

FONAMENTS DEL MAQUINARI. PRÀCTICA TEMA 5 ***INSTAL·LACIÓ PROGRAMES EN DISTRIBUCIONS DEBIAN I*** ***INTRODUCCIÓ A DOCKER***

Administració de Sistemes Informàtics en Xarxa. 1er Curs

Enrique Tortajada González

Cicles Formatius

Índex

1 INSTALACIÓ PROGRAMES EN DISTRO DEBIAN..... .1

Tema 1 1.1 Entorn gràfic..... .1

1.2 Línia de comandes amb apt..... .2

1.3 Instal·lació des de codi font..... .2

2 CREACIÓ D'UN SERVIDOR WEB AMB DOCKER I NGINX.....3

3 BIBLIOGRAFIA..... .6

1 INSTAL·LACIÓ PROGRAMES EN DISTRO DEBIAN

Objectius:

En esta pràctica, aprendràs a instal·lar tres programes en Ubuntu o una altra distribució tipus Debian de tres maneres diferents. La pràctica es divideix en tres parts, cadascuna de les quals es centra en una forma diferent d'instal·lar programes en Linux. Per tant has de disposar d'una màquina virtual o un equip amb Ubuntu o qualsevol distribució tipus Debian

1. **Atom:** Atom és un editor de text lleuger i que et permet escriure i editar textos.
2. **Wireshark:** Wireshark és un analitzador de xarxa que et permet capturar i analitzar el tràfic de xarxa en temps real.
3. **Aplicació a elegir:** Fes una recerca de aplicacions en codi font i fes la instal·lació. Has de tindre en compte les dependències i saber que el procés es més difícil. Tens exemples com vlc per a mint amb tots el detalls en [Cómo instalar y usar VLC Media Player en Linux Mint](#)
- [Linux Hint | Heading \(headinghometodinner.org\)](#) . Per a Ubuntu n'he tens al següent enllaç: [Ubuntu Linux: Como compilar e instalar un programa desde el código fuente en Ubuntu \(ubuntinux.blogspot.com\)](#). Pots descarregar qualsevol aplicació en codi Font, pensa que quan més dependències tinga major dificultat tindrà la instal·lació.

Els programes són una guia, pots escollir uns altres que siguen d'utilitat per a tú. La part interessant resideix en el procés d'instal·lació en que podràs vore de menor a major la dificultat.

Requisits previs: Sistema operatiu Ubuntu o màquina virtual d'Ubuntu (Pots fer ús de qualsevol distribució basada en Debian).

1.1 Entorn gràfic

Instal·la el programa **Atom** a través del centre de programari d'Ubuntu.

Passos de la pràctica (0,5 punt):

1. Obre el centre de programari d'Ubuntu.
2. Cerca "Atom" a la barra de cerca.
3. Fes clic a "Instal·la".
4. Espera que s'instal·le el programa

- **Què són els paquets snaps o snap packages?**

Es un formato de distribución similar a un AppImage en el que se pretende que sea un "paquete instalable universal" para desplegar software en sistemas Linux, los paquetes snap son paquetes que contienen todas las dependencias que necesitan para funcionar correctamente.

- **Fa ús l'entorn gràfic d'estes ferramentes?**

Si, dado que es la manera mas facil y principal de distribuir estos paquetes snap por ejemplo en la Ubuntu Store o uApp Explorer por ejemplo, estos no estan asociados a una distribucion concreta, ademas

- **Coneixes altres ferramentes gràfiques per a instal·lar programes en Ubuntu o altra distro Debian? Quines diferències trobes?**

UappExplorer por ejemplo es un poco mas completa que la ubuntu store que te trae por defecto, las principales diferencias que nos encontramos seria la distribucion de las categorias, la cantidad de paquetes snap que encontramos, en linux mint encontramos el Gestor de software que nos instala flatpak en contra de los snap que tenemos en los programas anteriores però basicamente es como un snap con la principal diferencia que los flatpak soportan otros repositorios remotos, la aplicacion en si esta sandboxed lo que nos ahorra un mal trago con los posibles errores en las dependencias.

1.2 Línia de comandes amb apt

Instal·la el programa **Wireshark** a través de la línia de comandes. Explica el procés i adjunta les captures que cregues convenient:

Passos de la pràctica (0,5):

1. Obre la terminal.
2. Executa la comanda **sudo add-apt-repository ppa:wireshark-dev/stable** per afegir el repositori PPA oficial de Wireshark
3. Executa la comanda **sudo apt-get update** per actualitzar la llista de paquets.

```

Archivo Editar Ver Buscar Terminal Ayuda
dislexia@dislexia-VirtualBox:~$ sudo su
[sudo] contraseña para dislexia:
root@dislexia-VirtualBox:~# add-apt-repository ppa:wireshark-dev/stable
Está a punto de añadir el PPA siguiente:
Latest stable Wireshark releases back-ported from Debian package versions.

Back-porting script is available at https://github.com/rbalint/pkg-wireshark-ubuntu-ppa

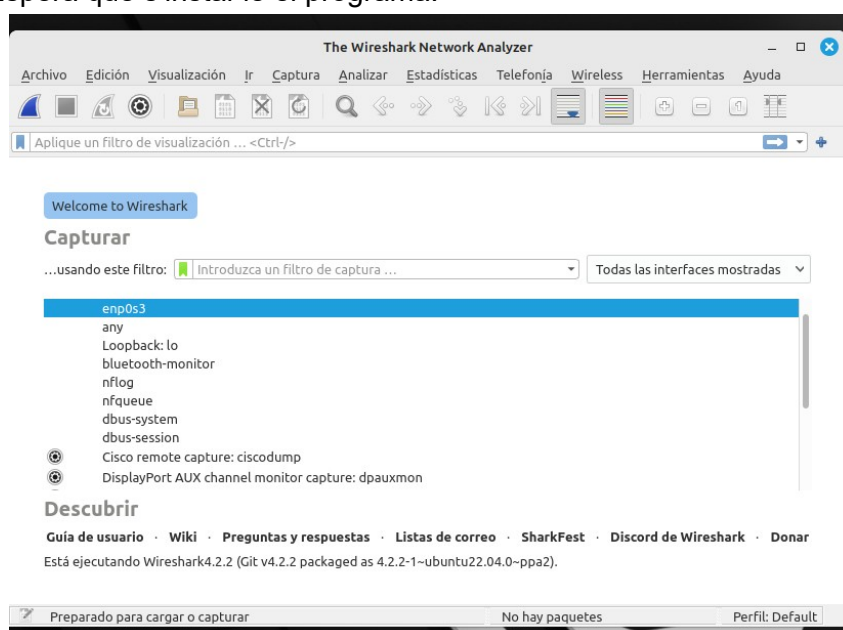
From Ubuntu 16.04 you also need to enable "universe" repository, see:
http://askubuntu.com/questions/148638/how-do-i-enable-the-universe-repository

The packaging repository for Debian and Ubuntu is at: https://salsa.debian.org/debian/wireshark
Más información: https://launchpad.net/~wireshark-dev/+archive/ubuntu/stable
Pulse Intro para continuar o Ctrl+C para cancelar

gpg: creado el directorio '/root/.gnupg'
gpg: caja de claves '/root/.gnupg/pubring.kbx' creada
gpg: /root/.gnupg/trustdb.gpg: se ha creado base de datos de confianza
gpg: caja de claves '/etc/apt/keys/A2E4028B5A4B70CD7808A3090875551314ECA0F0.Keyring' creada
gpg: clave D075551314ECA0F0: clave pública "Launchpad PPA for Wireshark Developers" importada
gpg: Cantidad total procesada: 1
gpg:   importadas: 1
root@dislexia-VirtualBox:~# apt-get update
Obj:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Obj:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Des:3 https://ppa.launchpadcontent.net/wireshark-dev/stable/ubuntu jammy InRelease [24,4 kB]
Obj:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign:5 http://packages.linuxmint.com virginia InRelease
Obj:6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Des:7 https://ppa.launchpadcontent.net/wireshark-dev/stable/ubuntu jammy/main i386 Packages [1.152 B]
Des:8 https://ppa.launchpadcontent.net/wireshark-dev/stable/ubuntu jammy/main amd64 Packages [4.300 B]
Des:9 https://ppa.launchpadcontent.net/wireshark-dev/stable/ubuntu jammy/main Translation-en [1.988 B]
Obj:10 http://packages.linuxmint.com virginia Release
Descargados 31,8 kB en 0s (73,0 kB/s)
Leyendo lista de paquetes... Hecho

```

4. Executa la comanda **sudo apt-get install wireshark** per instal·lar Wireshark.
5. Espera que s'instal·le el programa.



Preguntas de reflexión (1 punt):

- **Què és APT (Advanced Paquet Tools)?**

Es una Herramienta de la línea de comandos creado para darle a los usuarios el privilegio de gestionar y administrar los paquetes de su distribución Linux Debian, dispone de subcomandos para listar, liberar paquetes, configurar paquetes, eliminarlos, se puede utilizar apt sin el -get para mayor comodidad del usuario

- **Què són les dependències? Podem tenir problemes a l'hora d'instal·lar programes? Què harem de fer per a prevenir això?**

Las dependencias son super importantes ya que es una biblioteca o archivos complementarios requeridos para que otro programa pueda funcionar correctamente, no solo instalarlo como tal, es probable que nos topemos con que estas dependencias no esten instaladas en nuestro sistema por tanto esto nos puede llevar a errores o problemas, para prevenir esto deberiamos limpiar la memoria de instalacion de paquetes y hacer un update, luego resolver las dependencias rotas que existan en el sistema, puedes informarte de los repositorios necesarios en las propias paginas web de los programas, en foros como git-hub o en las propias distribuciones.

1.3 Instal·lació des de codi font

Instal·la el programa **VLC** a través de la opció de codi font. Explica el procés i adjunta les./ captures que cregues convenient. Pot ser el procés tinga més parts que les demandades als passos, s'ha de tindre en compte que molts programes requereixen

Passos de la pràctica (2 punt):

1. Creamos un directorio y hacemos un cd y haremos un wget a la version 3.0.8

```
root@dislexia-VirtualBox:/home/dislexia/Escritorio# mkdir -pv ~/vlc && cd ~/vlc
mkdir: se ha creado el directorio '/root/vlc'
root@dislexia-VirtualBox:~/vlc# wget http://get.videolan.org/vlc/3.0.8/vlc-3.0.8.tar.xz
--2024-02-12 14:56:51-- http://get.videolan.org/vlc/3.0.8/vlc-3.0.8.tar.xz
Resolviendo get.videolan.org (get.videolan.org)... 195.154.241.219
Conectando con get.videolan.org (get.videolan.org)[195.154.241.219]:80... conectado.
Petición HTTP enviada, esperando respuesta... 302 Moved Temporarily
Ubicación: https://mirrors.uni-ruse.bg/vlc/vlc/3.0.8/vlc-3.0.8.tar.xz [siguiente]
--2024-02-12 14:56:51-- https://mirrors.uni-ruse.bg/vlc/vlc/3.0.8/vlc-3.0.8.tar.xz
Resolviendo mirrors.uni-ruse.bg (mirrors.uni-ruse.bg)... 91.210.88.42, 2001:67c:220c::2a
Conectando con mirrors.uni-ruse.bg (mirrors.uni-ruse.bg)[91.210.88.42]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 26041520 (25M) [application/octet-stream]
Guardando como: 'vlc-3.0.8.tar.xz'

vlc-3.0.8.tar.xz 56%[=====
```

2. Necesitaremos descargar los repositorios necesarios que no vienen con la descarga oficial y apt build-dep vlc.

```
dislexia@dislexia-VirtualBox:~$ sudo su
[sudo] contraseña para dislexia:
root@dislexia-VirtualBox:/home/dislexia# apt install -y git build-essential pkg-config libtool automake autopoint gettext
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
gettext ya está en su versión más reciente (0.21-4ubuntu4).
pkg-config ya está en su versión más reciente (0.29.2-1ubuntu3).
Se instalarán los siguientes paquetes adicionales:
autoconf autotools-dev g++ g++-11 git-man libc-dev-bin libc-devtools
libc6-dev libcrypt-dev liberror-perl libltdl-dev libnsl-dev libstdc++11-dev
libtirpc-dev rpcsvc-proto
Paquetes sugeridos:
autoconf-archive gnu-stardards autoconf-doc g++-multilib g++-11-multilib
gcc-11-doc git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui
gitk gitweb git-cvs git-mediawiki git-svn glibc-doc libtool-doc
libstdc++11-doc gfortran | fortran95-compiler gcj-jdk
Se instalarán los siguientes paquetes NUEVOS:
autoconf automake autopoint autotools-dev build-essential g++ g++-11 git
git-man libc-dev-bin libc-devtools libc6-dev libcrypt-dev liberror-perl
libltdl-dev libnsl-dev libstdc++11-dev libtirpc-dev libtool rpcsvc-proto
0 actualizados, 20 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Se necesita descargar 22,0 MB de archivos.
```

```

root@dislexia-VirtualBox:/home/dislexia# apt build-dep vlc
Leyendo lista de paquetes... Hecho
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Los siguientes paquetes se ELIMINARÁN:
  libjack-jackd2-0
Se instalarán los siguientes paquetes NUEVOS:
  debhelper debugedit dh-autoreconf dh-strip-nondeterminism dwz
  glib2.0-rsvg-2.0 icu-devtools liba52-0.7.4-dev liba52-dev libarchive-dev
  libasound2-dev libass-dev libatk-bridge2.0-dev libatk1.0-dev

```

3. Luego en el mismo directorio que estamos ~/vlc, ejecutaremos `tar -zxvf vlc-3.0.8.tar.gz`, haremos un `cd vlc 3.0.8`, luego haremos un `./bootstrap` para preparar el script que ejecuta el comando `/configure.status` y despues un `make -j4` para que instale el programa. Comprobamos version con un `vlc` para ver que esta instalado.

```

root@dislexia-VirtualBox:~/vlc# cd vlc-3.0.8
root@dislexia-VirtualBox:~/vlc/vlc-3.0.8# ./bootstrap
autoreconf: export WARNINGS=
autoreconf: Entering directory '.'
autoreconf: running: autopoint --force
copying file m4/fcntl-o.m4
copying file m4/threadlib.m4
copying file po/Makevars.template
autoreconf: running: aclocal -I m4 --force -I m4
autoreconf: configure.ac: tracing

```

```

configure: creating ./config.status
config.status: creating Makefile
config.status: creating doc/Makefile
config.status: creating modules/Makefile
config.status: creating m4/Makefile
config.status: creating po/Makefile.in
config.status: creating share/Makefile
config.status: creating compat/Makefile
config.status: creating src/Makefile
config.status: creating lib/Makefile
config.status: creating bin/Makefile
config.status: creating test/Makefile
config.status: creating modules/hw/mmal/Makefile
config.status: creating share/vlc.appdata.xml.in
config.status: creating config.h
config.status: executing depfiles commands
config.status: executing libtool commands
config.status: executing po-directories commands
config.status: creating po/POTFILES
config.status: creating po/Makefile

libvlc configuration
-----
version      : 3.0.8
system      : linux
architecture : x86_64 mmx sse sse2
optimizations : yes
vlc aliases  : cvlc rvlc qvlc svlc nvlc

To build vlc and its plugins, type 'make', or './compile' if you like nice colors.

```

```

root@dislexia-VirtualBox:~/vlc/vlc-3.0.8# make -j4
make all-recursive
make[1]: se entra en el directorio '/root/vlc/vlc-3.0.8'
Making all in compat
make[2]: se entra en el directorio '/root/vlc/vlc-3.0.8/compat'
make all-am
make[3]: se entra en el directorio '/root/vlc/vlc-3.0.8/compat'
  CC      dummy.lo
  CC      strlcpy.lo
  CC      strnstr.lo
  CCLD    libcompat.la
make[3]: se sale del directorio '/root/vlc/vlc-3.0.8/compat'
make[2]: se sale del directorio '/root/vlc/vlc-3.0.8/compat'
Making all in doc
make[2]: se entra en el directorio '/root/vlc/vlc-3.0.8/doc'
make[2]: No se hace nada para 'all'.
make[2]: se sale del directorio '/root/vlc/vlc-3.0.8/doc'
Making all in po
make[2]: se entra en el directorio '/root/vlc/vlc-3.0.8/po'
make[2]: No se hace nada para 'all'.
make[2]: se sale del directorio '/root/vlc/vlc-3.0.8/po'
Making all in share
make[2]: se entra en el directorio '/root/vlc/vlc-3.0.8/share'
  GEN     vlc.appdata.xml
  GEN     vlc.desktop
  GEN     skins2/default.vlt
drwxr-xr-x root/root      0 2019-08-19 09:55 default/
-rw-r--r-- root/root 55484 2017-11-24 17:29 default/theme.xml

```

```

dislexia@dislexia-VirtualBox:~$ vlc
VLC media player 3.0.16 Vetinari (revision 3.0.13-8-g41878ff4f2)

```


Preguntes de reflexió (1 punt):

- **Què fem amb cada una de les comandes utilitzades?**
tar -xf nom_del_paquet.tar.xz: Descomprimir el archivo que contiene los archivos necesarios para proseguir al paso siguiente, aunque no siempre tiene todos.
Configure: Este comando nos permite configurar un script específico que contiene la carpeta descomprimida que nos garantizara que la app/proyecto se pueda construir en el sistema actual. Crea el makefile necesario, contiene instrucciones necesarias.
Make: Este comando nos permitira leer el archivo creado 'Makefile' y es el encargado de realizar las tareas necesarias para construir e instalar posteriormente la app.
Make Install: Este comando nos permite instalar el programa basado en el archivo 'makefile'
Per què es descomprimeix el arxiu tar amb les opcions -zxvf ?
Z: Comprime con gzip el paquete generado
x: Desempaqueta el archivo de un paquete
v: Nos muestra lo que va realizando
f: Envia a la salida estandar, este siempre va el ultimo.
Quines avantatges i desavantatges trobes amb este mètode? Es recomanable en algun context?

Nos permite instalar las ultimas versiones de las app en cuestion, incluimos funcionalidades que por defecto no pueden estar incluidas con el apt, tenemos mas libertad de gestionar los paquetes, repositorios y dependencias necesarias para ejecutar la app en cuestion.

Como desventajas puede ser un poco mas lioso y tedioso que el metodo mas actual 'APT', tienes que tener instaladas previamente las depencias/librerias necesarias para la compilacion del programa que a veces no suelen estar todas en el tar y tienes que buscarlas manualmente e instalarlas aunque hay mucha informacion para esto.

Este metodo es recomendable para instalar versiones en concreto de programas o de proyectos echos haciendo su manejo mas detallado y controlado.

2 CREACIÓ D'UN SERVIDOR WEB AMB DOCKER I NGINX

Objectius:

- Aprendre a utilitzar Docker per a crear i gestionar contenidors.
- Configurar un servidor web Nginx dins d'un contenidor Docker.
- Servir una pàgina web estàtica personalitzada.
- Al tindre molts problemes seguint el tutorial que posa en la practica he anat a la pagina principal de docker y en la seva wiki he seguit el seus pasos

Passos de la pràctica (1'5 punts):

1. Eliminamos del sistema cualquier pack que no sea oficial de docker tras haber realizado varios intentos en los pasos de la practica, para eso ejecutaremos:

```
$ for pkg in docker.io docker-doc docker-compose docker-compose-v2 podman-docker containerd runc; do sudo apt-get remove $pkg; done
```

```
listex@distexia-VirtualBox:/etc/apt/sources.list.d$ for pkg in docker.io docker-doc docker-compose docker-compose-v2 podman-docker containerd runc; do sudo apt-get remove $pkg; done
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
El paquete «docker.io» no está instalado, no se eliminará
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Leyendo lista de paquetes... Hecho
```

2. Ahora instalaremos los repositorios de docker mediante un script que puedes copiar de su pagina web creando un archivo .sh y ejecutandolo luego con bash con touch y luego el nano script.sh copiaremos lo siguiente:

```
# Add Docker's official GPG key:
```

```
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/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
# Add the repository to Apt sources:
```

```
echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

Ejecutaremos el comando bash script.sh y dejaremos que instale:

```
dislexia@dislexia-VirtualBox:~/Escritorio$ nano script.sh
dislexia@dislexia-VirtualBox:~/Escritorio$ bash script.sh
Ign:1 https://mirrors.cicku.me/linuxmint/packages virginia InRelease
Obj:2 http://ftp.linux.cz/pub/linux/ubuntu jammy InRelease
Obj:3 https://mirrors.cicku.me/linuxmint/packages virginia Release
Obj:4 http://ftp.linux.cz/pub/linux/ubuntu jammy-updates InRelease
Obj:5 http://ftp.linux.cz/pub/linux/ubuntu jammy-backports InRelease
Obj:6 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:8 https://download.docker.com/linux/ubuntu virginia InRelease
Err:9 https://download.docker.com/linux/ubuntu virginia Release
404 Not Found [IP: 3.160.231.8 443]
Leyendo lista de paquetes... Hecho
E: El repositorio «https://download.docker.com/linux/ubuntu virginia Release» no tiene un fichero de Publicación.
N: No se puede actualizar de un repositorio como este de forma segura y por tanto está deshabilitado por omisión.
N: Vea la página de manual apt-secure(8) para los detalles sobre la creación de repositorios y la configuración de usuarios.
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
ca-certificates ya está en su versión más reciente (20230311ubuntu0.22.04.1).
curl ya está en su versión más reciente (7.81.0-1ubuntu1.15).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Ign:1 https://mirrors.cicku.me/linuxmint/packages virginia InRelease
Des:7 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [26,7 kB]
Obj:8 http://ftp.linux.cz/pub/linux/ubuntu jammy-backports InRelease
Descargados 75,5 kB en 0s (174 kB/s)
Leyendo lista de paquetes... Hecho
```

3. Instalamos la version mas reciente:

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

```
dislexia@dislexia-VirtualBox:~/Escritorio$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  docker-ce-rootless-extras libslirp0 pigz slirp4netns
Paquetes sugeridos:
  aufs-tools cgroupfs-mount | cgroup-lite
Se instalarán los siguientes paquetes NUEVOS:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin libslirp0 pigz slirp4netns
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Se necesita descargar 117 MB de archivos.
Se utilizarán 420 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.6.23-1 [20,6 MB]
```

4. Ahora podremos hacer un: **sudo docker run hello-world**

```
dislexia@dislexia-VirtualBox:~/Escritorio$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:4bd7811b6914a99dbc560e6a20eab57ff6655aea4a80c50b0c5491968cbc2e6
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/
```

i

3. Haremos un git clone de docker awesome compose que viene con varias imagenes ademas de nginx

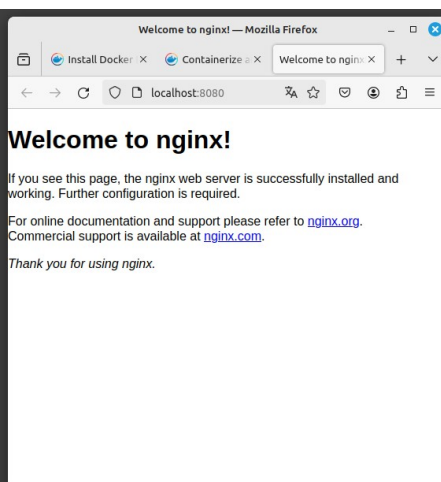
```
root@dislexia-VirtualBox:/home/dislexia/Escritorio# git clone https://github.com/docker/awesome-compose
Clonando en 'awesome-compose'...
remote: Enumerating objects: 2392, done.
remote: Total 2392 (delta 0), reused 0 (delta 0), pack-reused 2392
Recibiendo objetos: 100% (2392/2392), 5.90 MiB | 15.70 MiB/s, listo.
Resolviendo deltas: 100% (1072/1072), listo.
root@dislexia-VirtualBox:/home/dislexia/Escritorio# docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
c57ee500d61: Pull complete
9b0163235c08: Pull complete
f24a6f652778: Pull complete
9f3589a5fc50: Pull complete
f0bd99a47d4a: Pull complete
398157bc5c51: Pull complete
1ef1c1a36ec2: Pull complete
Digest: sha256:84c52dfd55c467e12ef85cad6a252c0990564f03c4850799bf41dd738738691f
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
root@dislexia-VirtualBox:/home/dislexia/Escritorio#
```

4. **Creació del Contenedor de Nginx:** Crea i executa un contenidor basat en la imatge de Nginx. Utilitza el següent comandament per a mapejar el port 80 del contenidor al port 8080 de la teva màquina host:

```
$ sudo docker run --name meu-nginx -p 8080:80 -d nginx
```

Amb això, el contenidor estarà en execució i el servidor web Nginx estarà escoltant en el port 8080 de la teva màquina.

```
Using default tag: latest
latest: Pulling from library/nginx
c57ee500d61: Pull complete
9b0163235c08: Pull complete
f24a6f652778: Pull complete
9f3589a5fc50: Pull complete
f0bd99a47d4a: Pull complete
398157bc5c51: Pull complete
1ef1c1a36ec2: Pull complete
Digest: sha256:84c52dfd55c467e12ef85cad6a252c0990564f03c4850799bf41dd738738691f
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
root@dislexia-VirtualBox:/home/dislexia/Escritorio# docker run --name meu-nginx -p 8080:80 -d nginx
8784e1e0648e4a66544583e5a5cad319440b5ef47088d19e28065e56308766e3
root@dislexia-VirtualBox:/home/dislexia/Escritorio#
```



5. **Comprovació del Servidor Web:** Obri un navegador i visita **http://localhost:8080**. Hauries de veure la pàgina de benvinguda de Nginx, indicant que el servidor web està funcionant correctament.

6. **Personalització de la Pàgina Web:** Crea un directori en la teua màquina host per a emmagatzemar el contingut web estàtic:

```
$ mkdir -p ~/meu-lloc-web
```

```
$ echo "Hola des de Docker!" > ~/meu-lloc-web/index.
```

7. **Vinculació del Contingut Estàtic al Contenidor:** Atura el contenidor actual i torna a executar-lo, aquesta vegada muntant el directori que conté la teua pàgina web estàtica. Això es fa amb l'opció `-v` per a muntar un volum:

```
$ sudo docker ps
```

```
$ sudo docker stop meu-nginx
```

```
$ docker run --name meu-nginx -p 8080:80 -v ~/meu-lloc-web:/usr/share/nginx/html:ro -d nginx
```

Aquest comandament munta el directori `~/meu-lloc-web` de la teua màquina host al directori `/usr/share/nginx/html` dins del contenidor, que és on Nginx espera trobar els fitxers per a servir. L'opció `:ro` indica que el volum es munta en mode de només lectura.

8. **Accés a la Pàgina Web Personalitzada:** Torna a obrir el navegador i visita `http://localhost:8080`. Ara hauries de veure el contingut de la teua pàgina web estàtica personalitzada, que diu "Hola des de Docker!".

9. **Neteja:** Un cop hages acabat amb la pràctica, pots aturar i eliminar el contenidor amb els següents comandaments:

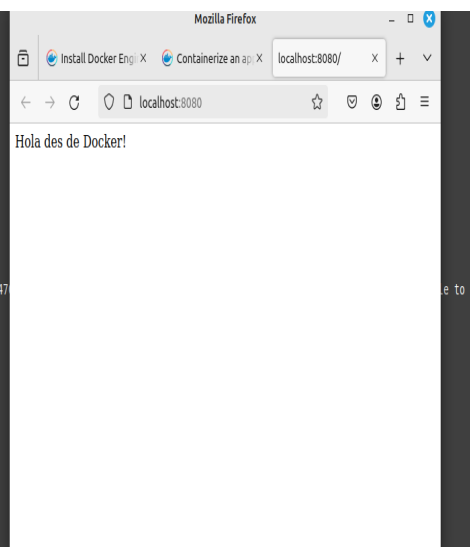
```
$ sudo docker stop meu-nginx
```

```
$ sudo docker rm meu-nginx
```

10. Si vols eliminar també la imatge de Nginx del teu sistema, pots fer-ho amb:

```
$ sudo docker rm --nginx
```

```
98157bc5c51: Pull complete
ef1c1a36ec2: Pull complete
Digest: sha256:84c52fd55c467e12ef85cad6a252c0990564f03c4850799bf41dd73738691f
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
root@dislexia-VirtualBox:/home/dislexia/Escritorio# docker run --name meu-nginx -p 8080:80 -d nginx
7784e1e0648e4a66544583e5a5cad319440b5ef47688d19e28065e56380766e3
root@dislexia-VirtualBox:/home/dislexia/Escritorio# mkdir -p ~/meu-lloc-web
root@dislexia-VirtualBox:/home/dislexia/Escritorio# echo "Hola des de Docker!" > ~/meu-lloc-web/index.html
root@dislexia-VirtualBox:/home/dislexia/Escritorio# docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED    STATUS    PORTS                               NAMES
7784e1e0648e   nginx     "/docker-entrypoint..." 3 minutes ago Up 3 minutes    0.0.0.0:8080->80/tcp, :::8080->80/tcp    meu-nginx
root@dislexia-VirtualBox:/home/dislexia/Escritorio# docker stop meu-nginx
meu-nginx
root@dislexia-VirtualBox:/home/dislexia/Escritorio# docker run --name meu-nginx -p 8080:80 -v ~/meu-lloc-web:/usr/share/nginx/html:ro -d nginx
docker: Error response from daemon: Conflict. The container name "/meu-nginx" is already in use by container "7784e1e0648e4a66544583e5a5cad319440b5ef47688d19e28065e56380766e3". You can re-use that name.
See 'docker run --help'.
root@dislexia-VirtualBox:/home/dislexia/Escritorio# docker run --name meu-nginx -p 8080:80 -v ~/meu-lloc-web:/usr/share/nginx/html:ro -d nginx
x81ca5b9636d93bc7b3be03348cc3abb678e7d8380a9360d23cf8183ad3d0b1
root@dislexia-VirtualBox:/home/dislexia/Escritorio#
```



Preguntes de reflexió (2'5 punts) :

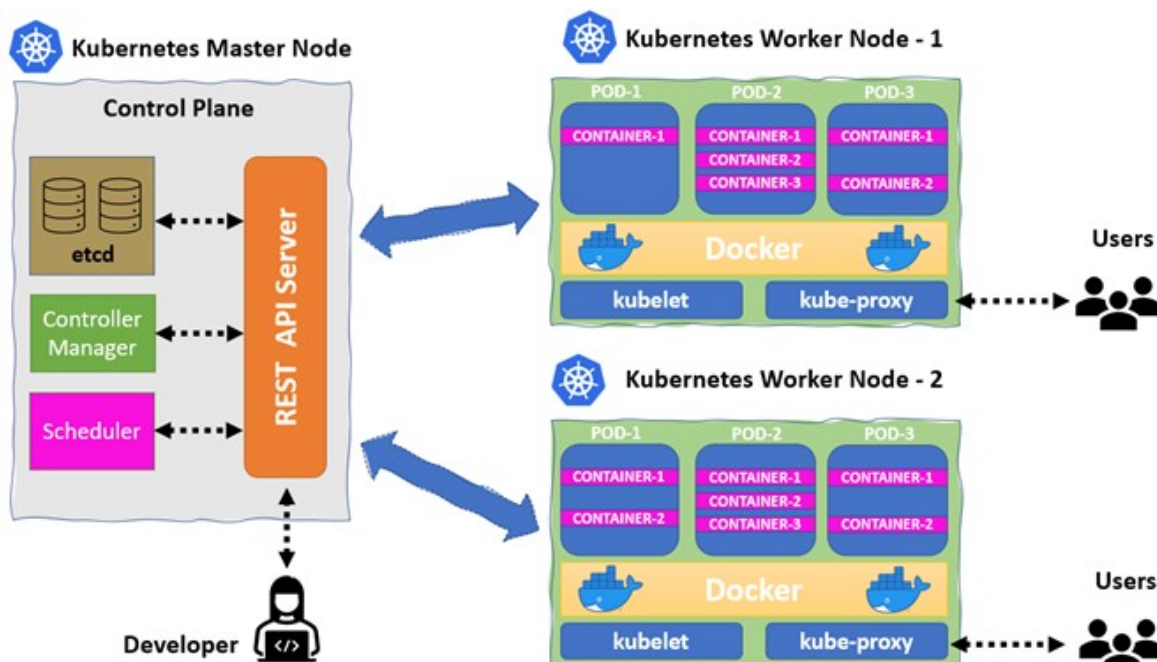
- Quins són els avantatges d'utilitzar Docker per a desplegar aplicacions?

Encapsular aplicacions junto con sus dependencias, aislar estas app en un contenedor además de ejecutarlo independientemente así no interfiere con otros contenedores, utiliza menos recursos en comparación con las máquinas virtuales o otro método para gestionar apps al final ahorras costes, puedes crear varias instancias de un contenedor y distribuir la carga de trabajo de manera eficiente con Kubernetes o Docker Swarm en entornos de alta carga sobre todo, los procesos de implementación se vuelven muy sencillos y automatizables.

En definitiva Docker nos Garantiza: Portabilidad, Aislamiento, Eficiencia, Escalabilidad, Desarrollo y despliegue más rápidos

- Com es pot utilitzar Docker en combinació amb eines com Kubernetes per a la gestió d'aplicacions a gran escala?

Podemos utilizar kubernetes como una herramienta de gestión de múltiples contenedores donde podremos escalar aplicaciones de manera vertical añadiendo más clusters u horizontal añadiendo más pods según demanda), automatización de estos mismos, asignación de recursos y el despliegue, mediante el cluster cada cual tiene dos tipos de nodos los worker nodes, donde se encuentra la aplicación (en el pod se encuentran los containers) y el master node tenemos la comunicación, automatización y gestión de recursos, (sea uno o más de un worker node se unen todos a un master node)



- Quines són les diferències entre muntar un volum i copiar fitxers directament al contenidor?

Cuando nos referimos a montar un volumen queremos decir que en el contenedor tenemos un espacio en nuestro equipo real donde se almacenaran los ficheros que copiamos a un contenedor, dependiendo de como montemos el volumen los datos seran persistentes hasta su borrado o por otra parte que cuando apaguemos el contenedor (no borrarlo) estos archivos se perderan.

docker volume create mis_datos (Persistente)

Volúmen bind / conectados - Es una manera de asociar una carpeta de nuestro equipo real y mapearla como una carpeta dentro de un contenedor. (Persistente)

docker run -d -it --name ubu3 --tmpfs /var/html/tempo ubuntu:17.10 (Persistente)

Aquest exercici t'ha permès practicar amb Docker, des de la descàrrega d'imatges fins a la configuració i desplegament d'un servidor web, i t'ha introduït a conceptes com els volums de Docker, que són essencials per a la gestió eficient dels fitxers i la persistència de dades dins dels contenidors.

3 BIBLIOGRAFIA

Admin. (2020, 21 decembre). *Cómo instalar y usar VLC Media Player en Linux Mint*. Heading. Recuperat el 26 de gener de 2024 de <https://headinghometodinner.org/es/c%C3%B3mo-instalar-y-usar-vlc-media-player-en-linux-mint>

Belinux. (s. f.). *Como compilar e instalar un programa desde el código fuente en Ubuntu*. Recuperat el 28 de gener de 2024 de <https://ubuntinux.blogspot.com/2020/10/instalar-desde--las-fuentes.html>

Braeuner, M. (2022, 18 juliol). *Guía de Docker para principiantes: Cómo crear tu primera aplicación Docker*. freeCodeCamp.org. Recuperat el 31 de gener de 2024 de <https://www.freecodecamp.org/espanol/news/guia-de-docker-para-principiantes-como-crear-tu-primera-aplicacion-docker/>

Equip editorial de IONOS. (2023, 13 setembre). *Docker: la tecnología de contenedores*. IONOS Digital Guide. Recuperat el 2 de febrer de 2024 de <https://www.ionos.es/digitalguide/servidores/know-how/que-es-docker/>

García, J. (2013, 17 març). Synaptic, un gestor debianita en Ubuntu. *Ubunlog*. Recuperat el 27 de gener de 2024 de <https://ubunlog.com/synaptic-un-gestor-debianita-en-ubuntu/>

Institut Obert de Catalunya. (n.d.). TEMA 5: Aplicacions de propòsit general. Cicle Formatiu de Grau Superior en Administració de Sistemes Informàtics en Xarxa. Departament d'Informàtica, 2n curs 2023 - 2024. Recuperat el 26 de gener de 2024 de : https://ioc.xtec.cat/materials/FP/Recursos/fp_asx_m05_/web/fp_asx_m05_htmlindex/index.html

Marco, B. S. (s. f.). *Ejercicios (1). Docker. Aplicaciones web*. Bartolomé Sintés Marco. *www.mclibre.org*. Recuperat el 1 de febrer de 2024 de <https://www.mclibre.org/consultar/webapps/lecciones/docker-1.html>

Pablinux. (2022, 15 novembre). *Cómo instalar un programa en Ubuntu*. *Ubunlog*. Recuperat el 27 de gener de 2024 de <https://ubunlog.com/como-instalar-un-programa-en-ubuntu/>