🏆 *Reto DÍA 8: ADMINISTRACIÓN BÁSICA DE LINUX SERVER* 🖥️🐧

Abraham Caamaño Martínez

✅ 1. Crear Grupos Específicos

Creamos los grupos que definirán los roles dentro de la empresa.

```
vboxuser@UbuntuServer:~$ sudo groupadd developers
[sudo] password for vboxuser:
vboxuser@UbuntuServer:~$ sudo groupadd admins
vboxuser@UbuntuServer:~$ sudo groupadd interns
```

✅ 2. Crear Usuarios y Asignarlos a sus Grupos

Ahora creamos los usuarios y los añadimos a sus grupos correspondientes.

```
vboxuser@UbuntuServer:~$ sudo adduser dev_user
info: Adding user `dev_user' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `dev_user' (1008) ...
info: Adding new user `dev_user' (1008) with group `dev_user (1008)' ...
info: Creating home directory `/home/dev_user' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
No password has been supplied.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for dev_user
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
info: Adding new user `dev_user' to supplemental / extra groups `users' ...
info: Adding user `dev_user' to group `users' ...
vboxuser@UbuntuServer:~$ sudo usermod -aG developers dev_user
```

```
vboxuser@UbuntuServer:~$ sudo adduser sysadmin
info: Adding user `sysadmin' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `sysadmin' (1009) ...
info: Adding new user `sysadmin' (1009) with group `sysadmin (1009)' ...
info: Creating home directory `/home/sysadmin' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for sysadmin
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
info: Adding new user `sysadmin' to supplemental / extra groups `users' ...
info: Adding user `sysadmin' to group `users' ...
vboxuser@UbuntuServer:~$ sudo usermod -aG admins,sudo sysadmin
vboxuser@UbuntuServer:~$ sudo adduser intern_user
info: Adding user `intern_user' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `intern_user' (1010) ...
info: Adding new user `intern_user' (1010) with group `intern_user (1010)' ...
info: Creating home directory `/home/intern_user' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for intern_user
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
info: Adding new user `intern_user' to supplemental / extra groups `users' ...
info: Adding user `intern_user' to group `users' ...
vboxuser@UbuntuServer:~$ sudo usermod -aG interns intern_user
```

✅ 3. Configurar Permisos de Acceso a Carpetas

```
vboxuser@UbuntuServer:~$ sudo mkdir -p /srv/data/projects
vboxuser@UbuntuServer:~$ sudo mkdir -p /srv/data/reports
vboxuser@UbuntuServer:~$ sudo mkdir -p /srv/data/shared_docs
vboxuser@UbuntuServer:~$
```

```
vboxuser@UbuntuServer:~$ sudo chown -R root:developers /srv/data/projects
vboxuser@UbuntuServer:~$ sudo chmod -R 2770 /srv/data/projects
vboxuser@UbuntuServer:~$ sudo chown -R root:admins /srv/data/reports
vboxuser@UbuntuServer:~$ sudo chmod -R 2770 /srv/data/reports
vboxuser@UbuntuServer:~$ sudo chown -R root:interns /srv/data/shared_docs
vboxuser@UbuntuServer:~$ sudo chmod -R 750 /srv/data/shared_docs
vboxuser@UbuntuServer:~$
```

🚀 Fase 2: Configuración de Tareas Automatizadas

✅ 1. Programar un Backup Automático con cron

Crear el directorio de backups

sudo mkdir -p /var/backups/projects

Editamos el crontab:

sudo crontab -e

Y añadimos la siguiente linea:

```
#
# 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/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
0 3 * * * /usr/bin/tar -czf /var/backups/projects/projects_backup_$(date +\%Y-\%m-\%d).tar.gz /srv/data/projects _
```

✅ 2. Script de Notificación de Actividad

Este script enviará un resumen de los últimos inicios de sesión a un archivo de log:

```
  GNU nano 7.2                                                    /usr/loc
#!/bin/bash

#Fichero donde guardo el reporte
LOG="/var/log/server_activity.log"

(
  echo "--- Reporte de actividad: $(date +"%Y-%m-%d %H:%M:%S") ---"
  echo ""
  echo ">> Ultimos 5 inicios de sesion:"
  last -n 5
  echo ""
  echo ">> Uso del disco:"
  df -h /
  echo "--- Fin del reporte ---"
  echo ""
) >> $LOG
```

Lo hacemos ejecutable:

```
vboxuser@UbuntuServer:~$ sudo chmod +x /usr/local/sbin/activity_monitor.sh
[sudo] password for vboxuser:
```

Y programamos su ejecución con cron:

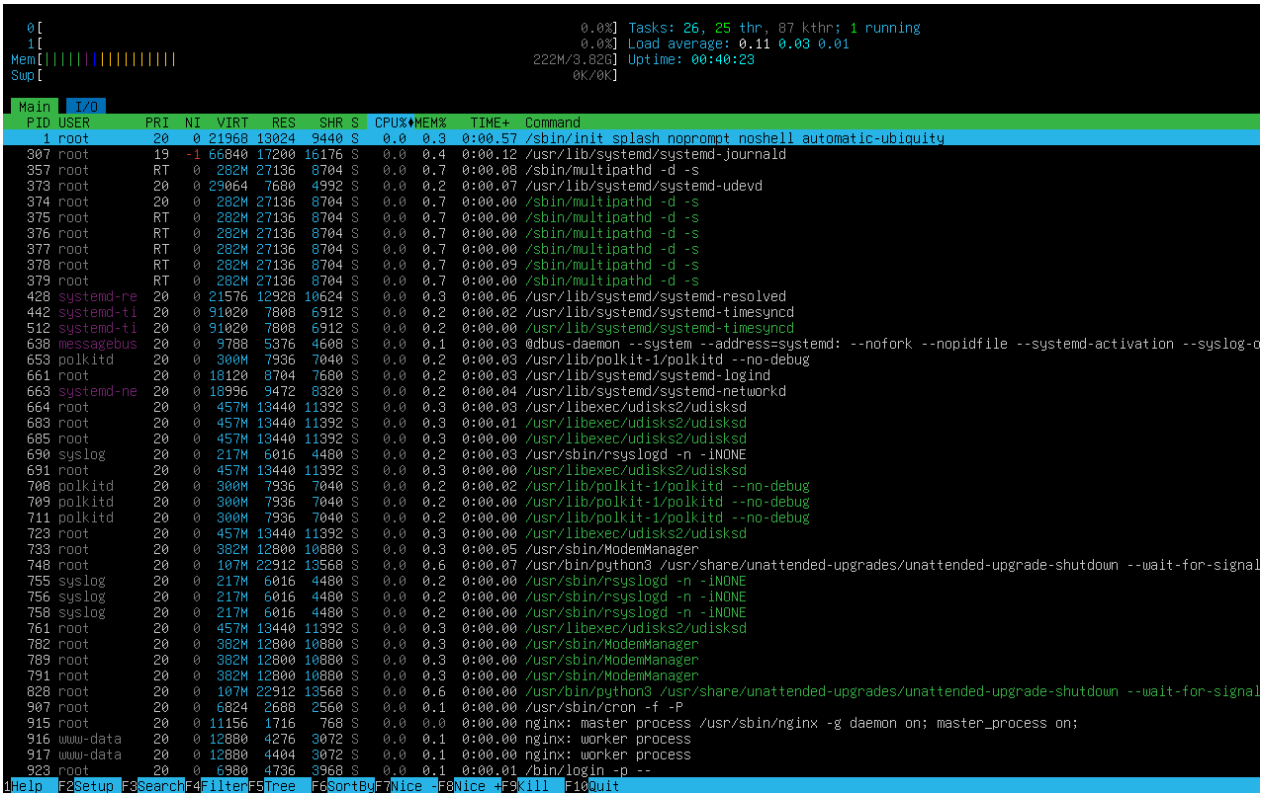
```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), 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/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
0 3 * * * /usr/bin/tar -czf /var/backups/projects/projects_backup_$(date +\%Y-\%m-\%d).tar.gz /srv/data/projects

0 * * * * /usr/local/sbin/activity_monitor.sh
```

🚀 Fase 3: Monitoreo y Optimización del Servidor

Instalamos htop con:

sudo apt install htop

Lo abrimos con: ***htop***



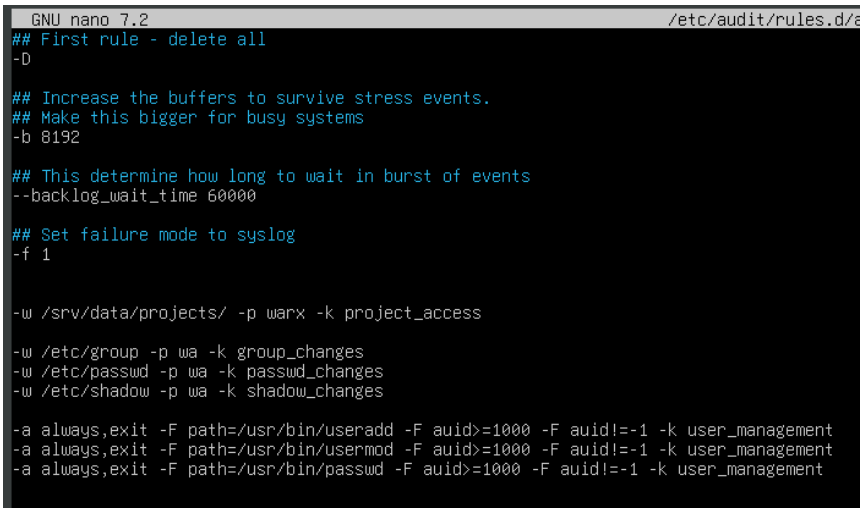
✅ 2. Configurar Logs de Auditoría

Lo instalamos con:

sudo apt install auditd

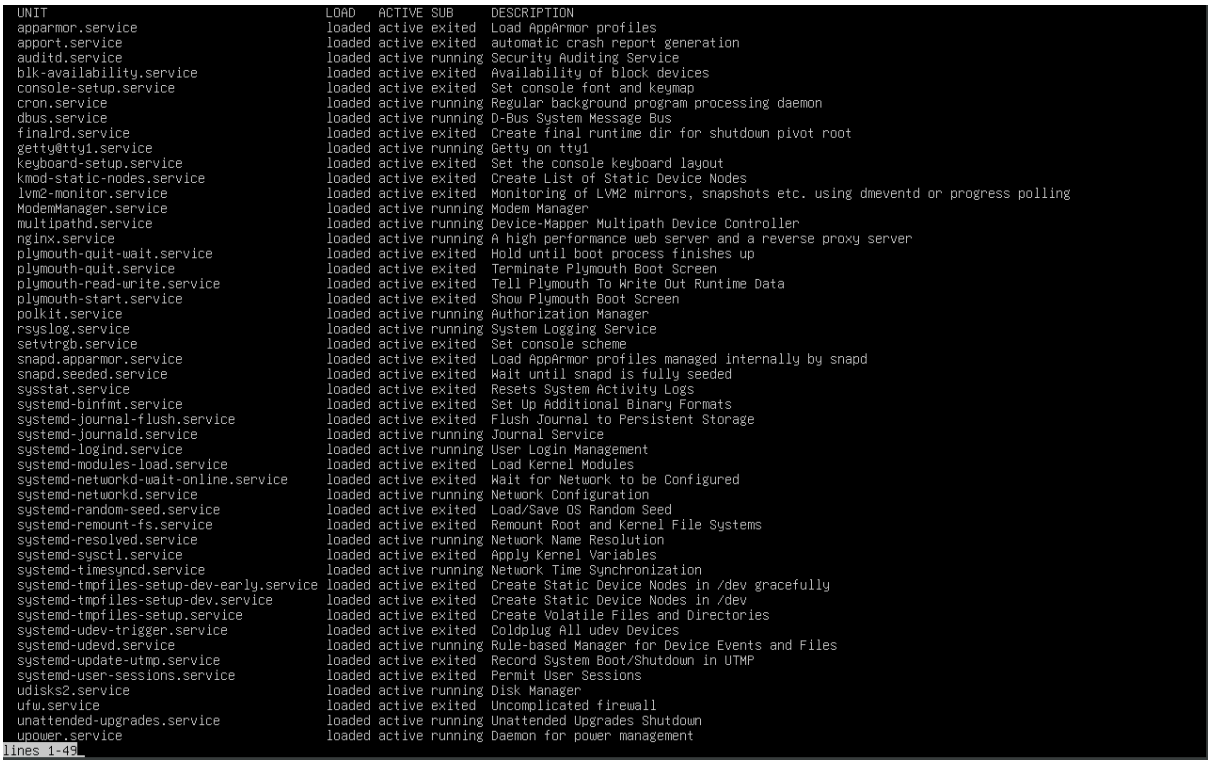
Editamos el archivo de auditoría con

sudo nano /etc/audit/rules.d/audit.rules

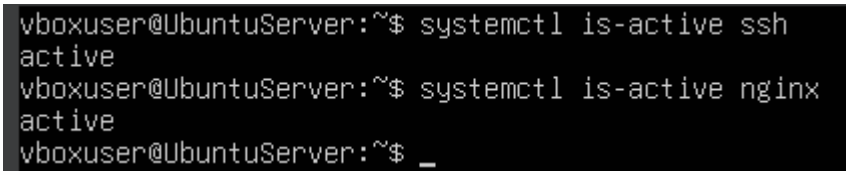


🚀 Fase 1: Análisis de Servicios del Sistema

✅ Listar todos los servicios activos del sistema usando systemctl list-units --type=service.



✅ Comprobar si el servidor web (instalado el día anterior) está activo, habilitado y funcionando.



🚀 Fase 2: Gestión avanzada de servicios con Systemd

✅ Detener, reiniciar y habilitar al arranque el servicio web (apache2 o nginx).

```
vboxuser@UbuntuServer:~$ sudo systemctl stop nginx
vboxuser@UbuntuServer:~$ sudo systemctl start nginx
vboxuser@UbuntuServer:~$ sudo systemctl restart nginx
vboxuser@UbuntuServer:~$ sudo systemctl disable nginx
Synchronizing state of nginx.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install disable nginx
Removed "/etc/systemd/system/multi-user.target.wants/nginx.service".
vboxuser@UbuntuServer:~$ sudo systemctl enable nginx
Synchronizing state of nginx.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable nginx
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
vboxuser@UbuntuServer:~$ _
```

✅ Modificar la configuración de uno de los servicios para que se reinicie automáticamente si falla (Restart=always en su .service).

✅ Crear un alias para reiniciar rápidamente el servicio desde .bashrc o .zshrc.

```
#Alias para el servicio de SSH
alias web-reiniciar='sudo systemctl restart nginx'
alias web-estado='sudo systemctl status nginx'
```

```
vboxuser@UbuntuServer:~$
vboxuser@UbuntuServer:~$ source ~/.bashrc
vboxuser@UbuntuServer:~$ web-reiniciar
vboxuser@UbuntuServer:~$ web-estado
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-06-23 09:54:24 UTC; 5s ago
       Docs: man:nginx(8)
    Process: 1519 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 1520 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 1522 (nginx)
      Tasks: 3 (limit: 4605)
     Memory: 2.3M (peak: 2.5M)
        CPU: 7ms
     CGroup: /system.slice/nginx.service
             ├─1522 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ├─1523 "nginx: worker process"
             └─1524 "nginx: worker process"

Jun 23 09:54:24 UbuntuServer systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Jun 23 09:54:24 UbuntuServer systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
vboxuser@UbuntuServer:~$
```

🚀 Fase 3: Creación de un servicio personalizado

✅ Crear un script Bash llamado saludo.sh que escriba "¡Servidor iniciado correctamente!" en un archivo /var/log/saludo.log.

```
  GNU nano 7.2                                                    /usr/l
#!/bin/bash
echo "Servidor iniciado correctamente - $(date)" >> /var/log/saludo.log
```

✅ Crear un nuevo servicio de systemd llamado saludo.service que ejecute ese script automáticamente al iniciar el sistema.

```
  GNU nano 7.2
[Unit]

After=network.target

[Service]

Type=simple
ExecStart=/usr/local/sbin/saludo.sh

[Install]

WantedBy=multi-user.target
```

✅ Comprobar que el servicio:

- **Está habilitado**
- **Se ejecuta al arrancar**
- **Crea el archivo de log correctamente**

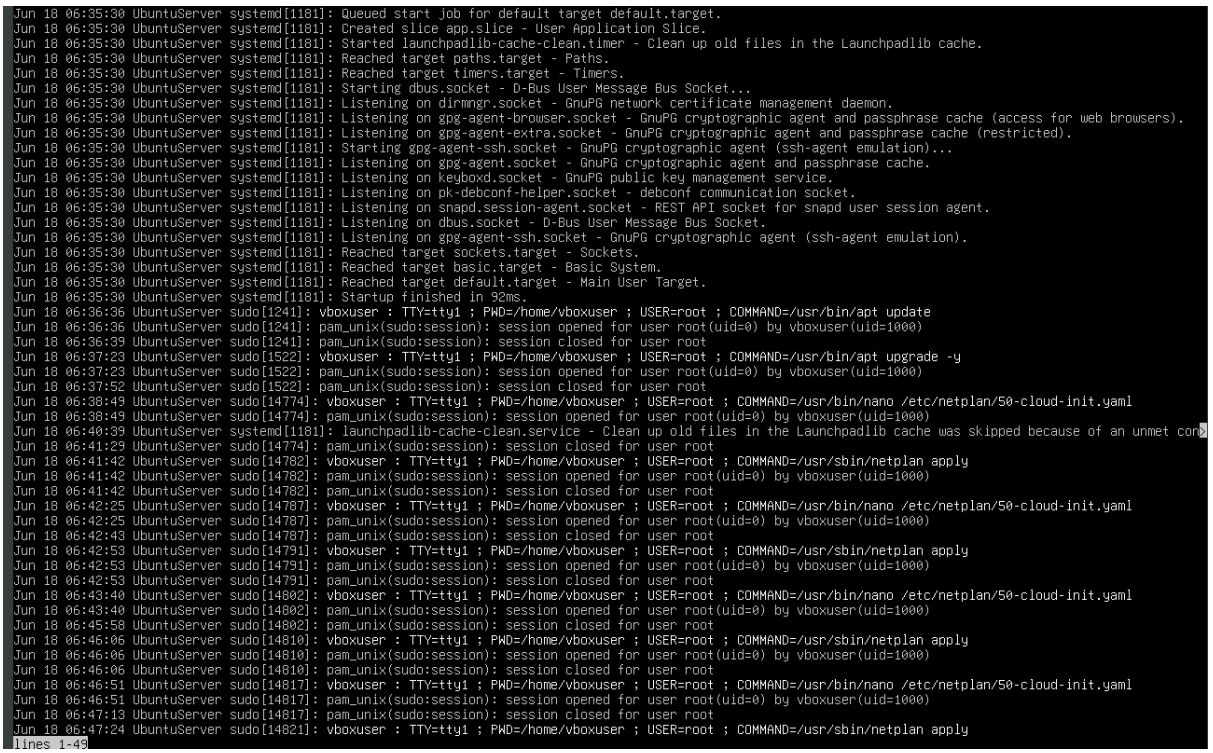
```
vboxuser@UbuntuServer:~$ cat /var/log/saludo.log
Servidor iniciado correctamente - Mon Jun 23 10:01:48 AM UTC 2025
Servidor iniciado correctamente - Mon Jun 23 10:03:52 AM UTC 2025
vboxuser@UbuntuServer:~$
```

```
vboxuser@UbuntuServer:~$ systemctl status saludo.service
● saludo.service
     Loaded: loaded (/etc/systemd/system/saludo.service; enabled; preset: enabled)
     Active: inactive (dead) since Mon 2025-06-23 10:03:52 UTC; 57s ago
   Duration: 62ms
    Process: 683 ExecStart=/usr/local/sbin/saludo.sh (code=exited, status=0/SUCCESS)
   Main PID: 683 (code=exited, status=0/SUCCESS)
        CPU: 4ms

Warning: some journal files were not opened due to insufficient permissions.
vboxuser@UbuntuServer:~$ _
```

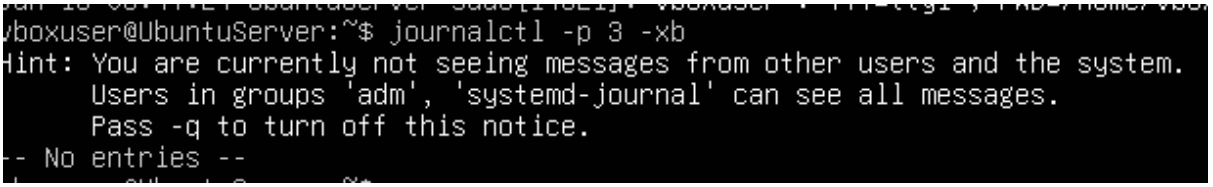
```
vboxuser@UbuntuServer:~$ systemctl status saludo
● saludo.service
     Loaded: loaded (/etc/systemd/system/saludo.service; enabled; preset: enabled)
     Active: inactive (dead) since Mon 2025-06-23 10:03:52 UTC; 8min ago
   Duration: 62ms
    Process: 683 ExecStart=/usr/local/sbin/saludo.sh (code=exited, status=0/SUCCESS)
   Main PID: 683 (code=exited, status=0/SUCCESS)
        CPU: 4ms
```

🚀 Fase 4: Monitorización y logs

✅ Visualizar los logs de los servicios anteriores con journalctl.



✅ Filtrar los mensajes de error o advertencia (journalctl -p 3 -xb).



✅ Registrar el estado del servicio saludo y guardar una copia del log en /srv/logs/saludo_journal.log.

(Se hace en el paso 3 de la fase 3)