correr los comandos de la imagen superior.



Comando apt-get update

```
root@acastillo:~# sudo apt-get install \
>     apt-transport-https \
>     ca-certificates \
>     curl \
>     gnupg-agent \
>     software-properties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20180409).
ca-certificates set to manually installed.
curl is already the newest version (7.58.0-2ubuntu3.8).
curl set to manually installed.
software-properties-common is already the newest version (0.96.24.32.11).
software-properties-common set to manually installed.
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  apt-transport-https gnupg-agent
0 upgraded, 2 newly installed, 0 to remove and 69 not upgraded.
Need to get 6576 B of archives.
After this operation, 195 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 apt-transport-https all 1.6.6ubuntu0.1 [1696 B]
Get:2 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 gnupg-agent all 2.2.4-1ubuntu1.2 [4880 B]
Fetched 6576 B in 0s (245 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 60017 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.6.6ubuntu0.1_all.deb ...
Unpacking apt-transport-https (1.6.6ubuntu0.1) ...
Selecting previously unselected package gnupg-agent.
Preparing to unpack .../gnupg-agent_2.2.4-1ubuntu1.2_all.deb ...
Unpacking gnupg-agent (2.2.4-1ubuntu1.2) ...
Setting up apt-transport-https (1.6.6ubuntu0.1) ...
Setting up gnupg-agent (2.2.4-1ubuntu1.2) ...
root@acastillo:~#
```

Comando apt-get install HTTPS

```
$ sudo apt-key fingerprint 0EBFCD88

pub   rsa4096 2017-02-22 [SCEA]
      9DC8 5822 9FC7 DD38 854A  E2D8 8D81 803C 0EBF CD88
uid           [ unknown] Docker Release (CE deb) <docker@docker.com>
sub   rsa4096 2017-02-22 [S]
```

4. Use the following command to set up the **stable** repository. To add the **nightly** or **test** repository, add the word `nightly` or `test` (or both) after the word `stable` in the commands below. Learn about **nightly** and **test** channels.

> **Note**: The `lsb_release -cs` sub-command below returns the name of your Ubuntu distribution, such as `xenial`. Sometimes, in a distribution like Linux Mint, you might need to change `$(lsb_release -cs)` to your parent Ubuntu distribution. For example, if you are using `Linux Mint Tessa`, you could use `bionic`. Docker does not offer any guarantees on untested and unsupported Ubuntu distributions.

x86_64 / amd64 | armhf | arm64 | ppc64le (IBM Power) | s390x (IBM Z)

```
$ sudo add-apt-repository \
   "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
   $(lsb_release -cs) \
   stable"
```

```
root@acastillo:~# sudo apt-get install docker-ce docker-ce-cli containerd.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  aufs-tools cgroupfs-mount libltdl7 pigz
The following NEW packages will be installed:
  aufs-tools cgroupfs-mount containerd.io docker-ce docker-ce-cli libltdl7 pigz
0 upgraded, 7 newly installed, 0 to remove and 56 not upgraded.
Need to get 85.8 MB of archives.
After this operation, 385 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://mirrors.digitalocean.com/ubuntu bionic/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Get:2 https://download.docker.com/linux/ubuntu bionic/stable amd64 containerd.io amd64 1.2.13-1 [20.1 MB]
Get:3 http://mirrors.digitalocean.com/ubuntu bionic/universe amd64 aufs-tools amd64 1:4.9+20170918-1ubuntu1 [104 kB]
Get:4 http://mirrors.digitalocean.com/ubuntu bionic/universe amd64 cgroupfs-mount all 1.4 [6320 B]
Get:5 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 libltdl7 amd64 2.4.6-2 [38.8 kB]
Get:6 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce-cli amd64 5:19.03.8~3-0~ubuntu-bionic [42.6 MB]
Get:7 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce amd64 5:19.03.8~3-0~ubuntu-bionic [22.9 MB]Fetched 85.8 MB in 2s (40.4 MB/s)
Selecting previously unselected package pigz.
(Reading database ... 90585 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.4-1_amd64.deb ...
Unpacking pigz (2.4-1) ...
Selecting previously unselected package aufs-tools.
Preparing to unpack .../1-aufs-tools_1%3a4.9+20170918-1ubuntu1_amd64.deb ...
Unpacking aufs-tools (1:4.9+20170918-1ubuntu1) ...
Selecting previously unselected package cgroupfs-mount.
Preparing to unpack .../2-cgroupfs-mount_1.4_all.deb ...
Unpacking cgroupfs-mount (1.4) ...
Selecting previously unselected package containerd.io.
Preparing to unpack .../3-containerd.io_1.2.13-1_amd64.deb ...
Unpacking containerd.io (1.2.13-1) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack .../4-docker-ce-cli_5%3a19.03.8~3-0~ubuntu-bionic_amd64.deb ...
Unpacking docker-ce-cli (5:19.03.8~3-0~ubuntu-bionic) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../5-docker-ce_5%3a19.03.8~3-0~ubuntu-bionic_amd64.deb ...
Unpacking docker-ce (5:19.03.8~3-0~ubuntu-bionic) ...
Selecting previously unselected package libltdl7:amd64.
Preparing to unpack .../6-libltdl7_2.4.6-2_amd64.deb ...
Unpacking libltdl7:amd64 (2.4.6-2) ...
Setting up aufs-tools (1:4.9+20170918-1ubuntu1) ...
Setting up containerd.io (1.2.13-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.Setting up cgroupfs-mount (1.4) ...
Setting up libltdl7:amd64 (2.4.6-2) ...
Setting up docker-ce-cli (5:19.03.8~3-0~ubuntu-bionic) ...
Setting up pigz (2.4-1) ...
```

Comando apt-get install Docker

Instalar docker compose

Dentro de la misma página de instalación de Docker Engine, se encuentran los comandos para instalar Docker Compose.

Install Compose on Linux systems

On Linux, you can download the Docker Compose binary from the Compose repository release page on GitHub. Follow the instructions from the link, which involve running the `curl` command in your terminal to download the binaries. These step-by-step instructions are also included below.

For `alpine`, the following dependency packages are needed: `py-pip`, `python-dev`, `libffi-dev`, `openssl-dev`, `gcc`, `libc-dev`, and `make`.

1. Run this command to download the current stable release of Docker Compose:

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.25.4/d
```

To install a different version of Compose, substitute `1.25.4` with the version of Compose you want to use.

If you have problems installing with `curl`, see Alternative Install Options tab above.

2. Apply executable permissions to the binary:

```
sudo chmod +x /usr/local/bin/docker-compose
```

```
root@acastillo:~# sudo curl -L "https://github.com/docker/compose/releases/download/1.25.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   617  100   617    0     0  10114      0 --:--:-- --:--:-- --:--:-- 10114
100 16.3M  100 16.3M    0     0  57.6M      0 --:--:-- --:--:-- --:--:-- 57.6M
root@acastillo:~# sudo chmod +x /usr/local/bin/docker-compose
root@acastillo:~# $ docker-compose --version
$: command not found
root@acastillo:~#  docker-compose --version
docker-compose version 1.25.4, build 8d51620a
root@acastillo:~# sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

Comando curl y chmod

GitHub Clonar repositorio

```
root@acastillo:/home# git clone https://github.com/TasiaCode2/mfc.git
Cloning into 'mfc'...
remote: Enumerating objects: 96, done.
remote: Counting objects: 100% (96/96), done.
remote: Compressing objects: 100% (28/28), done.
remote: Total 96 (delta 59), reused 96 (delta 59), pack-reused 0
Unpacking objects: 100% (96/96), done.
```

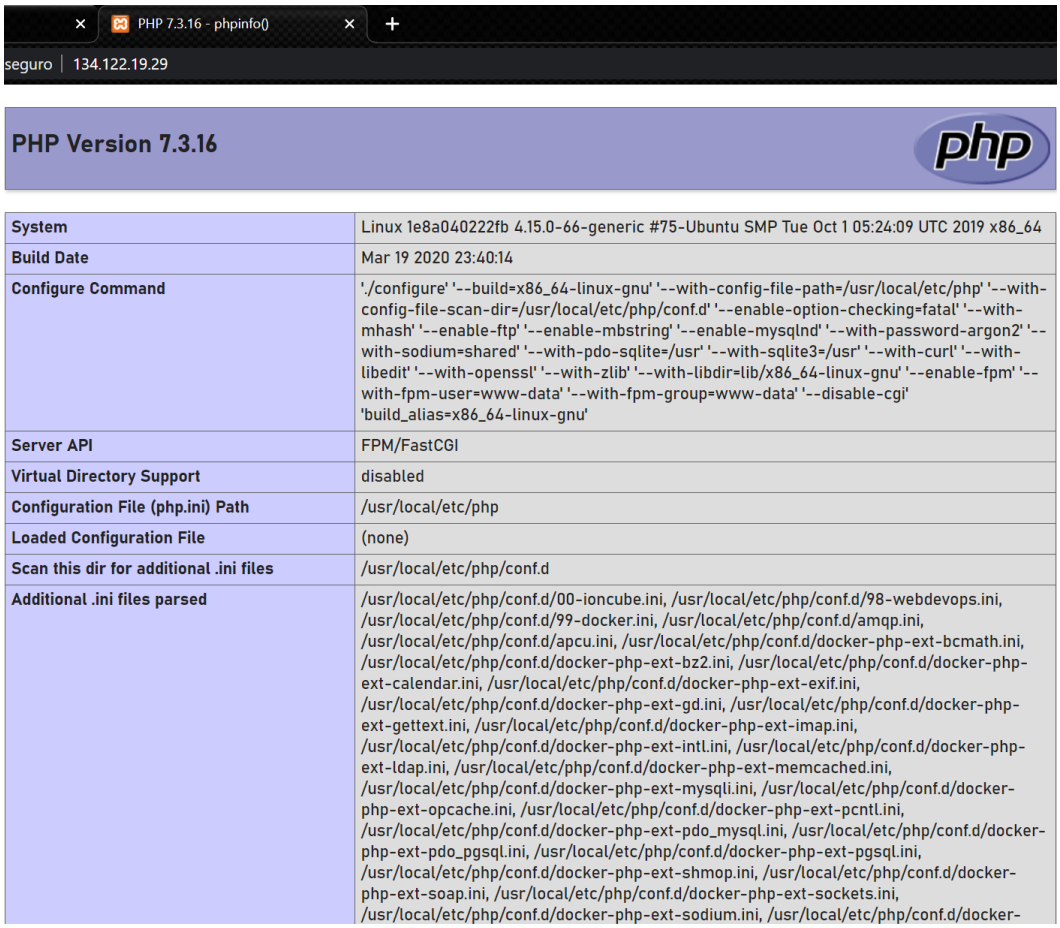
Cloné el repositorio de un contenedor que hice con imágenes de mariadb, php y apache.

```
root@acastillo:/home/mfc/mariapachepe# docker-compose up -d
Creating network "mariapachepe_default" with the default driver
Pulling db (mariadb/server:10.4)...
10.4: Pulling from mariadb/server
5c939e3a4d10: Pull complete
c63719cdbe7a: Pull complete
19a861ea6baf: Pull complete
651c9d2d6c4f: Pull complete
2db3b54cce92: Pull complete
ae9ce7e9adf4: Pull complete
d809091410c1: Pull complete
a0fcda8710e0: Pull complete
680ad6698c4c: Pull complete
0f325b2b4c13: Pull complete
1b3d2c17a886: Pull complete
1f5837b86ece: Pull complete
ebd63c8ad5ce: Pull complete
31dcb2c422e9: Pull complete
Digest: sha256:a89bc1f29d7521222224667d29ffb3ca4e424f97513099405ab279f87e28d20e
Status: Downloaded newer image for mariadb/server:10.4
Pulling phpache (webdevops/php-apache:7.3)...
7.3: Pulling from webdevops/php-apache
68ced04f60ab: Pull complete
1d2a5d8fa585: Pull complete
5d59ec4ae241: Pull complete
d42331ef4d44: Pull complete
a413239948bb: Pull complete
0e862908399b: Pull complete
2b149e595ce9: Pull complete
92fa4dca6f08: Pull complete
075eb11836cc: Pull complete
8851c85bc263: Pull complete
c67ad9c8a9df: Pull complete
b188ac6fcda2: Pull complete
86554a5bbe3d: Pull complete
98870994d731: Pull complete
36bbddd6f5bc: Pull complete
dcb873689169: Pull complete
f179a1681f5e: Pull complete
f5a946b8b05b: Pull complete
7c3747ecef9e: Pull complete
Digest: sha256:bdeda07524b9850b33a9c1c4d982cb4ec59f55a162854b14e1a72b828b6d01db
Status: Downloaded newer image for webdevops/php-apache:7.3
Creating mariapachepe_phpache_1 ... done
Creating mariapachepe_db_1      ... done
root@acastillo:/home/mfc/mariapachepe#
```

Levantar contenedor

Con el comando docker-compose up –d se levanta el contenedor y sale la consola

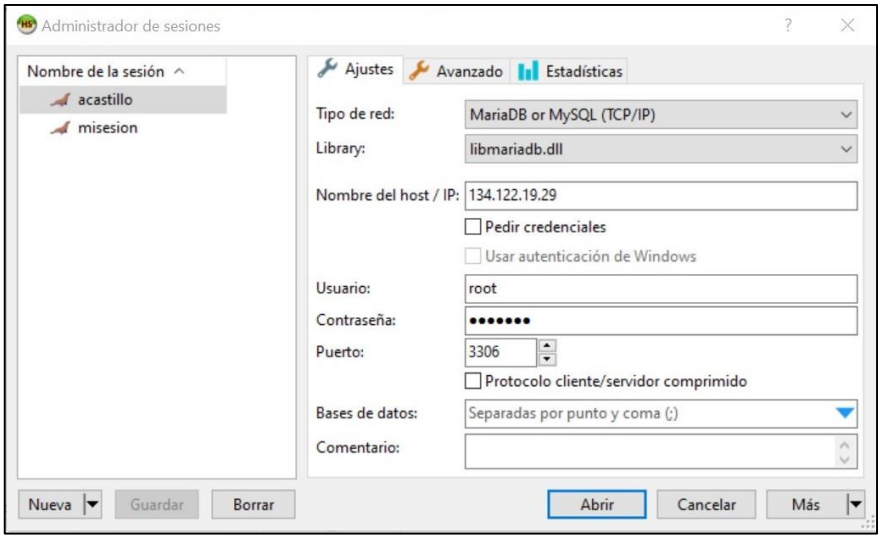
Acceso al contenedor via internet



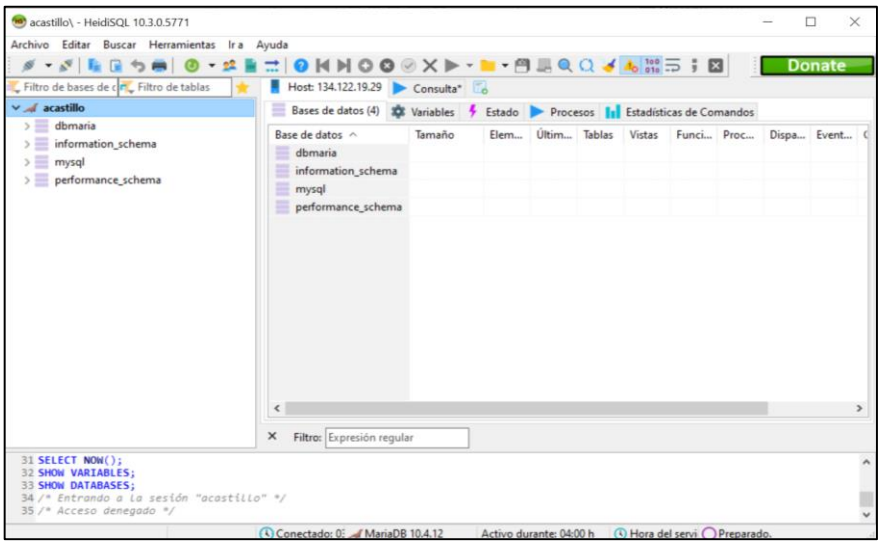
System	Linux 1e8a040222fb 4.15.0-66-generic #75-Ubuntu SMP Tue Oct 1 05:24:09 UTC 2019 x86_64
Build Date	Mar 19 2020 23:40:14
Configure Command	'./configure' '--build=x86_64-linux-gnu' '--with-config-file-path=/usr/local/etc/php' '--with-config-file-scan-dir=/usr/local/etc/php/conf.d' '--enable-option-checking=fatal' '--with-mhash' '--enable-ftp' '--enable-mbstring' '--enable-mysqlnd' '--with-password-argon2' '--with-sodium=shared' '--with-pdo-sqlite=/usr' '--with-sqlite3=/usr' '--with-curl' '--with-libedit' '--with-openssl' '--with-zlib' '--with-libdir=lib/x86_64-linux-gnu' '--enable-fpm' '--with-fpm-user=www-data' '--with-fpm-group=www-data' '--disable-cgi' 'build_alias=x86_64-linux-gnu'
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/usr/local/etc/php
Loaded Configuration File	(none)
Scan this dir for additional .ini files	/usr/local/etc/php/conf.d
Additional .ini files parsed	/usr/local/etc/php/conf.d/00-ioncube.ini, /usr/local/etc/php/conf.d/98-webdevops.ini, /usr/local/etc/php/conf.d/99-docker.ini, /usr/local/etc/php/conf.d/amqp.ini, /usr/local/etc/php/conf.d/apcu.ini, /usr/local/etc/php/conf.d/docker-php-ext-bcmath.ini, /usr/local/etc/php/conf.d/docker-php-ext-bz2.ini, /usr/local/etc/php/conf.d/docker-php-ext-calendar.ini, /usr/local/etc/php/conf.d/docker-php-ext-exif.ini, /usr/local/etc/php/conf.d/docker-php-ext-gd.ini, /usr/local/etc/php/conf.d/docker-php-ext-gettext.ini, /usr/local/etc/php/conf.d/docker-php-ext-imap.ini, /usr/local/etc/php/conf.d/docker-php-ext-intl.ini, /usr/local/etc/php/conf.d/docker-php-ext-ldap.ini, /usr/local/etc/php/conf.d/docker-php-ext-memcached.ini, /usr/local/etc/php/conf.d/docker-php-ext-mysqli.ini, /usr/local/etc/php/conf.d/docker-php-ext-opcache.ini, /usr/local/etc/php/conf.d/docker-php-ext-pcntl.ini, /usr/local/etc/php/conf.d/docker-php-ext-pdo_mysql.ini, /usr/local/etc/php/conf.d/docker-php-ext-pdo_pgsql.ini, /usr/local/etc/php/conf.d/docker-php-ext-pgsql.ini, /usr/local/etc/php/conf.d/docker-php-ext-shmop.ini, /usr/local/etc/php/conf.d/docker-php-ext-soap.ini, /usr/local/etc/php/conf.d/docker-php-ext-sockets.ini, /usr/local/etc/php/conf.d/docker-php-ext-sodium.ini, /usr/local/etc/php/conf.d/docker-

En este paso utilicé el navegador Chrome (con cualquiera funciona) y coloqué la ip de mi droplet en la barra de direcciones. El archivo dentro del repositorio estaba hecho para mostrar información del sistema.

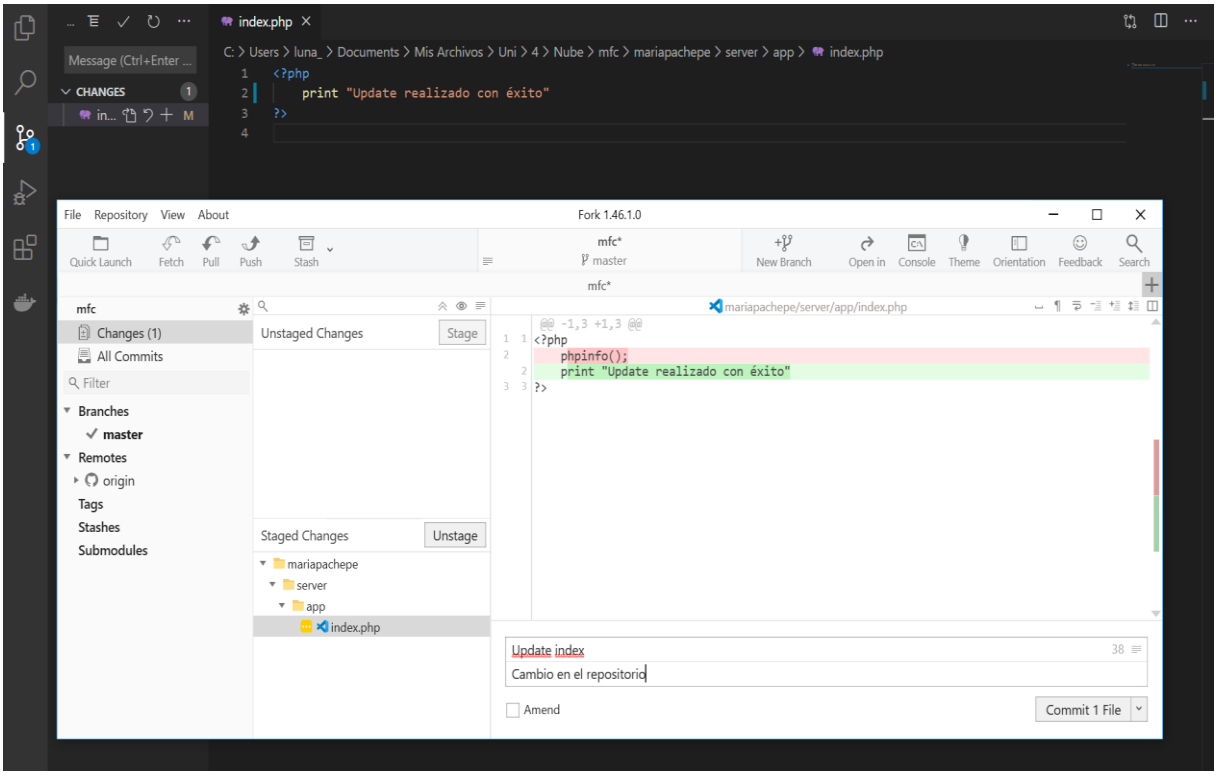
HeidiSQL Acceder a la BD



Para acceder a la base de datos dentro del droplet utilicé HeidiSQL. Los datos para ingresar son los que yo misma especifiqué en el archivo docker-compose de mi contenedor.



Fork Update al repositorio



Para realizar la modificación al repositorio, me apoyé en dos aplicaciones: Visual Studio Code y Fork. Utilicé Code para modificar el archivo index.php, y luego, con Fork, hice un commit y push al repositorio en la nube.

Sólo se realiza un pull desde el droplet y se recarga la página.

```
root@acastillo:/home/mfc# git pull origin master
From https://github.com/TasiaCode2/mfc
 * branch            master     -> FETCH_HEAD
Already up to date.
root@acastillo:/home/mfc#
root@acastillo:/home/mfc#
root@acastillo:/home/mfc# git pull origin master
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 6 (delta 1), reused 6 (delta 1), pack-reused 0
Unpacking objects: 100% (6/6), done.
From https://github.com/TasiaCode2/mfc
 * branch            master     -> FETCH_HEAD
   2ed5579..c7bafaf  master     -> origin/master
Updating 2ed5579..c7bafaf
Fast-forward
 mariapachepe/server/app/index.php | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
root@acastillo:/home/mfc# git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
root@acastillo:/home/mfc#
```