

Parte 1: Usuarios y permisos

1. ¿Cuál es tu **UID** y **GID**?

Mi **UID** es 1000 y mi **GID** es 1000 (si lo haces en modo root te pone que tu ID es 0), usamos los comandos `id -u` e `id -g`

2. ¿A qué grupos perteneces? ¿Qué permisos crees que tienes en cada grupo?

El usuario pertenece a los siguientes grupos →

```
alejandro-guerra@ldap:~$ id -Gn
alejandro-guerra adm cdrom sudo dip plugdev lxd
```

Cada grupo al que pertenece el usuario cuenta con los siguientes permisos:

El grupo adm → Es un grupo tradicional de administración, pero el grupo moderno y más común para otorgar permisos de super usuario.

El grupo cdrom → Es un grupo del sistema que tradicionalmente permitía a los usuarios acceder a dispositivos de CD/DVD

El grupo sudo → Es un grupo que permite otorgar permisos de super usuario a los miembros de ese grupo para ejecutar comandos administrativamente.

El grupo dip → Es un grupo del sistema que históricamente se usaba para permitir a usuarios sin privilegios administrar conexiones de acceso telefónico.

El grupo plugdev → Es un grupo especial para usuarios que necesitan acceder a dispositivos "enchufables" (como USBs, cámaras) sin permisos de root

El grupo lxd → Es un grupo que otorga acceso completo al demonio LXD, permitiendo a sus miembros crear, gestionar y destruir contenedores e instancias, interactuar con redes y almacenamiento, y ejecutar comandos `lxc` y `lxd` como si fueran administradores, pero con el sistema de permisos de LXD para control más granular por proyecto o identidad

3. Cambiar permisos de un archivo. Crea un archivo txt de prueba, muestra que permisos tiene por defecto, cambia los permisos para que solo pueda leer y escribir el propietario y muestra ahora los permisos que tiene.

```
-rw-r--r-- 1 root root 15 ene 12 16:14 prueba.txt
-rw-r--r-- 1 733 root 15 ene 12 16:14 prueba.txt
```

4. Explica cómo funcionan los permisos.

Los permisos funcionan de manera que definiendo estos últimos el usuario puede realizar ciertas acciones con ese fichero o incluso definir quien es el autor del mismo fichero.

Por ejemplo:

chmod 777 otorga permisos de lectura, escritura y ejecución a todos los grupos

chmod 775 otorga permisos de lectura, escritura y ejecución al propietario del archivo o directorio y al grupo asociado.

chmod 733 cambia el propietario del archivo de root al usuario que se está usando en ese momento.

Aunque también se pueden gestionar los permisos para que el usuario pueda realizar o no pueda realizar alguna característica en concreto.

5. Cambia el propietario y grupo de tu archivo de prueba

```
-rw----- 1 alejandro-guerra root 15 ene 12 16:14 prueba.txt
```

Parte 2: Servicios y demonios (Systemd)

1. ¿Qué servicios están activos?

```
alejandro-guerra@ldap:~$ systemctl list-units --type=service --state=running
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
cron.service                       loaded active running Regular background program processing daemon
dbus.service                       loaded active running D-Bus System Message Bus
fuupd.service                      loaded active running Firmware update daemon
getty@tty1.service                 loaded active running Getty on tty1
ModemManager.service              loaded active running Modem Manager
multipathd.service                loaded active running Device-Mapper Multipath Device Controller
polkit.service                    loaded active running Authorization Manager
rsyslog.service                   loaded active running System Logging Service
slapd.service                     loaded active running LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)
ssh.service                       loaded active running OpenBSD Secure Shell server
systemd-journald.service           loaded active running Journal Service
systemd-logind.service             loaded active running User Login Management
systemd-networkd.service           loaded active running Network Configuration
systemd-resolved.service           loaded active running Network Name Resolution
systemd-timesyncd.service          loaded active running Network Time Synchronization
systemd-udev.service              loaded active running Rule-based Manager for Device Events and Files
udisks2.service                   loaded active running Disk Manager
unattended-upgrades.service        loaded active running Unattended Upgrades Shutdown
upower.service                    loaded active running Daemon for power management
user@1000.service                  loaded active running User Manager for UID 1000

Legend: LOAD    → Reflects whether the unit definition was properly loaded.
          ACTIVE → The high-level unit activation state, i.e. generalization of SUB.
          SUB    → The low-level unit activation state, values depend on unit type.

20 loaded units listed.
```

2. ¿Cuál es el estado del servicio SSH?

```
alejandro-guerra@ldap:~$ systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Mon 2026-01-12 15:10:27 CET; 1h 21min ago
   TriggeredBy: ● ssh.socket
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 744 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 767 (sshd)
     Tasks: 1 (limit: 4602)
    Memory: 2.2M (peak: 2.4M)
       CPU: 294ms
   CGroup: /system.slice/ssh.service
           └─767 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

ene 12 15:10:26 ldap.AlejandroGuerra.local systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
ene 12 15:10:27 ldap.AlejandroGuerra.local sshd[767]: Server listening on 0.0.0.0 port 22.
ene 12 15:10:27 ldap.AlejandroGuerra.local sshd[767]: Server listening on :: port 22.
ene 12 15:10:27 ldap.AlejandroGuerra.local systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
alejandro-guerra@ldap:~$
```

3. Iniciar, detener y reiniciar un servicio

```

alejandro-guerra@ldap:~$ systemctl restart ssh
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'ssh.service'.
Authenticating as: Alejandro Guerra Abán (alejandro-guerra)
Password:
polkit-agent-helper-1: pam_authenticate failed: Authentication failure
==== AUTHENTICATION FAILED ====
Failed to restart ssh.service: Access denied
See system logs and 'systemctl status ssh.service' for details.
alejandro-guerra@ldap:~$ systemctl stop ssh
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to stop 'ssh.service'.
Authenticating as: Alejandro Guerra Abán (alejandro-guerra)
Password:
polkit-agent-helper-1: pam_authenticate failed: Authentication failure
==== AUTHENTICATION FAILED ====
Failed to stop ssh.service: Access denied
See system logs and 'systemctl status ssh.service' for details.
alejandro-guerra@ldap:~$ systemctl start ssh
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to start 'ssh.service'.
Authenticating as: Alejandro Guerra Abán (alejandro-guerra)
Password:
polkit-agent-helper-1: pam_authenticate failed: Authentication failure
==== AUTHENTICATION FAILED ====
Failed to start ssh.service: Access denied
See system logs and 'systemctl status ssh.service' for details.
alejandro-guerra@ldap:~$ _

```

4. ¿Cuál es la diferencia entre start y enable?

Start (iniciar) ejecuta un servicio o proceso inmediatamente, mientras que enable (habilitar) configura ese servicio para que se inicie automáticamente cada vez que el sistema se reinicia, sin iniciarlo en el momento actual.

5. Habilitar o deshabilitar inicio automático

```

alejandro-guerra@ldap:~$ sudo systemctl disable cron.service
Synchronizing state of cron.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install disable cron
Removed /etc/systemd/system/multi-user.target.wants/cron.service".
alejandro-guerra@ldap:~$ systemctl status cron.service
• cron.service - Regular background program processing daemon
  Loaded: loaded (/usr/lib/systemd/system/cron.service; disabled; preset: enabled)
  Active: active (running) since Mon 2026-01-12 15:10:26 CET; 1h 30min ago
  Docs: man:cron(8)
  Main PID: 739 (cron)
  Tasks: 1 (limit: 4602)
  Memory: 472.0K (peak: 2.2M)
  CPU: 287ms
  CGroup: /system.slice/cron.service
          └─739 /usr/sbin/cron -f -P

ene 12 16:15:01 ldap.AlejandroGuerra.local CRON[1354]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:15:01 ldap.AlejandroGuerra.local CRON[1354]: pam_unix(cron:session): session closed for user root
ene 12 16:17:01 ldap.AlejandroGuerra.local CRON[1407]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:17:01 ldap.AlejandroGuerra.local CRON[1408]: (root) CMD (cd / && run-parts --report /etc/cron.hourly)
ene 12 16:17:01 ldap.AlejandroGuerra.local CRON[1407]: pam_unix(cron:session): session closed for user root
ene 12 16:25:01 ldap.AlejandroGuerra.local CRON[1462]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:25:01 ldap.AlejandroGuerra.local CRON[1462]: pam_unix(cron:session): session closed for user root
ene 12 16:35:01 ldap.AlejandroGuerra.local CRON[1547]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:35:01 ldap.AlejandroGuerra.local CRON[1547]: (root) CMD (command -v debian-sa1 > /dev/null && debian-sa1 1 1)
ene 12 16:35:01 ldap.AlejandroGuerra.local CRON[1547]: pam_unix(cron:session): session closed for user root
alejandro-guerra@ldap:~$
alejandro-guerra@ldap:~$ sudo systemctl enable cron.service
Synchronizing state of cron.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable cron
Created symlink /etc/systemd/system/multi-user.target.wants/cron.service → /usr/lib/systemd/system/cron.service.
alejandro-guerra@ldap:~$ systemctl status cron.service
• cron.service - Regular background program processing daemon
  Loaded: loaded (/usr/lib/systemd/system/cron.service; enabled; preset: enabled)
  Active: active (running) since Mon 2026-01-12 15:10:26 CET; 1h 30min ago
  Docs: man:cron(8)
  Main PID: 739 (cron)
  Tasks: 1 (limit: 4602)
  Memory: 472.0K (peak: 2.2M)
  CPU: 287ms
  CGroup: /system.slice/cron.service
          └─739 /usr/sbin/cron -f -P

ene 12 16:15:01 ldap.AlejandroGuerra.local CRON[1354]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:15:01 ldap.AlejandroGuerra.local CRON[1354]: pam_unix(cron:session): session closed for user root
ene 12 16:17:01 ldap.AlejandroGuerra.local CRON[1407]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:17:01 ldap.AlejandroGuerra.local CRON[1408]: (root) CMD (cd / && run-parts --report /etc/cron.hourly)
ene 12 16:17:01 ldap.AlejandroGuerra.local CRON[1407]: pam_unix(cron:session): session closed for user root
ene 12 16:25:01 ldap.AlejandroGuerra.local CRON[1462]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:25:01 ldap.AlejandroGuerra.local CRON[1462]: pam_unix(cron:session): session closed for user root
ene 12 16:35:01 ldap.AlejandroGuerra.local CRON[1547]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
ene 12 16:35:01 ldap.AlejandroGuerra.local CRON[1547]: (root) CMD (command -v debian-sa1 > /dev/null && debian-sa1 1 1)
ene 12 16:35:01 ldap.AlejandroGuerra.local CRON[1547]: pam_unix(cron:session): session closed for user root

```

Parte 3: Logs del sistema

1. Explorar logs (journalctl), ¿Qué estás viendo exactamente?

Podemos observar registros del sistema

2. Mientras mantienes journalctl -f abierto, reinicia SSH o realiza algún cambio en un servicio.

```
alejandro-guerra@ldap:~$ systemctl restart ssh
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'ssh.service'.
Authenticating as: Alejandro Guerra Abán (alejandro-guerra)
Password:
polkit-agent-helper-1: pam_authenticate failed: Authentication failure
==== AUTHENTICATION FAILED ====
Failed to restart ssh.service: Access denied
See system logs and 'systemctl status ssh.service' for details.
alejandro-guerra@ldap:~$ journalctl -f
ene 12 16:42:19 ldap.AlejandroGuerra.local systemd[1]: Finished apt-daily.service - Daily apt download activities.
ene 12 16:42:19 ldap.AlejandroGuerra.local systemd[1]: apt-daily.service: Consumed 56.633s CPU time.
ene 12 16:45:00 ldap.AlejandroGuerra.local polkitd[690]: Registered Authentication Agent for unix-process:2370:568853 (system
t --notify-fd 5 --fallback), object path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale es_ES.UTF-8)
ene 12 16:45:00 ldap.AlejandroGuerra.local polkitd[690]: Identity 'unix-group:admin' is not valid, ignoring: No UNIX group wit
ene 12 16:45:00 ldap.AlejandroGuerra.local polkit-agent-helper-1[2380]: pam_unix(polkit-1:auth): authentication failure; logna
0 tty= ruser=alejandro-guerra rhost= user=alejandro-guerra
ene 12 16:45:01 ldap.AlejandroGuerra.local CRON[2381]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid
ene 12 16:45:01 ldap.AlejandroGuerra.local CRON[2382]: (root) CMD (command -v debian-sa1 > /dev/null && debian-sa1 1 1)
ene 12 16:45:01 ldap.AlejandroGuerra.local CRON[2381]: pam_unix(cron:session): session closed for user root
ene 12 16:45:03 ldap.AlejandroGuerra.local polkitd[690]: Operator of unix-process:2370:568853 FAILED to authenticate to gain a
sktop.systemd1.manage-units for system-bus-name::1.49 [systemctl restart ssh] (owned by unix-user:alejandro-guerra)
ene 12 16:45:03 ldap.AlejandroGuerra.local polkitd[690]: Unregistered Authentication Agent for unix-process:2370:568853 (syste
/freedesktop/PolicyKit1/AuthenticationAgent, locale es_ES.UTF-8) (disconnected from bus)
ene 12 16:45:04 ldap.AlejandroGuerra.local kernel: workqueue: drain_vmap_area_work hogged CPU for >100000us 8 times, consider s
```

3. Observa qué mensajes aparecen y explica qué indican.

Indican en su mayoría que el usuario a reiniciado un recurso pero no ha proporcionado una credenciales correcta.

Parte 4: Monitorización de recursos

1. Lanza top

```
top - 16:47:20 up 1:37, 1 user, load average: 0,00, 0,09, 0,06
Tasks: 116 total, 1 running, 115 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0,0 us, 0,1 sy, 0,0 ni, 94,7 id, 0,3 wa, 0,0 hi, 4,9 si, 0,0 st
MiB Mem : 3915,6 total, 2835,6 free, 425,8 used, 878,3 buff/cache
MiB Swap: 0,0 total, 0,0 free, 0,0 used. 3489,8 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
49	root	20	0	0	0	0	I	0,3	0,0	0:12.53	kworker/1:1-events
252	root	20	0	0	0	0	S	0,3	0,0	0:00.30	jbd2/dm-0-8
731	root	0	-20	0	0	0	I	0,3	0,0	0:00.61	kworker/0:2H-kblockd
1609	root	20	0	0	0	0	I	0,3	0,0	0:00.92	kworker/2:1-mm_percpu_wq
2389	alejand+	20	0	11944	5888	3712	R	0,3	0,1	0:00.02	top
1	root	20	0	22600	13552	9456	S	0,0	0,3	0:07.06	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.05	kthreadd
3	root	20	0	0	0	0	S	0,0	0,0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-rcu_g
5	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-rcu_p
6	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-slab_
7	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-netns
10	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/0:0H-events_highpri
12	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-mm_pe
13	root	20	0	0	0	0	I	0,0	0,0	0:00.00	rcu_tasks_kthread
14	root	20	0	0	0	0	I	0,0	0,0	0:00.00	rcu_tasks_rude_kthread
15	root	20	0	0	0	0	I	0,0	0,0	0:00.00	rcu_tasks_trace_kthread
16	root	20	0	0	0	0	S	0,0	0,0	0:00.27	ksoftirqd/0
17	root	20	0	0	0	0	I	0,0	0,0	0:01.14	rcu_preempt
18	root	rt	0	0	0	0	S	0,0	0,0	0:00.31	migration/0
19	root	-51	0	0	0	0	S	0,0	0,0	0:00.00	idle_inject/0
20	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/0
21	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/1
22	root	-51	0	0	0	0	S	0,0	0,0	0:00.00	idle_inject/1
23	root	rt	0	0	0	0	S	0,0	0,0	0:01.11	migration/1
24	root	20	0	0	0	0	S	0,0	0,0	0:01.43	ksoftirqd/1
26	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/1:0H-kblockd
27	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/2
28	root	-51	0	0	0	0	S	0,0	0,0	0:00.00	idle_inject/2
29	root	rt	0	0	0	0	S	0,0	0,0	0:00.98	migration/2
30	root	20	0	0	0	0	S	0,0	0,0	0:00.52	ksoftirqd/2
33	root	20	0	0	0	0	S	0,0	0,0	0:00.03	kdevtmpfs
34	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-jnet

2. Observa PID, %CPU, %MEM y usuario de los procesos. ¿Qué proceso consume más CPU?

El proceso que más CPU me consume sería el mismo htop con un 0,7%

3. ¿Qué proceso consume más memoria?

El proceso que más memoria me consume sería el mismo htop con un 0,7%

4. Filtra por usuario: top -u tu_usuario.

```
top - 16:51:52 up 1:41, 1 user, load average: 0,09, 0,07, 0,06
Tasks: 115 total, 2 running, 113 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0,0 us, 0,1 sy, 0,0 ni, 95,4 id, 0,0 wa, 0,0 hi, 4,5 si, 0,0 st
MiB Mem : 3915,6 total, 2835,8 free, 424,4 used, 879,9 buff/cache
MiB Swap: 0,0 total, 0,0 free, 0,0 used. 3491,3 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2408	alejand+	20	0	11952	5888	3712	R	0,3	0,1	0:00.08	top
989	alejand+	20	0	20132	11264	9344	S	0,0	0,3	0:00.17	systemd
990	alejand+	20	0	21152	3548	1920	S	0,0	0,1	0:00.00	(sd-pam)
998	alejand+	20	0	8920	5504	3840	S	0,0	0,1	0:01.09	bash

5. Finaliza un proceso.

Usamos el comando kill para finalizar un proceso

```
top - 16:52:49 up 1:42, 1 user, load average: 0,03, 0,06, 0,05
Tasks: 114 total, 1 running, 113 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0,0 us, 0,2 sy, 0,0 ni, 95,0 id, 0,0 wa, 0,0 hi, 4,7 si, 0,0 st
MiB Mem : 3915,6 total, 2835,7 free, 424,4 used, 880,0 buff/cache
MiB Swap: 0,0 total, 0,0 free, 0,0 used. 3491,3 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2415	alejand+	20	0	11952	5888	3712	R	0,3	0,1	0:00.04	top
998	alejand+	20	0	8920	5504	3840	S	0,0	0,1	0:01.12	bash

6. Htop ¿Qué ventajas encuentras frente a top?

htop ofrece ventajas significativas sobre top gracias a su **interfaz a todo color, soporte de ratón y gestión de procesos interactiva**, permitiendo desplazamiento vertical/horizontal, selección de procesos, y terminación/priorización con teclas de función (F1-F10) sin salir de la pantalla, facilitando la monitorización y administración de recursos de forma más intuitiva y visual que la versión básica y minimalista de top

7. ¿Qué procesos podrías finalizar y por qué?

Podemos finalizar procesos que consuman muchos recursos (CPU/Memoria), aplicaciones congeladas (que no responden) o tareas redundantes para liberar recursos y optimizar el rendimiento, usando comandos como kill, pkill, o killall con el PID o nombre del proceso, o mediante el monitor del sistema gráfico si lo tienes. Se finalizan procesos para solucionar cuelgues, depurar, o mantener el servidor ágil.