

Instal·lació Linux Server Ubuntu 20.04.3

[recepta](#)

instal·lació

configuració

```
sudo apt update -y && sudo apt upgrade -y
sudo apt install -y git vim
```

zsh

```
sudo apt install -y fonts-powerline zsh
sh -c "$(curl -fsSL
https://raw.githubusercontent.com/robbyrussell/oh-my-zsh/master/tools/install.sh)"
chsh -s $(which zsh)
```

```
sudo apt-get install -y zsh-syntax-highlighting
git clone https://github.com/zsh-users/zsh-syntax-highlighting.git
${ZSH_CUSTOM}/plugins/zsh-syntax-highlighting
```

```
git clone https://github.com/zsh-users/zsh-syntax-highlighting.git
${ZSH_CUSTOM}/plugins/zsh-syntax-highlighting
```

[; .zshrc](#)

```
ZSH_THEME="agnoster"
plugins=(git
    colored-man-pages
    colorize
    zsh-syntax-highlighting
    vagrant
    zsh-autosuggestions
    timer
)
```

ntp

```
sudo apt install -y ntp
sudo ln -s /usr/share/zoneinfo/Etc/UTC localtime_old
sudo unlink localtime
sudo ln -s /usr/share/zoneinfo/Europe/Andorra /etc/localtime
sudo systemctl restart ntp.service
```

docker

```
sudo apt-get install \
  ca-certificates \
  curl \
  gnupg \
  lsb-release
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/usr/share/keyrings/docker-archive-keyring.gpg
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
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io
sudo usermod -aG docker fidmag
```

seguridad

- instalar librería contraseñas en diccionario:

```
sudo apt install libpam-cracklib
```

- añadir/reemplazar:

```
/etc/pam.d/common-password
```

```
password requisite pam_cracklib.so retry=3 minlen=8 difok=3
```

- parámetros:
 - retry: número de intentos antes de que el sistema devuelva un error en la autenticación y nos expulse.
 - minlen: es la longitud mínima de la contraseña, por defecto está en 8 caracteres.
 - difok: número de caracteres diferentes que debe tener la nueva clave en comparación con la antigua.
 - ucredit: caracteres en mayúscula que debe tener como mínimo o máximo.
 - lcredit: caracteres en minúscula que debe tener como mínimo o máximo.
 - dcredit: el número de dígitos que debe tener como mínimo o máximo.
 - ocredit: el número de otros caracteres (símbolos) que debe tener la clave como mínimo o máximo.
 - para los credit:
 - lcredit=-2 : significa que como mínimo debe tener 2 caracteres en minúscula.
 - lcredit=+2 : significa que como máximo debe tener 2 caracteres en minúscula.
- expira la contraseña y obliga a cambio en próximo login:

```
passwd -e <USUARIO>
```

- caducidad:

```
passwd -w 5 -x 30 -i 1 <USUARIO>
```

- **w**: aviso X días antes de la caducidad
- **x**: expira cada X días
- **i**: desactiva la cuenta a los X días si no ha habido cambio de contraseña. Solo root puede reactivar.

/via:<https://www.redeszone.net/tutoriales/seguridad/configurar-politica-contrasenas-debian/>

ufw

```
sudo apt install -y ufw
ufw default deny incoming
ufw default allow outgoing
ufw allow ssh
ufw enable
ufw status
ufw app list
```

/via: <https://community.hetzner.com/tutorials/simple-firewall-management-with-ufw>

ssh

/via: <https://community.hetzner.com/tutorials/securing-ssh>

; /etc/ssh/sshd_config

```
AllowTcpForwarding no           # Disables port forwarding.
X11Forwarding no                # Disables remote GUI view.
AllowAgentForwarding no        # Disables the forwarding of the SSH
login.
MaxAuthTries 2
AllowUsers fidmag
ClientAliveInterval 300
ClientAliveCountMax 1
PermitRootLogin no
```

```
sudo sshd -t # test configuration
sudo systemctl restart sshd
```

fail2ban

```
sudo apt install -y fail2ban
sudo systemctl enable fail2ban
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
sudo vim /etc/fail2ban/jail.local # add enabled = true in [sshd] section
```

sudo

sudoreplay

```
sudo visudo
# add: Defaults log_output
sudo sudoreplay -l # list sessions
sudo sudoreplay <TSID>
```

su

```
sudo groupadd su
sudo usermod -a -G su fidmag
sudo dpkg-statoverride --update --add root admin 4750 /bin/su
```

IPv6 disable

```
# comprobar estado
ip a | grep inet6

# desactivar ya
sudo sysctl -w net.ipv6.conf.all.disable_ipv6=1
sudo sysctl -w net.ipv6.conf.default.disable_ipv6=1
sudo sysctl -w net.ipv6.conf.lo.disable_ipv6=1

# persistencia (solo activa tras un reboot)
echo "net.ipv6.conf.all.disable_ipv6 = 1" | sudo tee -a /etc/sysctl.conf
echo "net.ipv6.conf.default.disable_ipv6 = 1" | sudo tee -a /etc/sysctl.conf
echo "net.ipv6.conf.lo.disable_ipv6=1" | sudo tee -a /etc/sysctl.conf
```

updates

expand filesystem

```
parted -l /dev/sda
sudo cfdisk
sudo resize2fs /dev/sda2
```

canvi IP

```
sudo vim /etc/netplan/00-installer-config.yaml
sudo netplan apply
```

canvi hostname

```
sudo vim /etc/hostname
```

scripts

- [git://fidmag.org:/home/git/vmw-master](https://fidmag.org/)
 - setup.sh
 - post-clone.sh
 - new-docker-certificates.sh
- certificats:
 - en local:

```
scp CA_FIDMAG.crt CA_FIDMAG.key openssl.cnf fidmag@10.213.6.154:~
```

- en remot:

```
sudo mv CA_FIDMAG.crt CA_FIDMAG.key openssl.cnf /root
```

- en local:

```
mkdir ~/Documents/certificats-CA/10.213.6.154; scp vmware-master:~/ .docker/\* ~/Documents/certificats-CA/10.213.6.154
```

apache+php

```
sudo apt install apache2
```

```
sudo apt install ca-certificates apt-transport-https software-properties-common -y  
echo "deb https://packages.sury.org/php/ $(lsb_release -sc) main" | sudo tee  
/etc/apt/sources.list.d/sury-php.list  
wget -q0 - https://packages.sury.org/php/apt.gpg | sudo apt-key add -  
#wget -q0 - https://packages.sury.org/php/apt.gpg | gpg --dearmor | sudo tee  
/usr/share/keyrings/php-archive-keyring.gpg  
#echo "deb [signed-by=/usr/share/keyrings/php-archive-keyring.gpg]  
https://packages.sury.org/php/ $(lsb_release -sc) main" | sudo tee  
/etc/apt/sources.list.d/sury-php.list  
sudo apt update -y  
sudo apt install php8.0  
sudo apt install php8.0-{mysql,cli,common,xml,fpm,curl,mbstring,zip,gd}  
#apt install php php-common php-xml php-gd php-mbstring php-tokenizer php-json php-  
bcmath php-zip -y
```

[/etc/php/8.0/apache2/php.ini](#)

```
upload_max_filesize = 32M  
post_max_size = 48M  
memory_limit = 256M  
max_execution_time = 600  
max_input_vars = 3000  
max_input_time = 1000
```

```
systemctl restart apache2
```

default php

```
a2dismod php7.4 # disable 7.4  
a2enmod php8.0  
systemctl restart apache2
```

/via: <https://www.howtoforge.com/how-to-install-php-8-on-debian-11/>

/via: <https://computingforgeeks.com/how-to-install-php-on-debian-linux/>

From:
<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:
<https://miguelangel.torresegea.es/wiki/linux:install:server:ubuntu20043?rev=1642154997>

Last update: **14/01/2022 02:09**

