

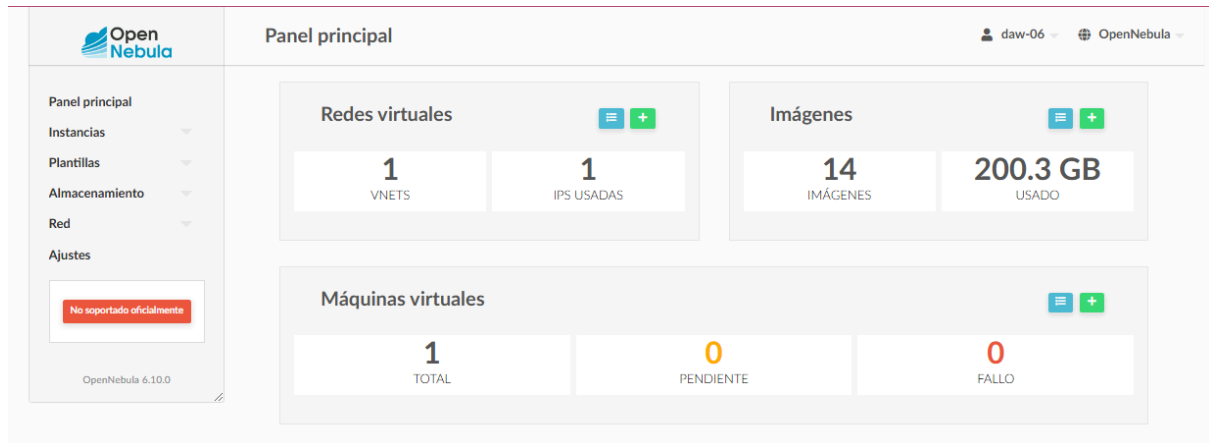
MANUAL DE DESPLIEGUE



Manual explicativo correspondiente al despliegue de nuestra aplicación mediante OpenNebula.

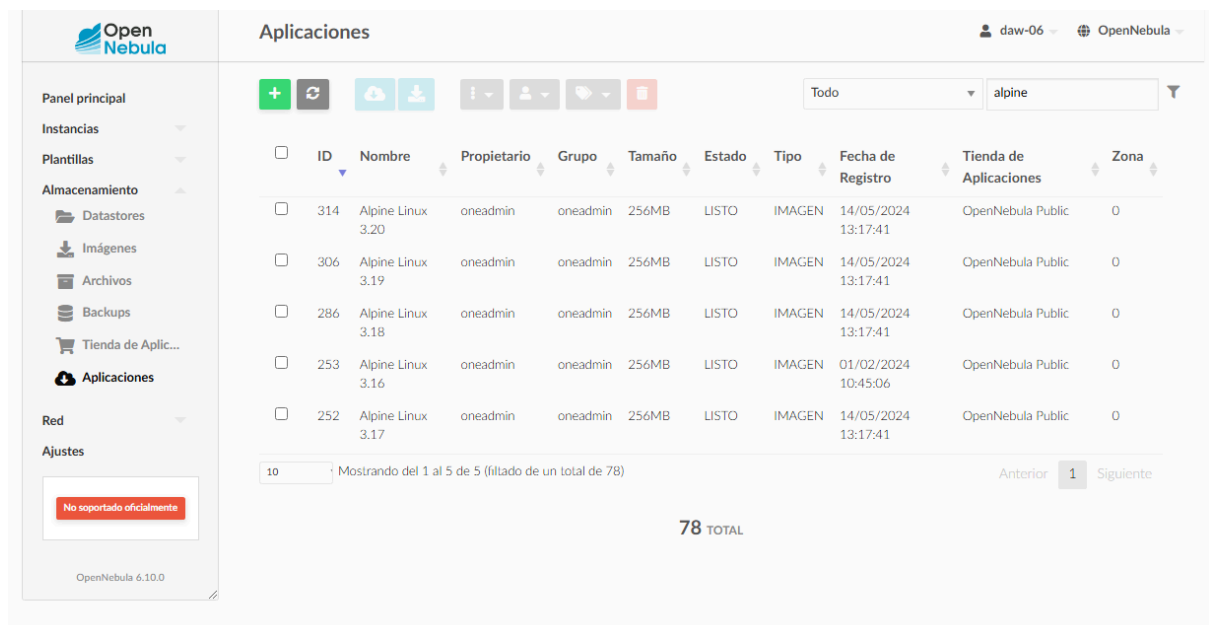
Puesta en marcha

Lo primero que haremos es entrar en OpenNebula para crear una máquina virtual con una de las plantillas que están ya creadas.



A continuación explicaremos los pasos a seguir con capturas de pantalla:

Primero entraremos en “Almacenamiento > Aplicaciones” y ahí descargamos la máquina virtual que queramos utilizar. Seleccionamos “Alpine Linux” y le damos click en la nube:



<input type="checkbox"/>	ID	Nombre	Propietario	Grupo	Tamaño	Estado	Tipo	Fecha de Registro	Tienda de Aplicaciones	Zona
<input checked="" type="checkbox"/>	314	Alpine Linux 3.20	oneadmin	oneadmin	256MB	LISTO	IMAGEN	14/05/2024 13:17:41	OpenNebula Public	0
<input type="checkbox"/>	306	Alpine Linux 3.19	oneadmin	oneadmin	256MB	LISTO	IMAGEN	14/05/2024 13:17:41	OpenNebula Public	0
<input type="checkbox"/>	286	Alpine Linux 3.18	oneadmin	oneadmin	256MB	LISTO	IMAGEN	14/05/2024 13:17:41	OpenNebula Public	0
<input type="checkbox"/>	253	Alpine Linux 3.16	oneadmin	oneadmin	256MB	LISTO	IMAGEN	01/02/2024 10:45:06	OpenNebula Public	0
<input type="checkbox"/>	252	Alpine Linux 3.17	oneadmin	oneadmin	256MB	LISTO	IMAGEN	14/05/2024 13:17:41	OpenNebula Public	0

Tendremos una plantilla con la máquina virtual.

Panel principal Instancias Plantillas MVs Servicios Routers Virtual Grupos MV Almacenamiento Red Ajustes <div>No soportado oficialmente</div> OpenNebula 6.10.0	<div> + Actualizar Instanciar Clonar </div> <div> <input type="checkbox"/> ID Nombre Propietario Grupo Fecha de registro </div> <div> <input type="checkbox"/> 409 Alpine Linux 3.20 daw-06 daw-06 26/01/2025 12:00:56 <input type="checkbox"/> 406 2A_Debian 12 bisasi oneadmin 15/01/2025 16:00:16 <input checked="" type="checkbox"/> 358 Debian 12 Desktop - 20241201 widemos oneadmin 01/12/2024 12:04:17 <input type="checkbox"/> 263 Ubuntu 22.04 - 20241124 widemos oneadmin 24/11/2024 19:08:37 <input type="checkbox"/> 262 Debian 12 - 20241124 widemos oneadmin 24/11/2024 18:51:05 <input type="checkbox"/> 258 2A_Windows Server 2022 - 20230221 bisasi oneadmin 22/10/2024 11:16:04 <input type="checkbox"/> 255 Kali 2024.3 jokin oneadmin 03/10/2024 12:40:44 <input type="checkbox"/> 162 Windows 10 Pro jokin oneadmin 05/10/2023 09:36:01 <input type="checkbox"/> 160 Ubuntu Desktop 22.04.3 jokin oneadmin 29/09/2023 09:47:29 <input type="checkbox"/> 122 Windows Server 2022 - 20230221 widemos oneadmin 21/03/2023 00:13:24 </div> <div> 10 Mostrando del 1 al 10 de 10 </div> <div> Anterior 1 Siguiente </div> <div>10 TOTAL</div>
---	---

La seleccionamos y elegimos el *datastore* “default”, le damos a guardar.

←
Descargar

Nombre
Alpine Linux 3.20

Nombre de Plantilla MV
Alpine Linux 3.20

☐ Do not import/export associated VM templates/images

Seleccione el Datastore para almacenar el recurso

Ha seleccionado este datastore: default

ID
Nombre
Propietario
Grupo
Capacidad
Clúster

1
default
oneadmin
oneadmin
331.7GB / 1.7TB (19%)
0

10 Mostrando del 1 al 1 de 1

Anterior 1 Siguiente

Entramos en nuestra plantilla.

Open Nebula

Panel principal

Instancias

Plantillas

MVs

Servicios

Routers Virtual

Grupos MV

Almacenamiento

Plantillas de MV

+

Actualizar

Instanciar

Clonar

Buscar

<input type="checkbox"/>	ID	Nombre	Propietario	Grupo	Fecha de registro
<input type="checkbox"/>	409	Alpine Linux 3.20	daw-06	daw-06	26/01/2025 12:00:56
<input type="checkbox"/>	406	2A_ Debian 12	bisasi	oneadmin	15/01/2025 16:00:16
<input type="checkbox"/>	358	Debian 12 Desktop - 20241201	widemos	oneadmin	01/12/2024 12:04:17
<input type="checkbox"/>	263	Ubuntu 22.04 - 20241124	widemos	oneadmin	24/11/2024 19:08:37
<input type="checkbox"/>	262	Debian 12 - 20241124	widemos	oneadmin	24/11/2024 18:51:05
<input type="checkbox"/>	258	2A_ Windows Server 2022 - 20230221	bisasi	oneadmin	22/10/2024 11:16:04

Open Nebula

Panel principal

Instancias

Plantillas

MVs

Servicios

Routers Virtual

Grupos MV

Almacenamiento

Red

Redes virtuales

Plantilla de Red

Modelo de Red

Grupos de segur...

Ajustes

No soportado oficialmente

OpenNebula 6.10.0

Plantilla de Máquina Virtual 409 Alpine Linux 3.20

←

↺

Actualizar


Instanciar

Clonar

Info

Plantilla

Información



ID409

NombreAlpine Linux 3.20

Tiempo de registro12:00:56 26/01/2025

Permisos


	Utilice	Administrar	Administrador
Propietario	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Grupo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Otros	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Propiedad

Propietario	daw-06
Grupo	daw-06

Le damos a “instanciar”.

Alpine Linux 3.20



Capacidad

Memoria128MB

CPU Físico0.5

CPU Virtual2

Discos

DISCO 0: Alpine Linux 3.20

1GB

Dev Dragons

Aritz Saiz, Lorena Arbé y Julen Corpas

Seleccionamos nuestra red y le damos a instanciar:

Red

Interfaz daw-06

Tipo de Interfaz

☐ Alias ?

Conexión RDP

☐ Activar ?

SSH connection

☐ Activar ?

Ha seleccionado las siguiente red: daw-06

ID	Nombre	Propietario	Grupo	Estado	Reserva	Clúster	Concesiones
1312	daw-06	daw-06	daw-06	READY	No	0	1 / 10

10 Mostrando del 1 al 1 de 1

Anterior 1 Siguiente

Forzar IPv4 ?

← Instanciar

☐ Instanciar como persistente ?

Nombre MV ?

alpine

Númer

3

Alpine Linux 3.20

Entramos en cualquiera de las máquinas que hemos creado con Alpine.

Panel principal

Instancias

MVs

Servicios

Routers Virtual

Plantillas

MVs

Servicios

ID	Nombre	Propietario	Grupo	Estado	Host	IPs
9073	alpine	daw-06	daw-06	BOOT	localhost	0: 172.20.228.63
9072	alpine	daw-06	daw-06	BOOT	localhost	0: 172.20.228.62
9071	alpine	daw-06	daw-06	BOOT	localhost	0: 172.20.228.61
8293	Debian	daw-06	daw-06	POWEROFF	localhost	0: 172.20.228.60

10 Mostrando del 1 al 4 de 4

Anterior 1 Siguiente

Panel principal

Instancias

MVs

Servicios

Routers Virtual

Plantillas

MVs

Servicios

Routers Virtual

Grupos MV

Almacenamiento

Red

Redes virtuales

Plantilla de Red

Modelo de Red

Grupos de segur...

Ajustes

No soportado oficialmente

OpenNebula 6.10.0

Info

Capacidad

Almacenamiento

Red


Instantáneas

Acciones

Configuración

Registro

Información



ID

9073

Nombre

alpine

Estado

ACTIVE

Estado LCM

RUNNING

Host

localhost

IP

0: 172.20.228.63

Hora de inicio

12:08:11 26/01/2025

ID del desplegado

281a6c3f-4606-40f9-af10-006f05a661bd

Reprogramar

no

Router Virtual

--

Permisos

Propietario

Utilice

Administrar

Administrador

Propiedad

Propietario

daw-06

Grupo

daw-06

Backups

Backup IDs

Atributos

LOGO

images/logos/linux.png

LXD_SECURITY_PRIVILEGED

true

Le damos a “Adjuntar interfaz de red”:

Info

Capacidad

Almacenamiento

Red

Instantáneas

Acciones

Configuración

Registro

ID	Red	IP	MAC	Dirección PCI	IPv6 ULA	Prefijo Global IPv6	Acciones
0	daw-06	172.20.228.63	02:00:ac:14:e4:3f	--	--	--	<div>Adjuntar Interfaz de Red</div>

Mostrando del 1 al 1 de 1

Anterior 1 Siguiente

Seleccionamos la nueva interfaz de red.

Adjuntar nueva interfaz de red

☐ RDP

☐ Attach as an alias

Ha seleccionado las siguiente red: daw-06

Buscar

ID	Nombre	Propietario	Grupo	Estado	Reserva	Clúster	Concesiones
1312	daw-06	daw-06	daw-06	READY	No	0	4 / 10

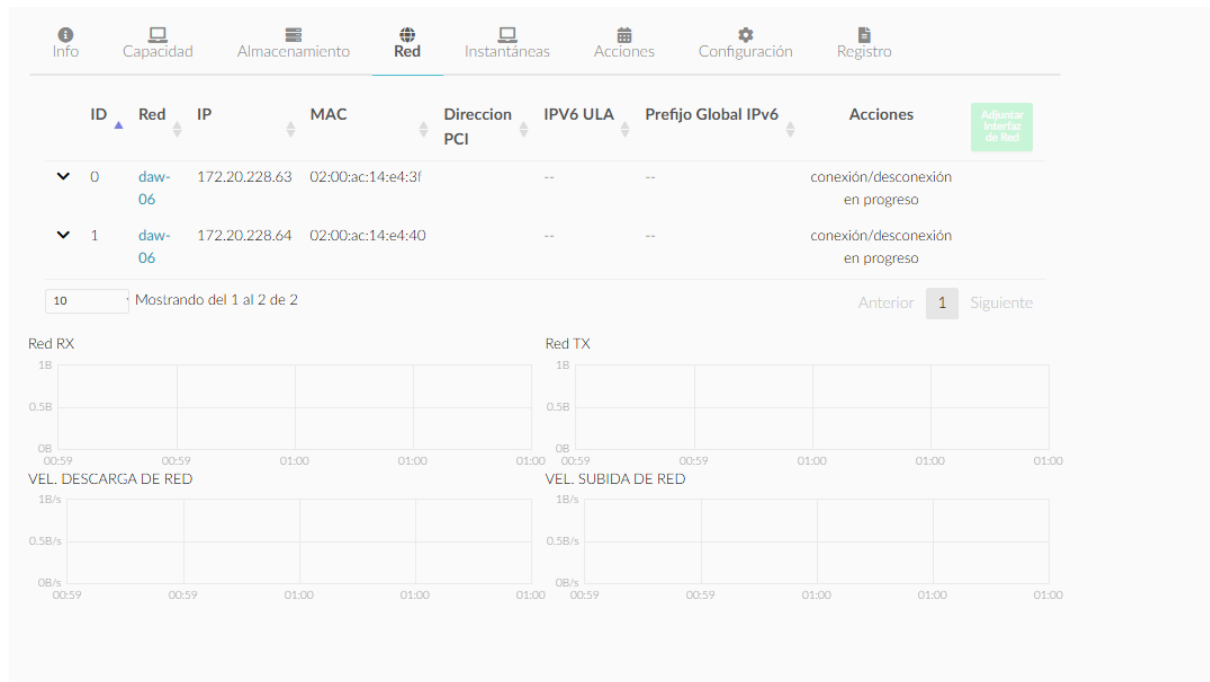
Mostrando del 1 al 1 de 1

Anterior 1 Siguiente

Dev Dragons

Aritz Saiz, Lorena Arbé y Julen Corpas

Ahora tendremos la interfaz de red adjuntada.




MVs
daw-06
OpenNebula

Panel principal
Instancias
Plantillas
Almacenamiento
Red
Ajustes

+
↺
🔒
▶
🔒
⏻
↺
📷
🗑️

<input type="checkbox"/>	ID	Nombre	Propietario	Grupo	Estado	Host	IPs	
<input checked="" type="checkbox"/>	9073	alpine	daw-06	daw-06	EJECUTANDO	localhost	0: 172.20.228.63	
<input checked="" type="checkbox"/>	9072	alpine	daw-06	daw-06	EJECUTANDO	localhost	0: 172.20.228.62	
<input checked="" type="checkbox"/>	9071	alpine	daw-06	daw-06	EJECUTANDO	localhost	0: 172.20.228.61	

Al darle a la papelera y a “Terminar” se borrarán las máquinas virtuales seleccionadas.

MVs

daw-06
 OpenNebula

Panel principal

Instancias

Plantillas

Almacenamiento

Red

Ajustes

+

↺

📁

▶

🔒

⏻

↺

🔄

👤

📄

<input type="checkbox"/>	ID	Nombre	Propietario	Grupo	Estado	Host	IPs
<input checked="" type="checkbox"/>	8293	Debian	daw-06	daw-06	POWEROFF	localhost	0: 172.20.228.60

Mostrando del 1 al 1 de 1

Anterior

1

Siguiente

1 TOTAL

0 ACTIVA

1 APAGADO

0 PENDIENTE

0 FALLO

No soportado oficialmente

Entramos en nuestra máquina virtual (*debian12*).

Panel principal

Instancias

Plantillas

MVs

Servicios

Routers Virtual

Grupos MV

Almacenamiento

Red

Ajustes

No soportado oficialmente

OpenNebula 6.10.0

+ ▾ ↺

Actualizar Instanciar Clonar 🔒 ▾ 👤 ▾ 🗑 ▾

Buscar 🔍

<input type="checkbox"/>	ID ▾	Nombre	Propietario	Grupo	Fecha de registro
<input type="checkbox"/>	409	Alpine Linux 3.20	daw-06	daw-06	26/01/2025 12:00:56
<input type="checkbox"/>	406	2A_Debian 12	bisasi	oneadmin	15/01/2025 16:00:16
<input type="checkbox"/>	358	Debian 12 Desktop - 20241201	widemos	oneadmin	01/12/2024 12:04:17
<input type="checkbox"/>	263	Ubuntu 22.04 - 20241124	widemos	oneadmin	24/11/2024 19:08:37
<input type="checkbox"/>	262	Debian 12 - 20241124	widemos	oneadmin	24/11/2024 18:51:05
<input type="checkbox"/>	258	2A_Windows Server 2022 - 20230221	bisasi	oneadmin	22/10/2024 11:16:04
<input type="checkbox"/>	255	Kali 2024.3	jokin	oneadmin	03/10/2024 12:40:44
<input type="checkbox"/>	162	Windows 10 Pro	jokin	oneadmin	05/10/2023 09:36:01
<input type="checkbox"/>	160	Ubuntu Desktop 22.04.3	jokin	oneadmin	29/09/2023 09:47:29
<input type="checkbox"/>	122	Windows Server 2022 - 20230221	widemos	oneadmin	21/03/2023 00:13:24

10 ▾

Mostrando del 1 al 10 de 10

Anterior 1 Siguiente

10 TOTAL

Seleccionamos nuestra red dentro de la máquina virtual y le damos a “Instanciar”.

^ 🏠 Red

^ Interfaz daw-06

Tipo de Interfaz

☐ Alias ?

Conexión RDP

☐ Activar ?

SSH connection

☐ Activar ?

Ha seleccionado las siguiente red: daw-06

ID ▾ Nombre

Propietario

Grupo

Estado

Reserva

Clúster

Concesiones

1312

daw-06

daw-06

daw-06

READY

No

0

1 / 10

10 ▾

Mostrando del 1 al 1 de 1

Anterior 1 Siguiente

Forzar IPv4: ?

Dentro de las instancias de máquinas virtuales veremos la nuestra.



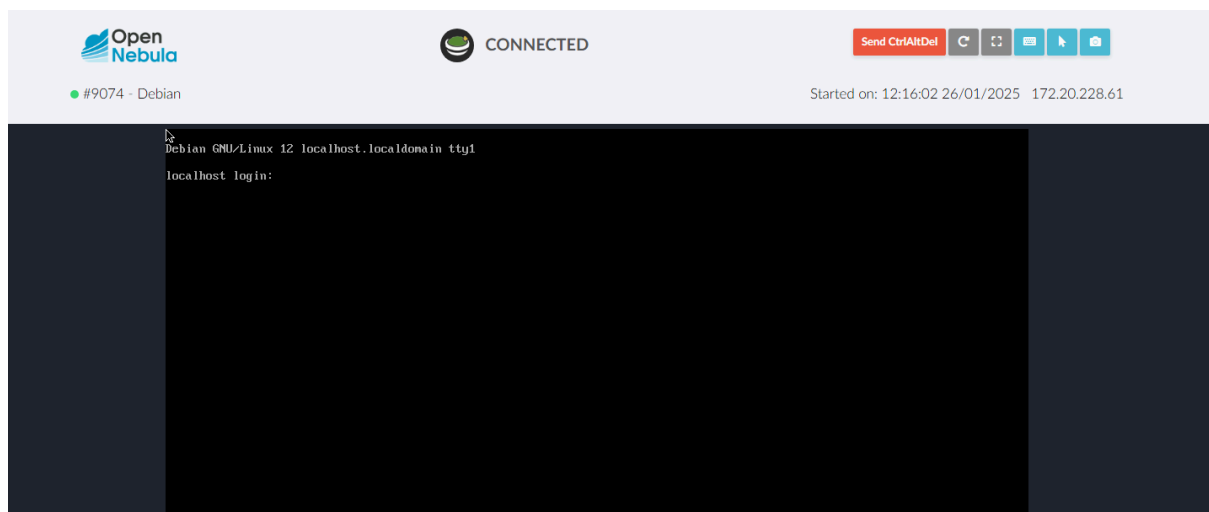
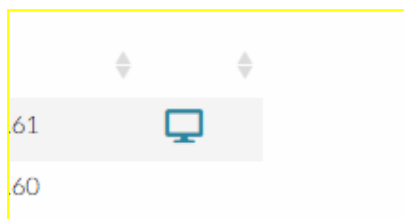
The screenshot shows the 'Instancias' (Instances) section of the OpenNebula web interface. On the left is a sidebar with navigation options: 'Panel principal', 'Instancias' (selected), 'Servicios', 'Routers Virtual', 'Plantillas', 'Almacenamiento', and 'Red'. The main area displays a table of virtual machines. Above the table is a toolbar with icons for adding, refreshing, locking, starting, stopping, restarting, cloning, and deleting. A search bar is in the top right. The table has columns for selection, ID, Name, Owner, Group, State, Host, and IPs. Two VMs are listed: ID 9074 (Debian, daw-06, BOOT state) and ID 8293 (Debian, daw-06, POWEROFF state). Below the table, it shows 'Mostrando del 1 al 2 de 2' and pagination links. At the bottom, a summary bar indicates: 2 TOTAL, 1 ACTIVA, 1 APAGADO, 0 PENDIENTE, 0 FALLO.

	ID	Nombre	Propietario	Grupo	Estado	Host	IPs
<input checked="" type="checkbox"/>	9074	Debian	daw-06	daw-06	BOOT	localhost	0: 172.20.228.61
<input type="checkbox"/>	8293	Debian	daw-06	daw-06	POWEROFF	localhost	0: 172.20.228.60

10 Mostrando del 1 al 2 de 2 Anterior 1 Siguiente

2 TOTAL 1 ACTIVA 1 APAGADO 0 PENDIENTE 0 FALLO

Si le damos click a la pantalla de la derecha nos llevará a la consola:



Dentro de la máquina vamos a configuración:

MV 9074 Debian EJECUTANDO
daw-06 OpenNebula

Info

Capacidad

Almacenamiento

Red

Instantáneas

Acciones

Configuración

Registro

Información		Permisos	Utilice	Administrar	Administrador
		Pietario	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
		Grupo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		Otros	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Propiedad					
ID	9074	Pietario	daw-06		
Nombre	Debian	Grupo	daw-06		
Estado	ACTIVE				

Le damos a “Actualizar Configuración”:

Info

Capacidad

Almacenamiento

Red

Instantáneas

Acciones

Configuración

Registro

Actualizar Configuración

BACKUPS

OS

ARCH

x86_64

FIRMWARE

FIRMWARE_SECURE

YES

UUID

6e4a5fe8-43d8-4fae-b67a-5e967726713f

GRAPHICS

LISTEN

0.0.0.0

PORT

14974

CONTEXT

DISK_ID

1

ETH0_DNS

192.168.10.1 1.1.1.1

SO & CPU

Entrada/Salida

Contexto

Etiquetas

Backups

Arrancar

Kernel

Ramdisk

Características

Modelo de CPU

Arquitectura

x86_64

Bus para discos SD

Tipo de Máquina


Dispositivo Principal

sda1

Orden de Arranque ?

☐

disk0




Debian 12 - 20241124

↑

↓

☐

nic0



daw-06

↑

↓

Parametros de arranque del Kernel

ro console=tty1

Ruta hacia el ejecutable que contiene el bootloader

Ponemos este comando (en una terminal de Windows para obtener la *clave pública*) y la ponemos en la configuración de la máquina.

```
C:\Users\Julen Corpas>type .ssh\id_rsa.pub
```

The screenshot shows the 'Configuración' (Configuration) window of a virtual machine. On the left, there is a sidebar with 'Archivos' (Files) and 'Variables personalizadas' (Custom variables). The main area is titled 'Configuración' and contains several settings. The 'Añadir contextualización de SSH' (Add SSH contextualization) checkbox is checked. Below it, the 'Clave ssh pública' (SSH public key) field is populated with the output of the Windows command: 'ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDzuy...'. The 'Script de inicio' (Startup script) field is empty. On the right side, there are additional checkboxes for 'Añadir' (Add), 'Añadir' (Add), and 'Inform' (Info).

La máquina pasará a estar encendida.

<input type="checkbox"/>	ID	Nombre	Propietario	Grupo	Estado	Host	IPs	
<input checked="" type="checkbox"/>	9074	Debian	daw-06	daw-06	HOTPLUG	localhost	0: 172.20.228.61	
<input type="checkbox"/>	8293	Debian	daw-06	daw-06	POWEROFF	localhost	0: 172.20.228.60	

10 Mostrando del 1 al 2 de 2 Anterior 1 Siguiente

Después se pondrá en “EJECUTANDO”

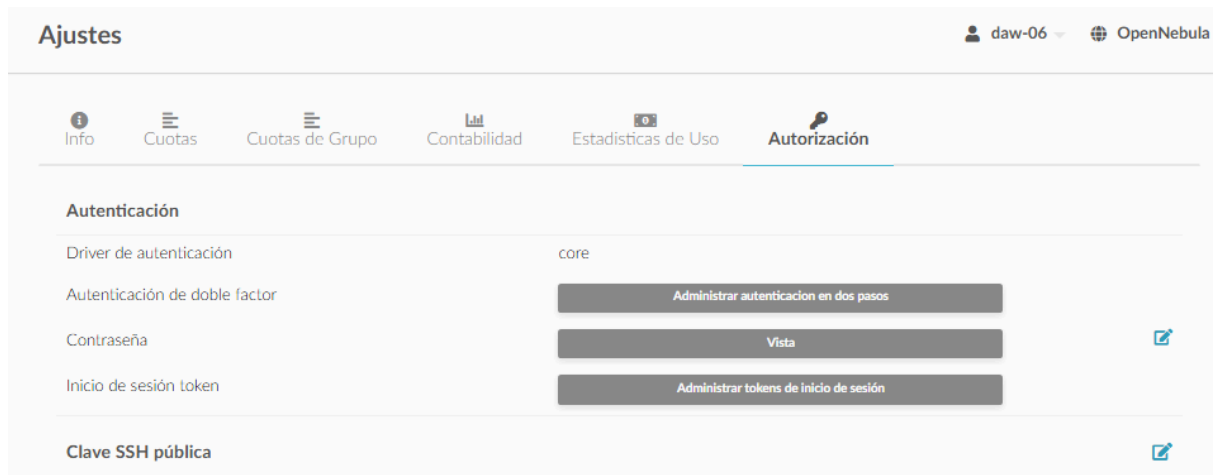
Nos conectamos mediante el comando **ssh root@172.20.228.43** (ip de la máquina virtual).

```
C:\Users\2gdaw04>ssh root@172.20.228.43
Linux localhost.localdomain 6.1.0-30-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.124-1 (2025-01-12) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb  4 09:39:14 2025 from 172.20.227.242
root@localhost:~#
```

Se puede cambiar la clave pública desde el perfil en “Autorización”:



Dentro del ssh se pondrían todos estos comandos:

```
root@localhost:~# apt update && apt upgrade -y
Get:1 file:/etc/apt/mirrors/debian.list Mirrorlist [30 B]
Get:5 file:/etc/apt/mirrors/debian-security.list Mirrorlist [39 B]
Hit:2 https://deb.debian.org/debian bookworm InRelease
Get:3 https://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:4 https://deb.debian.org/debian bookworm-backports InRelease [59.0 kB]
Get:6 https://deb.debian.org/debian-security bookworm-security InRelease [59.0 kB]
Get:7 https://deb.debian.org/debian-security bookworm-security/main amd64 Packages [59.0 kB]
Fetched 307 kB in 1s (30.7 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  linux-image-6.1.0-29-amd64
root@localhost:~# apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  linux-image-6.1.0-29-amd64
root@localhost:~# sudo apt-get update
* curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/debian/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/debian/gpg -o /etc/apt/keyrings/docker.asc
Get:1 file:/etc/apt/mirrors/debian.list Mirrorlist [30 B]
```

Y también los siguientes comandos:

```
sudo apt-get update
```

```
sudo apt-get install ca-certificates curl
```

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
sudo curl -fsSL https://download.docker.com/linux/debian/gpg -o  
/etc/apt/keyrings/docker.asc
```

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```

root@localhost:~# echo \
> "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings,
bian \
> $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
> sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
root@localhost:~# sudo apt-get update
Get:1 file:/etc/apt/mirrors/debian.list Mirrorlist [30 B]

```

echo \

"deb [arch=\$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]

https://download.docker.com/linux/debian \

\$(. /etc/os-release && echo "\$VERSION_CODENAME") stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

```

root@localhost:~# sudo mkdir -p /etc/apt/keyrings
root@localhost:~# curl -fsSL https://download.docker.com/linux/debian/gpg | sudo tee /etc/apt/keyrings/docker.asc > /dev/null
root@localhost:~# sudo apt-get update
Get:1 file:/etc/apt/mirrors/debian.list Mirrorlist [30 B]
Get:5 file:/etc/apt/mirrors/debian-security.list Mirrorlist [39 B]

```

sudo mkdir -p /etc/apt/keyrings

curl -fsSL https://download.docker.com/linux/debian/gpg | sudo tee /etc/apt/keyrings/docker.asc > /dev/null

sudo apt-get update

Después iría este y cuando haga una pregunta hay que decir que sí:

```

root@localhost:~# sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done

```

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

Para comprobar que está instalado:

```

root@localhost:~# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@localhost:~# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
e6590344b1a5: Pull complete
Digest: sha256:e0b569a5163a5e6be84e210a2587e7d447e08f87a0e90798363fa44a0464a1e8
Status: Downloaded newer image for hello-world:latest
$
Hello from Docker!

```

sudo docker run hello-world

En nuestra página de Moodle, OpenNebula entra en *Github:blog* y clona el repositorio con **root**.

Aquí clonaríamos nuestro repositorio:

```
root@localhost:~# git clone -v https://github.com/Lorena-Arbe-Sanchez/Reto3_equipo2.git
fatal: destination path 'Reto3_equipo2' already exists and is not an empty directory.
root@localhost:~# ls
Reto3_equipo2
root@localhost:~# cd
. ssh/      Reto3_equipo2/
root@localhost:~# cd Reto3_equipo2
root@localhost:~/Reto3_equipo2# ls
Base_datos LICENSE README.md reto3_equipo2
root@localhost:~/Reto3_equipo2# apt install make
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
```

En nuestro repositorio cambiaremos el nombre de la carpeta que contiene la aplicación a “laravel” para que funcione el workspace.

```
root@localhost:~/Reto3_equipo2# mv reto3_equipo2 laravel
root@localhost:~/Reto3_equipo2# ls
Base_datos LICENSE README.md docker laravel
root@localhost:~/Reto3_equipo2#
```

Dentro de la carpeta “docker” copiaremos y modificaremos el archivo “.env” para que sea como en el proyecto.

```
root@localhost:~/Reto3_equipo2# cd docker
root@localhost:~/Reto3_equipo2/docker# ls
Makefile docker-compose.override.yml docker-compose.yml docs env.example nginx php
root@localhost:~/Reto3_equipo2/docker#
root@localhost:~/Reto3_equipo2/docker# cp env.example .env
root@localhost:~/Reto3_equipo2/docker# nano .env
```

```

GNU nano 7.2 .env
# Versiones de los contenedores
PHP_VERSION=8.2
ALPINE_VERSION=3.19
MARIADB_VERSION=10.11
REDIS_VERSION=7.2
PHPMYADMIN_VERSION=latest
PHPREDISADMIN_VERSION=latest
NGINX_VERSION=mainline
HTTPS_PORTAL_VERSION=1

# Variables de configuración
MYSQL_ROOT_PASSWORD=
MYSQL_DATABASE=reto3
MYSQL_USER=root
MYSQL_PASSWORD=

# Prefijo para los contenedores
COMPOSE_PROJECT_NAME=reto3

# Sitios web
HTTPS_PORTAL_SITES=blog.test -> http://nginx:80, phpmyadmin.blog.test -> http://phpmyadmin:80,
HTTPS_PORTAL_STAGE=local

```

Dentro de la carpeta docker haremos un **make build** para poder usar los comandos make.

```

root@localhost:~/Reto3_equipo2/docker# ls
Makefile docker-compose.override.yml docker-compose.yml docs env.example nginx php
root@localhost:~/Reto3_equipo2/docker# make
----
Blog
----

Opciones:
-----
build
start / stop / restart / stop-all
install
seed
workspace / workspace-root
test
redis-cli / redis-flush
stats
clean
-----

root@localhost:~/Reto3_equipo2/docker# make build
WARN[0000] /root/Reto3_equipo2/docker/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please
remove it to avoid potential confusion
[+] Building 0.9s (1/2)                                                                                               docker:default
=> [php internal] load build definition from Dockerfile                                                         0.0s
=> => transferring dockerfile: 2.56kB                                                                           0.0s
=> [php internal] load metadata for docker.io/library/php:8.2-fpm-alpine3.19                                   0.9s

[+] Building 185.0s (14/27)                                                                                       docker:default
=> => sha256:eda2dcd66a84fa63a604eb9eb09285f9be42854566c5dfd7fb3e8aed43c8b8ed 19.49kB / 19.49kB 2.1s
=> => extracting sha256:afe8bdc9352f0c00e7bfcf3195ff6477335848dd5f69a5484079f65d36107bc1 0.1s
=> => extracting sha256:ec9ba93938c1e6819122e0e35e838c04aa5f97e51790efa5407603ed5c290305 0.0s
=> => sha256:4f4fb700ef54461cfa02571ae0db9a0dc1e0cdb5577484a6d75e68dc38e8acc1 32B / 32B 2.1s
=> => extracting sha256:4ca5eee9757946dd708f4c1f3f3474daca840ae65a8506ce54f1bc33641e5dfc 0.6s
=> => sha256:87571102b5ac222b392f8e5b4b70062e450e16f6378d6ee25cbe0388fb30d690 9.18kB / 9.18kB 2.2s
=> => extracting sha256:b9ff907fbfc9569b57aa28531be17241b6abb439297602856069c6680aacbec4 0.0s
=> => extracting sha256:eda2dcd66a84fa63a604eb9eb09285f9be42854566c5dfd7fb3e8aed43c8b8ed 0.0s
=> => extracting sha256:4f4fb700ef54461cfa02571ae0db9a0dc1e0cdb5577484a6d75e68dc38e8acc1 0.0s
=> => extracting sha256:87571102b5ac222b392f8e5b4b70062e450e16f6378d6ee25cbe0388fb30d690 0.0s
=> [php internal] load build context                                                                           0.0s
=> => transferring context: 213B                                                                                  0.0s
=> [php 2/23] WORKDIR /var/www/html                                                                           0.4s
=> [php 3/23] RUN apk add --no-cache build-base linux-headers shadow nano curl                               10.6s
=> [php 4/23] RUN docker-php-ext-install pdo pdo_mysql && docker-php-ext-enable pdo_mysql                    18.9s

```

Haremos **make start** para que todos los contenedores funcionen correctamente y nos dará la url de la web.

```
root@localhost:~/blog/docker# make start
[+] Running 16/60
  ⚙️ mariadb [ ] Pulling
  ⚙️ phpmyadmin [ ] Pulling
  ⚙️ redis [#####] 16.82MB / 16.82MB Pulling
  ⚙️ mailpit [ ] Pulling
  ⚙️ https-portal [#####] Pulling

root@localhost:~/Reto3_equipo2/docker# make start
WARN[0000] /root/Reto3_equipo2/docker/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 7/7
  ⚙️ Container reto3-redis-1      Running      0.0s
  ⚙️ Container reto3-mailpit-1    Running      0.0s
  ⚙️ Container reto3-mariadb-1    Started      0.7s
  ⚙️ Container reto3-phpmyadmin-1 Running      0.0s
  ⚙️ Container reto3-nginx-1      Started      1.6s
  ⚙️ Container reto3-php-1        Started      0.9s
  ⚙️ Container reto3-https-portal-1 Running      0.0s

---
Blog
---

Sitios disponibles:
-----
[Sitio web] https://blog.test
[phpMyAdmin] https://phpmyadmin.blog.test
[Mailpit] https://mailpit.blog.test
-----
root@localhost:~/Reto3_equipo2/docker#
```

Al hacer el make workspace entraremos en la app y veremos todas sus carpetas:

```
root@localhost:~/Reto3_equipo2/docker# make workspace
www-data@php:laravel$ ls
README.md      composer.json  package-lock.json  public          tailwind.config.js
app            composer.lock  package.json       resources       tests
artisan        config         phpunit.xml        routes          vite.config.js
bootstrap     database      postcss.config.js  storage

www-data@php:laravel$
```

En la ruta que aparece en la captura crearemos una carpeta back para el backend y dentro un **Dockerfile** y un **docker-compose.yml**

```
root@localhost:~# mkdir back
root@localhost:~# cd back/
root@localhost:~/back# nano Dockerfile
root@localhost:~/back# nano docker-compose.yml
root@localhost:~/back# nano docker-compose.yml
root@localhost:~/back# cd ..
```


Esto va dentro del Dockerfile:

```
# Usa la imagen base oficial de PHP con Apache y PHP 8.2
FROM php:8.2-apache
```

```
# Instala extensiones necesarias para Laravel y cron
```

```
RUN apt-get update && apt-get install -y \
    libzip-dev \
    unzip \
    git \
    cron \
    && docker-php-ext-install zip pdo pdo_mysql
```

```
# Instala Composer
```

```
COPY --from=composer:2.6 /usr/bin/composer /usr/bin/composer
```

```
# Habilita el módulo de reescritura de Apache
```

```
RUN a2enmod rewrite
RUN sed -i 's!/var/www/html!/var/www/html/public!g'
/etc/apache2/sites-available/000-default.conf
```

```
# Establece el directorio de trabajo
```

```
WORKDIR /var/www/html
```

```
# Copia los archivos del proyecto Laravel
```

```
COPY . /var/www/html
```

```
# Configura permisos
```

```
RUN chown -R www-data:www-data /var/www/html
```

```
# Expone el puerto 80
```

```
EXPOSE 80
```

```
# Comando para iniciar cron y Apache
```

```
CMD ["sh", "-c", "cron && apache2-foreground"]
```

```
GNU nano 7.2 docker-compose.yml
version: '3.8'

services:
  laravel-app:
    build:
      context: .
      dockerfile: Dockerfile
    ports:
      - "8080:80"
    volumes:
      - ./proyecto-backend:/var/www/html
    environment:
      APP_ENV: local
      APP_DEBUG: true
      APP_KEY: "base64:rhKHr2xitJBkC3MBj+PtBC3dMs9D4pCDOPz+tuVGtIk="
      DB_CONNECTION: mysql
      DB_HOST: "172.20.227.241"
      DB_PORT: 3306
      DB_DATABASE: "grupo2_2425"
      DB_USERNAME: "grupo2_2425"
      DB_PASSWORD: "Rn/AjQi[jgh5pxxz"
```

version: '3.8'

services:

laravel-app:

build:

context: .

dockerfile: Dockerfile

ports:

- "8080:80"

volumes:

- ./proyecto-backend:/var/www/html

environment:

APP_ENV: local

APP_DEBUG: true

APP_KEY: "base64:rhKHr2xitJBkC3MBj+PtBC3dMs9D4pCDOPz+tuVGtIk="

DB_CONNECTION: mysql

DB_HOST: "172.20.227.241"

DB_PORT: 3306

DB_DATABASE: "grupo2_2425"

DB_USERNAME: "grupo2_2425"

DB_PASSWORD: "Rn/AjQi[jgh5pxxz"

Ignoramos la carpeta front ya que es solo si utilizas Vue.

Dev Dragons

Aritz Saiz, Lorena Arbé y Julen Corpas

Crearemos una carpeta llamada **proyecto-backend** en back.

```
root@localhost:~/front# ls
Dockerfile  docker-compose.yml  nginx.conf
root@localhost:~/front# cd ..
root@localhost:~# cd Reto3_equipo2/
root@localhost:~/Reto3_equipo2# ls
Base_datos  LICENSE  README.md  docker  laravel
root@localhost:~/Reto3_equipo2# cd ..
root@localhost:~# rm -rf
.bash_history .cache/      .docker/      .profile      Reto3_equipo2/ front/
.bashrc       .config/     .local/       .ssh/         back/
root@localhost:~# rm -rf front
root@localhost:~# cd back/
root@localhost:~/back# ls
Dockerfile  docker-compose.yml
root@localhost:~/back# nano docker-compose.yml
root@localhost:~/back# nano docker-compose.yml
root@localhost:~/back# mkdir proyecto-backend
root@localhost:~/back# cd ..
root@localhost:~# ls
Reto3_equipo2  back
root@localhost:~# cd Reto3_equipo2/
root@localhost:~/Reto3_equipo2# ls
Base_datos  LICENSE  README.md  docker  laravel
```

Copiaremos todos los archivos de la carpeta “laravel” a la carpeta proyecto-backend.

```
root@localhost:~/Reto3_equipo2# cp -r laravel/* ../back/proyecto-backend/
root@localhost:~/Reto3_equipo2# cd ..
root@localhost:~# cd back/
root@localhost:~/back# ls
Dockerfile  docker-compose.yml  proyecto-backend
root@localhost:~/back# cd proyecto-backend/
root@localhost:~/back/proyecto-backend# ls
README.md  bootstrap  config  package.json  public  storage  vendor
app        composer.json  database  phpunit.xml  resources  tailwind.config.js  vite.config.js
artisan    composer.lock  package-lock.json  postcss.config.js  routes  tests
root@localhost:~/back/proyecto-backend# nano .env
root@localhost:~/back/proyecto-backend# ls
README.md  bootstrap  config  package.json  public  storage  vendor
app        composer.json  database  phpunit.xml  resources  tailwind.config.js  vite.config.js
artisan    composer.lock  package-lock.json  postcss.config.js  routes  tests
root@localhost:~/back/proyecto-backend# docker compose build
WARN[0000] /root/back/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to a
void potential confusion
[+] Building 69.2s (16/16) FINISHED                                docker:default
```

Crearemos un archivo “.env” dentro de la carpeta del proyecto-backend:

```
GNU nano 7.2 .env
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:rozbel/kabqs15ZhDPd7PQ62oR9Vgr/BF3mzr4k1lLs=
APP_DEBUG=true
APP_TIMEZONE=UTC
APP_URL=http://localhost

APP_LOCALE=en
APP_FALLBACK_LOCALE=en
APP_FAKER_LOCALE=en_US

APP_MAINTENANCE_DRIVER=file
# APP_MAINTENANCE_STORE=database

PHP_CLI_SERVER_WORKERS=4

BCRYPT_ROUNDS=12

LOG_CHANNEL=stack
LOG_STACK=single
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug

DB_CONNECTION=mysql
```

```
DB_CONNECTION=mysql
DB_HOST=172.20.227.241
DB_PORT=3306
DB_DATABASE=reto3
DB_USERNAME=grupo2_2425
DB_PASSWORD=Rn/AjQi[jgh5pxxz
```

Haremos un **make-build** y un **docker compose up -d** para que todos los contenedores se levanten.

```
[+] Building 1/1
  ⑆ laravel-app Built 0.0
root@localhost:~/back/proyecto-backend# docker compose up -d
WARN[0000] /root/back/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to
void potential confusion
[+] Running 2/2
  ⑆ Network back_default Created 0.1
  ⑆ Container back-laravel-app-1 Started 0.7
root@localhost:~/back/proyecto-backend# ifconfig
>
> ^C
root@localhost:~/back/proyecto-backend# ifconfig
-bash: ifconfig: command not found
root@localhost:~/back/proyecto-backend# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 02:00:ac:14:e4:2b brd ff:ff:ff:ff:ff:ff
        altname enp3s0
        inet 172.20.228.43/16 brd 172.20.255.255 scope global eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::acff:fe14:e42b/64 scope link
            valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
```

Pondremos el comando **docker exec -it back-laravel-app-1 bash** para poder entrar al proyecto como si fuese make workspace. Accederemos a la carpeta “laravel” y veremos nuestro proyecto.

```
[+] Building 1/1
  ⑆ laravel-app Built 0.0
root@localhost:~/back# docker compose up -d
WARN[0000] /root/back/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to
void potential confusion
[+] Running 1/1
  ⑆ Container back-laravel-app-1 Running 0.0
root@localhost:~/back# docker exec -it reto3-php-1 bash
Error response from daemon: No such container: reto3-php-1
root@localhost:~/back# docker ps -a
CONTAINER ID   IMAGE          NAMES                COMMAND                  CREATED        STATUS        PORTS
29a558236ba0   back-laravel-app "docker-php-entrypoi..." 4 minutes ago   Up 4 minutes   0.0.0.0:8080->80/t
, [::]:8080->80/tcp   back-laravel-app-1
3d445cab8a6a   hello-world    "/hello"              2 hours ago    Exited (0) 2 hours ago
zen_proskuriakova
b4573a5cb6dc   redis:7.2-alpine "docker-entrypoint.s..." 2 hours ago    Up 2 hours    6379/tcp
reto3-redis-1
root@localhost:~/back# docker exec -it back-laravel-app-1 bash
root@29a558236ba0:/var/www/html# cd laravel
bash: cd: laravel: No such file or directory
root@29a558236ba0:/var/www/html# cd /var/www/html/laravel
bash: cd: /var/www/html/laravel: No such file or directory
root@29a558236ba0:/var/www/html# ls
README.md  bootstrap  config      package.json  public  storage  vendor
app         composer.json  database    phpunit.xml   resources  tailwind.config.js  vite.config.js
artisan    composer.lock  package-lock.json  postcss.config.js  routes    tests
```

Desde la carpeta html le daremos permisos a “bootstrap” y al poner la ip con el nombre **blog.test** en la carpeta **hosts** de nuestro equipo podremos acceder al proyecto al poner la url en cualquier navegador.

```
root@29a558236ba0:/var/www/html# ls
README.md  bootstrap  config      package.json  public  storage  vendor
app         composer.json  database    phpunit.xml   resources  tailwind.config.js  vite.config.js
artisan    composer.lock  package-lock.json  postcss.config.js  routes    tests
root@29a558236ba0:/var/www/html# cd ..
root@29a558236ba0:/var/www# ls
html
root@29a558236ba0:/var/www# cd html/
root@29a558236ba0:/var/www/html# chmod -R 777 storage bootstrap/cache
root@29a558236ba0:/var/www/html# chown -R www-data:www-data storage bootstrap/cache
root@29a558236ba0:/var/www/html#
```