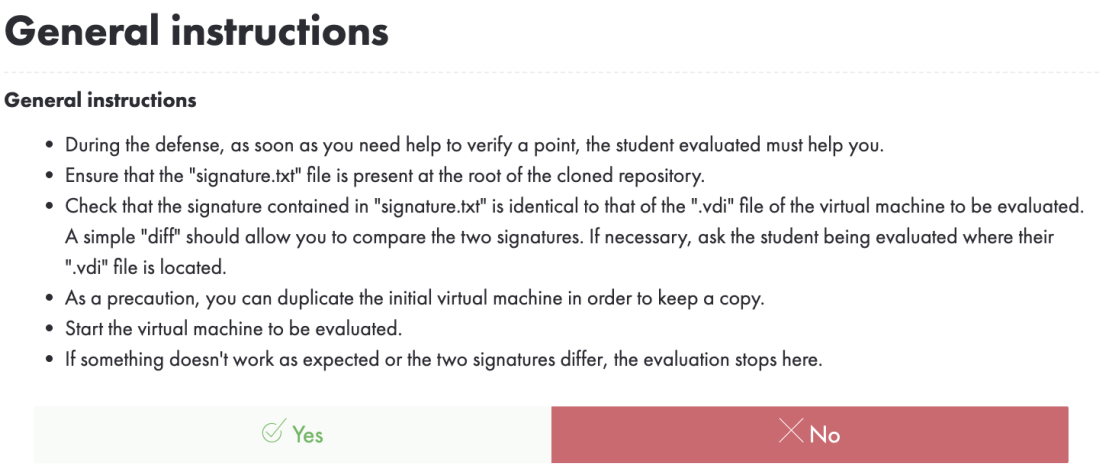
1. Firma:



General instructions

General instructions

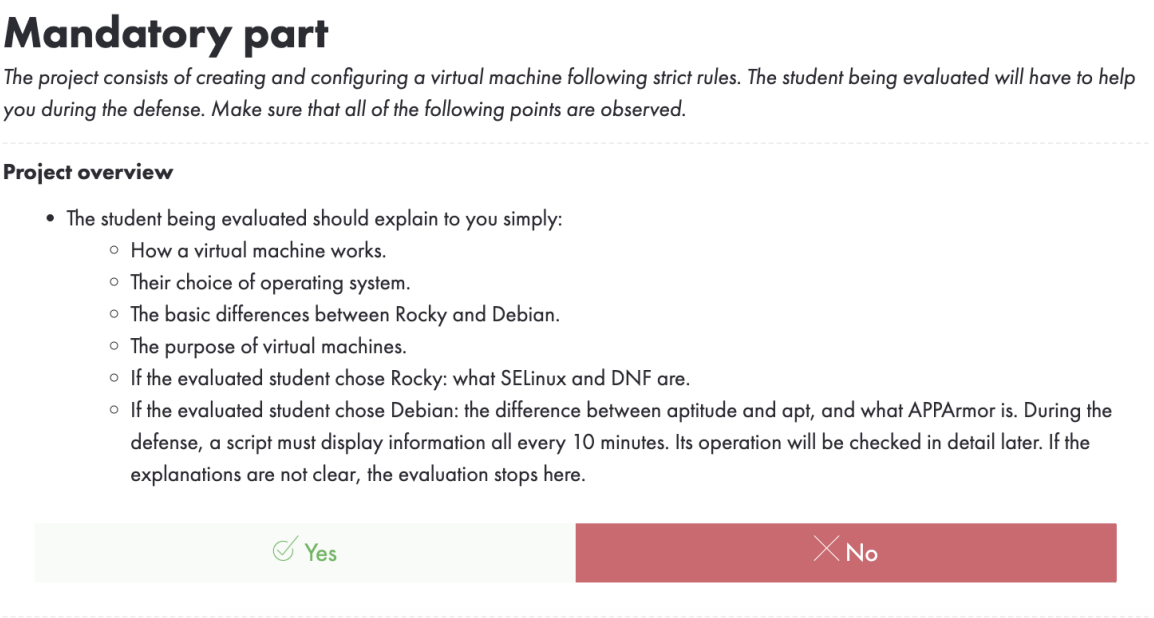
- During the defense, as soon as you need help to verify a point, the student evaluated must help you.
- Ensure that the "signature.txt" file is present at the root of the cloned repository.
- Check that the signature contained in "signature.txt" is identical to that of the ".vdi" file of the virtual machine to be evaluated. A simple "diff" should allow you to compare the two signatures. If necessary, ask the student being evaluated where their ".vdi" file is located.
- As a precaution, you can duplicate the initial virtual machine in order to keep a copy.
- Start the virtual machine to be evaluated.
- If something doesn't work as expected or the two signatures differ, the evaluation stops here.

Yes No

Comprobar la firma en el directorio donde está signature.txt

shasum Born2beeroot.vdi

2. Preguntas



Mandatory part

The project consists of creating and configuring a virtual machine following strict rules. The student being evaluated will have to help you during the defense. Make sure that all of the following points are observed.

Project overview

- The student being evaluated should explain to you simply:
  - How a virtual machine works.
  - Their choice of operating system.
  - The basic differences between Rocky and Debian.
  - The purpose of virtual machines.
  - If the evaluated student chose Rocky: what SELinux and DNF are.
  - If the evaluated student chose Debian: the difference between aptitude and apt, and what APPArmor is. During the defense, a script must display information all every 10 minutes. Its operation will be checked in detail later. If the explanations are not clear, the evaluation stops here.

Yes | No

1. Parte oral (Preguntas por parte del corrector/a):

- ¿Qué es una máquina virtual?

Es un software que simula un sistema de computación (una máquina) y por lo tanto puede ejecutar programas como si fuese una computadora real.

Permite crear múltiples entornos simulados con recursos concretos (Ram, disco duro, etc) desde un solo sistema de hardware físico.
Es muy útil para probar programas en diferentes sistemas operativos o entornos sin tener que cambiar de máquina física.

- ¿Por qué has escogido Debian?

Principalmente porque es una sugerencia del propio subject, además estoy más familiarizado (aunque sea vagamente) con Debian ya que he usado Raspberry Pi (Utiliza Raspbian que está basado en Debian).

- Diferencias básicas entre Rocky y Debian

Rocky es un proyecto de la comunidad Linux, impulsado por la empresa CloudLinux como alternativa a CentOS. Ya que este cambió su modelo de lanzamiento estable a uno de lanzamiento continuo.
Por otra parte **Debian** es uno de los sistemas operativos basados en Linux, más antiguos y se basa en un modelo de desarrollo comunitario (No está vinculado a ninguna empresa en particular).

Principalmente, **Rocky** está **más enfocado al tema de servidores y entornos empresariales**, por lo que **no es tan buena opción para estaciones de trabajo de escritorio como lo es Debian**.
Esto es debido a su gran estabilidad durante períodos más largos de tiempo, en cuanto al soporte de sus versiones, con lo cual suele ser más atractivo para organizaciones que necesitan mantener sus sistemas estables durante más tiempo.

Debian es más cambiante ya que **se adhiere a lo que se conoce como "el contrato social de Debian**" que enfatiza **la libertad del software, la calidad y la cooperación de la comunidad de software libre**.

Además esto **permite que existan una amplia gama de arquitecturas basadas en Debian** (por ejemplo Raspbian), **lo que lo hace más versátil y adecuado para una gran variedad de entornos y dispositivos de hardware diferentes.**
También **cuenta con un gran repositorio de paquetes, facilitando la instalación de una amplia variedad de software y aplicaciones de forma sencilla.**
Sin embargo también cabe enfatizar que Debian ofrece diferentes ramas (Stable, Testing y Unstable) Lo que realmente nos da a elegir entre estabilidad a largo plazo o las últimas actualizaciones. Lo cual también lo hace adecuado para servidores y estaciones de trabajo.

La intención de Rocky es ser compatible con las aplicaciones y cargas de trabajo diseñadas para CentOS, lo que facilita la migración de sistemas existentes.

Resumen:

Ventajas de Debian:

- Filosofía basada en la libertad del software.
- Amplia variedad de arquitecturas soportadas.
- Gran cantidad de paquetes disponibles.
- Flexibilidad en la elección de versiones y ramas.

Desventajas de Debian:

- Puede no estar tan centrado en la compatibilidad binaria con otros sistemas como Rocky Linux.
- La estabilidad de las versiones de Debian puede variar según la rama elegida.

En última instancia, la elección entre Rocky Linux y Debian dependerá de sus necesidades y preferencias específicas. Si buscas una alternativa a CentOS para entornos empresariales, Rocky Linux puede ser una elección sólida. Si valoras la filosofía de software libre y la versatilidad, Debian es una opción a considerar.

- ¿Cuál es el propósito de las máquinas virtuales?

Sobre todo se trata de tener un entorno de trabajo controlado e independiente de la plataforma de hardware (o máquina física) y sistema operativo “real” donde estamos trabajando.

-Esto puede ser por motivos de seguridad (ya que no ponemos en compromiso la máquina real sobre la que estamos trabajando).

-Puede ser para probar y desarrollar software para una máquina con unas características concretas (a nivel de SO o de hardware).

-Etc.

- Diferencias entre apt y aptitude

Básicamente Aptitude es una versión mejorada de apt.

APT es un administrador de paquetes de más **bajo nivel y aptitude e**s un administrador de paquetes de **alto nivel.**

Apt es una herramienta de línea de comandos con interfaz sencilla para gestionar paquetes, es **más fácil de usar para tareas básicas.** Se centra en tareas esenciales como i**nstalar, actualizar, eliminar y buscar paquetes**

Aptitude es más bien un administrador de paquetes que ofrece más cantidad de características y comandos. Es más poderoso y versátil en comparación con apt, pero también es más complicado en cuanto a su utilización. Ofrece **características más avanzadas como resolución de conflictos con dependencias, seguimiento y**

filtrado de paquetes de forma más detallada y compleja. Esto lo puede hacer apt pero tirando más de manual y comandos, lo que lo hace más complicado ya que no está diseñada para tal fin, tanto como aptitude. Aptitude, a diferencia de apt, mantiene un historial o registro de las operaciones realizadas en los paquetes lo cual puede ser útil para rastrear operaciones y deshacer cambios.

- ¿Qué es APPArmor?

Es un sistema o módulo de seguridad de Linux, para controlar restringir el acceso de los programas a diferentes partes del sistema u otros programas, por motivos de seguridad.
Añade capas de seguridad para sistemas como puede ser un servidor o sistemas de alta seguridad. Elimina riesgos en cuanto a posibles vulnerabilidades de sistema. Por si no lo había dicho antes: **seguridad.**

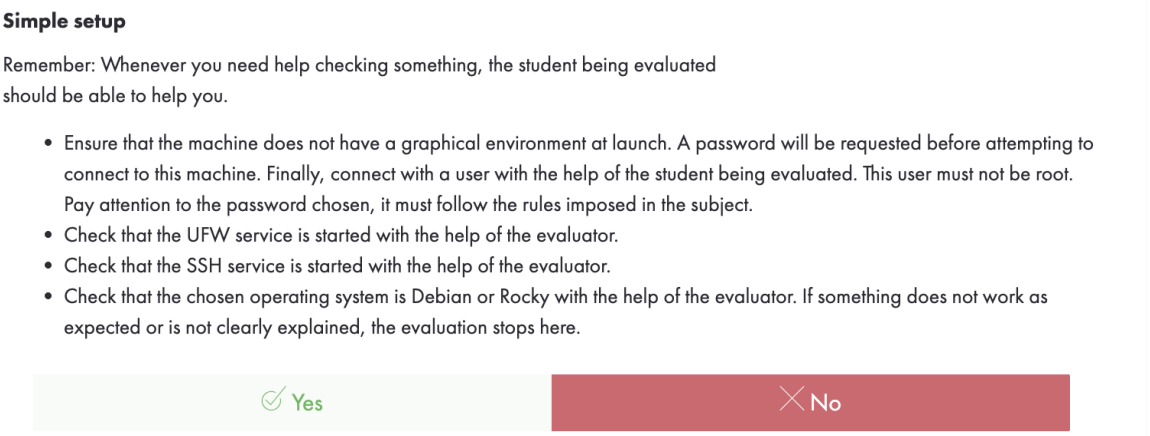
- ¿Qué es GRUB?

GNU GRUB (*GNU GRand Unified Bootloader*) es un **cargador de arranque** múltiple, desarrollado por el proyecto GNU que nos permite elegir qué Sistema Operativo arrancar de los instalados.

- ¿Qué es GID?

Es el identificador de grupo, es una abreviatura de Group.

3. Comprobaciones



SSH: (por si quiero hacerlo a través de iTerm).

Acceso	**ssh user@localhost -p 4242**
Acceder al usuario Root	**sudo su**
Salir de ssh…	**exit**
Comprobar el servicio	**sudo service ssh status**
Mostrar versión de ssh	**ssh -V**

Que no haya ninguna **interfaz gráfica** en uso **ls /usr/bin/*session**

Servicio **UFW** **sudo service ufw status** `sudo ufw status`

Servicio **SSH** **sudo service ssh status** `/etc/ssh/sshd_config /etc/ssh/ssh_config`

Sistema operativo **uname -v** (aquí “v” es en minúscula)

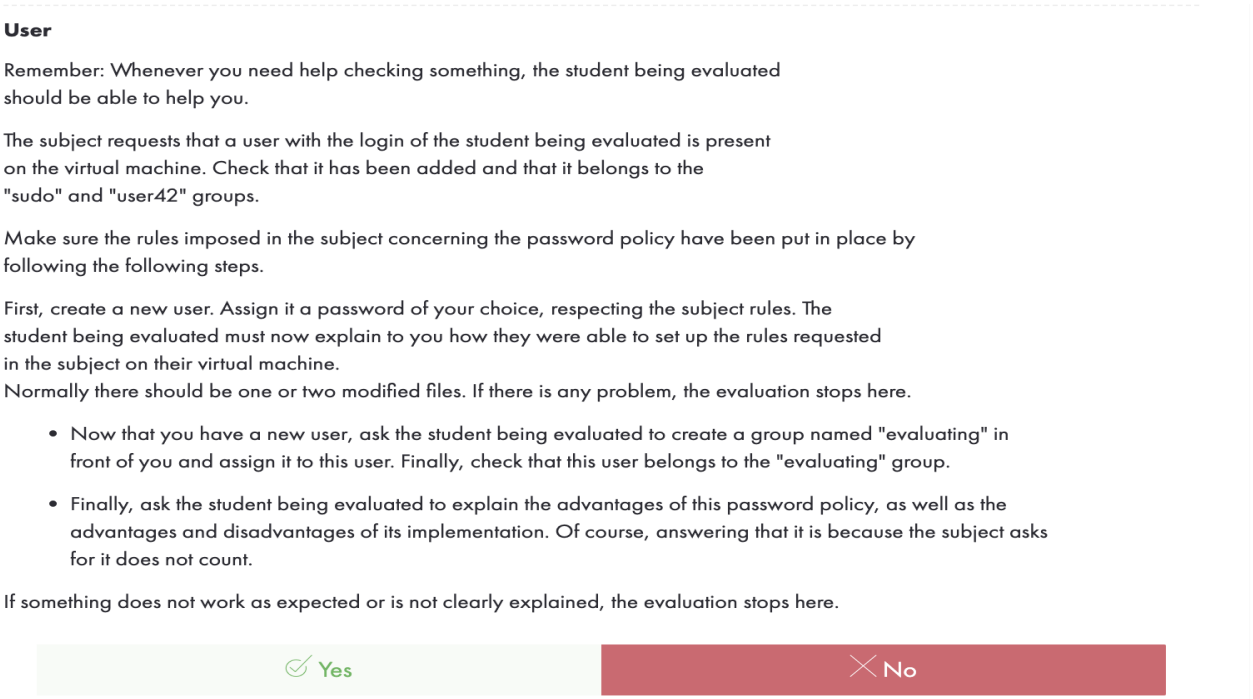
uname -a (versión más completa)

***nota-** si en una comprobación el texto es muy largo, puedo añadir **| more** detrás.

Por ejemplo si hago **sudo -V** y quiero avanzar poco a poco, pondría:

sudo -V | more

4. Usuario



User

Remember: Whenever you need help checking something, the student being evaluated should be able to help you.

The subject requests that a user with the login of the student being evaluated is present on the virtual machine. Check that it has been added and that it belongs to the "sudo" and "user42" groups.

Make sure the rules imposed in the subject concerning the password policy have been put in place by following the following steps.

First, create a new user. Assign it a password of your choice, respecting the subject rules. The student being evaluated must now explain to you how they were able to set up the rules requested in the subject on their virtual machine.
Normally there should be one or two modified files. If there is any problem, the evaluation stops here.

- Now that you have a new user, ask the student being evaluated to create a group named "evaluating" in front of you and assign it to this user. Finally, check that this user belongs to the "evaluating" group.
- Finally, ask the student being evaluated to explain the advantages of this password policy, as well as the advantages and disadvantages of its implementation. Of course, answering that it is because the subject asks for it does not count.

If something does not work as expected or is not clearly explained, the evaluation stops here.

Yes | No

***Nota: Recuerda estar en usuario root.** **sudo su**

Comprobar que tu usuario esté en el grupo "sudo" **getent group sudo**

Comprobar que tu usuario esté en el grupo "User 42" **getent group user42**

Crear un **nuevo usuario** y mostrar que sigue la política de contraseñas que hemos creado. **sudo adduser "name_user"** e introducimos una contraseña que siga la política.

Creamos un nuevo grupo llamado "evaluating". sudo addgroup evaluating

Añadimos el nuevo usuario al nuevo grupo. sudo adduser name_user evaluating

Comprobar que se haya introducido correctamente. getent group evaluating

Comprobar todos los grupos creados con el siguiente comando cat /etc/group | more

Ventajas y desventajas de la política de contraseñas fuerte

Que es más segura, difícil de vulnerar// que es más difícil de recordar (WTF?)

Comprobación con la política de contraseñas fuerte

- Configuración de **política de contraseñas fuerte** (script con los días) nano /etc/login.defs

-Script con los requisitos de la contraseña nano /etc/pam.d/common-password

minlen=10 // Cantidad mínima de caracteres.

ucredit=-1 // Mínimo debe contener una mayúscula. (" -" ya que debe contener como mínimo)

dcredit=-1 // Mínimo debe contener un dígito.

lcredit=-1 // Mínimo debe contener una minúscula.

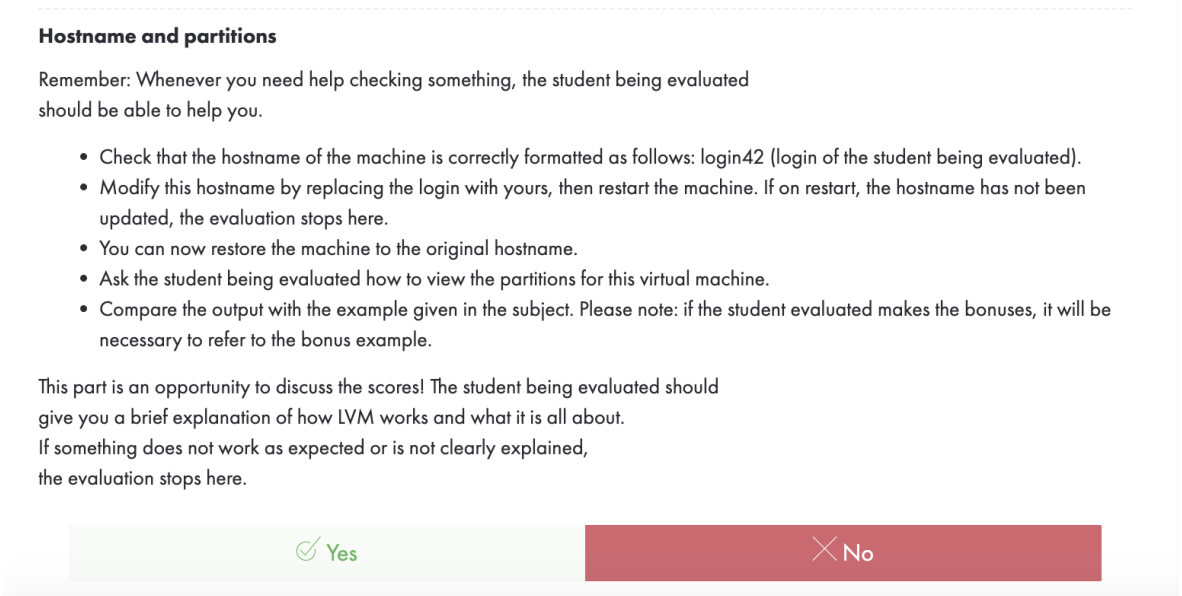
maxrepeat=3 // No puede tener más de 3 veces seguidas el mismo carácter.

reject_username // No puede contener el nombre del usuario.

difok=7 // Debe tener al menos 7 caracteres que no sean parte de la antigua contraseña.

enforce_for_root // Implementaremos esta política para el usuario root.

5. Particiones



Hostname and partitions

Remember: Whenever you need help checking something, the student being evaluated should be able to help you.

- Check that the hostname of the machine is correctly formatted as follows: login42 (login of the student being evaluated).
- Modify this hostname by replacing the login with yours, then restart the machine. If on restart, the hostname has not been updated, the evaluation stops here.
- You can now restore the machine to the original hostname.
- Ask the student being evaluated how to view the partitions for this virtual machine.
- Compare the output with the example given in the subject. Please note: if the student evaluated makes the bonuses, it will be necessary to refer to the bonus example.

This part is an opportunity to discuss the scores! The student being evaluated should give you a brief explanation of how LVM works and what it is all about.
If something does not work as expected or is not clearly explained, the evaluation stops here.

Yes | No

- Comprobar que el hostname es correcto. **hostname**

- Modificar hostname para reemplazar tu login por el del evaluador.

sudo nano /etc/hostname *y reemplazamos nuestro*

sudo nano /etc/hosts *y reemplazamos nuestro login por el nuevo.*

Reiniciar la máquina. sudo reboot

Una vez nos hemos logueado de nuevo podemos ver como el hostname se ha cambiado correctamente. **hostname**

- Como ver las particiones de la máquina virtual **lsblk**

- ¿Qué es LVM? "Logical Volume Manager", es un gestor de volúmenes lógicos (discos o particiones) usado en Linux.

Administración de espacio flexible: permite agregar o eliminar dispositivos de almacenamiento sin afectar la estructura de los sistemas de archivos.

Redimensionamiento en caliente: Puedes aumentar o reducir el tamaño de los volúmenes lógicos en tiempo real sin necesidad de detener el sistema o desmontar particiones..

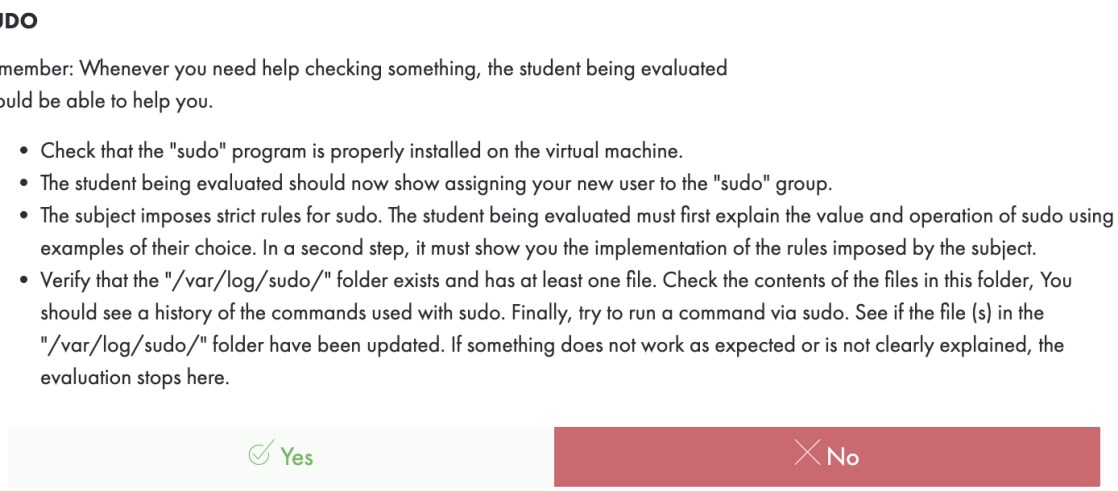
Snapshots: LVM permite crear instantáneas (snapshots) de los volúmenes lógicos.

Rendimiento y tolerancia a fallos: LVM ofrece opciones para implementar configuraciones RAID.

Mejora la organización del almacenamiento: LVM permite gestionar dispositivos de almacenamiento físico a través de conceptos más lógicos, como volúmenes físicos, grupos de volúmenes y volúmenes lógicos.

Portabilidad: Puedes mover volúmenes lógicos entre sistemas LVM compatibles, lo que facilita la migración de datos y la recuperación de sistemas.

6. Sudo



SUDO

Remember: Whenever you need help checking something, the student being evaluated should be able to help you.

- Check that the "sudo" program is properly installed on the virtual machine.
- The student being evaluated should now show assigning your new user to the "sudo" group.
- The subject imposes strict rules for sudo. The student being evaluated must first explain the value and operation of sudo using examples of their choice. In a second step, it must show you the implementation of the rules imposed by the subject.
- Verify that the "/var/log/sudo/" folder exists and has at least one file. Check the contents of the files in this folder, You should see a history of the commands used with sudo. Finally, try to run a command via sudo. See if the file (s) in the "/var/log/sudo/" folder have been updated. If something does not work as expected or is not clearly explained, the evaluation stops here.

Yes | No

-Comprobar que sudo está instalado **which sudo**

->Aunque es más correcto usar este **dpkg -s sudo**

-Introducimos usuario dentro de sudo **sudo adduser name_user sudo**

-Comprobar que está dentro del grupo **getent group sudo**

Sudo o Super User Do

Sudo sirve para elevar privilegios temporalmente, con el fin de ejecutar comandos o archivos que requieren permisos especiales, generalmente el usuario root.

Cuando un usuario utiliza el comando sudo, se le solicitará que ingrese su propia contraseña para confirmar su identidad antes de ejecutar el comando con privilegios elevados. Esto ayuda a garantizar que solo usuarios autorizados realicen cambios significativos en el sistema.

Por ejemplo, si un usuario normal quiere instalar un nuevo programa que afecta a todo el sistema, podría usar sudo para ejecutar el comando de instalación con los privilegios de superusuario. Esto proporciona un nivel adicional de seguridad al restringir el acceso a funciones críticas del sistema solo a usuarios autorizados.

`Ejemplo:` **`sudo apt-get update`** En este caso, `apt-get update` se utiliza para actualizar la lista de paquetes disponibles, pero al anteponerse con `sudo`, estás ejecutando la actualización con los privilegios de superusuario. Cuando ingreses este comando, se te pedirá que ingreses tu contraseña para confirmar que tienes los permisos necesarios para realizar la acción.

-Muestra la aplicación de las reglas impuestas para sudo por el subject (script).

nano /etc/sudoers.d/sudo_config

Defaults passwd_tries=3 // número de intentos para introducir una contraseña errónea

Defaults badpass_message="badpass, try again" // mensaje de error en caso de poner una mala contraseña

Defaults logfile="/var/log/sudo/sudo_config" // archivo donde quedan registrados todos los comandos sudo (este mismo)

Defaults log_input, log_output // Para que cada comando quede archivado en el directorio especificado (input y output)

Defaults iolog_dir="/var/log/sudo" // Para que cada comando quede archivado en el directorio especificado (input y output)

Defaults requiretty // Para activar el modo TTY (texto)

Defaults secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

// Para restringir los directorios utilizables por sudo

-Mostrar que la ruta /var/log/sudo/ existe cd /var/log/sudo

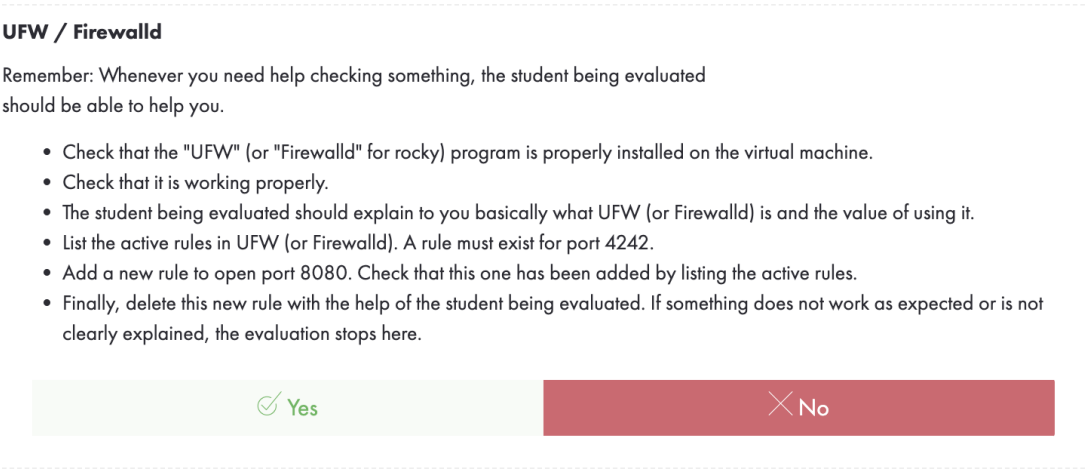
y contiene al menos un fichero, ls

Ver historial de los comandos utilizados con sudo cat sudo_config

Ejecuta un comando con sudo sudo nano pruebaSudo

y comprueba que se actualiza el fichero cat sudo_config

7. UFW /Firewall



UFW / Firewalld

Remember: Whenever you need help checking something, the student being evaluated should be able to help you.

- Check that the "UFW" (or "Firewalld" for rocky) program is properly installed on the virtual machine.
- Check that it is working properly.
- The student being evaluated should explain to you basically what UFW (or Firewalld) is and the value of using it.
- List the active rules in UFW (or Firewalld). A rule must exist for port 4242.
- Add a new rule to open port 8080. Check that this one has been added by listing the active rules.
- Finally, delete this new rule with the help of the student being evaluated. If something does not work as expected or is not clearly explained, the evaluation stops here.

Yes | No

-Comprueba que UFW está instalado **dpkg -s ufw**

-Comprueba que funciona correctamente **sudo service ufw status**

- ¿Qué es UFW?

(Uncomplicated Firewall) Es un firewall sencillo que utiliza la línea de comandos para configurar las iptables usando un pequeño número de comandos simples.

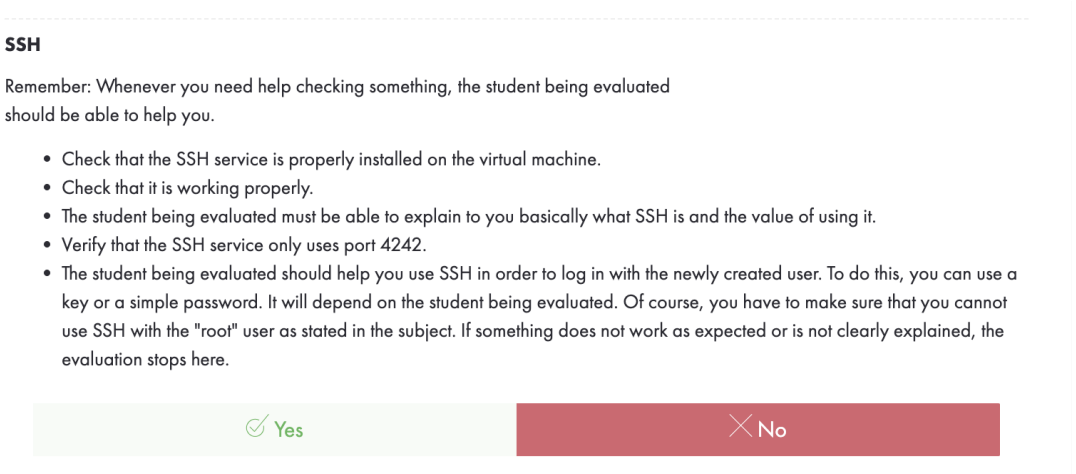
-Lista las reglas activas en UFW **sudo ufw status numbered**

-Crea una nueva regla para el puerto 8080 **sudo ufw allow 8080**

y comprueba que se ha añadido **sudo ufw status numbered**

-A continuación la borramos **sudo ufw delete** *“num_rule”*

8. SSH



SSH

Remember: Whenever you need help checking something, the student being evaluated should be able to help you.

- Check that the SSH service is properly installed on the virtual machine.
- Check that it is working properly.
- The student being evaluated must be able to explain to you basically what SSH is and the value of using it.
- Verify that the SSH service only uses port 4242.
- The student being evaluated should help you use SSH in order to log in with the newly created user. To do this, you can use a key or a simple password. It will depend on the student being evaluated. Of course, you have to make sure that you cannot use SSH with the "root" user as stated in the subject. If something does not work as expected or is not clearly explained, the evaluation stops here.

Yes | No

-Comprueba que el servicio ssh está instalado which ssh

-Comprueba que funciona correctamente y que solo funciona por el puerto 4242. sudo service ssh status

-Qué es ssh?

(Secure Shell) Es un protocolo de administración remota que le permite a los usuarios controlar y modificar sus servidores remotos a través de Internet o red local a través de un mecanismo de autenticación.

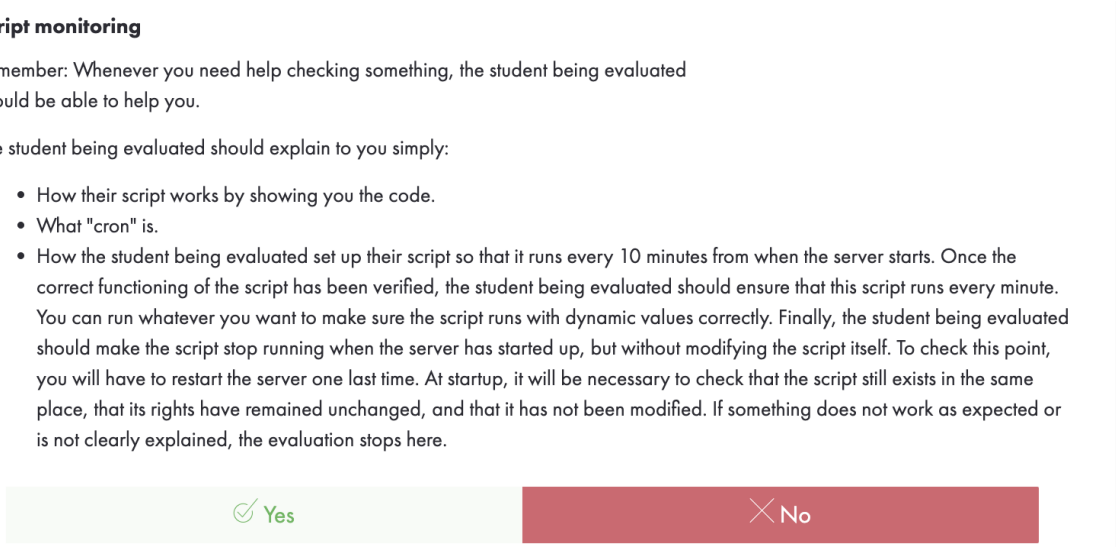
-Usa ssh para iniciar sesión con el usuario recién creado.

ssh evaluator@localhost -p 4242

-Asegúrate de que no puedes usar ssh con el usuario root.

ssh root@localhost -p 4242 *(nos tiene que dar error).*

10.Script



Script monitoring

Remember: Whenever you need help checking something, the student being evaluated should be able to help you.

The student being evaluated should explain to you simply:

- How their script works by showing you the code.
- What "cron" is.
- How the student being evaluated set up their script so that it runs every 10 minutes from when the server starts. Once the correct functioning of the script has been verified, the student being evaluated should ensure that this script runs every minute. You can run whatever you want to make sure the script runs with dynamic values correctly. Finally, the student being evaluated should make the script stop running when the server has started up, but without modifying the script itself. To check this point, you will have to restart the server one last time. At startup, it will be necessary to check that the script still exists in the same place, that its rights have remained unchanged, and that it has not been modified. If something does not work as expected or is not clearly explained, the evaluation stops here.

Yes | No

-Abrir Script para analizarlo nano monitoring.sh

Ejecutar Script sh monitoring.sh

-Modifica el tiempo de ejecución del script de 10 minutos a 1.

sudo crontab -u root -e

Haz que el script deje de ejecutarse cuando el servidor se haya iniciado, pero sin modificar el script.

sudo /etc/init.d/cron stop

Si queremos que vuelva a ejecutarse:

sudo /etc/init.d/cron start

- ¿Que es crontab?

Es un **administrador de procesos** en segundo plano. Los procesos indicados serán ejecutados en el momento que especifiques en el fichero crontab. Es el archivo que modificamos para que nuestro script "monitoring.sh" se ejecute cada 10 minutos.

-Explicación del script

-Arquitectura del SO y su versión de kernel

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

// se almacena en **arch** y luego la usaremos para imprimirla (similar a lo demás.)

-Numero de nucleos fisicos

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

// **grep** sirve para buscar coincidencias en cadenas de texto // Busca ese apartado en el archivo **/proc/cpuinfo** // **wc -l**: Este comando cuenta el número de líneas en la salida proporcionada por el comando grep

-mostrar el número de núcleos virtuales

```
cpuv=$(grep "processor" /proc/cpuinfo | wc -l)
```

// idem que en el anterior

-Memoria RAM

```
ram_total=$(free --mega | awk '$1 == "Mem:" {print $2}')

ram_use=$(free --mega | awk '$1 == "Mem:" {print $3}')

ram_percent=$(free --mega | awk '$1 == "Mem:" {printf("%.2f"), $3/$2*100}')
```

// **free** es un comando que nos arroja información varia sobre la RAM del ordenador (usada, libre, total, etc…)

//**–mega** nos da la información de free en megas, que es lo que nos pide el subject

//**awk '$1 == "Mem:"** sirve para filtrar la información de la primera fila llamada “mem” obtenida por **free**

// **print** y el número que va detrás corresponde con la columna de la fila que hemos seleccionado previamente.

// en el último caso **{printf("%.2f"), $3/$2*100}** se hace el porcentaje de la memoria usada con formato de dos decimales.

-Memoria del disco

```
disk_total=$(df -m | grep "/dev/" | grep -v "/boot" | awk '{disk_t += $2}
END {printf ("%.1fGb\n"), disk_t/1024}')

disk_use=$(df -m | grep "/dev/" | grep -v "/boot" | awk '{disk_u += $3} END
{print disk_u}')

disk_percent=$(df -m | grep "/dev/" | grep -v "/boot" | awk '{disk_u += $3}
{disk_t+= $2} END {printf("%d"), disk_u/disk_t*100}')
```

//**df -m** es un comando que se utiliza para obtener el tamaño del disco en megas (-m)

// usamos otra vez **grep** primero para buscar dev (info sobre discos) y después en su versión grep -v (para excluir) boot (líneas con partición de arranque).

// usamos **awk** para obtener información de las columnas:

{disk_t += $2}: Suma el valor de cada línea en la segunda columna (espacio total en disco en megabytes).
END {printf ("%.1fGb\n"), disk_t/1024}: Al llegar al final del procesamiento de todas las líneas, imprime el resultado total en gigabytes con un formato específico que muestra un decimal. entre 1024 para convertir megabytes a gigabytes.

-Porcentaje uso de CPU

```
cpul=$(vmstat 1 2 | tail -1 | awk '{printf $15}')

cpu_op=$(expr 100 - $cpul)

cpu_fin=$(printf "%.1f" $cpu_op)
```

// **vmstat** comando muestra estadísticas sobre la actividad del sistema.

//opciones **1 2**: 1 cada cuanto debe muestrear en segundos, 2 cantidad de muestras que debe tomar.

//**tail -1**: Muestra solo la última línea de la salida generada por vmstat

//**awk '{printf $15}'**: Extrae el valor de la columna 15. En las estadísticas de vmstat, la columna 15 representa la carga promedio de la CPU. (id)

// las otras dos variables son cálculos para extraer el porcentaje, restándole 100 y formateando con un decimal. lo que nos daría la carga de CPU

-Último reinicio

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

// **awk** nos está filtrando la fila que ponga sistem e imprimirá las columnas 3 y 4 (fecha y hora)

alternadas por un espacio (“ “)

-Uso de LVM

```
lvmu=$(if [ $(lsblk | grep "lvm" | wc -l) -gt 0 ]; then echo yes; else echo no; fi)
```

// **lsblk** muestra las particiones y los dispositivos de almacenamiento.

//**grep "lvm"**: Filtra las líneas que contienen la cadena "lvm".

// **wc -l:** Cuenta el número de líneas en la salida del comando anterior.

// **[... -gt 0]** para verificar si el número de líneas que contienen la cadena "lvm" es mayor que cero.

//then echo yes; else echo no; fi: En caso de que la condición sea verdadera (es decir, si se encuentra "lvm"), se imprime "yes"; de lo contrario, se imprime "no". Lo cual nos dice si se usa LVM

-Conexiones TCP

```
tcpc=$(ss -ta | grep ESTAB | wc -l)
```

//**ss -ta, ss** muestra información sobre las conexiones de red// **-t** filtra las conexiones TCP// **-a** muestra tanto las conexiones activas como las pasivas.

// **grep ESTAB**: Filtra las líneas que contienen la cadena "ESTAB", que indica conexiones establecidas.

// **wc -l**: Cuenta el número de líneas en la salida del comando anterior (wc comando usado para contar en unix, -l cuenta líneas)

IMPORTANTE: otras opciones con wc -l: Muestra número de líneas. -w**:** Muestra número de palabras. -c: Muestra solo el número de bytes.

-Número de usuarios

```
ulog=$(users | wc -w)
```

// users nos dice los usuarios, y con wc -w contamos la cantidad de palabras para entregar un número

-Dirección IP y MAC

```
ip=$(hostname -I)
```

// **hostname -I** nos da la IP , -I sirve para ver la dirección porque sin esta nos da el nombre del host.

```
mac=$(ip link | grep "link/ether" | awk '{print $2}')
```

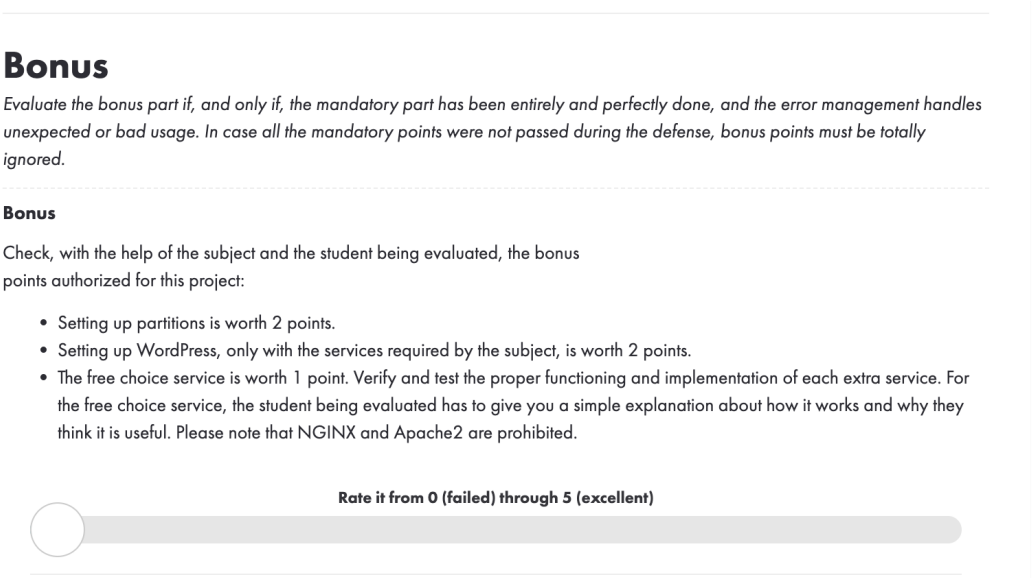
// **ip link** nos da info sobre las interfaces de red //**grep** filtra las palabras seleccionadas// **awk** selecciona la segunda columna

-Número de comandos ejecutados con sudo

```
cmnd=$(journalctl _COMM=sudo | grep COMMAND | wc -l)
```

//Journalctl es una herramienta que se encarga de recopilar y administrar los registros del sistema,// **COMM=sudo,** filtra y muestra solo los relacionados con sudo. // grep filtra las líneas con el texto coincidente COMMAND // wc -l cuenta las líneas (línea por comando).

11. Bonus



Bonus

Evaluate the bonus part if, and only if, the mandatory part has been entirely and perfectly done, and the error management handles unexpected or bad usage. In case all the mandatory points were not passed during the defense, bonus points must be totally ignored.

Bonus

Check, with the help of the subject and the student being evaluated, the bonus points authorized for this project:

- Setting up partitions is worth 2 points.
- Setting up WordPress, only with the services required by the subject, is worth 2 points.
- The free choice service is worth 1 point. Verify and test the proper functioning and implementation of each extra service. For the free choice service, the student being evaluated has to give you a simple explanation about how it works and why they think it is useful. Please note that NGINX and Apache2 are prohibited.

Rate it from 0 (failed) through 5 (excellent)

- Ver particiones como en el subject **lsblk**

- Ver página de Wordpress:

    -Meter en el **navegador localhost**

        Nombre de usuario:

        Contraseña:

    -Ver lighttpd

    -Ver mariaDB teclear en terminal mariadb

        una vez dentro si tecleamos SHOW DATABASES;

    -Ver PHP

    -Ver OpenLiteSpeed poner en el navegador **localhost:7080**

        Nombre de usuario:

        Contraseña:

BONUS (Explicaciones de cada cosa).

1. **OpenLiteSpeed.**

Es un servidor web de código abierto diseñado para servir sitios web de manera eficiente y rápida. Es una alternativa de código abierto a LiteSpeed Web Server, que es un servidor web comercial conocido por su alta velocidad y eficiencia.

Las características clave de OpenLiteSpeed incluyen:

Rendimiento: OpenLiteSpeed está diseñado para ser rápido y eficiente, lo que lo hace ideal para servir sitios web de alto tráfico.

Escalabilidad: OpenLiteSpeed es escalable y puede manejar una gran cantidad de conexiones simultáneas.

Seguridad: OpenLiteSpeed ofrece varias características de seguridad, como protección contra ataques DDoS, filtrado de solicitudes, y la capacidad de configurar reglas de seguridad personalizadas.

Panel de control web: OpenLiteSpeed ofrece un panel de control web llamado "WebAdmin Console" que facilita la administración y configuración del servidor.

Compatibilidad con PHP: OpenLiteSpeed es compatible con PHP y se integra fácilmente con aplicaciones web que utilizan PHP.

SSL/TLS: Ofrece soporte para SSL/TLS, lo que permite la implementación de conexiones seguras a través de HTTPS.

Control de ancho de banda: OpenLiteSpeed proporciona opciones para limitar el ancho de banda de las conexiones entrantes, lo que es útil para gestionar la velocidad de transferencia de datos.

Para ver la página de OpenLiteSpeed

Poner en el navegador localhost:7080

Nombre de usuario: wp-login

Contraseña: …

*Con el comando **sudo ufw status** podemos **comprobar que puertos tenemos abiertos** (ya que debemos abrir el puerto 80 después de instalar OpenLiteSpeed).

2. **WordPress**

Sistema de gestión de contenidos (CMS por sus siglas en inglés) enfocado a la creación de cualquier tipo de página web.

Permite administrar contenido de forma sencilla, se integra bien con herramientas SEO, tiene escalabilidad, comunidad activa, actualizaciones regulares… etc

*Para instalar la última versión de WordPress primero debemos instalar **wget y zip.**

wget Es una herramienta sencilla de línea de comandos que se utiliza para descargar archivos de la web.

zip Es una utilidad de línea de comandos para comprimir y descomprimir archivos en formato ZIP.

Para ver la página de Wordpress

Poner en el navegador http://localhost

Para acceder al panel de administración de Wordpress

Escribimos en el navegador: localhost/wp-admin
Nombre de usuario: wp-login
Contraseña: …

3. **Mariadb**

Es un sistema de gestión de bases de datos (DBMS).

MariaDB está diseñado para ser una alternativa compatible con MySQ, pero a menudo incluye mejoras de rendimiento en comparación con MySQL.

Tiene otras características como almacenamiento de alto rendimiento (motores de almacenamiento avanzados), seguridad mejorada (funciones como el cifrado de datos en reposo y en tránsito, autenticación avanzada y auditoría de eventos.

Además es de código abierto y gratuito

*Para acceder a Mariadb y comprobar que está en funcionamiento, podemos acceder a través de la terminal y teclear simplemente mariadb , una vez dentro si tecleamos SHOW DATABASES;

4. **PHP**

Es un lenguaje de programación. Se utiliza principalmente para desarrollar aplicaciones web dinámicas y sitios web interactivos. PHP se ejecuta en el lado del servidor.

Con esto instalaremos los paquetes necesarios para poder ejecutar aplicaciones web escritas en lenguaje PHP y que necesiten conectarse a una base de datos MySQL (una base de datos MySQL

es como un archivo electrónico que organiza datos de manera que sea fácil de buscar, actualizar y administrar. Es especialmente útil para aplicaciones web y otras aplicaciones que requieren un almacenamiento de datos sólido y estructurado).

Recursos útiles:

-Comandos Linux
https://youtu.be/gd7BXuUQ91w?si=2yBdOmsbLGxrZahJ

-Tutorial de instalación de Openlitespeed (Alternativa a servicio adicional en el bonus).
https://imperioweb.net/como-instalar-openlitespeed-en-debian-12-bookworm