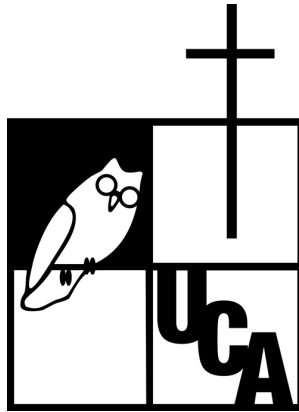


Universidad Centroamericana
José Simeón Cañas



digidoro

Despliegue de API

Alumna

Fernanda Camila Vásquez Meléndez

00065521

Fecha de entrega

viernes 30 de junio de 2023



Guía para el despliegue de API

En este documento se ofrece una guía detallada sobre el proceso de publicación de una aplicación Node JS. La implementación se llevará a cabo en servidores de DigitalOcean utilizando un dominio personalizado. Además, se realizarán configuraciones fundamentales del firewall mediante el uso de un proxy de Nginx, junto con la implementación de encriptación HTTPS.

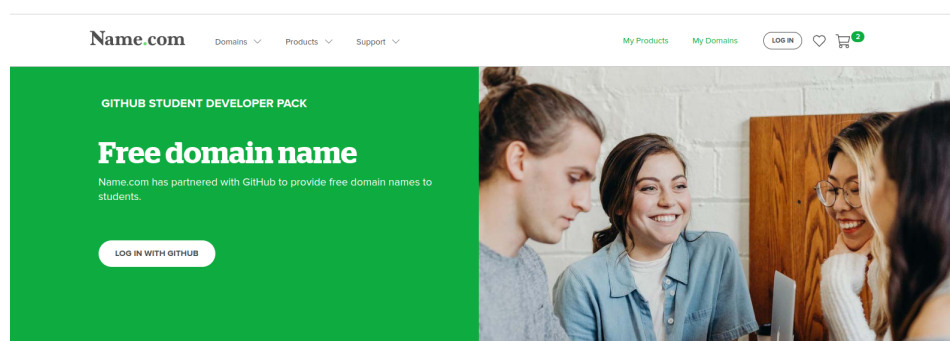
Preparando nuestro entorno

Compra de dominio

Utilizaremos alguna de las tres opciones para la compra de dominios que incluye el paquete de desarrollador para estudiantes asociado a nuestra cuenta institucional de la UCA.

 About Namecheap Affordable registration, hosting, and domain management Benefit 1 year domain name registration on the .me TLD. Get access by connecting your GitHub account on Namecheap > Get help at Namecheap support Benefit 1 SSL certificate free for 1 year. Request your offer code to get access > Get help at Namecheap support	Name.com About Name.com Domains, Google Workspace, WordPress, DigitalOcean hosting, and more. Benefit Build your online presence with a free .LIVE domain name, free privacy protection, and a free SSL certificate. Get access by connecting your GitHub account on Name.com > Get help at Name.com support	 About .TECH A powerful domain extension to convey that you belong to the technology industry. Benefit One standard .TECH domain free for 1 year and 2 free email accounts with 100 MB free storage Get access by connecting your GitHub account on .TECH > Get help at .TECH support
---	--	---

Seleccionamos Name.com y nos registramos con nuestra cuenta de Github para hacer valer nuestros beneficios.



The screenshot shows the Name.com website with a green banner for the "GITHUB STUDENT DEVELOPER PACK". The banner text reads: "Free domain name", "Name.com has partnered with GitHub to provide free domain names to students.", and a button "LOG IN WITH GITHUB". The website header includes "Name.com", navigation links for "Domains", "Products", and "Support", and user options for "My Products", "My Domains", "LOG IN", a heart icon, and a shopping cart icon with a "2" badge. The background of the banner features a photo of three young women smiling and looking at a laptop.

Buscamos nuestro dominio deseado y lo agregamos al carrito de compras. Procedemos al checkout y comenzamos a realizar el pago de nuestro dominio.

Carrito > Iniciar sesión > Pago > Confirmation

ARTÍCULOS	DURACIÓN	PRECIO	RESUMEN DEL PEDIDO (2 artículos)
galacticshop.studio Registro Seguridad avanzada + Privacidad <small> ☑ Whois Privacy: evita las llamadas no deseadas y oculta la información personal del directorio público WHOIS ☑ Bloqueo Plus de dominio: evita las transferencias o los cambios no autorizados. ☑ Certificado SSL: protege tu sitio web. </small>	1 año	\$29.99 GRATIS FREE <small>Renews at \$4.99/yr</small>	Subtotal: \$33.98 Ahorro: - \$33.98 (Ahorras 100 %) Total: \$0.00 SIGUIENTE PASO: INICIO DE SESIÓN >
ADDITIONAL OFFERINGS <div> galacticshop.studio for \$0 with Google Workspace Business Starter Yearly 36 US\$ </div> <div> 30 DAY FREE TRIAL Titan Email <small> ☑ 10 GB account ☑ Cifrado de datos ☑ Built in Calendar and Contacts ☑ Apps for iOS and Android </small> </div>			

[VACIAR CARRITO](#)

CÓDIGO DE PROMOCIÓN

Llenamos los datos necesarios para el pago.

Create a Name.com Account

Already have a Name.com account? [Log in](#)

DATOS DE LA CUENTA

ESCOGER UN NOMBRE DE USUARIO

CREAR UNA CONTRASEÑA

☑ Extensión mínima de 7 caracteres
 ☑ Debe tener 1 número/símbolo.
 ☑ Debe tener 1 letra.

DATOS DE CONTACTO

NOMBRE

APELLIDO

ORGANIZACIÓN

LINEA DE DIRECCIÓN 1

LINEA DE DIRECCIÓN 2

CIUDAD

ESTADO/PROVINCIA

CÓDIGO POSTAL

PAÍS

Select a Country

☐ Acepto el [Acuerdo de registro](#) y la [Política de privacidad](#)

☒ Envíeme la información más reciente sobre servicios y promociones de Name.com (puedes darte de baja en cualquier momento).

[CREAR MI CUENTA](#)

ADDITIONAL OFFERINGS

galacticshop.studio for \$0
 with Google Workspace Business Starter Yearly
 \$36

30 DAY FREE TRIAL
Titan Email


 ☑ 10 GB account ☑ Data encryption ☑ Built in Calendar and Contacts ☑ Apps for iOS and Android

Porcedemos a registrar nuestro metodo de pago y completaremos nuestra orden.






PAYMENT OPTIONS

NAME ON CARD

NICKNAME

☐

Google Pay

☒
New Credit/Debit Card

CARD NUMBER

EXPIRY DATE

mm/yy

CVV/CVC

This credit card will be saved as your account default. Any product on auto-renewal will renew with this default

BILLING INFORMATION

Edit

Fernanda Vásquez

Universidad Centroamericana José Simeón Cañas

San Salvador, El Sañvador

San Salvador, San Salvador 01101 SV

+503.73931458

ORDER SUMMARY (2 Items)

Subtotal: \$33.98

Savings: - \$33.98
(You save 100%)

Total: \$0.00

☒
I agree to the [Registration Agreement](#) and [Privacy Policy](#)

☒
I have read and acknowledge the [Additional Registry Requirements](#)

Verificamos nuestra información de contacto.

ENTER VERIFICATION CODE

Please check your email for your verification code and enter it below. Sometimes it can take a minute or two for our email to make it across the internet, remember to check your spam folder. Contact information won't be updated until the changes have been verified.

VERIFICATION CODE

9021F1

VERIFYING:

Fernanda Vásquez
Universidad Centroamericana José Simeón Cañas
00065221@uca.edu.sv
+503.73931458
San Salvador, El Sañvador
San Salvador, San Salvador 01101

CONTACT INFORMATION VALIDATION

You have unverified contact information associated with this account. ICANN requires that contact information be validated within 15 days of any registration or the domain name will be disabled. Please verify your contact information below.

Verify my contact information via:

☒ Email

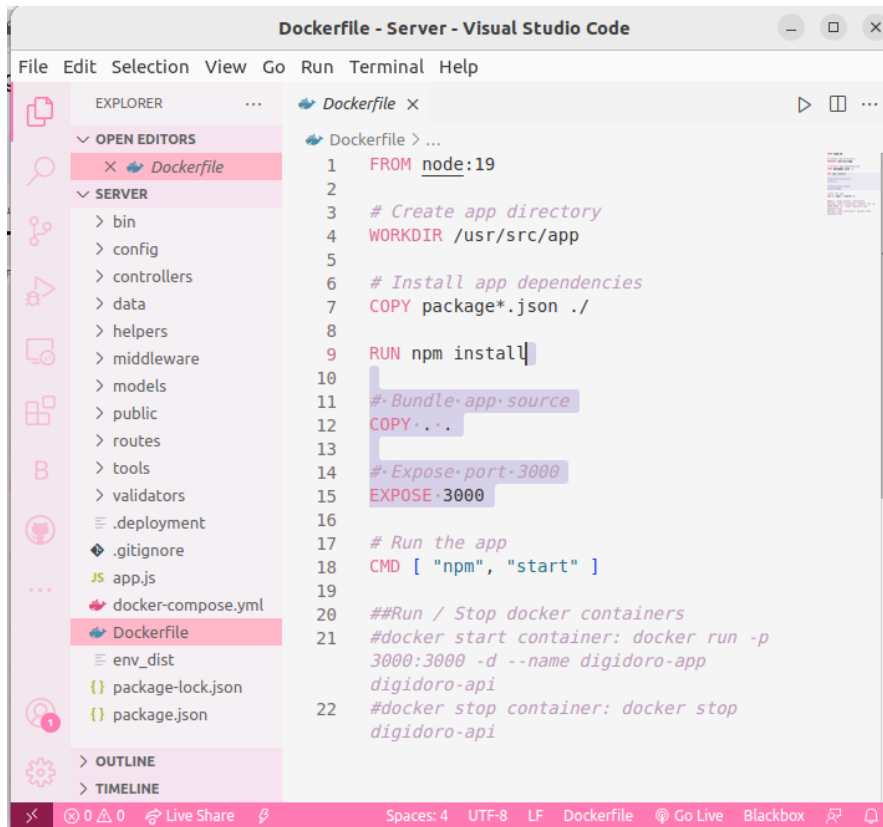
Configuración de entorno para app Node.js

Crear archivo Dockerfile

Este archivo se utiliza para configurar la construcción de imágenes de contenedores en Docker. Este archivo opera mediante instrucciones y comandos para copiar archivos, paquetes y gestionar variables de entorno de nuestra aplicación.

En esta guía, haremos uso de los siguientes comandos:

1. **FROM:** Indica la imagen base que se utilizará para trabajar en nuestro entorno. Aquí se especifica que se trabajara con la version 19 de Node.
2. **WORKDIR:** Define el directorio de trabajo en el cual se ejecutarán los comandos siguientes del archivo Dockerfile.
3. **COPY:** Copia paquetes o archivos desde nuestro equipo al contenedor.
4. **RUN:** Ejecuta los comandos indicados dentro del contenedor. Aquí se copian las dependencias y se ejecuta el comando `npm install` para crear un entorno adecuado para nuestra aplicación Node.js.
5. **COPY:** Este segundo comando COPY se utiliza para copiar el contenido del directorio, como archivos fuente, a un directorio dentro del contenedor.
6. **EXPOSE:** Establece el puerto que utilizará nuestro contenedor.
7. **CMD:** Define el comando predeterminado al iniciar el contenedor de la imagen. En este caso, deseamos que al iniciar el contenedor se ejecute "npm run start", que es el comando para iniciar nuestra aplicación Node.js con Express.



```

1 FROM node:19
2
3 # Create app directory
4 WORKDIR /usr/src/app
5
6 # Install app dependencies
7 COPY package*.json ./
8
9 RUN npm install
10
11 # Bundle app source
12 COPY . .
13
14 # Expose port 3000
15 EXPOSE 3000
16
17 # Run the app
18 CMD [ "npm", "start" ]
19
20 ##Run / Stop docker containers
21 #docker start container: docker run -p
22 3000:3000 -d --name digidoro-app
23 digidoro-api
24 #docker stop container: docker stop
25 digidoro-api

```

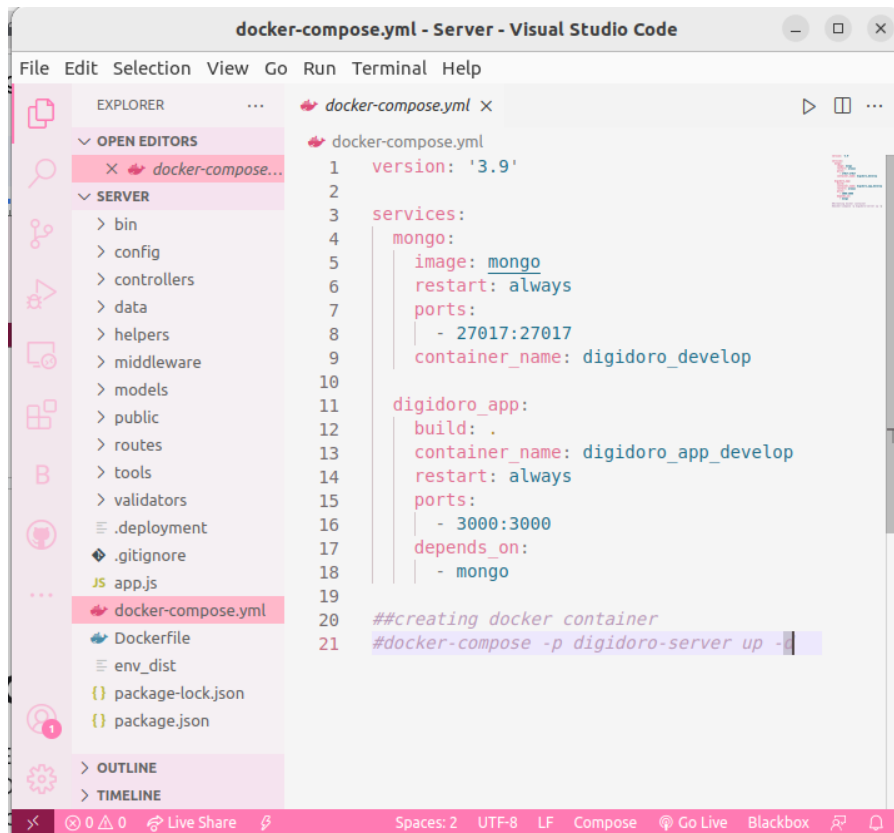
Crear archivo docker-compose.yml

Este archivo nos permite gestionar múltiples servicios dentro del entorno de Docker, lo que facilita la administración de aplicaciones multicontenedor. De lo contrario, tendríamos que utilizar una imagen y contenedores separados para cada componente de nuestra aplicación, como la base de datos y la API.

Haremos uso de los siguientes comandos:

8. **version:** Indica la versión de docker-compose que estaremos utilizando.
9. **services:** Define los servicios que forman parte de la aplicación. En nuestro caso, incluimos el servicio "mongo".
10. **image:** Especifica la imagen que se utilizará para el servicio agregado anteriormente, en este caso, el servicio "mongo".

11. **container_name:** Indica el nombre del contenedor para nuestro servicio. En nuestro caso, utilizamos "digidoro_develop" para la imagen de MongoDB y "digidoro_app_develop" para la imagen del servicio de construcción (aplicación).
12. **restart:** Indica que el contenedor se reiniciará automáticamente en caso de fallos. Esto se aplica tanto al servicio "mongo" como al servicio de construcción de la aplicación.
13. **ports:** Establece los puertos para el contenedor. En el caso del servicio "mongo", se utiliza el puerto 27017 del host y el puerto 27017 del contenedor de MongoDB. Para el servicio de construcción, se utiliza el puerto 3000 del host y el puerto 3000 del contenedor.
14. **depends_on:** Especifica las dependencias de un servicio con respecto a otro. En este caso, se indica que el servicio "mongo" se iniciará antes del servicio "digidoro_app" para evitar problemas de conexión.



```
docker-compose.yml - Server - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
X docker-compose...
SERVER
  > bin
  > config
  > controllers
  > data
  > helpers
  > middleware
  > models
  > public
  > routes
  > tools
  > validators
  > .deployment
  > .gitignore
  > app.js
  > docker-compose.yml
  > Dockerfile
  > env_dist
  > package-lock.json
  > package.json
  > OUTLINE
  > TIMELINE
docker-compose.yml
1 version: '3.9'
2
3 services:
4   mongo:
5     image: mongo
6     restart: always
7     ports:
8       - 27017:27017
9     container_name: digidoro_develop
10
11   digidoro_app:
12     build: .
13     container_name: digidoro_app_develop
14     restart: always
15     ports:
16       - 3000:3000
17     depends_on:
18       - mongo
19
20 ##creating docker container
21 #docker-compose -p digidoro-server up -d
```

Creando Droplet

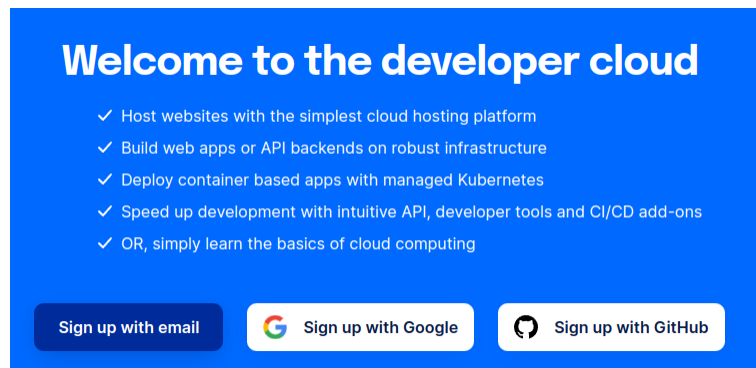
Creando y configurando Droplets en DigitalOcean

DigitalOcean es una reconocida empresa proveedora de servicios de infraestructuras en la nube. Su plataforma permite a los desarrolladores implementar, administrar y escalar sus aplicaciones en la nube de manera eficiente y sencilla.

Un enfoque distintivo de DigitalOcean es su metodología de trabajo basada en **droplets**, que son máquinas virtuales creadas y configuradas específicamente para satisfacer las necesidades de una aplicación determinada. Esta flexibilidad y adaptabilidad brinda a los usuarios la capacidad de personalizar y optimizar su entorno de desarrollo de acuerdo con sus requisitos específicos.

Paso 1: Crear usuario e iniciar sesión en DigitalOcean

1. Seguiremos el [link oficial](#) de DigitalOcean y nos registraremos usando nuestra cuenta GitHub vinculada a nuestra cuenta institucional de la UCA.



2. Si ya estamos registrados, iniciaremos sesión y completaremos la información que se nos es solicitada.

Log in to your account

email*

password*

Log In

Sign In with Google

Sign In with GitHub

[Forgot Password?](#)

What is your role or business type?

Hobbyist or Student

What do you plan to build on DigitalOcean?

A web or mobile application

What is your monthly spend on cloud infrastructure across cloud platforms? (Provide an estimate)

\$0 - \$50

How many employees work at your company?

2-9

Submit

3. De manera automática se generará un nuevo proyecto llamado first-project al iniciar sesión en nuestra cuenta.

Search by resource name or public IP (Ctrl+B)

Create

My Team
Estimated costs: \$0.00

first-project DEFAULT

Update your project information under Settings

Move Resources

Resources Activity Settings

Welcome to DigitalOcean!

When you build on DigitalOcean, you can have full control of your infrastructure (with products like Droplets and reserved IPs) or let us handle the infrastructure for you (with products like App Platform, load balancers, and managed databases).

Spin up a Droplet

Droplets are virtual machines that anyone can setup in seconds.

Deploy your web app

Build, deploy, and scale apps quickly using a simple, fully-managed solution.

LaunchKit BETA

Create a production-ready, scalable infrastructure with multiple resources.

Getting Started

Not sure where to start? Let us help you understand where to begin.

Paso 2: Configuración de proyecto

Entraremos a settings del proyecto y editaremos los campos:

1. **Name:** digidoro.
2. **Add a description:** digidoro is the perfect app for anyone looking to stay focused, manage their time, and achieve their goals. Try it out today and see how digidoro improve your I
3. **What's it for?** Class project
4. **Environment:** Production
5. Damos click en guardar.

Resources

Activity

Settings

Project details

Name

Enter name

digidoro

✓

Add a description

Enter description

digidoro is the perfect app for anyone looking to stay focused, manage their time, and achieve their goals. Tr

What's it for?

Environment

Class project / Educational purposes

* ▼

Production

▼

☒ This is my default project

Resources created without a designated project will be added to your default

Save

Cancel

Paso 3: Creación de droplet


En el apartado de droplets, se selecciona la opción "Crear un nuevo Droplet". Luego, se elige la región más cercana, en este caso, San Francisco, que es el datacenter más cercano disponible. Esta elección se


realiza con el objetivo de minimizar la cantidad de saltos que las peticiones deben realizar, optimizando así la latencia y el rendimiento de la aplicación.


Create ^


?


Esti


 **Droplets**
Create cloud servers


 **Kubernetes**
Create Kubernetes clusters


 **Apps**
Deploy your code

 **Functions**
Create Cloud Functions


 **Databases**
Create database clusters


 San Francisco


 London

 Bangalore

OS Marketplace Custom images

 Ubuntu


 Fedora

 Debian

Version
22.10 x64

Choose Size
Droplet Type
SHARED CPU
Basic
(Plan selected)
General Purpose

CPU options

 **Regular**
Disk type: SSD

\$4/mo
\$0.006/hour

512 MB / 1 CPU
10 GB SSD Disk
500 GB transfer

\$6/mo
\$0.009/hour

1 GB / 1 CPU
25 GB SSD Disk
1000 GB transfer

```

feri@feri-System-Product-Name: ~/Escritorio
feri@feri-System-Product-Name:~/Escritorio$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/feri/.ssh/id_rsa): virtual_ubuntu_dig
idoro
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in virtual_ubuntu_digidoro
Your public key has been saved in virtual_ubuntu_digidoro.pub
The key fingerprint is:
SHA256:fROU/JFpyVo13bZOS7nevUgG7j7S7DZyvNxY+RegUI0 feri@feri-System-Product-Name
The key's randomart image is:
+---[RSA 3072]-----+
|      .+o  *+|
|      Eo.O  =|
|      .  = oo|
|      o  .o.=|
|      S o.+ = o|
|      .o...=|
|      +.  =.  +|
|      ooO*  o.+|
|      BB+o  oo|
+-----[SHA256]-----+

```

```

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCmbvqIVJGS5mJAIQ2mfU4g1kQEBpT72oPgaTdrO
TrwPDxCH4W0vSmPkCYw4IHvfVpLp4cq8f5qw1CgZPqWaslUpD4Wgn3uybhrBajMm35ZqWMYP0j9qkm
stPaGwM00ymILGEzMLMoZNNYzgxC1oTkC2qBAP3XE+TCnkfqDHdhOJddpdhExyBxQi/j9Rr4Kbv0z7
0Lotv9BdaNKko30CXG32I149eJMyimxCC75aSsZShmjGctiNRWipZqMTsG7CB9by+xAYRAAcBexjNE
+pQisiRqR0GkLtLTxDaczEBQ+pNXMO1A0mKgCW1+pqUiI07wOTL9vVqogOkY2/TVTx3sn/osW/d0iv
FfvgkZjBjZE5yfGPjhK3U/fMH6d4M8Wueyj2o0OKj5pg/gmQ5ka8Dm0BGUr4pR8Nz0XxsOA9L275xO
fMY2qMPj/qUAPlo6Tsvz/Pub8EVsl7b/8BB3IFESTfUz50fitwJxvMz2YV7c7g6DHaluWVmN2U5o+W
0u0XU= feri@feri-System-Product-Name
feri@feri-System-Product-Name:~/Escritorio$ ^C

```

Add public SSH key

Copy your public SSH key and paste it in the space below. For instructions on how, follow the steps on the right.

SSH key content

NYzgxClOtKc2qBAP3XE+TCnkfqDHdhOJddpdhExyBxQi/j9Rr4Kbv0z70Lotv
9BdaNKko30CXG32I149eJMyimxCC75aSsZShmjGctiNRWipZqMTsG7CB9by
+xAYRAAcBexjNE+pQisiRqR0GkLtLTxDaczEBQ+pNXMO1A0mKgCW1+pqUiI0
7wOTL9vVqogOkY2/TVTx3sn/osW/d0ivFfvgkZjBjZE5yfGPjhK3U/fMH6d4M8
Wueyj2o0OKj5pg/gmQ5ka8Dm0BGUr4pR8Nz0XxsOA9L275xOfMY2qMPj/
qUAPlo6Tsvz/Pub8EVsl7b/8BB3IFESTfUz50fitwJxvMz2YV7c7g6DHaluWVm
N2U5o+W0u0XU= feri@feri-System-Product-Name

Name

virtual_ubuntu_digidoro

Add SSH Key

Enter passphrase (empty for no
passphrase):
Enter same passphrase again:

This will generate two files, by default called
`id_rsa` and `id_rsa.pub`. Next, add this
public key.

Add the public key

Copy and paste the contents of the `.pub` file,
typically `id_rsa.pub`, into the SSH key
content field on the left.

cat ~/.ssh/id_rsa.pub

Copy

For more detailed guidance, see [How to
Add SSH Keys to Droplets](#)

We recommend these options



Add improved metrics monitoring and alerting (free)

Collect and graph expanded system-level metrics, track performance, and set up alerts instantly within the control panel.

Finalize Details

Quantity

Deploy multiple Droplets with the same configuration.

—

1 Droplet

+

Hostname

Give your Droplets an identifying name you will remember them by.

virtual-ubuntu-digidoro-app

Tags

digidoro GalacticStudio deploy API Class-Project Type tags here

Project

digidoro

\$4.00/month

\$0.006/hour

[CREATE VIA COMMAND LINE](#)

Create Droplet



digidoro DEFAULT

Class project / Educational purposes / Production / digidoro is the perfect app for anyone looking to stay focused, ...

→ Move Resources

Resources Activity Settings

DROPLETS (1)

		virtual-ubuntu-digidoro-app	159.223.193.49	digidoro + 4			
---	---	-----------------------------	----------------	--------------	---	---	---

Configuración de espacio virtual

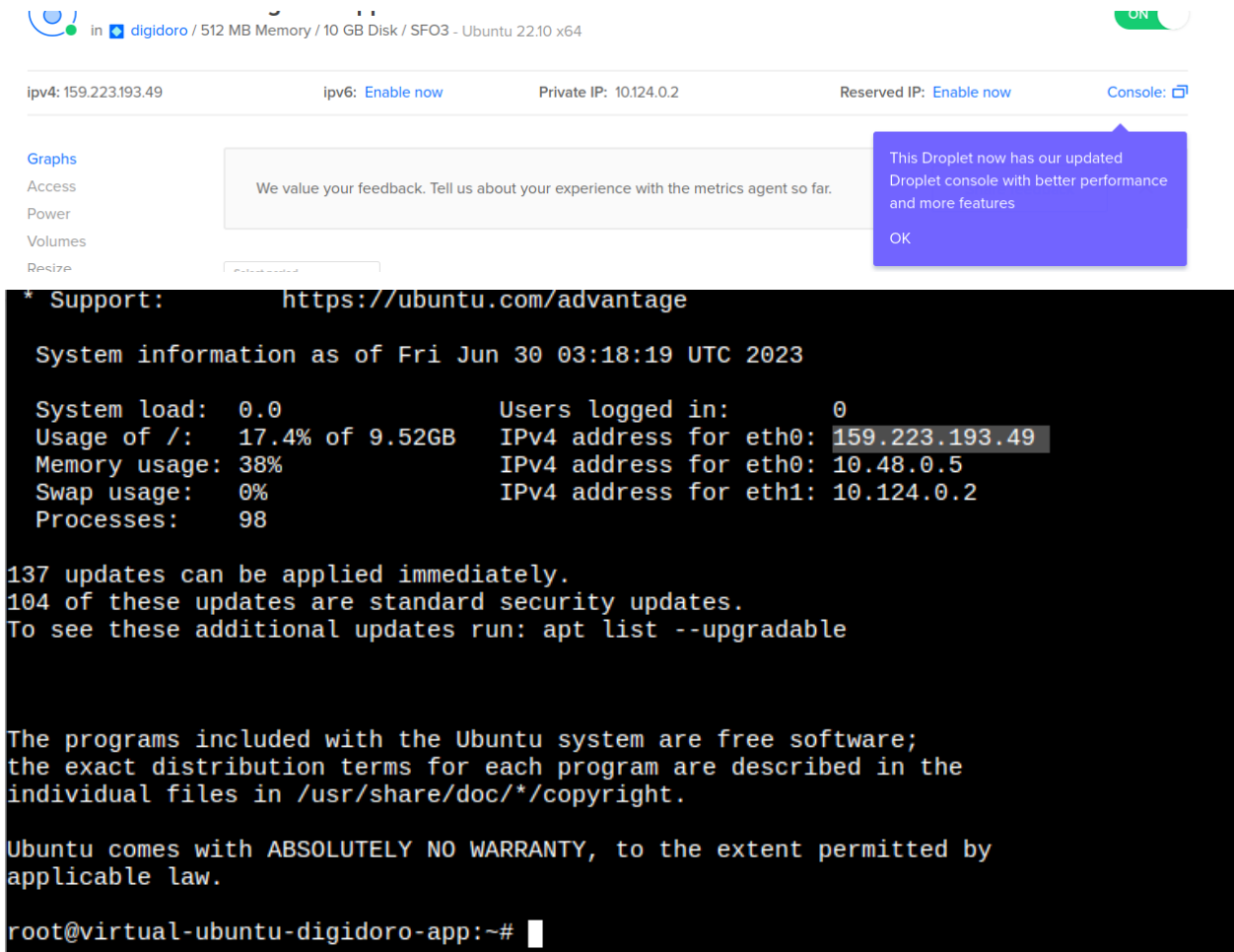
Una vez creado el droplet, podemos comenzar la configuración de nuestro espacio virtual. Para ello, iniciaremos sesión como root utilizando el siguiente comando: **root@server_ip_address**.

Donde "server_ip_address" representa la dirección IP del droplet que hemos creado. Al ejecutar este comando, se nos solicitará la contraseña de root para acceder al espacio virtual y realizar las configuraciones necesarias.

Paso 4: Crear nuevo usuario

Una vez iniciada la sesión como root en el droplet, se procede a crear un nuevo usuario para evitar realizar todas las tareas desde la cuenta de root. En este caso, se crea un usuario llamado "digidoro_admin". La creación de usuarios adicionales ayuda a mejorar la seguridad y la administración de permisos y tareas.

Abrimos nuestra consola.



The screenshot shows the Digidoro console interface. At the top, it displays the droplet's status as 'ON' and its specifications: 512 MB Memory / 10 GB Disk / SFO3 - Ubuntu 22.10 x64. Below this, there are links to enable IPv6, Private IP, and Reserved IP, along with a 'Console' button. A sidebar on the left lists various management options: Graphs, Access, Power, Volumes, and Resize. A central message box states: 'We value your feedback. Tell us about your experience with the metrics agent so far.' A blue notification bubble on the right says: 'This Droplet now has our updated Droplet console with better performance and more features' with an 'OK' button. The main terminal window shows the following output:

```
* Support: https://ubuntu.com/advantage

System information as of Fri Jun 30 03:18:19 UTC 2023

System load: 0.0          Users logged in: 0
Usage of /: 17.4% of 9.52GB IPv4 address for eth0: 159.223.193.49
Memory usage: 38%        IPv4 address for eth0: 10.48.0.5
Swap usage: 0%           IPv4 address for eth1: 10.124.0.2
Processes: 98

137 updates can be applied immediately.
104 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@virtual-ubuntu-digidoro-app:~#
```

Reemplaza "server_ip_address" con la dirección IP de tu droplet. En nuestro caso: 159.223.193.49.

Una vez que se ha iniciado sesión como root en el droplet, puedes utilizar el siguiente comando para crear un nuevo usuario: **adduser digidoro_admin**.

Se solicitará establecer una contraseña para el nuevo usuario y proporcionar información adicional, como el nombre completo y detalles de contacto. Después de ingresar la información requerida, se habrá creado exitosamente el nuevo usuario "digidoro_admin".

```
Adding user `digidoro_admin' ...
Adding new group `digidoro_admin' (1000) ...
Adding new user `digidoro_admin' (1000) with group `digidoro_admin' ...
Creating home directory `/home/digidoro_admin' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for digidoro_admin
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
root@virtual-ubuntu-digidoro-app:~#
```

Una vez creado el usuario "digidoro_admin", se pueden asignar permisos para facilitar las tareas y permitirle realizarlas con privilegios de root. Para ello, se utiliza el comando "usermod -aG sudo user" con el fin de asignar privilegios de root al nuevo usuario.

De esta manera, el usuario "digidoro_admin" será agregado al grupo sudo, lo cual le permitirá ejecutar comandos con privilegios de root utilizando el comando sudo.

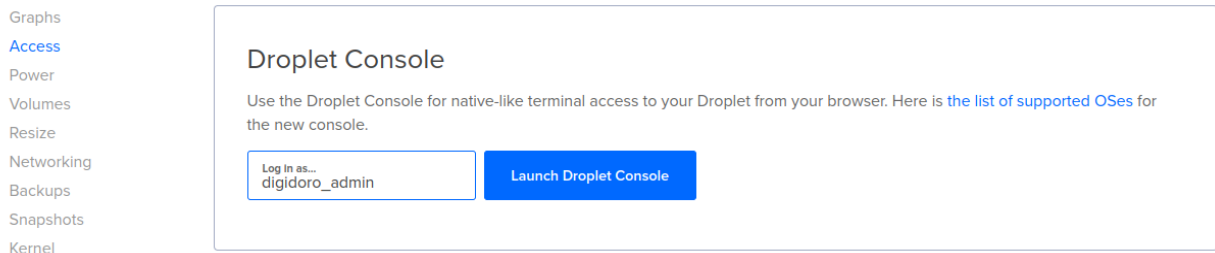
```
Is the information correct? [Y/n] y
root@virtual-ubuntu-digidoro-app:~# usermod -aG sudo digidoro_admin
root@virtual-ubuntu-digidoro-app:~#
```

A partir de este momento, el usuario "digidoro_admin" tendrá los privilegios necesarios para llevar a cabo tareas administrativas. Si no se requiere realizar más tareas como root, se puede cerrar la conexión SSH.

Con estos pasos, el usuario "digidoro_admin" podrá utilizar el comando sudo antes de ejecutar comandos, lo que le permitirá realizar tareas con privilegios de root. Esta configuración mejora la seguridad y facilita la realización eficiente de las tareas en el droplet.

Paso 5: Asignar SSH Key a digidoro_admin

En Acces, iniciamos sesión con el usuario digidoro_admin.



```
digidoro_admin@virtual-ubuntu-digidoro-app:~$
```

Es importante verificar que la estructura de directorios adecuada esté presente en el entorno de usuario y que el archivo "authorized_keys" exista. Si no está presente, es necesario crearlos. A continuación, se detallan los pasos para realizar esta verificación y, en caso necesario, crear los directorios y archivos:

```
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

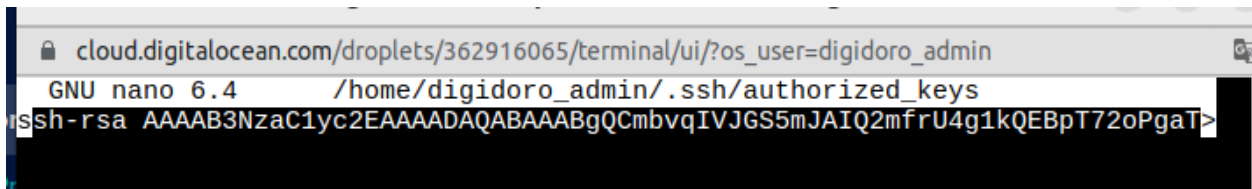
digidoro_admin@virtual-ubuntu-digidoro-app:~$ ls -a
.  .bash_logout  .cache  .profile
.. .bashrc      .cloud-locale-test.skip  .ssh
digidoro_admin@virtual-ubuntu-digidoro-app:~$ cd .ssh
digidoro_admin@virtual-ubuntu-digidoro-app:~/.ssh$ ls
authorized_keys
digidoro_admin@virtual-ubuntu-digidoro-app:~/.ssh$
```

Una vez dentro del droplet, se puede utilizar el editor de texto "nano" para abrir el archivo "authorized_keys" en la carpeta ".ssh/":

A continuación, se puede reemplazar el contenido del archivo "authorized_keys" con la clave SSH deseada. En este caso, se copiará y pegará la clave SSH llamada "virtual_ubuntu_digidoro" en el archivo utilizando el editor "nano".

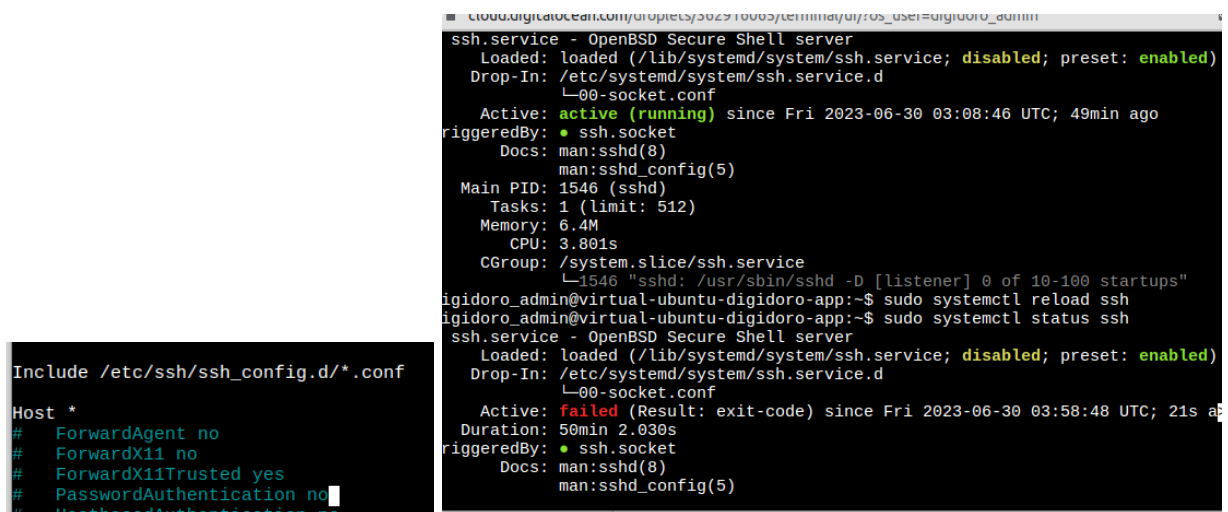
Después de reemplazar el contenido, se puede guardar y cerrar el archivo en el editor "nano".

Al seguir estos pasos, se reescribirá el archivo "authorized_keys" con la clave SSH deseada, en este caso, la clave llamada "virtual_ubuntu_digidoro". Esto permitirá la autenticación mediante dicha clave al acceder al droplet a través de SSH.



```
cloud.digitalocean.com/droplets/362916065/terminal/ui/?os_user=digidoro_admin
GNU nano 6.4 /home/digidoro_admin/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCmbvqIVJGS5mJAIQ2mfrU4g1kQEBpT72oPgaT
```

Por último, se cargan los cambios realizados en el sistema mediante el comando "sudo systemctl reload ssh ", lo cual permite que los cambios en el archivo "authorized_keys" surtan efecto. A continuación, se puede acceder al usuario mediante SSH para verificar que la autenticación con la nueva clave SSH funcione correctamente.



```
cloud.digitalocean.com/droplets/362916065/terminal/ui/?os_user=digidoro_admin
Include /etc/ssh/ssh_config.d/*.conf
Host *
# ForwardAgent no
# ForwardX11 no
# ForwardX11Trusted yes
# PasswordAuthentication no
# HostbasedAuthentication no

cloud.digitalocean.com/droplets/362916065/terminal/ui/?os_user=digidoro_admin
ssh.service - OpenBSD Secure Shell server
Loaded: loaded (/lib/systemd/system/ssh.service; disabled; preset: enabled)
Drop-In: /etc/systemd/system/ssh.service.d
└─00-socket.conf
Active: active (running) since Fri 2023-06-30 03:08:46 UTC; 49min ago
TriggeredBy: ● ssh.socket
Docs: man:sshd(8)
man:sshd_config(5)
Main PID: 1546 (sshd)
Tasks: 1 (limit: 512)
Memory: 6.4M
CPU: 3.801s
CGroup: /system.slice/ssh.service
└─1546 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
digidoro_admin@virtual-ubuntu-digidoro-app:~$ sudo systemctl reload ssh
digidoro_admin@virtual-ubuntu-digidoro-app:~$ sudo systemctl status ssh
ssh.service - OpenBSD Secure Shell server
Loaded: loaded (/lib/systemd/system/ssh.service; disabled; preset: enabled)
Drop-In: /etc/systemd/system/ssh.service.d
└─00-socket.conf
Active: failed (Result: exit-code) since Fri 2023-06-30 03:58:48 UTC; 21s ago
Duration: 50min 2.030s
TriggeredBy: ● ssh.socket
Docs: man:sshd(8)
man:sshd_config(5)
```

Configuración básica de Firewall e instalación de Nginx

La configuración del Firewall permite mantener un control y restricción del tráfico de red que ingresa y sale de la aplicación, brindando una capa adicional de seguridad. Por otro lado, Nginx actúa como un servidor web y un proxy que permite enrutar y balancear las solicitudes entrantes hacia los diferentes componentes de la aplicación. Esta combinación de Firewall y Nginx contribuye a garantizar la seguridad y el rendimiento de la aplicación en entornos de producción.

Paso 5: Instalación de Nginx

Se ejecutarán los siguientes comandos para actualizar los paquetes locales y realizar la instalación de Nginx en el entorno:

Para actualizar los paquetes locales, se utiliza el comando: **sudo apt update**. Este comando descarga la lista actualizada de paquetes disponibles en los repositorios y sincroniza la información local con los servidores de paquetes.

Una vez actualizados los paquetes, se puede proceder a la instalación de Nginx mediante el siguiente comando: **sudo apt install nginx**.

Este comando instala el servidor web Nginx en el entorno. Durante el proceso de instalación, se descargan los paquetes necesarios y se realiza la configuración inicial de Nginx. Al ejecutar estos comandos, se actualizarán los paquetes locales y se instalará Nginx en el entorno, lo cual permitirá utilizarlo como servidor web y proxy en el entorno de despliegue de la aplicación.

```
Last login: Fri Jun 30 03:49:07 2023 from 162.243.188.66
root@virtual-ubuntu-digidoro-app:~# sudo apt update
Get:1 http://mirrors.digitalocean.com/ubuntu kinetic InRelease [267 kB]
Hit:2 https://repos.insights.digitalocean.com/apt/do-agent main InRelease
Hit:3 http://mirrors.digitalocean.com/ubuntu kinetic-updates InRelease
Hit:4 http://mirrors.digitalocean.com/ubuntu kinetic-backports InRelease
Hit:5 http://security.ubuntu.com/ubuntu kinetic-security InRelease
Hit:6 https://repos.droplet.digitalocean.com/apt/droplet-agent main InRelease
Fetched 267 kB in 1s (461 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
133 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@virtual-ubuntu-digidoro-app:~# sudo apt install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.22.0-1ubuntu1.1).
0 upgraded, 0 newly installed, 0 to remove and 133 not upgraded.
```

Paso 6: Configuración de Firewall

Verificaremos el status con `sudo ufw status` y observaremos las opciones aplicables con `sudo ufw app list`. Luego habilitamos las opciones deseadas con `sudo ufw allow "option"` y por último habilitamos el firewall con `sudo ufw enable`.

```
root@virtual-ubuntu-digidoro-app:~# sudo ufw status
Status: inactive
root@virtual-ubuntu-digidoro-app:~# sudo ufw app list
Available applications:
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
  OpenSSH
root@virtual-ubuntu-digidoro-app:~# sudo ufw allow "Nginx Full"
Rules updated
Rules updated (v6)
root@virtual-ubuntu-digidoro-app:~# sudo ufw enable.
```

```
root@virtual-ubuntu-digidoro-app:~# sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@virtual-ubuntu-digidoro-app:~# █
```


Configurando DNS

Con el dominio que compramos anteriormente [galacticshop.studio](#) configuraremos nuestro DNS.

Paso 7: Añadir dominio a DigitalOcean

Networking

[Domains](#)
[Reserved IPs](#)
[Load Balancers](#)
[VPC](#)
[Firewalls](#)
[PTR records](#)



Looks like there are no domains here.

You can add one easily, though. Enter a domain below.

HOSTNAME

WILL DIRECT TO

TTL (SECONDS)

DNS records

Type	Hostname	Value	TTL (seconds)	
NS	www.galacticshop.studio	Will direct to 159.223.193.49	Enter TTL 1800	Save
NS	www.galacticshop.studio	directs to ns2.digitalocean.com.	1800	More ▾
NS	www.galacticshop.studio	directs to ns3.digitalocean.com.	1800	More ▾

Paso 7: Configurar nameserver en Name.com

USE DEFAULT NAMESERVERS
NAMESEVER TEMPLATES
DELETE ALL

NAMESEVER	CREATED	ACTIONS
ns1.name.com	2023-06-29	EDIT DELETE
ns2.digitalocean.com	2023-06-29	EDIT DELETE
ns3.digitalocean.com	2023-06-29	EDIT DELETE
ns4.name.com	2023-06-29	EDIT DELETE

ADD NAMESERVER

Changes Saved Successfully!
Update applied successfully

Configuración de HTTPS/SSL

Paso 8: Instalación de Certbot y configuración

Se utilizará la herramienta Certbot, que simplifica el proceso de obtención de certificados SSL/TLS gratuitos de Let's Encrypt. El objetivo es habilitar HTTPS/SSL en el sitio web a través de Nginx, utilizando los certificados SSL válidos proporcionados por Let's Encrypt.

Certbot es una herramienta ampliamente utilizada y confiable que automatiza gran parte del proceso de obtención, renovación y configuración de certificados SSL/TLS. Al utilizar Certbot con Let's Encrypt, se pueden obtener certificados válidos y confiables de forma gratuita, lo

que permite establecer una conexión segura y cifrada mediante HTTPS en el sitio web. La configuración de Certbot y la integración con Nginx permitirán habilitar HTTPS en el sitio web, asegurando las comunicaciones y brindando confianza a los usuarios al mostrar que el sitio web utiliza un certificado SSL válido. Esto es especialmente importante para proteger la privacidad de los datos transmitidos entre el servidor y los usuarios, así como para evitar advertencias de seguridad en los navegadores.

Para una correcta instalación de Certbot en nuestro entorno seguir los siguientes pasos de la [guia oficial](#) de DigitalOcean.

```
Last login: Fri Jun 30 04:08:30 UTC 2023 from 162.243.190.66 on pts/0
root@virtual-ubuntu-digidoro-app:~# sudo snap install core; sudo snap refresh co
re
core 16-2.59.4 from Canonical✓ installed
snap "core" has no updates available
root@virtual-ubuntu-digidoro-app:~# sudo apt remove certbot
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package 'certbot' is not installed, so not removed
0 upgraded, 0 newly installed, 0 to remove and 133 not upgraded.
root@virtual-ubuntu-digidoro-app:~# sudo snap install --classic certbot
certbot 2.6.0 from Certbot Project (certbot-eff✓) installed
root@virtual-ubuntu-digidoro-app:~# sudo ln -s /snap/bin/certbot /usr/bin/certbo
t
root@virtual-ubuntu-digidoro-app:~#
```

Paso 9: Configuración de Nginx para habilitar HTTPS.

Gracias a la herramienta Certbot, podemos configurar de manera automática el certificado SSL. Solo necesitamos proporcionar la dirección correcta correspondiente a nuestro dominio.

Editamos el archivo de configuración de Nginx con el siguiente comando: **sudo nano /etc/nginx/sites-available/default** y escribimos el dominio al que estamos pidiendo certificación.

```
# Don't use them in a production server!
#
# include snippets/snakeoil.conf;

root /var/www/html;

# Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;

server_name galacticshop.studio www.galacticshop.studio;

location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    try_files $uri $uri/ =404;
}

# pass PHP scripts to FastCGI server
#
#location ~ \.php$ {
#    include snippets/fastcgi-php.conf;
```

^G Help
^X Exit

^O Write Out
^R Read File

^W Where Is
^_ Replace

^K Cut
^U Paste

^T Execute
^J Justify

^C Location
^_ Go To Line

Verificamos que los cambios sean correctos haciendo uso de **sudo nginx -t**

Por último, recargamos nuestro servicio nginx con **sudo systemctl reload nginx**.

```
root@virtual-ubuntu-digidoro-app:~# sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
root@virtual-ubuntu-digidoro-app:~# sudo systemctl reload nginx
sudo: systemctl reload: command not found
root@virtual-ubuntu-digidoro-app:~# sudo systemctl reload nginx
root@virtual-ubuntu-digidoro-app:~# _
```

Paso 10: Obtener SSL con Certbot

Haremos uso de un plugin para nginx que nos permitirá reconfigurar y recargar Nginx de manera automática. Con el comando **sudo certbot --nginx -d server_name.com -d www.server_name.com**.

En este caso **--nginx** es el plugin que nos ayuda a ejecutar de manera sencilla las configuraciones, **-d** especifica el dominio al que queremos certificar.

Se nos solicitarán permisos e información personal. Por ultimo, utilizamos **sudo certbot renew --dry-run**

Deploy

Instalación de Docker

Además al estar haciendo uso de dockerfiles y docker-compose debemos de tener los servicios de docker instalados en nuestro entorno virtual. Haremos uso de la guía de instalación oficial de docker.

```
root@virtual-ubuntu-digidoro-app:~# sudo apt update
Get:1 http://mirrors.digitalocean.com/ubuntu kinetic InRelease [267 kB]
Hit:2 https://repos.insights.digitalocean.com/apt/do-agent main InRelease
Hit:3 http://mirrors.digitalocean.com/ubuntu kinetic-updates InRelease
Hit:4 http://mirrors.digitalocean.com/ubuntu kinetic-backports InRelease
Get:5 http://security.ubuntu.com/ubuntu kinetic-security InRelease [109 kB]
Hit:6 https://repos-droplet.digitalocean.com/apt/droplet-agent main InRelease
Fetched 376 kB in 7s (54.0 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
133 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@virtual-ubuntu-digidoro-app:~# sudo apt install apt-transport-https ca-certificates curl software-properties-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apt-transport-https is already the newest version (2.5.3).
ca-certificates is already the newest version (20230311ubuntu0.22.10.1).
curl is already the newest version (7.85.0-1ubuntu0.5).
software-properties-common is already the newest version (0.99.27).
software-properties-common set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 133 not upgraded.
root@virtual-ubuntu-digidoro-app:~# ~_
```



```

500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:20.10.3~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:20.10.2~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:20.10.1~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:20.10.0~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:19.03.15~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:19.03.14~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:19.03.13~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:19.03.11~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:19.03.10~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
5:19.03.9~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
root@virtual-ubuntu-digidoro-app:~# ~

```

```

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@virtual-ubuntu-digidoro-app:~# sudo systemctl status docker
• docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; preset: enabl
   Active: active (running) since Fri 2023-06-30 05:45:18 UTC; 12s ago
   TriggeredBy: • docker.socket
   Docs: https://docs.docker.com
   Main PID: 3332 (dockerd)
   Tasks: 7
   Memory: 41.5M
   CPU: 395ms
   CGroup: /system.slice/docker.service
           └─3332 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/cont
Jun 30 05:45:17 virtual-ubuntu-digidoro-app systemd[1]: Starting Docker Applica
Jun 30 05:45:17 virtual-ubuntu-digidoro-app dockerd[3332]: time="2023-06-30T05:
Jun 30 05:45:17 virtual-ubuntu-digidoro-app dockerd[3332]: time="2023-06-30T05:
Jun 30 05:45:18 virtual-ubuntu-digidoro-app dockerd[3332]: time="2023-06-30T05:
Jun 30 05:45:18 virtual-ubuntu-digidoro-app dockerd[3332]: time="2023-06-30T05:
Jun 30 05:45:18 virtual-ubuntu-digidoro-app dockerd[3332]: time="2023-06-30T05:
Jun 30 05:45:18 virtual-ubuntu-digidoro-app dockerd[3332]: time="2023-06-30T05:
Jun 30 05:45:18 virtual-ubuntu-digidoro-app dockerd[3332]: time="2023-06-30T05:
Jun 30 05:45:18 virtual-ubuntu-digidoro-app systemd[1]: Started Docker Applicat
lines 1-21/21 (END)

```

Despliegue de Landing Page.

Si tenemos nuestro landing.html en un repositorio de github fácilmente podemos clonar nuestra landing dentro de nuestro entorno virtual. Nosotros lo colocaremos en /var/www/html.

Para hacer visible nuestro index.html, debemos cambiar la dirección de nuestro index.html que está por defecto en Nginx, apuntando en nuestro caso a la dirección de nuestra landing page.

Accedemos al documento de configuración y sobre escribimos con nano al documento default, añadiendo en root nuestro nuevo directorio y en Index el nombre de nuestro nuevo html.

Guardamos y verificamos que los cambios sean correctos con nginx -, si los cambios son correctos reiniciamos nuestro servicio.

```
root@virtual-ubuntu-Digidoro-app:/var/www/html# sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
root@virtual-ubuntu-Digidoro-app:/var/www/html# sudo systemctl restart nginx.service
```

```
GNU nano 6.4 /etc/nginx/sites-available/default
# Read up on ssl_ciphers to ensure a secure configuration.
# See: https://bugs.debian.org/765782
#
# Self signed certs generated by the ssl-cert package
# Don't use them in a production server!
#
# include snippets/snakeoil.conf;

#root /var/www/html;
root /var/www/html/landing/digidoro-landing-page/landing

# Add index.php to the list if you are using PHP
#index index.html index.htm index.nginx-debian.html;
index index.html

server_name galacticshop.studio www.galacticshop.studio;

location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
```

