



# **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/local/sbin/activity_monitor.sh
#!/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:

```
GNU nano 7.2 /etc/crontab
# 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 -czf /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***

```
0[          0.0%] Tasks: 26, 25 thr, 87 kthr: 1 running
1[          0.0%] Load average: 0.11 0.03 0.01
Mem[|||||] 222M/3.82G Uptime: 00:40:23
Sup[          0K/0K]

Main I/O
PID USER      PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
1 root         20    0 21968 13024  9440  S   0.0  0.3   0:00.57 /sbin/init splash noprompt noshell automatic-ubiquity
307 root        19   -1 66840 17200 16176  S   0.0  0.4   0:00.12 /usr/lib/systemd/systemd-journald
357 root        RT    0 282M 27136  8704  S   0.0  0.7   0:00.08 /sbin/multipathd -d -s
373 root        20    0 29064 7680  4992  S   0.0  0.2   0:00.07 /usr/lib/systemd/systemd-udevd
374 root        20    0 282M 27136  8704  S   0.0  0.7   0:00.00 /sbin/multipathd -d -s
375 root        RT    0 282M 27136  8704  S   0.0  0.7   0:00.00 /sbin/multipathd -d -s
376 root        RT    0 282M 27136  8704  S   0.0  0.7   0:00.00 /sbin/multipathd -d -s
377 root        RT    0 282M 27136  8704  S   0.0  0.7   0:00.00 /sbin/multipathd -d -s
378 root        RT    0 282M 27136  8704  S   0.0  0.7   0:00.09 /sbin/multipathd -d -s
379 root        RT    0 282M 27136  8704  S   0.0  0.7   0:00.00 /sbin/multipathd -d -s
428 systemd-re  20    0 21576 12928 10624  S   0.0  0.3   0:00.06 /usr/lib/systemd/systemd-resolved
442 systemd-tl  20    0 91020 7808  6912  S   0.0  0.2   0:00.02 /usr/lib/systemd/systemd-timesyncd
512 systemd-tl  20    0 91020 7808  6912  S   0.0  0.2   0:00.00 /usr/lib/systemd/systemd-timesyncd
638 messagebus  20    0 9788 5376  4608  S   0.0  0.1   0:00.03 @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog=0
653 polkitd     20    0 300M 7936  7040  S   0.0  0.2   0:00.03 /usr/lib/polkit-1/polkitd --no-debug
661 root        20    0 18120 8704  7680  S   0.0  0.2   0:00.03 /usr/lib/systemd/systemd-logind
663 systemd-ne  20    0 18996 9472  8320  S   0.0  0.2   0:00.04 /usr/lib/systemd/systemd-networkd
664 root        20    0 457M 13440 11392  S   0.0  0.3   0:00.03 /usr/libexec/udisks2/udisksd
683 root        20    0 457M 13440 11392  S   0.0  0.3   0:00.01 /usr/libexec/udisks2/udisksd
685 root        20    0 457M 13440 11392  S   0.0  0.3   0:00.00 /usr/libexec/udisks2/udisksd
690 syslog      20    0 217M 6016  4480  S   0.0  0.2   0:00.03 /usr/sbin/rsyslogd -n -iNONE
691 root        20    0 457M 13440 11392  S   0.0  0.3   0:00.00 /usr/libexec/udisks2/udisksd
708 polkitd     20    0 300M 7936  7040  S   0.0  0.2   0:00.02 /usr/lib/polkit-1/polkitd --no-debug
709 polkitd     20    0 300M 7936  7040  S   0.0  0.2   0:00.00 /usr/lib/polkit-1/polkitd --no-debug
711 polkitd     20    0 300M 7936  7040  S   0.0  0.2   0:00.00 /usr/lib/polkit-1/polkitd --no-debug
723 root        20    0 457M 13440 11392  S   0.0  0.3   0:00.00 /usr/libexec/udisks2/udisksd
733 root        20    0 382M 12800 10880  S   0.0  0.3   0:00.05 /usr/sbin/ModemManager
748 root        20    0 107M 22912 13568  S   0.0  0.6   0:00.07 /usr/bin/python3 /usr/share/unattended-upgrades/unattended-upgrade-shutdown --wait-for-signal
755 syslog      20    0 217M 6016  4480  S   0.0  0.2   0:00.00 /usr/sbin/rsyslogd -n -iNONE
756 syslog      20    0 217M 6016  4480  S   0.0  0.2   0:00.00 /usr/sbin/rsyslogd -n -iNONE
758 syslog      20    0 217M 6016  4480  S   0.0  0.2   0:00.00 /usr/sbin/rsyslogd -n -iNONE
761 root        20    0 457M 13440 11392  S   0.0  0.3   0:00.00 /usr/libexec/udisks2/udisksd
782 root        20    0 382M 12800 10880  S   0.0  0.3   0:00.00 /usr/sbin/ModemManager
789 root        20    0 382M 12800 10880  S   0.0  0.3   0:00.00 /usr/sbin/ModemManager
791 root        20    0 382M 12800 10880  S   0.0  0.3   0:00.00 /usr/sbin/ModemManager
828 root        20    0 107M 22912 13568  S   0.0  0.6   0:00.00 /usr/bin/python3 /usr/share/unattended-upgrades/unattended-upgrade-shutdown --wait-for-signal
907 root        20    0 6824 2688  2560  S   0.0  0.1   0:00.00 /usr/sbin/cron -f -P
915 root        20    0 11156 1716  768  S   0.0  0.0   0:00.00 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
916 www-data    20    0 12880 4276  3072  S   0.0  0.1   0:00.00 nginx: worker process
917 www-data    20    0 12880 4404  3072  S   0.0  0.1   0:00.00 nginx: worker process
923 root        20    0 5980 4736  3968  S   0.0  0.1   0:00.01 /bin/login -p --

Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit
```

## ✓ 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
```

```
GNU nano 7.2 /etc/audit/rules.d/audit.rules
## First rule - delete all
-D

## Increase the buffers to survive stress events.
## Make this bigger for busy systems
-b 8192

## This determine how long to wait in burst of events
--backlog_wait_time 60000

## Set failure mode to syslog
-f 1

-w /srv/data/projects/ -p warx -k project_access

-w /etc/group -p wa -k group_changes
-w /etc/passwd -p wa -k passwd_changes
-w /etc/shadow -p wa -k shadow_changes

-a always,exit -F path=/usr/bin/useradd -F auid>=1000 -F auid!=1 -k user_management
-a always,exit -F path=/usr/bin/usermod -F auid>=1000 -F auid!=1 -k user_management
-a always,exit -F path=/usr/bin/passwd -F auid>=1000 -F auid!=1 -k user_management
```



## Fase 1: Análisis de Servicios del Sistema



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

```
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
apparmor.service                   loaded active exited Load AppArmor profiles
apport.service                     loaded active exited automatic crash report generation
auditd.service                     loaded active running Security Auditing Service
blk-availability.service            loaded active exited Availability of block devices
console-setup.service              loaded active exited Set console font and keymap
cron.service                       loaded active running Regular background program processing daemon
dbus.service                       loaded active running D-Bus System Message Bus
finalrd.service                    loaded active exited Create final runtime dir for shutdown pivot root
getty@tty1.service                 loaded active running Getty on tty1
keyboard-setup.service              loaded active exited Set the console keyboard layout
kmod-static-nodes.service           loaded active exited Create List of Static Device Nodes
lvm2-monitor.service               loaded active exited Monitoring of LVM2 mirrors, snapshots etc. using dmeventd or progress polling
ModemManager.service               loaded active running Modem Manager
multipathd.service                 loaded active running Device-Mapper Multipath Device Controller
nginx.service                      loaded active running A high performance web server and a reverse proxy server
plymouth-quit-wait.service          loaded active exited Hold until boot process finishes up
plymouth-quit.service              loaded active exited Terminate Plymouth Boot Screen
plymouth-read-write.service         loaded active exited Tell Plymouth To Write Out Runtime Data
plymouth-start.service              loaded active exited Show Plymouth Boot Screen
polkit.service                     loaded active running Authorization Manager
rsyslog.service                   loaded active running System Logging Service
setvtrgb.service                   loaded active exited Set console scheme
snapd.apparmor.service              loaded active exited Load AppArmor profiles managed internally by snapd
snapd.seeded.service               loaded active exited Wait until snapd is fully seeded
sysstat.service                    loaded active exited Resets System Activity Logs
systemd-binfmt.service              loaded active exited Set Up Additional Binary Formats
systemd-journal-flush.service       loaded active exited Flush Journal to Persistent Storage
systemd-journald.service            loaded active running Journal Service
systemd-logind.service              loaded active running User Login Management
systemd-modules-load.service         loaded active exited Load Kernel Modules
systemd-networkd-wait-online.service loaded active exited Wait for Network to be Configured
systemd-networkd.service            loaded active running Network Configuration
systemd-random-seed.service          loaded active exited Load/Save OS Random Seed
systemd-remount-fs.service           loaded active exited Remount Root and Kernel File Systems
systemd-resolved.service            loaded active running Network Name Resolution
systemd-sysctl.service              loaded active exited Apply Kernel Variables
systemd-timesyncd.service            loaded active running Network Time Synchronization
systemd-tmpfiles-setup-dev-early.service loaded active exited Create Static Device Nodes in /dev gracefully
systemd-tmpfiles-setup-dev.service  loaded active exited Create Static Device Nodes in /dev
systemd-tmpfiles-setup.service       loaded active exited Create Volatile Files and Directories
systemd-udev-trigger.service         loaded active exited Coldplug All udev Devices
systemd-udevd.service               loaded active running Rule-based Manager for Device Events and Files
systemd-update-utmp.service          loaded active exited Record System Boot/Shutdown in UTMP
systemd-user-sessions.service        loaded active exited Permit User Sessions
udisks2.service                    loaded active running Disk Manager
ufw.service                         loaded active exited Uncomplicated firewall
unattended-upgrades.service          loaded active running Unattended Upgrades Shutdown
upower.service                     loaded active running Daemon for power management
```

lines 1-49



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

```
vboxuser@UbuntuServer:~$ systemctl is-active ssh
active
vboxuser@UbuntuServer:~$ systemctl is-active nginx
active
vboxuser@UbuntuServer:~$ _
```



## 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).**

```
#
# [Service]
# Type=forking
# PIDFile=/run/nginx.pid
# ExecStartPre=/usr/sbin/nginx -t
# ExecStart=/usr/sbin/nginx
# ExecReload=/usr/sbin/nginx -s reload
# ExecStop=-/sbin/systemctl stop nginx
# TimeoutStopSec=5
# KillMode=mixed
# Restart=always
#
```



**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.

```

Jun 18 06:35:30 UbuntuServer systemd[1181]: Queued start job for default target default.target.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Created slice app.slice - User Application Slice.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Started launchpadlib-cache-clean.timer - Clean up old files in the Launchpadlib cache.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Reached target paths.target - Paths.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Reached target timers.target - Timers.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Starting dbus.socket - D-Bus User Message Bus Socket...
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on dirmgr.socket - GnuPG network certificate management daemon.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on gpg-agent-browser.socket - GnuPG cryptographic agent and passphrase cache (access for web browsers).
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on gpg-agent-extra.socket - GnuPG cryptographic agent and passphrase cache (restricted).
Jun 18 06:35:30 UbuntuServer systemd[1181]: Starting gpg-agent-ssh.socket - GnuPG cryptographic agent (ssh-agent emulation)...
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on gpg-agent.socket - GnuPG cryptographic agent and passphrase cache.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on keyboxd.socket - GnuPG public key management service.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on pk-debconf-helper.socket - debconf communication socket.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on snapd.session-agent.socket - REST API socket for snapd user session agent.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on dbus.socket - D-Bus User Message Bus Socket.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Listening on gpg-agent-ssh.socket - GnuPG cryptographic agent (ssh-agent emulation).
Jun 18 06:35:30 UbuntuServer systemd[1181]: Reached target sockets.target - Sockets.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Reached target basic.target - Basic System.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Reached target default.target - Main User Target.
Jun 18 06:35:30 UbuntuServer systemd[1181]: Startup finished in 92ms.
Jun 18 06:36:36 UbuntuServer sudo[1241]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/bin/apt update
Jun 18 06:36:36 UbuntuServer sudo[1241]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:36:39 UbuntuServer sudo[1241]: pam_unix(sudo:session): session closed for user root
Jun 18 06:37:23 UbuntuServer sudo[1522]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/bin/apt upgrade -y
Jun 18 06:37:23 UbuntuServer sudo[1522]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:37:52 UbuntuServer sudo[1522]: pam_unix(sudo:session): session closed for user root
Jun 18 06:38:49 UbuntuServer sudo[14774]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/bin/nano /etc/netplan/50-cloud-init.yaml
Jun 18 06:38:49 UbuntuServer sudo[14774]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:40:39 UbuntuServer systemd[1181]: launchpadlib-cache-clean.service - Clean up old files in the Launchpadlib cache was skipped because of an unmet condition
Jun 18 06:41:29 UbuntuServer sudo[14774]: pam_unix(sudo:session): session closed for user root
Jun 18 06:41:42 UbuntuServer sudo[14782]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/sbin/netplan apply
Jun 18 06:41:42 UbuntuServer sudo[14782]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:41:42 UbuntuServer sudo[14782]: pam_unix(sudo:session): session closed for user root
Jun 18 06:42:25 UbuntuServer sudo[14787]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/bin/nano /etc/netplan/50-cloud-init.yaml
Jun 18 06:42:25 UbuntuServer sudo[14787]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:42:43 UbuntuServer sudo[14787]: pam_unix(sudo:session): session closed for user root
Jun 18 06:42:53 UbuntuServer sudo[14791]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/sbin/netplan apply
Jun 18 06:42:53 UbuntuServer sudo[14791]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:42:53 UbuntuServer sudo[14791]: pam_unix(sudo:session): session closed for user root
Jun 18 06:43:48 UbuntuServer sudo[14802]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/bin/nano /etc/netplan/50-cloud-init.yaml
Jun 18 06:43:48 UbuntuServer sudo[14802]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:45:58 UbuntuServer sudo[14802]: pam_unix(sudo:session): session closed for user root
Jun 18 06:46:06 UbuntuServer sudo[14810]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/sbin/netplan apply
Jun 18 06:46:06 UbuntuServer sudo[14810]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:46:06 UbuntuServer sudo[14810]: pam_unix(sudo:session): session closed for user root
Jun 18 06:46:51 UbuntuServer sudo[14817]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/bin/nano /etc/netplan/50-cloud-init.yaml
Jun 18 06:46:51 UbuntuServer sudo[14817]: pam_unix(sudo:session): session opened for user root(uid=0) by vboxuser(uid=1000)
Jun 18 06:47:13 UbuntuServer sudo[14817]: pam_unix(sudo:session): session closed for user root
Jun 18 06:47:24 UbuntuServer sudo[14821]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/sbin/netplan apply
lines 1-49

```

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

```

Jun 18 06:47:24 UbuntuServer sudo[14821]: vboxuser : TTY=ttty1 ; PWD=/home/vboxuser ; USER=root ; COMMAND=/usr/sbin/netplan apply
vboxuser@UbuntuServer:~$ journalctl -p 3 -xb
Hint: You are currently not seeing messages from other users and the system.
      Users in groups 'adm', 'systemd-journal' can see all messages.
      Pass -q to turn off this notice.
-- No entries --

```

✓ **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)