



Born2beRoot - Debian

YOU CAN DO ANYTHING YOU WANT TO DO

**VIRTUAL
MACHINE**

THIS IS YOUR WORLD

Contents

1.	Resize Logical Volumes (LVM)	4
	Parte Obligatoria.....	4
	Parte Obligatoria + Bonus.....	4
	Configuración parte + obligatoria.....	5
	Redimensionamiento de los volúmenes lógicos.....	5
	Configuración del área de intercambio (swap)	5
2.	Configuración del sudo	6
3.	Instalación y Configuración de SSH	7
4.	Instalación y configuración de UFW	12
5.	Conexión al VirtualBox vía SSH	13
6.	Políticas de contraseña.....	18
7.	Creación de usuarios y contraseñas	23
8.	Monitorización con CRONTAB CONFIGURATION	24
9.	Instalación y configuración de LLMP STACK	29
10.	Instalación y configuración de <i>LIGHTTPD</i>	30
11.	Instalación y configuración de MariaDB.....	31
12.	Instalación de <i>PHP & EXTENSIONS</i>	32
13.	WORDPRESS SETUP	33
14.	CONFIGURAR UN SERVICIO DE WORDPRESS A TU ELECCION.....	34
15.	Pasos prohibidos y verificaciones	36
15.1.	Instalar interfaz gráfica.....	36
15.2.	Verificar última versión	36
15.3.	Verificar particiones (Obligatoria)	36
15.4.	Preguntas.....	36
15.5.	Revisar SSH.....	37
15.6.	Revisar UFW o FIREWALLD	37
15.7.	Revisión 1.....	37
15.8.	Creación usuario y grupo en tu delante	38
15.9.	Script Monitoring.sh.....	38

1. Resize Logical Volumes (LVM)

“Resize Logical Volumes (LVM)” significa **cambiar el tamaño de una partición lógica** en un sistema que usa **LVM (Logical Volume Manager)** — es decir, **ampliar o reducir el espacio disponible** en una parte del disco sin tener que recrear todo el sistema.

De acuerdo con el formato que nos dieron, debe presentarse de esta forma o parecido, para eso debes configurar bien la maquina virtual, para que se parezca los más exacto.

Parte Obligatoria

```
wil@wil:~$ lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda        8:0    0   8G  0 disk
└─sda1     8:1    0 487M 0 part  /boot
└─sda2     8:2    0   1K  0 part
└─sda5     8:5    0 7.5G 0 part
  └─sda5_crypt 254:0  0 7.5G 0 crypt
    ├─wil--vg-root 254:1  0 2.8G 0 lvm   /
    ├─wil--vg-swap_1 254:2  0 976M 0 lvm   [SWAP]
    ├─wil--vg-home  254:3  0 3.8G 0 lvm   /home
sr0       11:0   1 1024M 0 rom

wil@wil:~$ _
```

Parte Obligatoria + Bonus

```
# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda        8:0    0 30.8G 0 disk
└─sda1     8:1    0 500M 0 part  /boot
└─sda2     8:2    0   1K  0 part
└─sda5     8:5    0 30.3G 0 part
  └─sda5_crypt 254:0  0 30.3G 0 crypt
    ├─LVMGroup-root 254:1  0 10G  0 lvm   /
    ├─LVMGroup-swap 254:2  0 2.3G 0 lvm   [SWAP]
    ├─LVMGroup-home 254:3  0 5G   0 lvm   /home
    ├─LVMGroup-var  254:4  0 3G   0 lvm   /var
    ├─LVMGroup-srv  254:5  0 3G   0 lvm   /srv
    ├─LVMGroup-tmp  254:6  0 3G   0 lvm   /tmp
    └─LVMGroup-var--log 254:7  0 4G   0 lvm   /var/log
sr0       11:0   1 1024M 0 rom
```

Cuando creas la máquina virtual, se genera automáticamente con una configuración base. Como todas se parecen bastante, no hay problema — todo tranqui.

Aun así, te recomiendo que al crearla elijas desde el principio los valores correctos de:

- Base Memory (RAM)
- Processors (CPU)
- Disk Size (tamaño del disco)

Así te aseguras de que tu máquina se parezca lo más posible al formato exigido, tanto en la **parte obligatoria** como en la **parte bonus**. Si planeas hacer la parte bonus, lo mejor es que configures el tamaño del disco según los requisitos del bonus desde el inicio, directamente en la configuración de

VirtualBox. De esa forma, evitarás problemas después con el espacio o con tener que redimensionar el disco más adelante.

Recomendación:

Parte Obligatoria

Esta configuración es la más pequeña y básica, enfocada en la eficiencia.

- Base Memory (RAM): 1024 MB (o 1 GB)
- Processors (CPU): 1
- Disk Size (Tamaño del disco): 8 GB

Parte Obligatoria + Bonus

Esta configuración es más grande y compleja, para acomodar múltiples particiones LVM.

- Base Memory (RAM): 2048 MB (o 2 GB)
- Processors (CPU): 2
- Disk Size (Tamaño del disco): 35 GB (Mínimo; 40 GB es más seguro)

Configuración parte + obligatoria

Para obtener el resultado mostrado anteriormente, debemos configurar lo siguiente:

Recuerda: todos estos comandos deben ejecutarse como **root**, o bien anteponiendo **sudo** si no lo eres.

Redimensionamiento de los volúmenes lógicos

```
Sudo lvresize -r -L 10G /dev/LVMGroup/root      # 1  
Sudo lvresize -r -L 5G /dev/LVMGroup/home       # 2  
Sudo lvresize -r -L 3G /dev/LVMGroup/var        # 3  
Sudo lvresize -r -L 3G /dev/LVMGroup/srv         # 4  
Sudo lvresize -r -L 3G /dev/LVMGroup/tmp         # 5  
Sudo lvresize -r -L 4G /dev/LVMGroup/var/log     # 6
```

Configuración del área de intercambio (swap)

Comandos previos

```
sudo vgs  
sudo swapoff -a
```

Redimensionar el volumen lógico del swap

```
lvresize -r -L 2.29G /dev/LVMGroup/swap  
sudo mkswap /dev/LVMGroup/swap
```

Verificación final - Para comprobar la estructura de los volúmenes

```
Lsblk
```

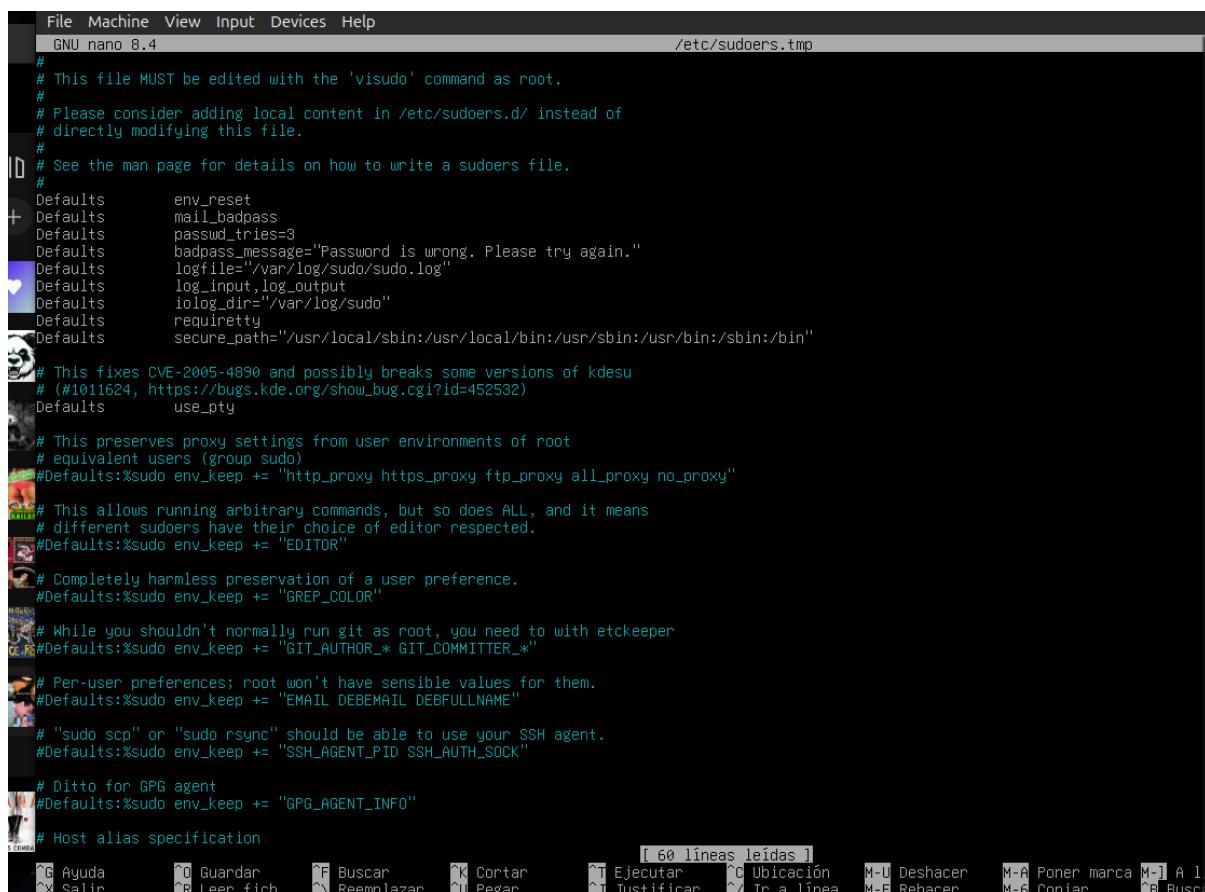
Este comando mostrará los distintos volúmenes lógicos creados y sus tamaños, confirmando que la configuración se aplicó correctamente.

2. Configuración del sudo

- Defaults passwd_tries=3
- Defaults badpass_message="Password is wrong. Please try again".
- Defaults logfile="/var/log/sudo/sudo.log"
- Defaults log_input,log_output
- Defaults iolog_dir="/var/log/sudo"
- Defaults requiretty

Ahora creamos el archivo de logs -> sudo mkdir -p /var/log/sudo

Sudo touch /var/log/sudo/sudo.log



```
File Machine View Input Devices Help
GNU nano 8.4                                     /etc/sudoers.tmp

#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset
Defaults    mail_badpass
Defaults    passwd_tries=3
Defaults    badpass_message="Password is wrong. Please try again."
Defaults    logfile="/var/log/sudo/sudo.log"
Defaults    log_input,log_output
Defaults    iolog_dir="/var/log/sudo"
Defaults    requiretty
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"

# This fixes CVE-2005-4890 and possibly breaks some versions of kdesu
# (#1011624, https://bugs.kde.org/show_bug.cgi?id=452532)
Defaults    use_pty

# This preserves proxy settings from user environments of root
# & equivalent users (group sudo)
#Defaults:@sudo env_keep += "http_proxy https_proxy ftp_proxy all_proxy no_proxy"
#Defaults:@sudo env_keep += "EDITOR"
#Defaults:@sudo env_keep += "GIT_AUTHOR_* GIT_COMMITTER_"
#Defaults:@sudo env_keep += "EMAIL DEBEMAIL DEBFULLNAME"
#Defaults:@sudo env_keep += "SSH_AGENT_PID SSH_AUTH_SOCK"
#Defaults:@sudo env_keep += "GPG_AGENT_INFO"
#Host alias specification
[G Ayuda   [O Guardar   [F Buscar   [K Cortar   [T Ejecutar   ^D Ubicación   M-U Deshacer   M-A Poner marca [M-J A 1
[X Salir   [R Leer fich. [Reemplazar [Pegar   [Justificar   ^I Ir a línea   M-F Rehacer   M-G Copiar   ^R Busca
```

3. Instalación y Configuración de SSH

Sudo apt install openssh-server -y

```
File Machine View Input Devices Help
Preparando para desempaquetar .../15-libx11-6_2%3a1.8.12-1_amd64.deb ...
Desempaquetando libx11-6:amd64 (2:1.8.12-1) ...
Seleccionando el paquete libxext6:amd64 previamente no seleccionado,
Preparando para desempaquetar .../16-libxext6_2%3a1.3.4-1+b3_amd64.deb ...
Desempaquetando libxext6:amd64 (2:1.3.4-1+b3) ...
Seleccionando el paquete libxmuu1:amd64 previamente no seleccionado,
Preparando para desempaquetar .../17-libxmuu1_2%3a1.1.3-3+b4_amd64.deb ...
Desempaquetando libxmuu1:amd64 (2:1.1.3-3+b4) ...
Seleccionando el paquete xauth previamente no seleccionado,
Preparando para desempaquetar .../18-xauth_1%3a1.1.2-1.1_amd64.deb ...
Desempaquetando xauth (1:1.1.2-1.1) ...
Configurando runit-helper (2.16.4) ...
Configurando libxau6:amd64 (1:1.0.11-1) ...
Configurando libxdmcp6:amd64 (1:1.1.5-1) ...
Configurando libxcb1:amd64 (1.17.0-2+b1) ...
Configurando libcbor0.10:amd64 (0.10.2-2) ...
Configurando liburap0:amd64 (7.6.q-36) ...
Configurando libx11-data (2:1.8.12-1) ...
Configurando libpam-systemd:amd64 (257.8-1~deb13u2) ...
Configurando libx11-6:amd64 (2:1.8.12-1) ...
Configurando libutmpd0:amd64 (0.73.0-3) ...
Configurando libtalloc2:1:amd64 (1.15.0-1+b1) ...
Configurando libxmuu1:amd64 (2:1.1.3-3+b4) ...
Configurando ncurses-term (6.5+20250216-2) ...
Configurando openssh-client (1:10.0p1-7) ...
Created symlink /etc/systemd/user/sockets.target.wants/ssh-agent.socket → '/usr/lib/systemd/user/ssh-agent.socket'.
Configurando libxext6:amd64 (2:1.3.4-1+b3) ...
Configurando dbus-user-session (1.16.2-2) ...
Configurando xauth (1:1.1.2-1.1) ...
Configurando openssh-sftp-server (1:10.0p1-7) ...
Configurando openssh-server (1:10.0p1-7) ...
Creating config file /etc/ssh/sshd_config with new version
Creating SSH2 RSA key; this may take some time ...
3072 SHA256:NNKVlQ2rSxItH01L1KXueY+001LaY4KpUbgSeppMBE root@serromer42 (RSA)
Creating SSH2 ECDSA key; this may take some time ...
3072 SHA256:zX87ffFrm2z+u xoOKJc2hNdpR0luApupizDnsSiUs+06Ic root@serromer42 (ECDSA)
Creating SSH2 ED25519 key; this may take some time ...
3072 SHA256:0kTlD4fMb2/XQQCnEUEqLEGLhysSaPWNDR0EscMiClU root@serromer42 (ED25519)
Creating user 'sshd' (sshd user) with UID 990 and GID 65534.
Created symlink '/etc/systemd/system/sshd.service' → '/usr/lib/systemd/system/ssh.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/ssh.service' → '/usr/lib/systemd/system/ssh.service'.
ssh.socket is a disabled or a static unit, not starting it.
Created symlink '/etc/systemd/system/ssh.service.wants/sshd-keygen.service' → '/usr/lib/systemd/system/sshd-keygen.service';
Created symlink '/etc/systemd/system/sshd.service.wants/sshd-keygen.service' → '/usr/lib/systemd/system/sshd-keygen.service';
Created symlink '/etc/systemd/system/sshd@.service.wants/sshd-keygen.service' → '/usr/lib/systemd/system/sshd-keygen.service';
Created symlink '/etc/systemd/system/ssh.socket.wants/sshd-keygen.service' → '/usr/lib/systemd/system/sshd-keygen.service'.
Procesando disparadores para libc-bin (2.41-12) ...
root@serromer42: "# rm -rf sudo.loh"
root@serromer42: "# sudo apt install openssh-server -y"
```

Nano /etc/ssh/sshd_config

```
File Machine View Input Devices Help
root@serromer42:~# pwd
/root
root@serromer42:~# cd /etc/
root@serromer42:/etc# ls
alternatives      cron.daily      dpkg      hosts.allow    locale.conf    network     rc3.d      ssh
anacrontab        cron.hourly    e2scrub.conf hosts.deny     locale.gen     networks    rc4.d      ssl
apparmor          cron.monthly   emacs      ifplugd      localtime    nftables.conf rc5.d      subgid
apparmor.d        cron.weekly    environment init.d      login.defs   nsswitch.conf rc6.d      subgid-
apt              cron.yearly    ethertypes initramfs-tools logrotate.conf opt       rc8.d      subuid
+ bash.bashrc      cron.daily      fstab      inputrc      logrotate.d os-release  resolv.conf subuid-
bindresvport.blacklist crypttab      gai.conf    issue       lvm          pam.conf    rmt       sudo.conf
binfmt.d          cron.hourly    group      issue.net    machine-id  pam.d      rpc       sudoers
bluetooth         cron.monthly   grub.d    kernel      mke2fs.conf  passwd     runit     sudoers.d
ca-certificates   cron.weekly    gshadow    ld.so.cache  modules     profile.d  selinux   supercat
ca-certificates.conf default      gshadow-   ld.so.conf   modules-load.d protocols  services  sv
console-setup     deluser.conf   gss       ld.so.conf.d motd       rc0.d      shadow-  systemctl
credstore         dhcpcd.conf   host.conf  libaudit.conf mtab       rc1.d      shadow-  systemd
credstore.encrypted dhcpcd.conf  hostname  libnl-3     nanorc     rc2.d      shells   terminfo
cron.d            dictionaries-common hosts     locale.alias netconfig  skel      tmpfiles.d
root@serromer42:~# cd
Display all 141 possibilities? (y or n)
root@serromer42:~# cd ssh
root@serromer42:~/ssh# pwd
/etc/ssh
root@serromer42:~/ssh# ls
moduli      ssh_config.d  sshd_config.d  ssh_host_ecdsa_key  ssh_host_ecdsa_key.pub  ssh_host_ed25519_key  ssh_host_ed25519_key.pub  ssh_host_rsa_key  ssh_host_rsa_key.pub
ssh_config  sshd_config   ssh_host_ecdsa_key  ssh_host_ecdsa_key      ssh_host_ecdsa_key      ssh_host_ed25519_key      ssh_host_ed25519_key      ssh_host_rsa_key      ssh_host_rsa_key
root@serromer42:~/ssh# nano ssh_config
```

Colocamos el puerto que nos dicen 4242 y le ponemos que esta prohibido conectarse como root.

Sudo nano /etc/ssh/sshd_config

```
GNU nano 8.4                                         sshd_config *
```

```
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 4242_
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile      .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

^G Ayuda          ^O Guardar        ^F Buscar        ^K Cortar        ^T Ejecutar       ^C Ubicación     M-U D
^X Salir          ^R Leer fich.    ^A Reemplazar    ^U Pegar         ^J Justificar    ^/ Ir a línea   M-E R
```

Sudo nano /etc/ssh/sshd_config

```
GNU nano 8.4                               /etc/ssh/ssh_config *
```

```
# This is the ssh client system-wide configuration file. See
# ssh_config(5) for more information. This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
# 1. command line options
# 2. user-specific file
# 3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options. For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.

Include /etc/ssh/ssh_config.d/*.conf

Host *
# ForwardAgent no
# ForwardX11 no
# ForwardX11Trusted yes
# PasswordAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIDelegateCredentials no
# GSSAPIKeyExchange no
# GSSAPITrustDNS no
# BatchMode no
# CheckHostIP no
# AddressFamily any
# ConnectTimeout 0
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/id_rsa
# IdentityFile ~/.ssh/id_dsa
# IdentityFile ~/.ssh/id_ecdsa
# IdentityFile ~/.ssh/id_ed25519
Port 4242
PermitRootLogin no
# Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,3des-cbc
# MACs hmac-md5,hmac-sha1,umac-64@openssh.com
# EscapeChar ~
# Tunnel no
# TunnelDevice any:any
```

```
File Machine View Input Devices Help
root@serromer42:/etc/ssh# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
  Active: active (running) since Sun 2025-11-02 16:49:41 CET; 5min ago
    Invocation: 400dd1f794144f8ba0181acb89c06657
      Docs: man:sshd(8)
             man:sshd_config(5)
    Main PID: 1719 (sshd)
       Tasks: 1 (limit: 1106)
      Memory: 1.9M (peak: 2M)
        CPU: 43ms
       CGroup: /system.slice/ssh.service
               └─1719 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

nov 02 16:49:41 serromer42 systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
nov 02 16:49:41 serromer42 sshd[1719]: Server listening on 0.0.0.0 port 22.
nov 02 16:49:41 serromer42 sshd[1719]: Server listening on :: port 22.
nov 02 16:49:41 serromer42 systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
root@serromer42:/etc/ssh# service ssh status
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
  Active: active (running) since Sun 2025-11-02 16:49:41 CET; 5min ago
    Invocation: 400dd1f794144f8ba0181acb89c06657
      Docs: man:sshd(8)
             man:sshd_config(5)
    Main PID: 1719 (sshd)
       Tasks: 1 (limit: 1106)
      Memory: 1.9M (peak: 2M)
        CPU: 43ms
       CGroup: /system.slice/ssh.service
               └─1719 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

nov 02 16:49:41 serromer42 systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
nov 02 16:49:41 serromer42 sshd[1719]: Server listening on 0.0.0.0 port 22.
nov 02 16:49:41 serromer42 sshd[1719]: Server listening on :: port 22.
nov 02 16:49:41 serromer42 systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
root@serromer42:/etc/ssh#
```

4. Instalación y configuración de UFW

Sudo apt install ufw -y – Instalamos el servicio.

Sudo ufw enable -> activamos el servicio.

Sudo ufw allow 4242 -> Habilitamos el puerto 4242

Sudo ufw status -> Vemos el estado del puerto

```
Born2beRoot1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
serromer@serromer42:~$ sudo ufw allow 4242
Rule added
Rule added (v6)
serromer@serromer42:~$ sudo ufw status
Status: active

To           Action    From
--           ----     ---
4242          ALLOW     Anywhere
4242 (v6)      ALLOW     Anywhere (v6)

serromer@serromer42:~$ sudo apt install ufw -y
ufw ya está en su versión más reciente (0.36.2-9).
Summary:
Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 0
serromer@serromer42:~$ sudo ufw enable
Firewall is active and enabled on system startup
serromer@serromer42:~$ sudo ufw allow 4242
Skipping adding existing rule
Skipping adding existing rule (v6)
serromer@serromer42:~$ sudo ufw status
Status: active

To           Action    From
--           ----     ---
4242          ALLOW     Anywhere
4242 (v6)      ALLOW     Anywhere (v6)

serromer@serromer42:~$
```

5. Conexión al VirtualBox vía SSH

La conexión dentro del terminal de maquina virtual debe estar activa en el puerto 4242, es decir ssh
`<username>@127.0.0.1 -p 4242`

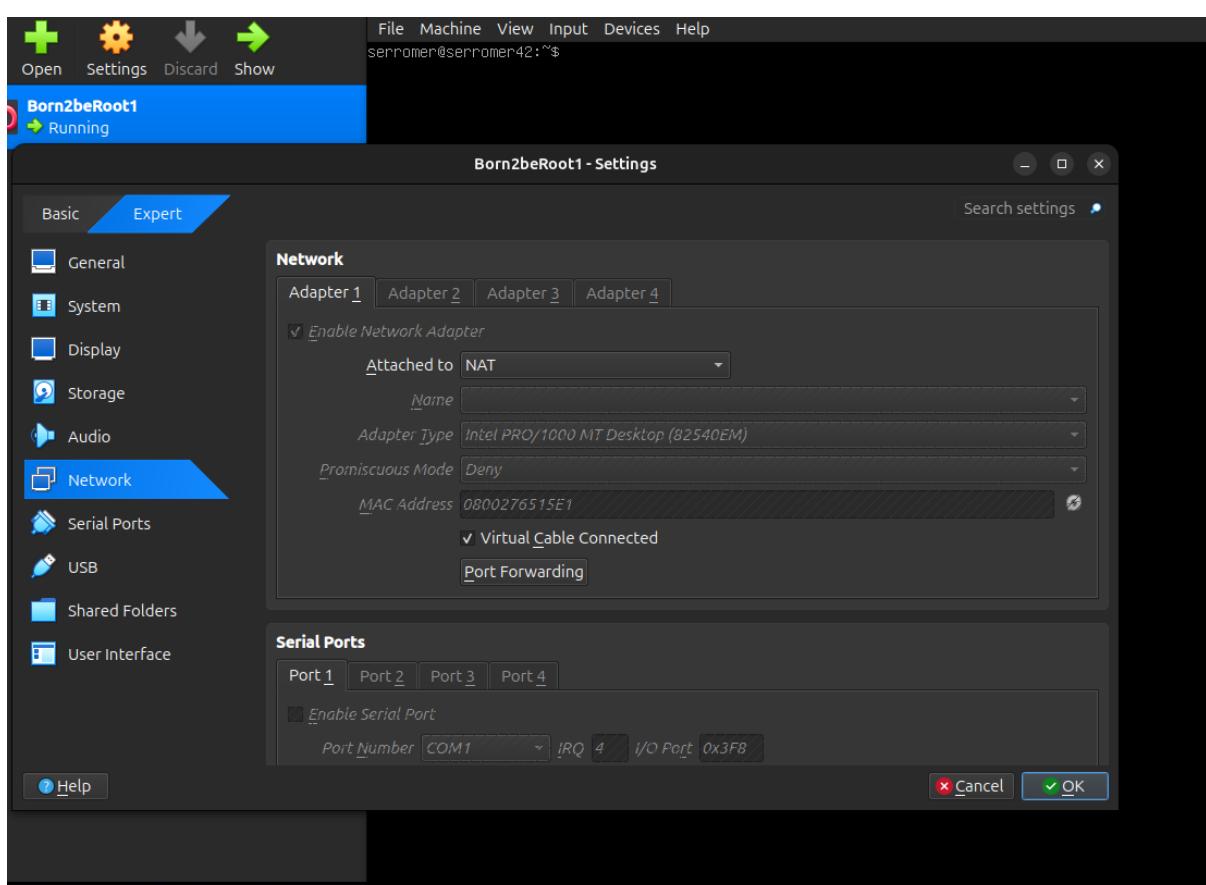
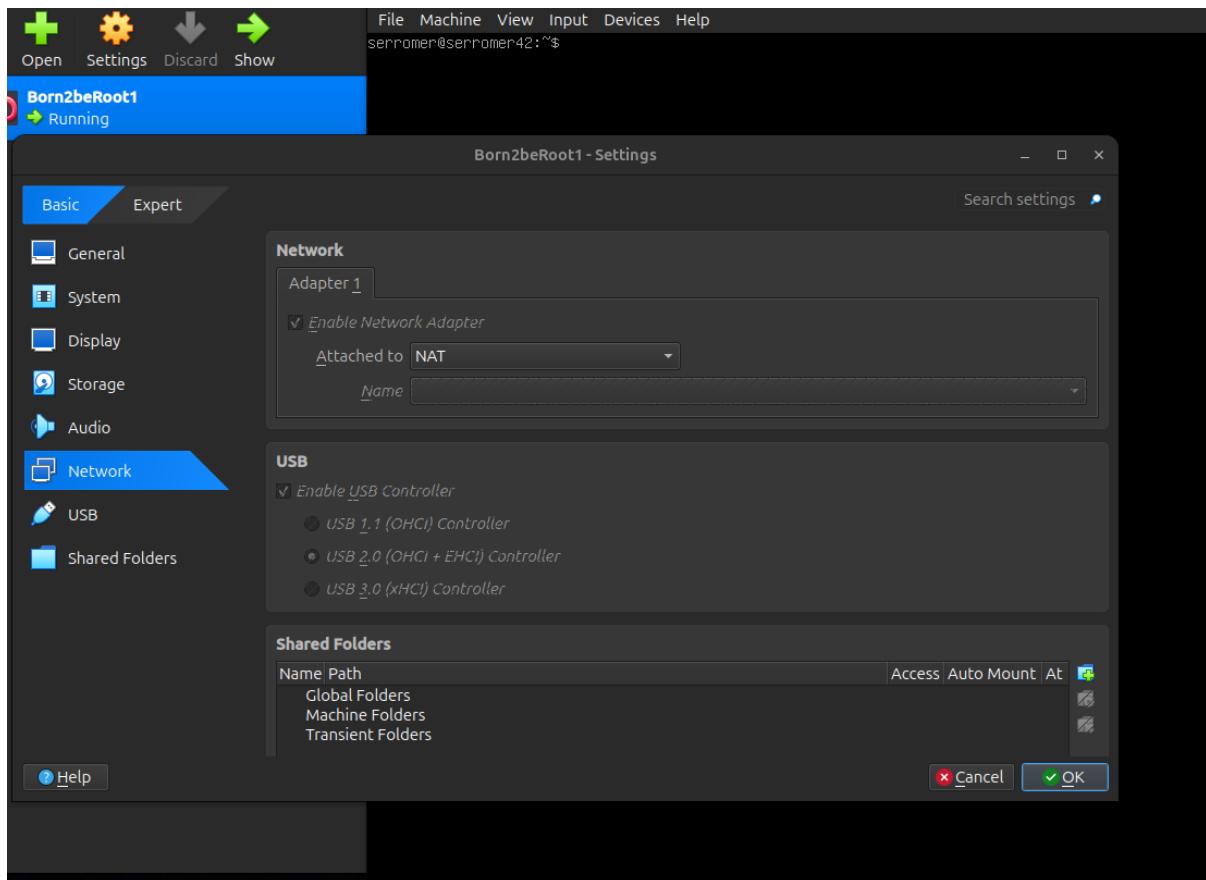
Conexión por dentro de la maquina

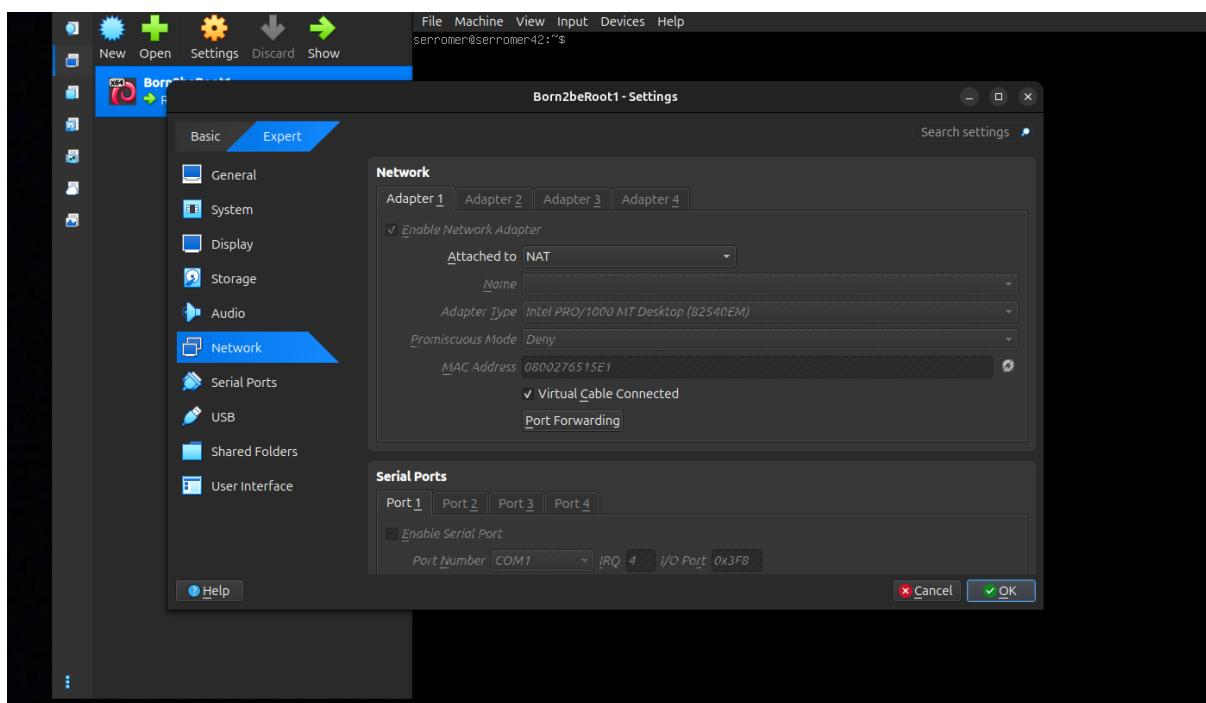
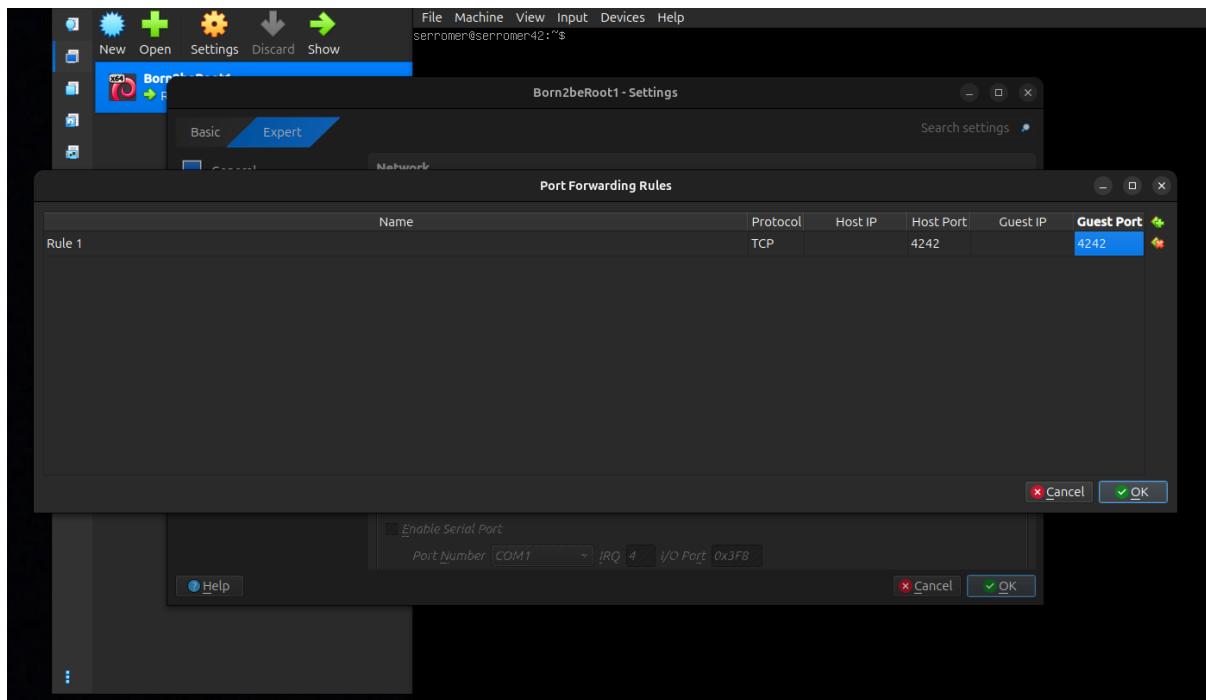
```
File Machine View Input Devices Help
serromer@serromer42:~$ ssh serromer@127.0.0.1 -p 4242
The authenticity of host '[127.0.0.1]:4242 ([127.0.0.1]:4242)' can't be established.
ED25519 key fingerprint is SHA256:0kT1D4fMb2/XQQCnEUEqLEGhysSaPKNDR0EscMiClU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[127.0.0.1]:4242' (ED25519) to the list of known hosts.
serromer@127.0.0.1's password:
Linux serromer42 6.12.48+deb13-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.12.48-1 (2025-09-20) 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.
serromer@serromer42:~$ ca
```

Conexión por fuera de la maquina





sudo apt install net-tools -> Si no tienes instalado

Netstat -tuln | grep – 4242 para verificar si el puerto 4242 esta escuchando para conexiones fuera.

A screenshot of a terminal window titled "sergio-alejandro@sergiodevelop:~". The terminal shows the command "netstat -tuln | grep 4242" being run, with the output:

```
tcp        0      0 0.0.0.0:4242          0.0.0.0:*                LISTEN
```

Below the terminal, there is explanatory text:

[sudo apt install net-tools](#) -> Si no tienes instalado
[Netstat -tuln | grep -i 4242](#) para verificar si el puerto 4242 [esta](#) escuchando para conexiones fuera.

Comando parecido ss -tuln | grep 4242

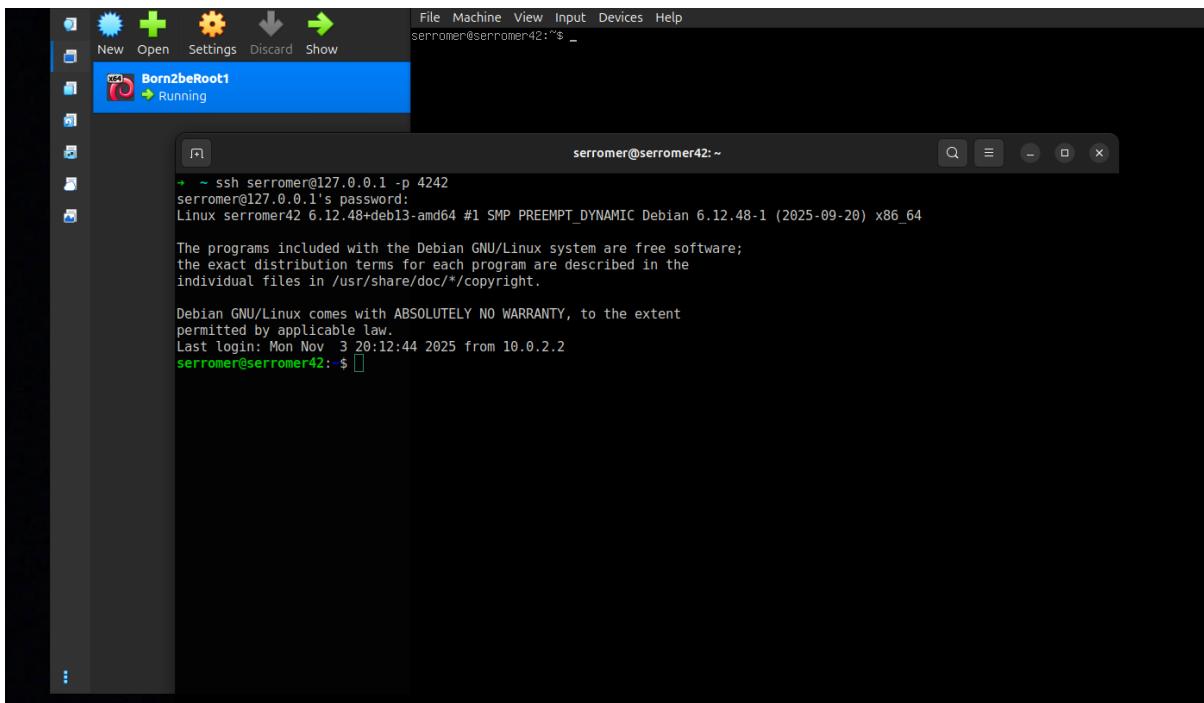
A screenshot of a terminal window titled "sergio-alejandro@sergiodevelop:~". The terminal shows the command "ss -tuln | grep -i 4242" being run, with the output:

```
tcp        0      0 0.0.0.0:4242          0.0.0.0:*                LISTEN
```

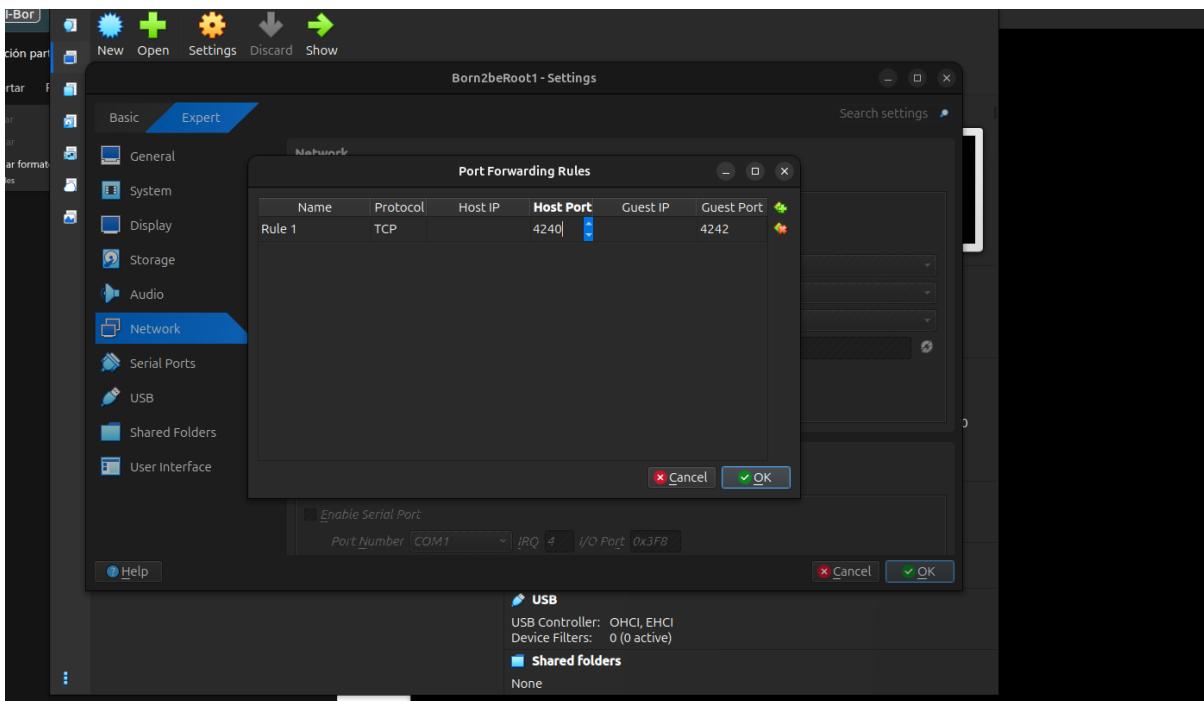
Puedes conectarte, ya.

Ssh [serromer@127.0.0.1](ssh://serromer@127.0.0.1) port 4242

Ssh [serromer@127.0.0.1](ssh://serromer@127.0.0.1) -p 4242

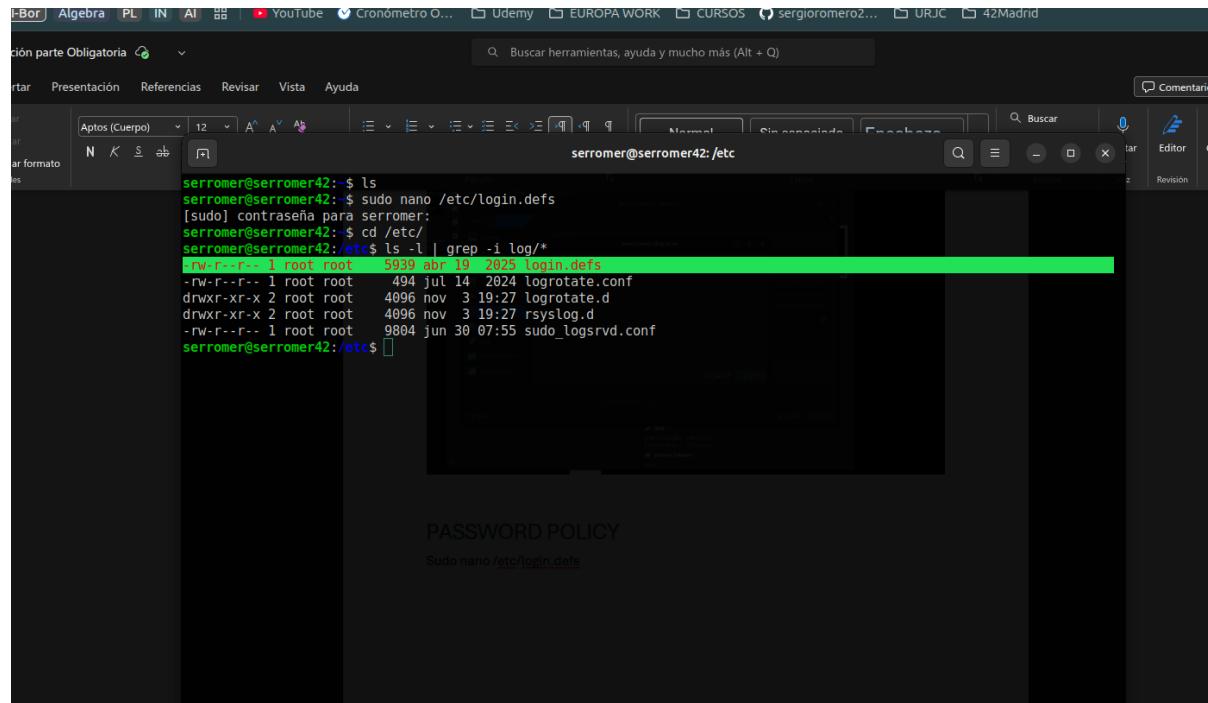


Si cambias el puerto en el virtual box, cambias el puerto con el que te conectas por SSH. Por ejemplo



6. Políticas de contraseña

Sudo nano /etc/login.defs

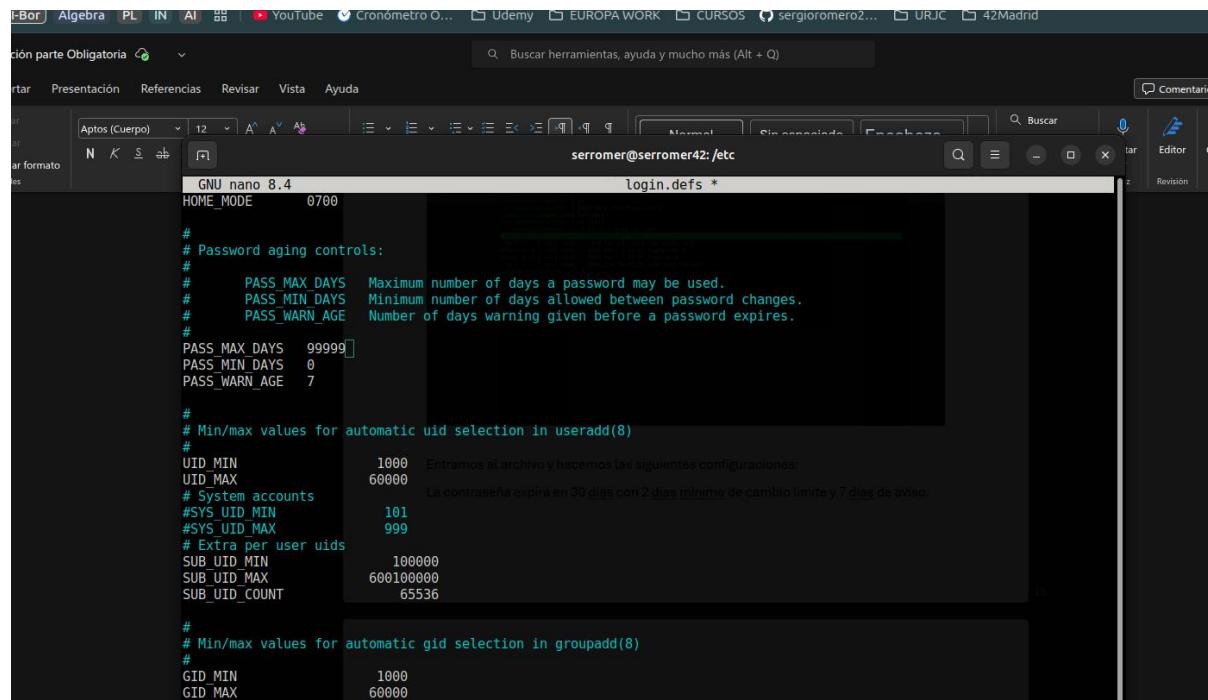


```
serromer@serromer42:~$ ls
serromer@serromer42:~$ sudo nano /etc/login.defs
[sudo] contraseña para serromer:
serromer@serromer42:~$ cd /etc/
serromer@serromer42:~/etc$ ls -l | grep -i log/*
-rw-r--r-- 1 root root 5939 abr 19 2025 login.defs
-rw-r--r-- 1 root root 494 jul 14 2024 logrotate.conf
drwxr-xr-x 2 root root 4096 nov  3 19:27 logrotate.d
drwxr-xr-x 2 root root 4096 nov  3 19:27 syslog.d
-rw-r--r-- 1 root root 9804 jun 30 07:55 sudo_logsrvd.conf
serromer@serromer42:~/etc$
```

PASSWORD POLICY
Sudo nano /etc/login.defs

Entramos al archivo y hacemos las siguientes configuraciones:

La contraseña expira en 30 días con 2 días mínimo de cambio límite y 7 días de aviso.



```
GNU nano 8.4                               login.defs *
HOME_MODE          0700

#
# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be used.
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#      PASS_WARN_AGE    Number of days warning given before a password expires.
#
#      PASS_MAX_DAYS  99999
#      PASS_MIN_DAYS  0
#      PASS_WARN_AGE   7
#
#      # Min/max values for automatic uid selection in useradd(8)
#
#      UID_MIN          1000  Entramos al archivo y hacemos las siguientes configuraciones:
#      UID_MAX          60000
#      # System accounts
#      #SYS_UID_MIN     101
#      #SYS_UID_MAX     999
#      # Extra per user uids
#      SUB_UID_MIN       100000
#      SUB_UID_MAX      600100000
#      SUB_UID_COUNT     65536
#
#      # Min/max values for automatic gid selection in groupadd(8)
#
#      GID_MIN          1000
#      GID_MAX          60000
```

```

GNU nano 8.4          login.defs *
HOME_MODE      0700

#
# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be used.
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#      PASS_WARN_AGE    Number of days warning given before a password expires.
#
#      PASS_MAX_DAYS   30
#      PASS_MIN_DAYS   2
#      PASS_WARN_AGE    7           Entramos al archivo y hacemos las siguientes configuraciones:
#                                La contraseña expira en 30 días con 2 días mínimo de cambio límite y 7 días de aviso.

#
# Min/max values for automatic uid selection in useradd(8)
#
UID_MIN        1000
UID_MAX        60000
# System accounts
#SYS_UID_MIN   101
#SYS_UID_MAX   999
# Extra per user uids
SUB_UID_MIN    100000
SUB_UID_MAX    600100000
SUB_UID_COUNT  65536

#
# Min/max values for automatic gid selection in groupadd(8)
#
GID_MIN        1000
GID_MAX        60000

```

Ahora pasamos a Instalar la biblioteca de comprobación de la calidad de las contraseñas.

Sudo apt-get install libpam-pwquality -y

```

serromer@serromer42:~$ sudo apt install libpam-pwquality -y
Installing:
  libpam-pwquality

Installing dependencies:
  cracklib-runtime  file  libcrack2  libmagic-mgc  libmagic1t64  libpwquality-common  libpwquality

Summary:
  Upgrading: 0, Installing: 8, Removing: 0, Not Upgrading: 0
  Download size: 757 kB
  Space needed: 12,1 MB / 6.410 MB available

Des:1 http://deb.debian.org/debian trixie/main amd64 libmagic-mgc amd64 1:5.46-5 [338 kB]
Des:2 http://deb.debian.org/debian trixie/main amd64 libmagic1t64 amd64 1:5.46-5 [109 kB]
Des:3 http://deb.debian.org/debian trixie/main amd64 file amd64 1:5.46-5 [43,6 kB]
Des:4 http://deb.debian.org/debian trixie/main amd64 libcrack2 amd64 2.9.6-5.2+b1 [44,4 kB]
Des:5 http://deb.debian.org/debian trixie/main amd64 cracklib-runtime amd64 2.9.6-5.2+b1 [143 kB]
Des:6 http://deb.debian.org/debian trixie/main amd64 libpwquality-common all 1:4.5-5 [51,9 kB]
Des:7 http://deb.debian.org/debian trixie/main amd64 libpwquality1 amd64 1:4.5-5 [13,2 kB]
Des:8 http://deb.debian.org/debian trixie/main amd64 libpam-pwquality amd64 1:4.5-5 [13,1 kB]
Descargados 757 kB en 0s (1.988 kB/s)
Seleccionando el paquete libmagic-mgc previamente no seleccionado.
(Leyendo la base de datos ... 30902 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../0-libmagic-mgc_1%3a5.46-5_amd64.deb ...
Desempaquetando libmagic-mgc (1:5.46-5) ...
Seleccionando el paquete libmagic1t64-amd64 previamente no seleccionado.
Preparando para desempaquetar .../1-libmagic1t64_amd64_1%3a5.46-5_amd64.deb ...
Desempaquetando libmagic1t64-amd64 (1:5.46-5) ...
Seleccionando el paquete file previamente no seleccionado.
Preparando para desempaquetar .../2-file_1%3a5.46-5_amd64.deb ...
Desempaquetando file (1:5.46-5) ...

```

Vamos a la ruta /etc/pam.d/common-password *

```

serromer@serromer42:~$ ls -l | grep -i pam/*
-rw-r--r-- 1 root root 552 jun 29 19:40 pam.conf
drwxr-xr-x 2 root root 4096 nov  3 20:25 pam.d
serromer@serromer42:~$ cd pam.d
serromer@serromer42:~/pam.d$ ls
chfn common-account common-session login passwd sshd sudo-i
chpasswd common-auth common-session-noninteractive newusers runuser su su-l
cshh common-password cron other runuser-l sudo
serromer@serromer42:~/pam.d$ sudo nano common-password

```

Ahora pasaremos a instalar la biblioteca de comprobación de la calidad de las contraseñas.

Sudo apt-get install libpam-pwquality-y

password requisite pam_pwquality.so retry=3 minlen=10 ucredit=-1 lcredit=-1 dcredit=-1 maxrepeat=3
reject_username difok=7 enforce_for_root

```

serromer@serromer42:~$ nano /etc/pam.d/common-password
# /etc/pam.d/common-password - password-related modules common to all services
#
# This file is included from other service-specific PAM config files,
# and should contain a list of modules that define the services to be
# used to change user passwords. The default is pam_unix.

# Explanation of pam_unix options:
# The "yescrypt" option enables
# hashed passwords using the yescrypt algorithm, introduced in Debian
# #11. Without this option, the default is Unix crypt. Prior releases
# used the option "sha512"; if a shadow password hash will be shared
# between Debian 11 and older releases replace "yescrypt" with "sha512"
# for compatibility. The "obscure" option replaces the old
# "#OBFUSCATE CHECKS ENAB" option in login.defs. See the pam_unix manpage
# for other options.

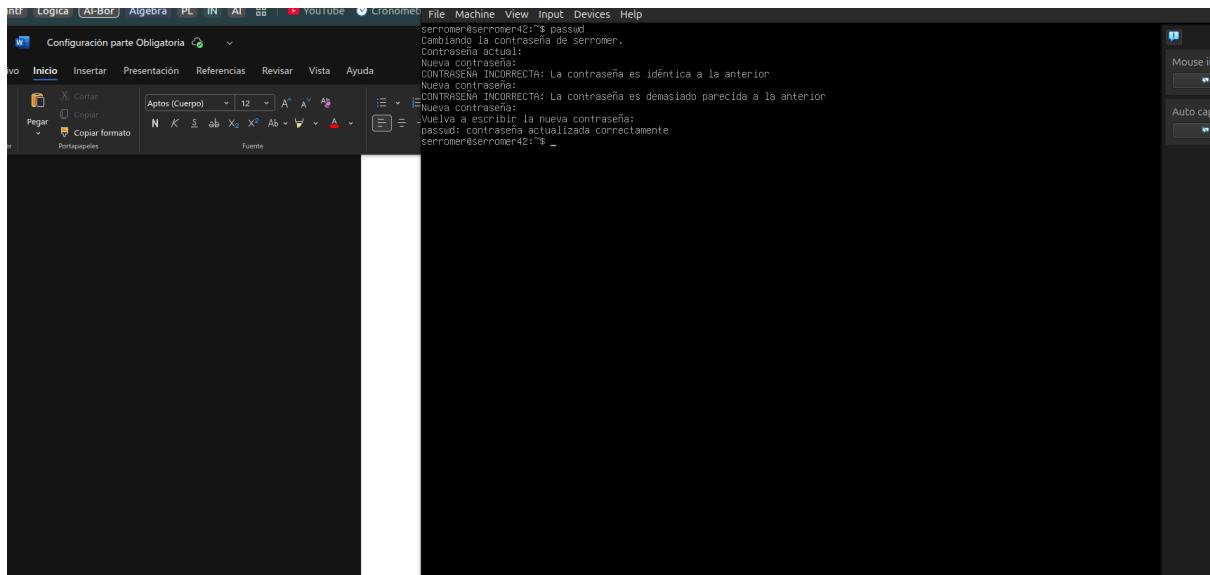
# As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
# To take advantage of this, it is recommended that you configure any
# local modules either before or after the default block, and use
# pam-auth-update to manage selection of other modules. See
# pam-auth-update(8) for details.

# here are the per-package modules (the "Primary" block)
password  requisite          pam_pwquality.so retry=3 minlen=10 ucredit=-1 lcredit=-1 dcredit=-1 maxrepeat=3 reject_username difok=7 enforce_for_root
password  [success=1 default=ignore]  pam_unix.so obscure use_authtok try_first_pass yescrypt
# here's the fallback if no module succeeds
password  requisite          pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password  required           pam_permit.so
# and here are more per-package modules (the "Additional" block)
# end of pam-auth-update config

```

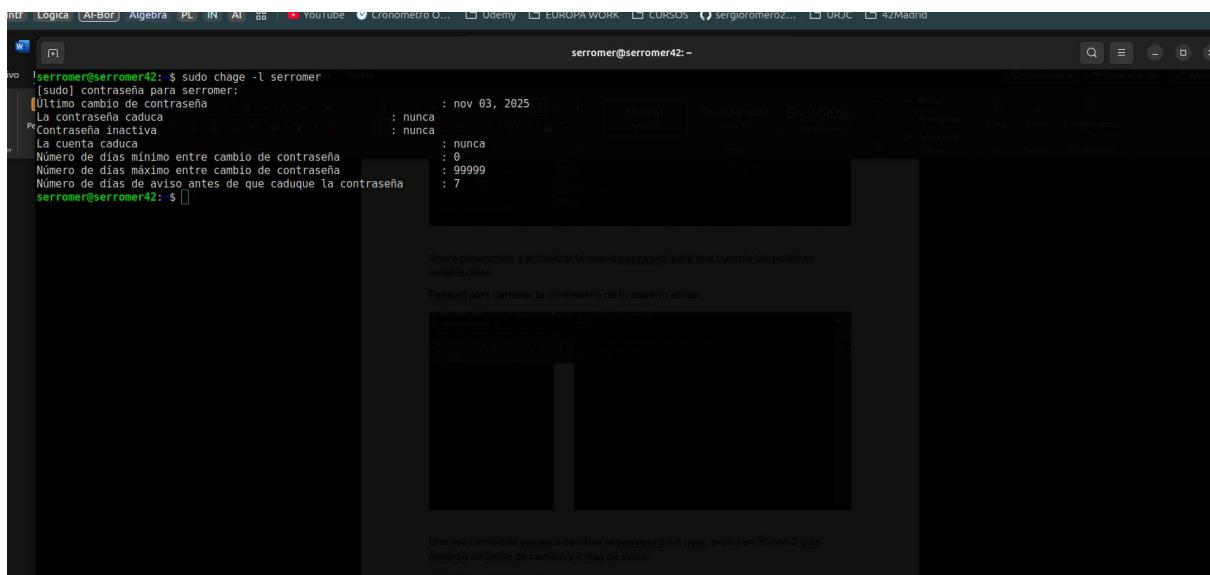
Ahora pasaremos a actualizar la nueva password, para que cumpla las políticas establecidas

Passwd para cambiar la contraseña de tu usuario actual.



Una vez cambiada vamos a cambiar la password del user, expira en 30 con 2 dias minimo de limite de cambio y 7 dias de aviso.

Sudo chage -l serromer



```
serromer@serromer42:~$ sudo chage -M 30 -m 2 -W 7 serromer
serromer@serromer42:~$ sudo chage -l serromer
Último cambio de contraseña : nov 03, 2025
La contraseña caduca : dic 03, 2025
La cuenta inactiva : nunca
Número de días mínimo entre cambio de contraseña : 2
Número de días máximo entre cambio de contraseña : 30
Número de días de aviso antes de que caduque la contraseña : 7
serromer@serromer42:~$
```

Una vez cambiada vamos a cambiar la password del user, expira en 30 con 2 días mínimo de límite de cambio y 7 días de aviso.

Sudo chage -l serromer

Ahora vamos a cambiar la contraseña de root, para que cumpla las políticas establecidas y hacemos los mismo pasos anteriores.

```
serromer@serromer42:~$ sudo chage -M 30 -m 2 -W 7 serromer
serromer@serromer42:~$ sudo chage -l serromer
Último cambio de contraseña : nov 03, 2025
La contraseña caduca : dic 03, 2025
La cuenta inactiva : nunca
Número de días mínimo entre cambio de contraseña : 2
Número de días máximo entre cambio de contraseña : 30
Número de días de aviso antes de que caduque la contraseña : 7
serromer@serromer42:~$ sudo passwd root
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
password: contraseña actualizada correctamente
serromer@serromer42:~$ sudo chage -l root
Último cambio de contraseña : nov 03, 2025
La contraseña caduca : dic 03, 2025
Contraseña inactiva : nunca
La cuenta caduca : nunca
Número de días mínimo entre cambio de contraseña : 0
Número de días máximo entre cambio de contraseña : 99999
Número de días de aviso antes de que caduque la contraseña : 7
serromer@serromer42:~$ sudo chage -M 30 -m 2 -W 7 root
serromer@serromer42:~$ sudo chage -l root
Último cambio de contraseña : nov 03, 2025
La contraseña caduca : dic 03, 2025
Contraseña inactiva : nunca
La cuenta caduca : nunca
Número de días mínimo entre cambio de contraseña : 2
Número de días máximo entre cambio de contraseña : 30
Número de días de aviso antes de que caduque la contraseña : 7
serromer@serromer42:~$
```

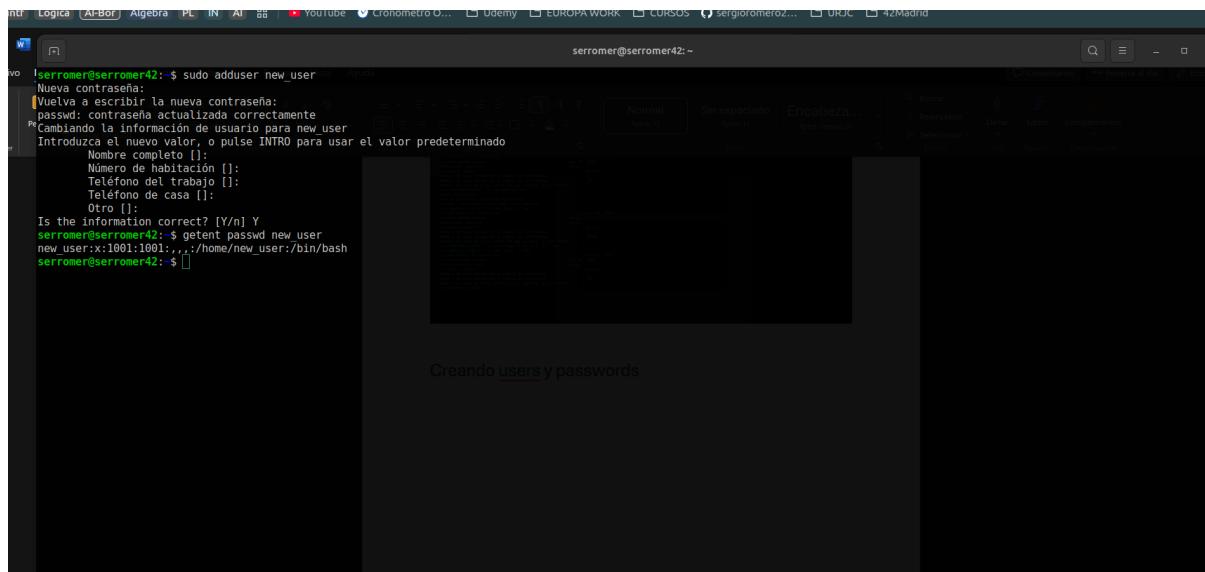
Ahora vamos a cambiar la contraseña de root, para que cumpla las [políticas](#) establecidas y hacemos los [mismo](#) pasos anteriores.

7. Creación de usuarios y contraseñas

Creando usuario new_user

Sudo adduser new_user

Password: Minim0Regla#



```
serromer@serromer42:~$ sudo adduser new_user
[Nueva contraseña]
[Vuelva a escribir la nueva contraseña:
password: contraseña actualizada correctamente
[Cambiando la información de usuario para new_user]
Introduzca el nuevo valor, o pulse INTRO para usar el valor predeterminado
    Nombre completo []:
    Número de habitación []:
    Teléfono del trabajo []:
    Teléfono de casa []:
    Otro []
Is the information correct? [Y/n] Y
serromer@serromer42:~$ getent passwd new_user
new_user:x:1001:1001::/home/new_user:/bin/bash
serromer@serromer42:~$
```

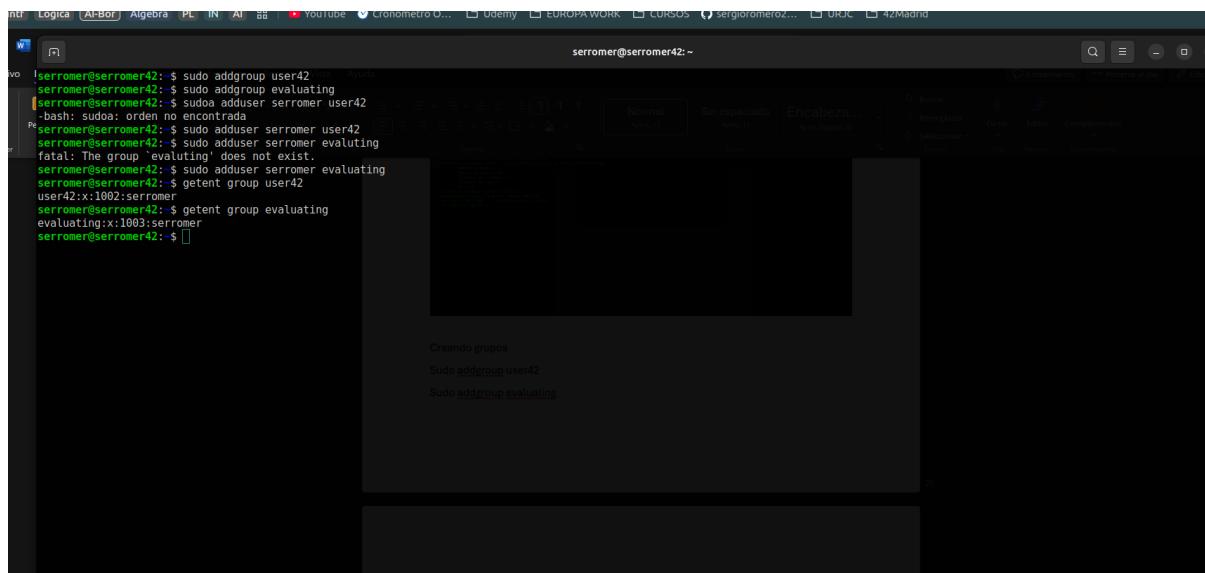
Creando grupos

Sudo addgroup user42

Sudo addgroup evaluating

Sudo adduser serromer user42 -> Agregando al grupo

Sudo adduser serromer evaluating -> Agregando al grupo



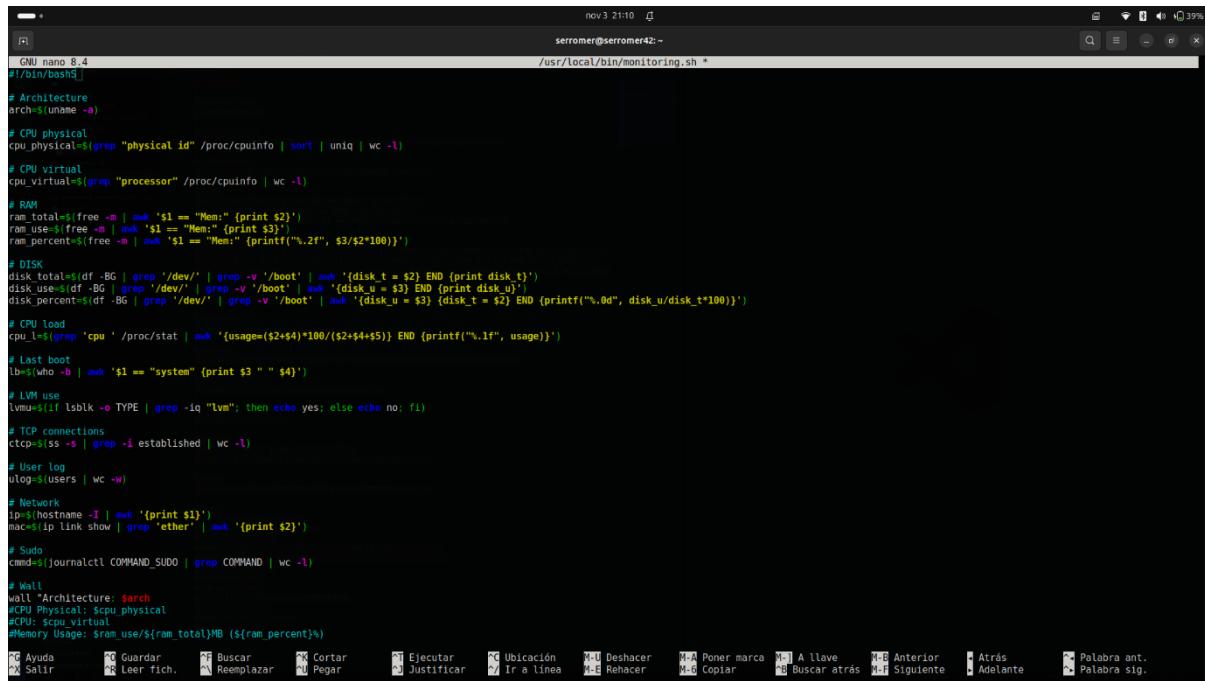
```
serromer@serromer42:~$ sudo addgroup user42
serromer@serromer42:~$ sudo addgroup evaluating
[sudo] password for serromer: orden no encontrada
serromer@serromer42:~$ sudo adduser serromer user42
serromer@serromer42:~$ sudo adduser serromer evaluating
fatal: The group 'evaluating' does not exist.
serromer@serromer42:~$ sudo adduser serromer evaluating
serromer@serromer42:~$ getent group user42
user42:x:1002:serromer
serromer@serromer42:~$ getent group evaluating
evaluating:x:1003:serromer
serromer@serromer42:~$
```

8. Monitorización con CRONTAB CONFIGURATION

Create the script file

Sudo touch /usr/local/bin/monitoring.sh

Sudo chmod 755 /usr/local/bin/monitoring.sh



```
GNU nano 8.4
serromer@serromer42:~ /usr/local/bin/monitoring.sh *
# Architecture
arch=$(uname -a)

# CPU physical
cpu_physical=$(grep "physical id" /proc/cpuinfo | sort | uniq | wc -l)

# CPU virtual
cpu_virtual=$(grep "processor" /proc/cpuinfo | wc -l)

# RAM
ram_total=$(free -m | awk '$1 == "Mem:" {print $2}')
ram_use=$(free -m | awk '$1 == "Mem:" {print $3}')
ram_percent=$(free -m | awk '$1 == "Mem:" {printf("%.2f", $3/$2*100)}')

# DISK
disk_total=$(df -BG | grep '/dev/' | grep -v '/boot' | awk '{disk_t = $2} END {print disk_t}')
disk_use=$(df -BG | grep '/dev/' | grep -v '/boot' | awk '{disk_u = $3} END {print disk_u}')
disk_percent=$(df -BG | grep '/dev/' | grep -v '/boot' | awk '{disk_u = $3} {disk_t = $2} END {printf("%.0d", disk_u/disk_t*100)})')

# CPU load
cpu_load=$(grep 'cpu' /proc/stat | awk '{usage=($2+$4)*100/($2+$4+$5)} END {printf("%.1f", usage)}')

# Last boot
lb=$(who -b | awk '$1 == "system" {print $3 " " $4}')

# LVM use
lvmuse=$(if lsb_release -o | grep -iq "lvm"; then echo yes; else echo no; fi)

# TCP connections
tcp=$(ss -a | grep -i established | wc -l)

# User log
vlog=$(users | wc -w)

# Network
ip=$(hostname -I | awk '{print $1}')
mac=$(ip link show | grep 'ether' | awk '{print $2}')

# Sudo
cmd=$(journalctl COMMAND_SUDO | grep COMMAND | wc -l)

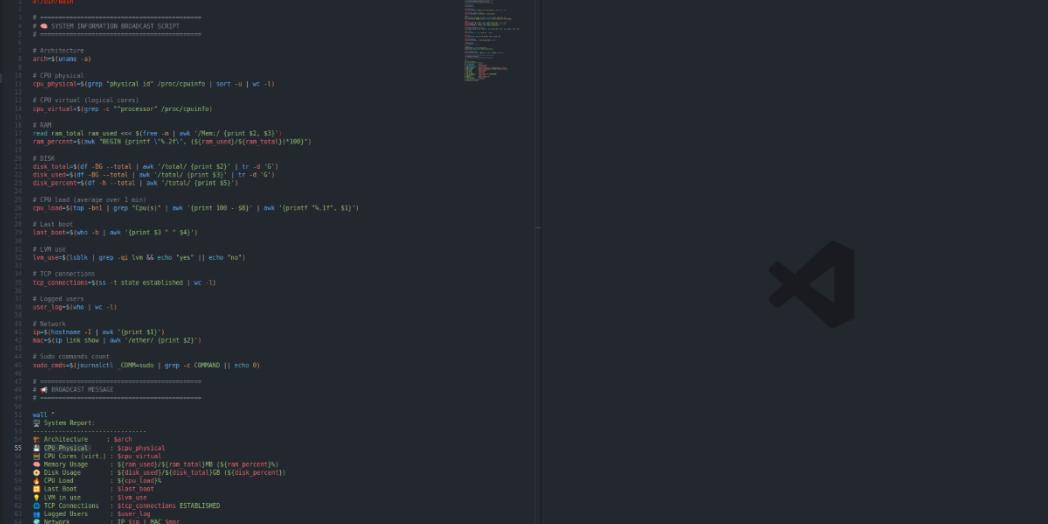
# Wall
wall $(Architecture: search
$CPU_Physical: $cpu_physical
$CPU_Virtual: $cpu_virtual
#Memory Usage: $ram_use/${ram_total}MB. ${ram_percent}%)
```

```

1 monitoring.bash
2 #!/bin/bash
3
4 # Architecture
5 arch=$(uname -a)
6
7 # CPU physical
8 cpu_physical=$(grep "physical id" /proc/cpuinfo | sort | uniq | wc -l)
9
10 # CPU virtual
11 cpu_virtual=$(grep "processor" /proc/cpuinfo | wc -l)
12
13 # RAM
14 ram_total=$(free -m | awk '$1 == "Mem:" {print $2}')
15 ram_use=$(free -m | awk '$1 == "Mem:" {print $3}')
16 ram_percent=$(free -m | awk '$1 == "Mem:" {printf("%.2f", $3/$2*100)}')
17
18 # DISK
19 disk_total=$(df -BG | grep '/dev/' | grep -v '/boot' | awk '{disk_t = $2} END {print disk_t}')
20 disk_use=$(df -BG | grep '/dev/' | grep -v '/boot' | awk '{disk_u = $3} END {print disk_u}')
21 disk_percent=$(df -BG | grep '/dev/' | grep -v '/boot' | awk '{disk_u = $3} {disk_t = $2} END {
22
23 # CPU load
24 cpu_l=$(grep 'cpu ' /proc/stat | awk '{usage=($2+$4)*100/($2+$4+$5)} END {printf("%.1f", usage)}
25
26 # Last boot
27 lb=$(who -b | awk '$1 == "system" {print $3 " " $4}')
28
29 # LVM use
30 lvmu=$(if lsblk -o TYPE | grep -iq "lvm"; then echo yes; else echo no; fi)
31
32 # TCP connections
33 ctcp=$(ss -s | grep -i established | wc -l)
34
35 # User log
36 ulog=$(users | wc -w)
37
38 # Network
39 ip=$(hostname -I | awk '{print $1}')
40 mac=$(ip link show | grep 'ether' | awk '{print $2}')
41
42 # Sudo
43 cmmnd=$(journalctl COMMAND_SUDO | grep COMMAND | wc -l)
44
45 # Wall
46 wall "Architecture: $arch
47 #CPU Physical: $cpu_physical
48 #CPU Virtual: $cpu_virtual
49 #Memory Usage: $ram_use/${ram_total}MB (${ram_percent}%)"
50 #Disk Usage: $disk_use/${disk_total}GB (${disk_percent}%)"
51 #CPU load: $cpu_l%
52 #Last boot: $lb
53 #LVM use: $lvmu
54 #Connections TCP: $ctcp ESTABLISHED
55 #User log: $ulog
56 #Network: IP $ip ($mac)
57 #Sudo: $cmmnd cmd"

```

Version mejorada:



```
nov 3 22:48
```

```
Archivo Editor Selección Ver Ir Ejecutar Terminal Ayuda
```

```
monitoring_bash
```

```
#!/bin/bash
```

```
## SYSTEM INFORMATION BROADCAST SCRIPT
```

```
##
```

```
## Architecture
```

```
arch=$(uname -a)
```

```
## CPU physical
```

```
cpu_physical=$(grep "physical id" /proc/cpuinfo | sort -u | wc -l)
```

```
## CPU virtual, (logical cores)
```

```
cpu_virtual=$(grep -c "processor" /proc/cpuinfo)
```

```
##
```

```
## RAM
```

```
ram_free=$(awk '{print $1}' /proc/meminfo | awk '{print $2*$3}')
```

```
ram_percent=$(awk '{BEGIN{printf "%0.2f", ($1/$2)*100}')
```

```
##
```

```
## DISK
```

```
disk_free=$(df -h | awk '{print $2}' | awk '{print $1-$2-$3}')
```

```
disk_percent=$(df -h | awk '{print $2}' | awk '{print $3-$2}' | awk '{print $1-$2-$3}')
```

```
##
```

```
## CPU load (average over 1 min)
```

```
cpu_load=$(top -b -n 1 | grep "loadavg" | awk '{print 100-$1}' | awk '{print "%-1f", $1}')
```

```
##
```

```
## Network
```

```
last_boot=$(who -b | awk '{print $3 " " $4}')
```

```
## LVM
```

```
lvm_use=$(lsblk | grep -o lvm 64 echo "yes" || echo "no")
```

```
## TCP connections
```

```
tcp_connections=$(ss -t state established | wc -l)
```

```
##
```

```
## Logged users
```

```
user_log=$(w | wc -l)
```

```
##
```

```
## IP
```

```
ip4=$(hostname -I | awk '{print $3}')
```

```
mac=$(ifconfig | awk '/ether/ {print $2}')
```

```
## Sudo commands count
```

```
sudo_cmds=$(netstat -C | grep -e COMMAND || echo 0)
```

```
##
```

```
##
```

```
## BROADCAST MESSAGE
```

```
##
```

```
wall >
```

```
## System Report:
```

```
##
```

```
## 55 Arch:Architecture : Sarsh
```

```
## 56 CPU:CPU physical
```

```
## 57 CPU Cores (virt.):cpu.virtual
```

```
## 58 Disk Usage:df -h | awk '{print $1" "$2" "$3" "$4" "$5" }'
```

```
## 59 Disk I/O:df -B -k | awk '{print $1" "$2" "$3" "$4" }'
```

```
## 60 CPU Load:top -b -n 1 | grep "loadavg" | awk '{print 100-$1}' | awk '{print "%-1f", $1}'
```

```
## 61 LVM:LVM in use
```

```
## 62 TCP Connections:ss -t state established
```

```
## 63 Logged Users:who -l
```

```
## 64 Networks:IP:IP | MAC:MAC
```

```
## 65 Sudo Commands:netstat -C | grep -e COMMAND || echo 0
```

```
##
```

```
## Generated on $(date)
```

Ahora debemos hacer que el script se ejecute sin password

Your_username ALL=(ALL) NOPASSWD:/usr/local/bin/monitoring.sh

```
GNU nano 8.4
Defaults        log input,log output
Defaults        ioilog dir="/var/log/sudo"
Defaults        requiretty
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"

# This fixes CVE-2005-4099 and possibly breaks some versions of kdesu
# (#1011624, https://bugs.kde.org/show_bug.cgi?id=452532)
Defaults        use_py

# This preserves proxy settings from user environments of root
# equivalent users (group sudo)
#Defaults:%sudo env_keep += "http_proxy https_proxy ftp_proxy all_proxy no_proxy"

# This allows running arbitrary commands, but so does ALL, and it means
# different sudoers have their choice of editor respected.
#Defaults:%sudo env_keep += "EDITOR"

# Completely harmless preservation of a user preference.
#Defaults:%sudo env_keep += "GREP_COLORbin/"

# While you shouldn't normally run git as root, you need to with etkeeper
#Defaults:%sudo env_keep += "GIT_AUTHOR_ GIT_COMMITTER_"

# Per-user preferences: root won't have sensible values for them.
#Defaults:%sudo env_keep += "EMAIL DEBEMAIL DEBFULLNAME"

# "sudo scp" or "sudo rsync" should be able to use your SSH agent.
#Defaults:%sudo env_keep += "SSH_AGENT_PID SSH_AUTH_SOCK"

# Ditto for GPG agent
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"

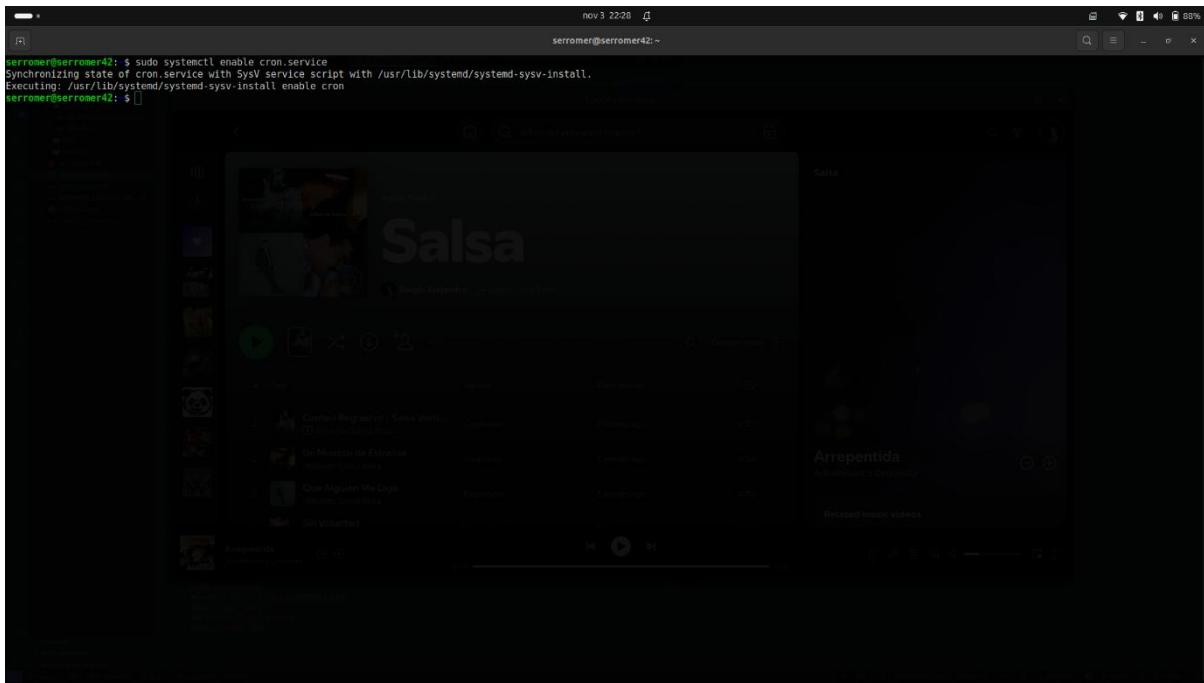
# Host alias specification
# Host alias specification
# User alias specification
# Cmd alias specification

# User privilege specification
#sudo  ALL=(ALL:ALL)  ALL
serromer ALL=(ALL) NOPASSWD: /usr/local/bin/monitoring.sh
# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL)  ALL

# See sudoers(5) for more information on "@include" directives:

@includeincluded /etc/sudoers.d
```

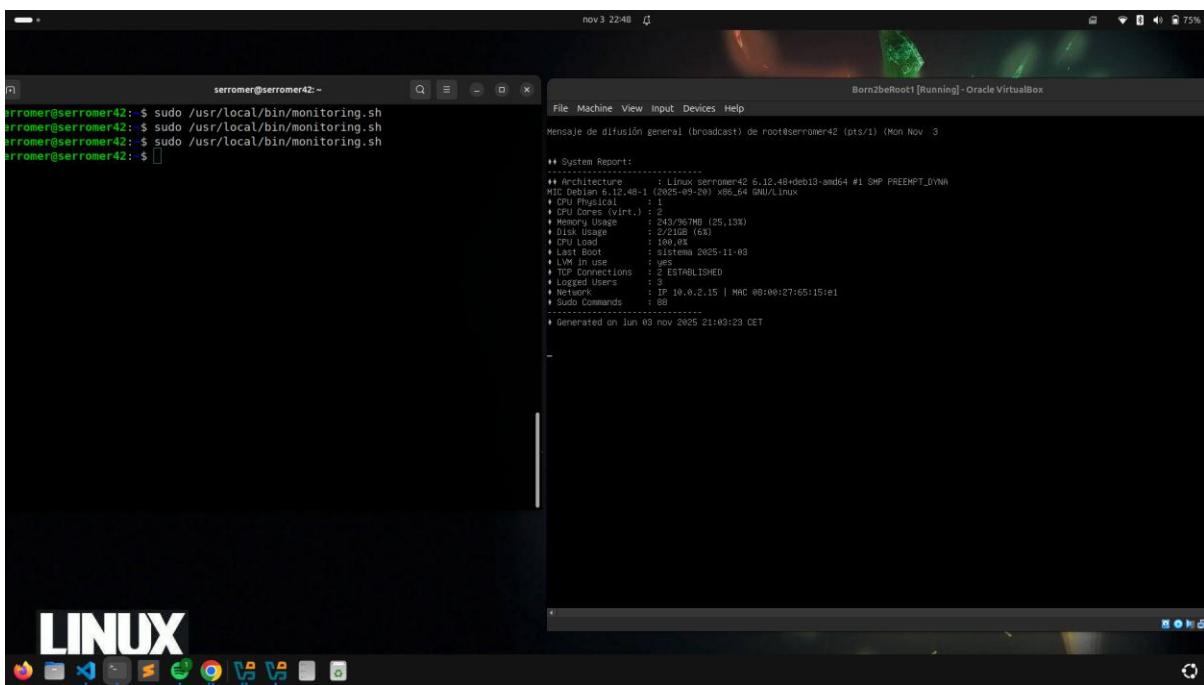
```
sudo systemctl enable cron.service
```



Ahora reiniciamos para que los cambios se hagan válidos.

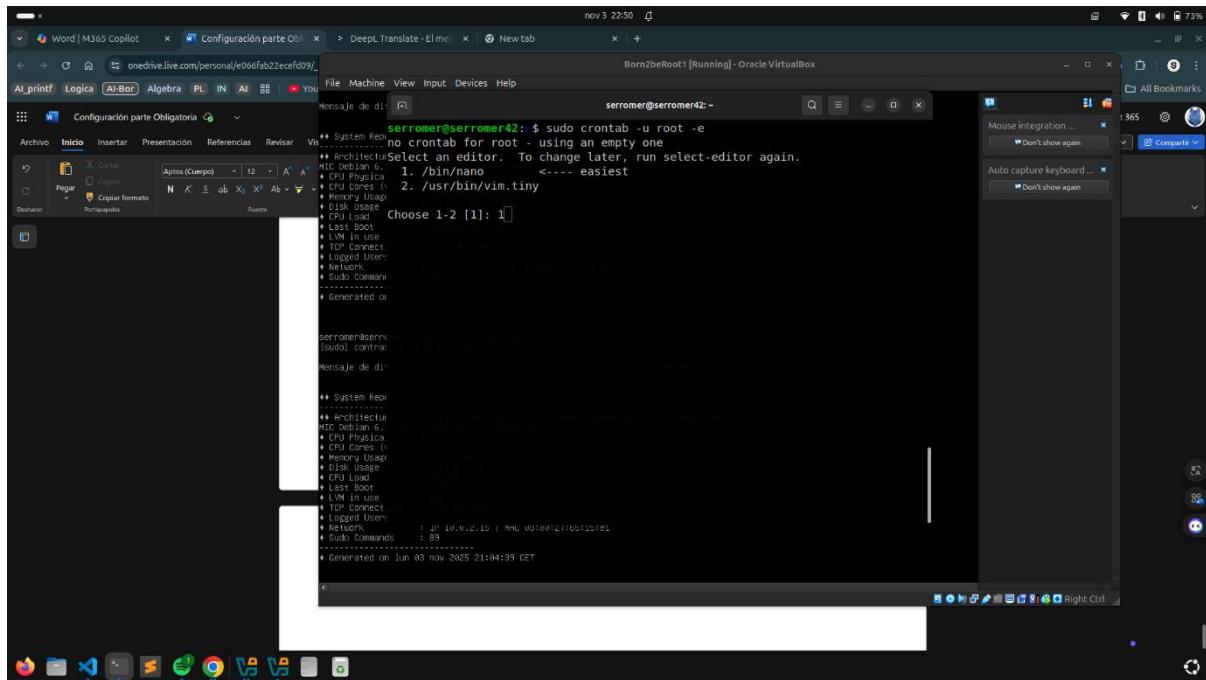
Ahora una vez dentro de nuevo y reiniciado la maquina, ejecutamos nuestro script:

Sudo /usr/local/bin/monitoring.sh



Programa esto en el crontab de root.

Sudo crontab -u root -e



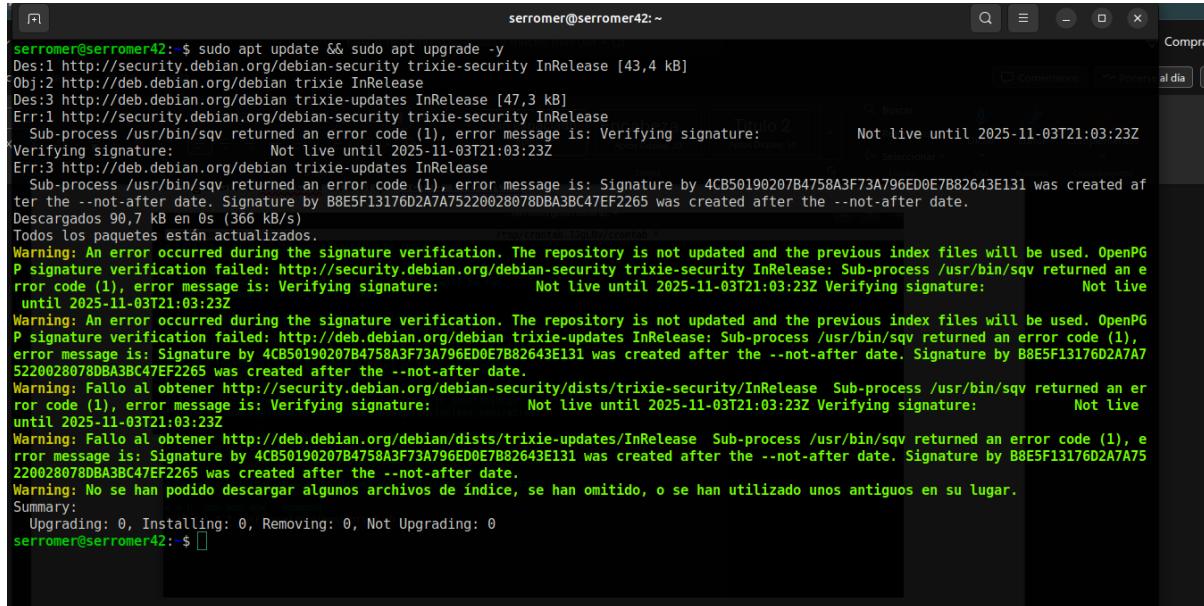
Lo siguiente correra el script cada minutos.

A screenshot of a terminal window titled 'serromer@serromer42: ~'. The user is viewing a crontab file. The terminal shows the following text:

```
GNU nano 8.4
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#ante correra el script cada minutos.
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
*/10 * * * * /usr/local/bin/monitoring.sh
```

9. Instalación y configuración de LLMP STACK

Sudo apt update && sudo apt upgrade -y



```
serromer@serromer42: ~
serromer@serromer42: $ sudo apt update && sudo apt upgrade -y
Des:1 http://security.debian.org/debian-security trixie-security InRelease [43,4 kB]
Obj:2 http://deb.debian.org/debian trixie InRelease
Des:3 http://deb.debian.org/debian trixie-updates InRelease [47,3 kB]
Err:1 http://security.debian.org/debian-security trixie-security InRelease
      Sub-process /usr/bin/sqv returned an error code (1), error message is: Verifying signature:
      Verifying signature:          Not live until 2025-11-03T21:03:23Z
      Err:3 http://deb.debian.org/debian trixie-updates InRelease
      Sub-process /usr/bin/sqv returned an error code (1), error message is: Signature by 4CB50190207B4758A3F73A796ED0E7B82643E131 was created after the --not-after date. Signature by B8E5F13176D2A7A75220028078DBA3BC47EF2265 was created after the --not-after date.
      Descargados 90,7 kB en 0s (366 kB/s)
      Todos los paquetes están actualizados.
      Warning: An error occurred during the signature verification. The repository is not updated and the previous index files will be used. OpenPGP signature verification failed: http://security.debian.org/debian-security trixie-security InRelease: Sub-process /usr/bin/sqv returned an error code (1), error message is: Verifying signature:          Not live until 2025-11-03T21:03:23Z Verifying signature:          Not live until 2025-11-03T21:03:23Z
      Warning: An error occurred during the signature verification. The repository is not updated and the previous index files will be used. OpenPGP signature verification failed: http://deb.debian.org/debian trixie-updates InRelease: Sub-process /usr/bin/sqv returned an error code (1), error message is: Signature by 4CB50190207B4758A3F73A796ED0E7B82643E131 was created after the --not-after date. Signature by B8E5F13176D2A7A75220028078DBA3BC47EF2265 was created after the --not-after date.
      Warning: Fallo al obtener http://security.debian.org/debian-security/dists/trixie-security/InRelease Sub-process /usr/bin/sqv returned an error code (1), error message is: Verifying signature:          Not live until 2025-11-03T21:03:23Z Verifying signature:          Not live until 2025-11-03T21:03:23Z
      Warning: Fallo al obtener http://deb.debian.org/debian/dists/trixie-updates/InRelease Sub-process /usr/bin/sqv returned an error code (1), error message is: Signature by 4CB50190207B4758A3F73A796ED0E7B82643E131 was created after the --not-after date. Signature by B8E5F13176D2A7A75220028078DBA3BC47EF2265 was created after the --not-after date.
      Warning: No se han podido descargar algunos archivos de índice, se han omitido, o se han utilizado unos antiguos en su lugar.
      Summary:
      Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 0
serromer@serromer42: $
```

Me sale ese aviso, pero lo ignorare.

10. Instalación y configuración de *LIGHTTPD*

Sudo apt install lighttpd -y

Sudo ufw allow 80

Sudo ufw status

Reiniciar el lighttpd

Systemctl reload lighttpd

Systemctl enable lighttpd

Systemctl start lighttpd

11. Instalación y configuración de MariaDB

Sudo apt install mariadb_server -y

Sudo mysql_secure_instalattion

Te pedira password de root.

DAle n a 2 y luego yes a todo.

Entrar a mariaDB

Sudo mariadb

Crear tu database.

CREATE DATABASE database_name;

Example: CREATE DATABASE serromer;

GRANT ALL ON database_name.* TO 'user_name'@'localhost' IDENTIFIED BY 'db_password' WITH
GRANT OPTION;

EXAMPLE: GRANT ALL ON serromer.* TO '[serromer](#)'@'localhost' IDENTIFIED BY 'Fuerza#123A' WITH
GRANT OPTION;

FLUSH PRIVILEGES;

Ahora ejecutamos mariadb -u serromer -p

SHOW DATABASES una vez dentro.

12. Instalación de *PHP & EXTENSIONS*

Sudo apt install php-fpm php-mysql php-curl php-gd php-zip -y

VERIFICAR PHP INSTALLATION

Dpkg -l | grep php

Php -v

Edit FASTCGI Configuration (OPen the configuration file):

Con la version de arriba debes editar este archivo a su versión correspondiente.

Sudo nano /etc/lighttpd/conf-available/15-fastcgi-php.conf

Sudo lighty-enable-mod fastcgi

Sudo systemctl reload lighttpd

Sudo systemctl status lighttpd

13. WORDPRESS SETUP

Sudo apt install wget -y

Sudo wget https://wordpress.org/latest.tar.gz -P /var/www/html

Descromprimimos:

Sudo tar -xvzf /var/www/html/ latest.tar.gz -C /var/www/html

Eliminamos:

Sudo rm -rf /var/www/html/ latest.tar.gz

sudo cp -r /var/www/html/wordpress/* /var/www/html

sudo rm -rf /var/www/html/wordpress

sudo cp /var/www/html/wp-config-sample.php /var/www/html/wp-config.php

sudo chown -R www-data:www-data /var/www/html

sudo chmod -R 755 /var/www/html

Ahora editamos el archivo

Nano /var/www/html/wp-config.php

Este archivo lo editamos para configurar la base de datos

Sudo systemctl reload lighttpd

Ahora entramos al navegador y verificamos que todo funciona

Localhost:8080

14. CONFIGURAR UN SERVICIO DE WORDPRESS A TU ELECCION

Yo escogi REDIS

REDIS SETUP

Sudo apt install redis-server -y

Sudo apt install php-redis -y

Sudo systemctl restart lighttpd

Sudo nano /etc/redis/redis.conf

Para el apartado anterior una vez dentro establecemos una contraseña,

requirepass Fuerza#123a

Enable snapshots:

Save 900 1

Save 300 10

Save 60 10000

Nada mas con el archivo anterior,

Ahora aseguramos el snapshot directorio

Sudo mkdir -p /var/lib/redis

Sudo chown redis:redis /var/lib/redis

Sudo systemctl restart redis-server

CONFIGURAR WORDPRESS PARA USAR REDIS

Sudo nano /var/www/html/wp-config.php

Define('WP_REDIS_HOST','127.0.0.1');

Define ('WP_REDIS_PASSWORD','Fuerza#123A');

Test:

Redis-pi

Auth redis_password

Ping

Expected output:

PONG

Redis-cli

Aut@password

Ping

Exitgfdgd

Solo falta la parte ultima

15. Pasos prohibidos y verificaciones

15.1. Instalar interfaz gráfica



Como consiste en configurar un servidor, deberás instalar el número mínimo de servicios. Por este motivo, una interfaz gráfica no tiene sentido. Está prohibido por tanto instalar X.org o cualquier servidor gráfico equivalente. En caso de hacerlo, tu nota será 0.

15.2. Verificar última versión

Deberás elegir como sistema operativo la última versión estable de **Debian** (no testing/unstable), o la última versión estable de **Rocky**. Se recomienda encarecidamente **Debian** si no tienes experiencia en administración de sistemas.

15.3. Verificar particiones (Obligatoria)

```
wil@wil:~$ lsblk
NAME           MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda            8:0    0   8G  0 disk
└─sda1          8:1    0 487M  0 part  /boot
  └─sda2          8:2    0   1K  0 part
    └─sda5          8:5    0 7.5G  0 part
      └─sda5_crypt 254:0    0 7.5G  0 crypt
        ├─wil--vg-root 254:1    0 2.8G  0 lvm   /
        ├─wil--vg-swap_1 254:2    0 976M  0 lvm   [SWAP]
        └─wil--vg-home 254:3    0 3.8G  0 lvm   /home
sr0           11:0    1 1024M 0 rom
wil@wil:~$ _
```

15.4. Preguntas



Durante la defensa, se te harán unas preguntas sobre el sistema operativo que has elegido. Debes saber, por lo tanto, las diferencias entre aptitude y apt, o qué son SELinux y AppArmor. En definitiva, ¡entiende lo que estás utilizando!

15.5. Revisar SSH

El servicio SSH se ejecutará obligatoriamente en el puerto 4242 de tu máquina virtual. Por seguridad, no debe ser posible conectarte a través de SSH como root.



El uso de SSH será comprobado durante la defensa creando un nuevo usuario. Por lo tanto, debes entender cómo funciona.

15.6. Revisar UFW o FIREWALLD

Debes configurar tu sistema operativo con el firewall UFW, (o firewalld en Rocky) dejando solamente el puerto 4242 abierto en tu máquina virtual.



Tu firewall debe estar activo cuando ejecutes la máquina virtual.
Para Para Rocky, debes usar firewalld en lugar de UFW

15.7. Revisión 1

- El **hostname** de tu máquina virtual debe ser tu login terminado en 42 (por ejemplo, wil42). Deberás modificar este **hostname** durante tu evaluación.
- Debes implementar una política de contraseñas fuerte.
- Debes instalar y configurar **sudo** siguiendo reglas estrictas.
- Además del usuario root, un usuario con tu login como nombre debe existir.
- Este usuario debe pertenecer a los grupos **user42** y **sudo**.

Para configurar una política de contraseñas fuerte, deberás cumplir los siguientes requisitos:

- Tu contraseña debe expirar cada 30 días.
- El número mínimo de días permitido antes de modificar una contraseña deberá ser 2.
- El usuario debe recibir un mensaje de aviso 7 días antes de que su contraseña expire.
- Tu contraseña debe tener como mínimo 10 caracteres de longitud. Debe contener una mayúscula, una minúscula y un número. Por cierto, no puede tener más de 3 veces consecutivas el mismo carácter.



Después de preparar tus archivos de configuración, deberás cambiar la contraseña de todas las cuentas presentes en la máquina virtual, root incluida.

Para configurar una contraseña fuerte para tu grupo sudo, debes cumplir con los siguientes requisitos:

- Autenticarte con sudo debe estar limitado a tres intentos en el caso de introducir una contraseña incorrecta.
- Un mensaje personalizado de tu elección debe mostrarse en caso de que la contraseña introducida sea incorrecta cuando se utilice sudo.
- Para cada comando ejecutado con sudo, tanto el input como el output deben quedar archivados en el directorio /var/log/sudo/.
- El modo TTY debe estar activado por razones de seguridad.
- Por seguridad, los directorios utilizables por sudo deben estar restringidos. Por ejemplo:
`/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin`

15.8. Creación usuario y grupo en tu delante



Durante la defensa, deberás crear un usuario y asignárselo a un grupo.

15.9. Script Monitoring.sh