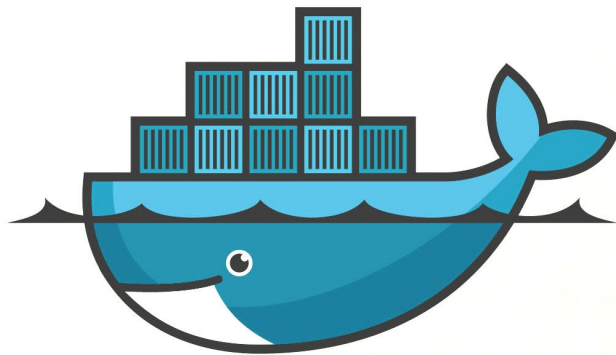


Instalación

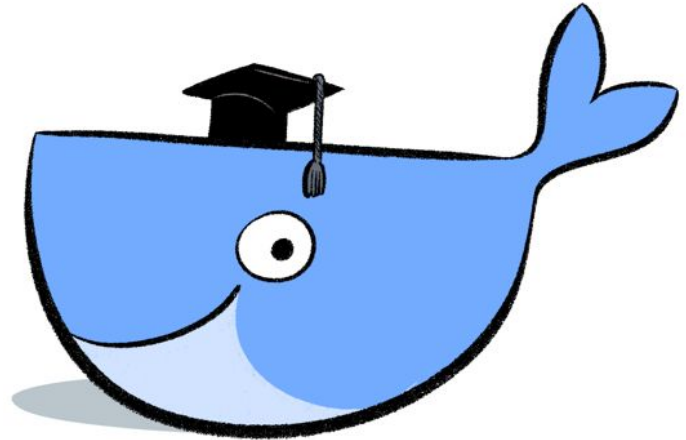


docker

Contenido

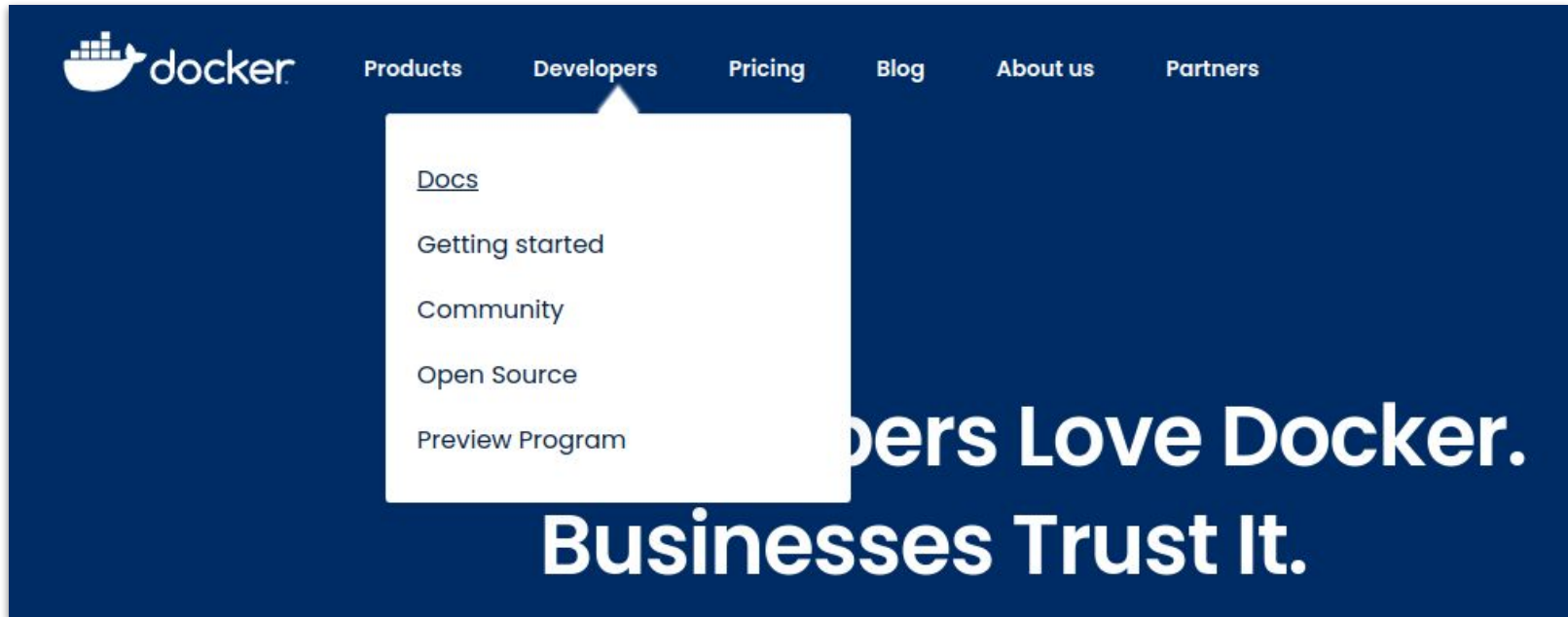
1. Documentación de la instalación
2. Proceso de instalación
3. Comprobación de la instalación
4. Operaciones de mantenimiento

Documentación de la instalación



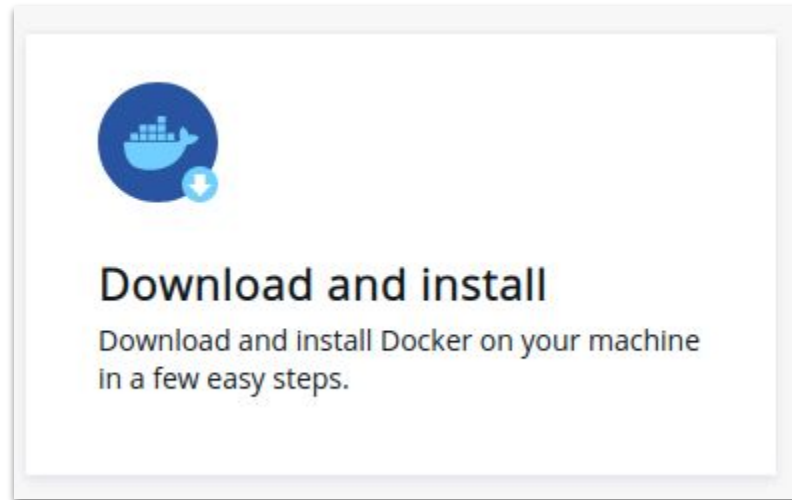
1. Documentación de la instalación

<https://www.docker.com/> —————> **Developers / Docs**



1. Documentación de la instalación

<https://docs.docker.com/> —————> **Download and install**



1. Documentación de la instalación

Seleccionamos **Docker for Linux**



Docker Desktop for Mac

A native application using the macOS sandbox security model which delivers all Docker tools to your Mac.



Docker Desktop for Windows

A native Windows application which delivers all Docker tools to your Windows computer.



Docker for Linux

Install Docker on a computer which already has a Linux distribution installed.

1. Documentación de la instalación

Server

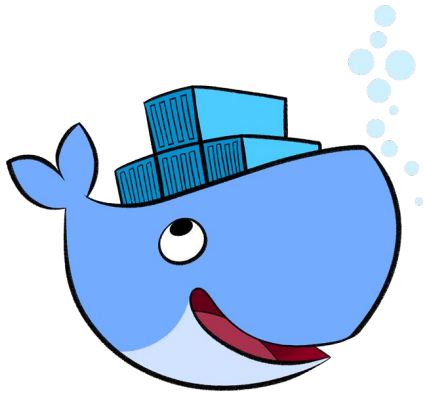
Docker provides `.deb` and `.rpm` packages for

Platform	x86_64 / amd64
CentOS	✓
Debian	✓
Fedora	✓
Raspbian	
RHEL	
SLES	
Ubuntu	✓

Seleccionamos **Ubuntu**



Proceso de instalación



Proceso de instalación

1. Prerrequisitos
 - a. Comprobar compatibilidad con el SO
 - b. Desinstalar versiones anteriores de los paquetes

2. Instalación
 - a. Añadir el repositorio
 - b. Instalar docker

Comprobar compatibilidad con el SO

Requisitos

Prerequisites

OS requirements

To install Docker Engine, you need the 64-bit version of

- Ubuntu Impish 21.10
- Ubuntu Hirsute 21.04
- **Ubuntu Focal 20.04 (LTS)**
- Ubuntu Bionic 18.04 (LTS)

Tenemos



Kubuntu 20.04

<https://www.kubuntu.org>

```
pepe@pepe-VBox:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04.4 LTS
Release:        20.04
Codename:       focal
```

Desinstalar versiones anteriores

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

```
pepe@pepe-VBox:~$ sudo apt remove docker docker-engine docker.io containerd runc
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
E: No se ha podido localizar el paquete docker-engine
pepe@pepe-VBox:~$ sudo apt remove docker docker.io containerd runc
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete «docker» no está instalado, no se eliminará
El paquete «runc» no está instalado, no se eliminará
El paquete «docker.io» no está instalado, no se eliminará
El paquete «containerd» no está instalado, no se eliminará
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
```

Añadir el repositorio

1. Actualizar índice de paquetes e instalar prerequisites

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

```
$ sudo apt-get install \  
ca-certificates \  
curl \  
gnupg \  
lsb-release
```

```
pepe@pepe-VBox:~$ sudo apt update  
Obj:1 http://es.archive.ubuntu.com/ubuntu focal InRelease  
Obj:2 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease  
Obj:3 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease  
Obj:4 http://security.ubuntu.com/ubuntu focal-security InRelease  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Todos los paquetes están actualizados.
```

```
pepe@pepe-VBox:~$ sudo apt install ca-certificates curl gnupg lsb-release  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
lsb-release ya está en su versión más reciente (11.1.0ubuntu2).  
fijado lsb-release como instalado manualmente.  
ca-certificates ya está en su versión más reciente (20210119~20.04.2).  
fijado ca-certificates como instalado manualmente.  
gnupg ya está en su versión más reciente (2.2.19-3ubuntu2.1).  
fijado gnupg como instalado manualmente.  
Se instalarán los siguientes paquetes NUEVOS:  
curl
```

Añadir el repositorio

2. Añadir clave GPG del repositorio de Docker

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-
```

```
pepe@pepe-VBox:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg  
pepe@pepe-VBox:~$
```

Añadir el repositorio

3. Añadir el repositorio de Docker

```
$ echo \  
"deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \  
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
pepe@pepe-VBox:~$ echo \  
> "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \  
> $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
pepe@pepe-VBox:~$ cat /etc/apt/sources.list.d/docker.list  
deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu focal stable
```


Instalar Docker

```
$ sudo apt-get update  
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
```

```
pepe@pepe-VBox:~$ sudo apt update  
Obj:1 http://es.archive.ubuntu.com/ubuntu focal InRelease  
Obj:2 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease  
Obj:3 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease  
Des:4 https://download.docker.com/linux/ubuntu focal InRelease [57,7 kB]  
Obj:5 http://security.ubuntu.com/ubuntu focal-security InRelease  
Des:6 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [14,8 kB]  
Descargados 72,5 kB en 1s (119 kB/s)  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Todos los paquetes están actualizados.
```

Instalar Docker

```
$ sudo apt-get update  
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
```

```
pepe@pepe-VBox:~$ sudo apt install docker-ce docker-ce-cli containerd.io  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Se instalarán los siguientes paquetes adicionales:  
  docker-ce-rootless-extras docker-scan-plugin git git-man liberror-perl patch pigz  
  slirp4netns  
Paquetes sugeridos:  
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit  
  git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn  
  diffutils-doc  
Se instalarán los siguientes paquetes NUEVOS:  
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras docker-scan-plugin  
  git git-man liberror-perl patch pigz slirp4netns  
0 actualizados, 11 nuevos se instalarán, 0 para eliminar y 0 no actualizados.  
Se necesita descargar 104 MB de archivos.  
Se utilizarán 448 MB de espacio de disco adicional después de esta operación.  
¿Desea continuar? [S/n]  
Des:1 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1 [57,4 k  
B]
```


Comprobación de la instalación



Comprobación de la instalación

1. Visualizar versiones instaladas
2. Visualizar información de docker
3. Ejecutar contenedor de ejemplo

Visualizar versiones instaladas

sudo docker version

```
pepe@pepe-VBox:~$ sudo docker version
Client: Docker Engine - Community
 Version:      20.10.13
 API version:  1.41
 Go version:   go1.16.15
 Git commit:   a224086
 Built:        Thu Mar 10 14:07:51 2022
 OS/Arch:     linux/amd64
 Context:      default
 Experimental: true

Server: Docker Engine - Community
 Engine:
  Version:      20.10.13
  API version:  1.41 (minimum version 1.12)
  Go version:   go1.16.15
  Git commit:   906f57f
  Built:        Thu Mar 10 14:05:44 2022
  OS/Arch:     linux/amd64
  Experimental: false
 containerd:
  Version:      1.5.10
  GitCommit:    2a1d4dbdb2a1030dc5b01e96fb110a9d9f150ecc
 runc:
  Version:      1.0.3
  GitCommit:    v1.0.3-0-gf46b6ba
 docker-init:
  Version:      0.19.0
  GitCommit:    de40ad0
```

Visualizar información de Docker

sudo docker info

```
pepe@pepe-VBox:~$ docker info
Client:
 Context:    default
 Debug Mode: false
 Plugins:
  app: Docker App (Docker Inc., v0.9.1-beta3)
  buildx: Docker Buildx (Docker Inc., v0.8.0-docker)
  scan: Docker Scan (Docker Inc., v0.17.0)

Server:
 Containers: 3
  Running: 0
  Paused: 0
  Stopped: 3
 Images: 2
 Server Version: 20.10.13
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
  userxattr: false
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Cgroup Version: 1
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: inactive
 Runtimes: io.containerd.runtime.v1.linux runc io.containerd.runc.v2
 Default Runtime: runc
```

Ejecutar contenedor de ejemplo

```
sudo docker run hello-world
```

```
pepe@pepe-VBox:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:4c5f3db4f8a54eb1e017c385f683a2de6e06f75be442dc32698c9bbe6c861edd
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

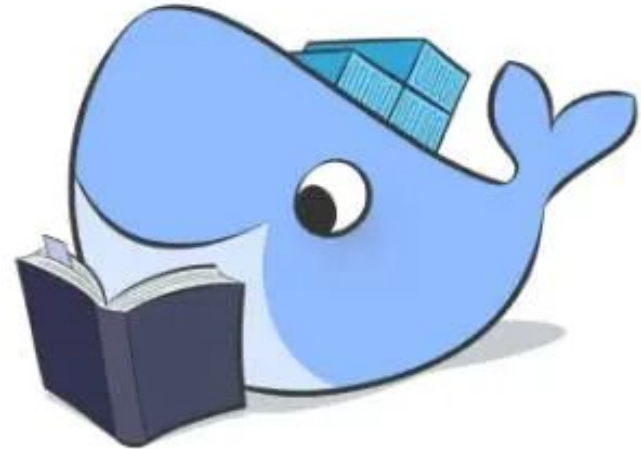
To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
```

Operaciones de mantenimiento



Operaciones de mantenimiento

1. Actualizar Docker
2. Eliminar Docker
3. Ejecutar Docker como un usuario normal (sin sudo)

Actualizar Docker

Al instalarse con un repositorio, Docker se actualizará **con el resto del sistema**.

```
sudo apt update  
sudo apt upgrade
```


Eliminar Docker

Para eliminar **Docker**, hay que borrar los paquetes correspondientes:

```
sudo apt-get purge docker-ce docker-ce-cli containerd.io
```

Para eliminar los **contenedores, la imágenes y los volúmenes**, hay que borrar los directorios que las contienen:

```
sudo rm -rf /var/lib/docker  
sudo rm -rf /var/lib/containerd
```

Ejecutar Docker como un usuario normal (sin sudo)

Si un usuario normal quiere usar docker, se debe añadir al grupo **docker**:

```
usermod -aG docker username
```

Se deberá volver a iniciar sesión para que el cambio de grupo surta efecto.

Los contenedores que ejecute ese usuario, heredarán sus permisos en vez de los del root.

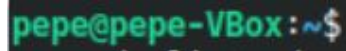
Ejercicio 1



Ejercicio 1

Instala Docker en una máquina virtual y haz un manual con capturas de pantalla y explicaciones sobre el proceso.

El prompt de la máquina tiene que identificarte, como por ejemplo:

A screenshot of a terminal window showing a green prompt string 'pepe@pepe-VBox:~\$' on a black background. The text is in a monospaced font, with the username 'pepe' and host 'pepe-VBox' in green, and the tilde '~' and dollar sign '\$' in white.

La instalación debe contemplar 4 fases:

1. Prerrequisitos.
2. Instalación.
3. Comprobación.
4. Añadir usuario al grupo docker.