



# S.I.G.T.

Sistemas Operativos III

APPTec

Rol	Apellido	Nombre	C.I	Email	Tel/Cel.
Coordinador	Pereyra	Emiliano	4.774.396-6	epereyra@apptecuy.com	092324130
Sub-Coordinador	Varela	Michael	4.543.461-8	mvarela@apptecuy.com	099297255
Integrante 1	González	Mauro	5.251.060-7	mgonzalez@apptecuy.com	094866094
Integrante 2	Otero	Gonzalo	5.014.881-8	gotero@apptecuy.com	094762305

**Docente: Rodríguez, Carlos**

**Fecha de culminación**

**13/11/2023**

**TERCERA ENTREGA**

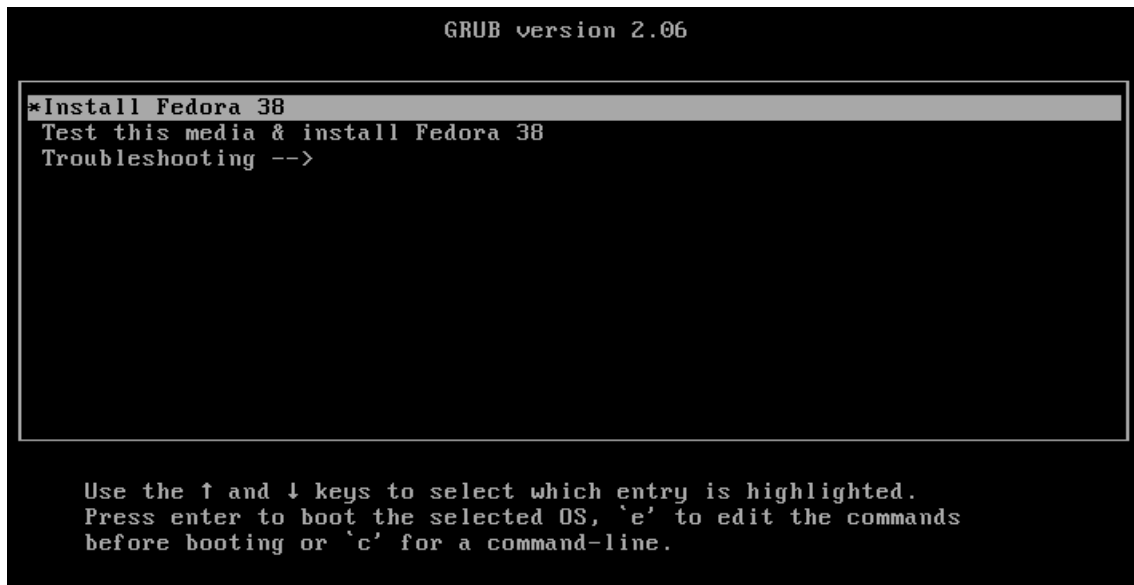
**I.S.B.O.**

**3BH**

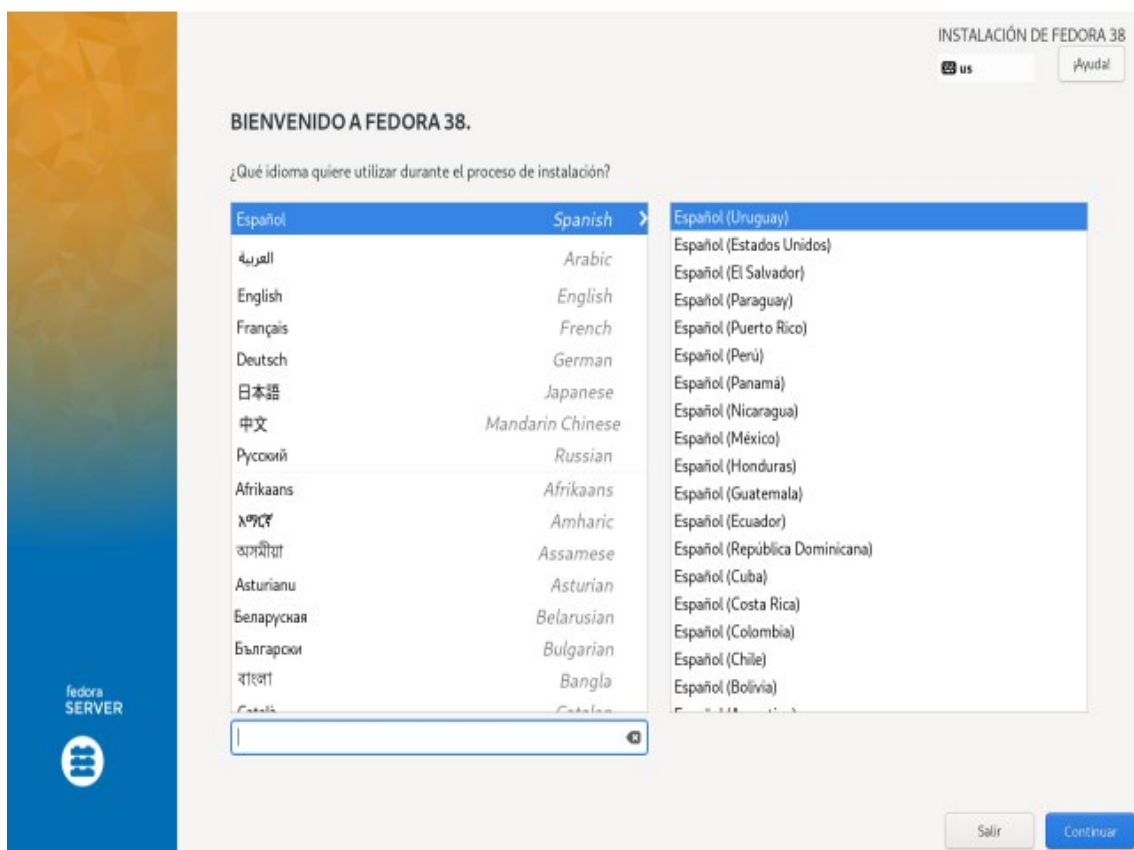
## Índice

Instalación del servidor.....	3
Instalación de docker.....	7
Instalación de servidor LAMP.....	8
Instalación del servidor web Apache.....	8
Instalación MARIADB.....	10
Instalación PHP.....	11
Script que permita manejar los logs del sistema operativo.....	12
Logs de auditoría creados por el equipo de trabajo.....	13
Estudio de los diferentes roles de los usuarios del servidor.....	15
Usuarios necesarios en el sistema operativo creados de acuerdo al estudio de roles.....	15
Administrador.....	15
Administrador de base de datos.....	16
Operador.....	17
Menú operador.....	17
Menú de alta usuario.....	18
Alta.....	18
Baja.....	18
Modificación.....	19
Menú de alta grupo.....	19
Alta grupo.....	19
Baja grupo.....	20
Modificación grupo.....	20
Servicios.....	20
RespalDOS.....	21
RED.....	21
Firewall.....	21
Logs.....	22
Configuración de red en las terminales y servidor.....	22
Configuración del servicio SSH.....	22
Archivos crontab con rutinas de backup.....	24

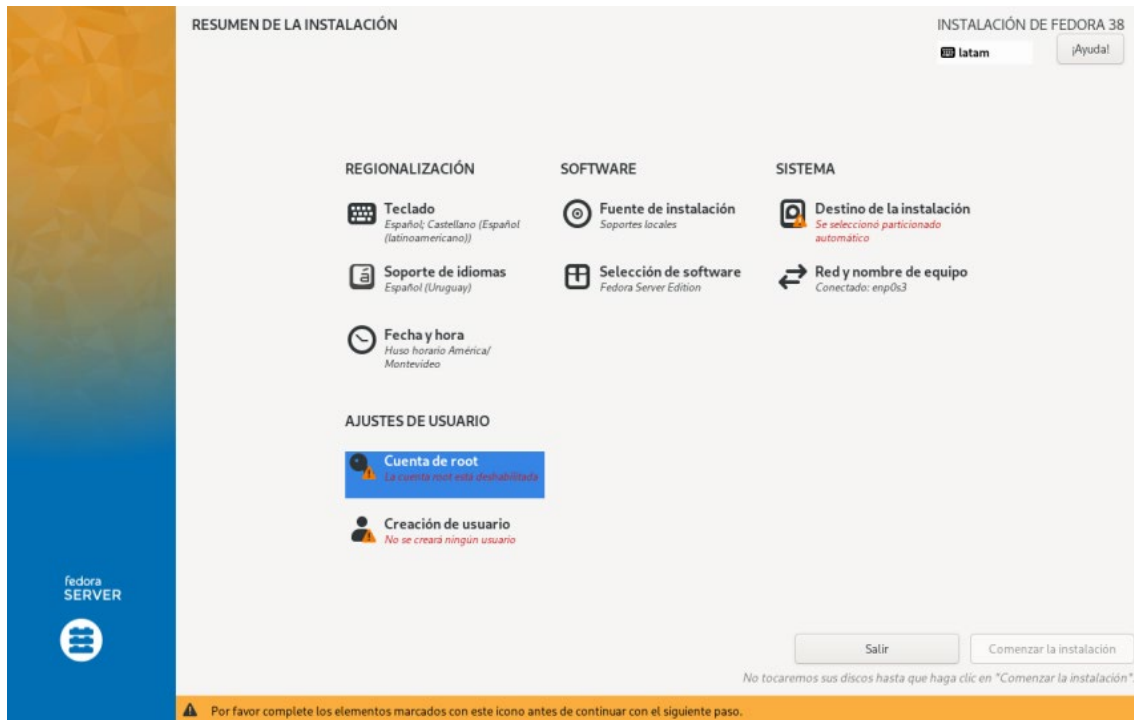
## Instalación del servidor.



Aquí elegimos el idioma deseado, en este caso elegiremos español.



En este paso, deberemos elegir el disco destino de la instalación, elegir si activar la cuenta root o no y decidir si ya crear un usuario.



Aquí elegimos el disco donde se instalara.



Activación de la cuenta de root.

**CUENTA DE ROOT** INSTALACIÓN DE FEDORA 38

Hecho latam ¡Ayuda!

La cuenta root se utiliza para administrar el sistema.

El usuario root (también conocido como superusuario) tiene acceso completo a todo el sistema. Por esta razón, es mejor iniciar sesión en este sistema como usuario root sólo para realizar el mantenimiento o la administración del sistema.

☐ Desactivar la cuenta de root

Desactivar la cuenta de root bloqueará la cuenta y desactivará el acceso remoto con la cuenta de root. Esto evitará el acceso administrativo involuntario al sistema.

☒ Activar la cuenta de root

Habilitar la cuenta de root le permitirá establecer una contraseña de root y, opcionalmente, habilitar el acceso remoto a la cuenta de root en este sistema.

Contraseña administrativa:  Longitud insuficiente

Confirmar:

☐ Permitir el acceso SSH de root con contraseña

Creación del usuario que tendrá privilegios para administrar el sistema.

**CREAR USUARIO** INSTALACIÓN DE FEDORA 38

Hecho latam ¡Ayuda!

Nombre completo:

Nombre de usuario:

☒ Añadir privilegios administrativos a esta cuenta de usuario (membresía al grupo wheel)

☒ Se requiere una contraseña para usar esta cuenta

Contraseña:  Débil

Confirmar la contraseña:

Una vez que queden prontos estos pasos le damos a “Comenzar la instalación”.

**RESUMEN DE LA INSTALACIÓN** INSTALACIÓN DE FEDORA 38

latam ¡Ayuda!

**REGIONALIZACIÓN**

- Teclado**  
Español; Castellano (Español latinoamericano)
- Soporte de idiomas**  
Español (Uruguay)
- Fecha y hora**  
Huso horario América/ Montevideo

**SOFTWARE**

- Fuente de instalación**  
Soportes locales
- Selección de software**  
Fedora Server Edition

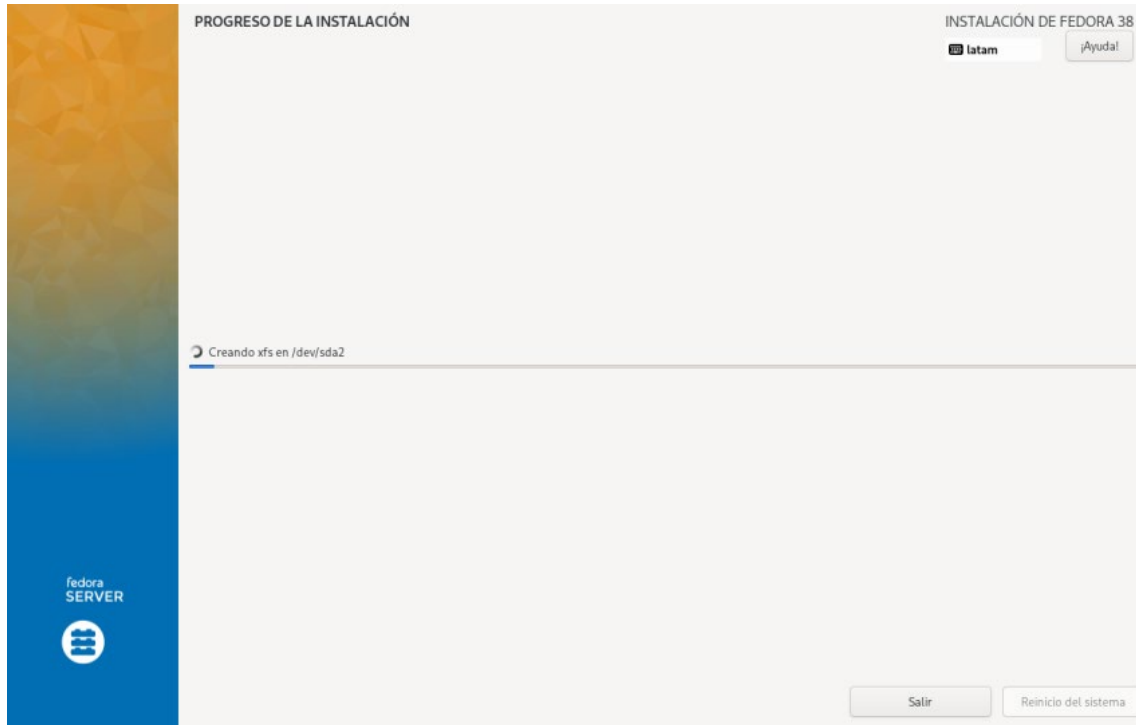
**SISTEMA**

- Destino de la instalación**  
Se seleccionó particionado automático
- Red y nombre de equipo**  
Conectado: enp0s3

**AJUSTES DE USUARIO**

- Cuenta de root**  
Contraseña de root establecida
- Creación de usuario**  
Se creará el usuario administrador adminSistema

No tocaremos sus discos hasta que haga clic en "Comenzar la instalación".



Una vez que termina le damos a reiniciar.



Lo siguiente es opcional, se agregó un entorno grafico de GNOME solo para que sea más “amigable” con la persona que está encargada de manejar el servidor.

Instalar el entorno gráfico GNOME:

```
sudo dnf install @gnome-desktop
sudo systemctl set-default graphical.target
```

## Instalación de docker.

sudo dnf install docker

```
[AdministradorSistema@192 ~]$ sudo dnf install docker
Última comprobación de caducidad de metadatos hecha hace 3:04:22, el sáb 11 nov 2023 20:17:02.
Dependencias resueltas.
=====
Paquete                Arquitectura  Versión                Repositorio            Tam.
-----
Instalando:
moby-engine             x86_64        24.0.5-1.fc38          updates                 28 M
Instalando dependencias:
pigz                    x86_64        2.7-3.fc38             fedora                   83 k
=====
Resumen de la transacción
=====
Instalar 2 Paquetes

Tamaño total de la descarga: 28 M
Tamaño instalado: 109 M
¿Está de acuerdo [s/N]?: s
Descargando paquetes:
(1/2): pigz-2.7-3.fc38.x86_64.rpm                                427 kB/s | 83 kB  00:00
(2/2): moby-engine-24.0.5-1.fc38.x86_64.rpm                     19 MB/s | 28 MB  00:01
Total                                                            6.5 MB/s | 28 MB  00:04
```

```
=====
Instalando:
moby-engine             x86_64        24.0.5-1.fc38          updates                 28 M
Instalando dependencias:
pigz                    x86_64        2.7-3.fc38             fedora                   83 k
=====
Resumen de la transacción
=====
Instalar 2 Paquetes

Tamaño total de la descarga: 28 M
Tamaño instalado: 109 M
¿Está de acuerdo [s/N]?: s
Descargando paquetes:
(1/2): pigz-2.7-3.fc38.x86_64.rpm                                427 kB/s | 83 kB  00:00
(2/2): moby-engine-24.0.5-1.fc38.x86_64.rpm                     19 MB/s | 28 MB  00:01
Total                                                            6.5 MB/s | 28 MB  00:04
=====
Ejecutando verificación de operación
Verificación de operación exitosa.
Ejecutando prueba de operaciones
Prueba de operación exitosa.
Ejecutando operación
Preparando : pigz-2.7-3.fc38.x86_64 1/1
Instalando : pigz-2.7-3.fc38.x86_64 1/2
Ejecutando scriptlet: moby-engine-24.0.5-1.fc38.x86_64 2/2
Instalando : moby-engine-24.0.5-1.fc38.x86_64 2/2
Ejecutando scriptlet: moby-engine-24.0.5-1.fc38.x86_64 2/2
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket + /usr/lib/systemd/system/docker.socket.
Verificando : pigz-2.7-3.fc38.x86_64 1/2
Verificando : moby-engine-24.0.5-1.fc38.x86_64 2/2
Instalado:
moby-engine-24.0.5-1.fc38.x86_64 pigz-2.7-3.fc38.x86_64
¡Listo!
```

Instalar el dnf-plugins-corepaquete (que proporciona los comandos para administrar sus repositorios DNF) y configure el repositorio.

Comandos: sudo dnf -y install dnf-plugins-core  
 sudo dnf config-manager --add-repo  
<https://download.docker.com/linux/fedora/docker-ce.repo>

```
[AdministradorSistema@192 ~]$ sudo dnf config-manager --add-repo https://download.docker.com/linux/fedora/docker-ce.repo
Agregando repositorio de: https://download.docker.com/linux/fedora/docker-ce.repo
[AdministradorSistema@192 ~]$
```

Instalar Docker Engine, containerd y Docker Compose:

```
sudo dnf install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin
```

Iniciar Docker: `sudo systemctl start docker`  
`sudo systemctl enable docker`

```
[AdministradorSistema@192 ~]$ sudo systemctl start docker
[AdministradorSistema@192 ~]$ sudo systemctl enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[AdministradorSistema@192 ~]$
```

Verificar que la instalación de Docker Engine se haya realizado correctamente ejecutando el siguiente comando.

Comando: `sudo docker run hello-world`

```
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
```

## Instalación de servidor LAMP.

Instalación del servidor web Apache.

Comando: `sudo dnf -y install httpd`

```
[AdministradorSistema@192 ~]$ sudo dnf install httpd
Última comprobación de caducidad de metadatos hecha hace 0:02:53, el sáb 11 nov 2023 23:45:15.
Dependencias resueltas.
=====
Paquete                Arquitectura  Versión      Repositorio  Tam.
-----
Instalando:
httpd                  x86_64       2.4.58-1.fc38 updates      50 k
Instalando dependencias:
apr                    x86_64       1.7.2-2.fc38 fedora       127 k
apr-util              x86_64       1.6.3-2.fc38 fedora       96 k
fedora-logos-httpd    x86_64       38.1.0-1.fc38 fedora       16 k
httpd-core             x86_64       2.4.58-1.fc38 updates     1.4 M
httpd-filesystem       noarch       2.4.58-1.fc38 updates      12 k
httpd-tools            x86_64       2.4.58-1.fc38 updates      80 k
Instalando dependencias débiles:
apr-util-bdb          x86_64       1.6.3-2.fc38 fedora       13 k
apr-util-openssl       x86_64       1.6.3-2.fc38 fedora       15 k
julietaula-montserrat-fonts noarch      1:7.222-4.fc38 fedora       1.6 M
mod_http2              x86_64       2.0.25-1.fc38 updates     161 k
mod_lua                x86_64       2.4.58-1.fc38 updates      58 k
Resumen de la transacción
=====
Instalar 12 Paquetes

Tamaño total de la descarga: 3.7 M
Tamaño instalado: 10 M
¿Está de acuerdo [s/N]?
```

Paquete	Arquitectura	Versión	Repositorio	Tam.
apr	x86_64	1.7.2-2.fc38	fedora	127 k
apr-util	x86_64	1.6.3-2.fc38	fedora	96 k
fedora-logos-httpd	x86_64	38.1.0-1.fc38	fedora	16 k
httpd-core	x86_64	2.4.58-1.fc38	updates	1.4 M
httpd-filesystem	noarch	2.4.58-1.fc38	updates	12 k
httpd-tools	x86_64	2.4.58-1.fc38	updates	80 k
apr-util-bdb	x86_64	1.6.3-2.fc38	fedora	13 k
apr-util-openssl	x86_64	1.6.3-2.fc38	fedora	15 k
julietaula-montserrat-fonts	noarch	1:7.222-4.fc38	fedora	1.6 M
mod_http2	x86_64	2.0.25-1.fc38	updates	161 k
mod_lua	x86_64	2.4.58-1.fc38	updates	58 k

```

Instalado:
apr-1.7.2-2.fc38.x86_64          apr-util-1.6.3-2.fc38.x86_64          apr-util-bdb-1.6.3-2.fc38.x86_64
apr-util-openssl-1.6.3-2.fc38.x86_64  fedora-logos-httpd-38.1.0-1.fc38.noarch  httpd-2.4.58-1.fc38.x86_64
httpd-core-2.4.58-1.fc38.x86_64  httpd-filesystem-2.4.58-1.fc38.noarch  httpd-tools-2.4.58-1.fc38.x86_64
julietaula-montserrat-fonts-1:7.222-4.fc38.noarch  mod_http2-2.0.25-1.fc38.x86_64  mod_lua-2.4.58-1.fc38.x86_64

¡Listo!
```

Comandos básicos para administrar el servidor:

Iniciar:



sudo systemctl start httpd

Parar:

sudo systemctl stop httpd

Habilitar:

sudo systemctl enable httpd

Reiniciar:

sudo systemctl restart httpd

Status:

sudo systemctl status httpd

Recargar:

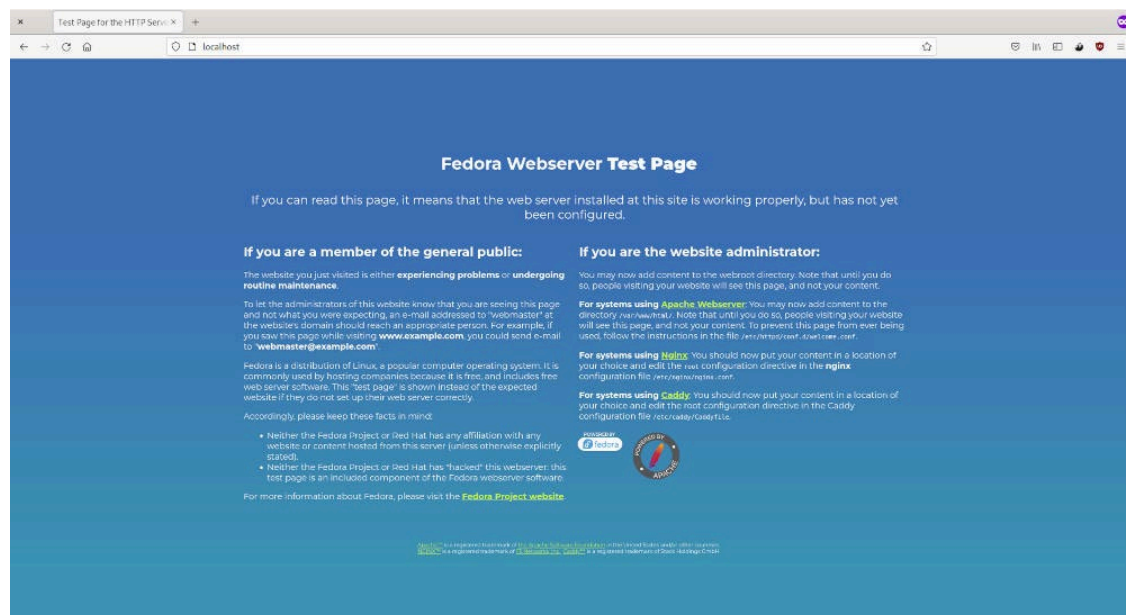
sudo systemctl reload httpd

Una vez instalado, ejecutamos los comandos de habilitar, iniciar y status:

```
[AdministradorSistema@192 ~]$ sudo systemctl start httpd
[AdministradorSistema@192 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service + /usr/lib/systemd/system/httpd.service.
[AdministradorSistema@192 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
           └─10-timeout-abort.conf
   Active: active (running) since Sat 2023-11-11 23:50:11 -03; 1min 16s ago
     Docs: man:httpd.service(8)
   Main PID: 4158 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
     Tasks: 177 (limit: 6873)
    Memory: 14.1M
       CPU: 276ms
   CGroup: /system.slice/httpd.service
           └─4158 /usr/sbin/httpd -DFOREGROUND
             4159 /usr/sbin/httpd -DFOREGROUND
             4160 /usr/sbin/httpd -DFOREGROUND
             4161 /usr/sbin/httpd -DFOREGROUND
             4162 /usr/sbin/httpd -DFOREGROUND

nov 11 23:50:11 192.168.1.11 systemd[1]: Starting httpd.service - The Apache HTTP Server...
nov 11 23:50:11 192.168.1.11 httpd[4158]: Server configured, listening on: port 80
nov 11 23:50:11 192.168.1.11 systemd[1]: Started httpd.service - The Apache HTTP Server.
[AdministradorSistema@192 ~]$
```

Nos dirigimos a un navegador web y entramos en <http://localhost>



## Instalación MARIADB.

Para empezar, ejecutamos: `sudo dnf install mariadb-server`

```
[AdministradorSistema@192 ~]$ sudo dnf install mariadb-server
Última comprobación de caducidad de metadatos hecha hace 0:08:18, el sáb 11 nov 2023 23:45:15.
Dependencias resueltas.
=====
Paquete                               Arquitectura Versión                               Repositorio                               Tam.
-----
Instalando:
mariadb-server                        x86_64      3:10.5.22-1.fc38                       updates                                   11 M
Instalando dependencias:
mariadb                               x86_64      3:10.5.22-1.fc38                       updates                                   1.6 M
mariadb-common                        x86_64      3:10.5.22-1.fc38                       updates                                   33 k
mariadb-connector-c                    x86_64      3.3.5-1.fc38                           updates                                   208 k
mariadb-connector-c-config             noarch      3.3.5-1.fc38                           updates                                   8.6 k
mariadb-errmsg                         x86_64      3:10.5.22-1.fc38                       updates                                   217 k
mysql-selinux                         noarch      1.0.7-2.fc38                           updates                                   35 k
perl-DBD-MariaDB                      x86_64      1.22-4.fc38                            updates                                   151 k
perl-DBI                              x86_64      1.642-15.fc38                          updates                                   709 k
perl-File-Copy                         noarch      2.39-497.fc38                          updates                                   21 k
perl-Math-BigInt                      noarch      1:1.9998.39-1.fc38                     updates                                   203 k
perl-Math-BigRat                      noarch      0.2624-3.fc38                          updates                                   41 k
perl-Math-Complex                     noarch      1.59-497.fc38                          updates                                   48 k
perl-Sys-Hostname                     x86_64      1.24-497.fc38                          updates                                   18 k
Instalando dependencias débiles:
mariadb-backup                        x86_64      3:10.5.22-1.fc38                       updates                                   6.5 M
mariadb-cracklib-password-check        x86_64      3:10.5.22-1.fc38                       updates                                   15 k
mariadb-gssapi-server                 x86_64      3:10.5.22-1.fc38                       updates                                   16 k
mariadb-server-utils                  x86_64      3:10.5.22-1.fc38                       updates                                   216 k
Resumen de la transacción
=====
Instalar 18 Paquetes
Tamaño total de la descarga: 21 M
Tamaño instalado: 118 M
¿Está de acuerdo [s/N]?:
```

```
Instalado:
mariadb-3:10.5.22-1.fc38.x86_64      mariadb-backup-3:10.5.22-1.fc38.x86_64      mariadb-common-3:10.5.22-1.fc38.x86_64
mariadb-connector-c-3.3.5-1.fc38.x86_64  mariadb-connector-c-config-3.3.5-1.fc38.noarch  mariadb-cracklib-password-check-3:10.5.22-1.fc38.x86_64
mariadb-errmsg-3:10.5.22-1.fc38.x86_64  mariadb-gssapi-server-3:10.5.22-1.fc38.x86_64  mariadb-server-3:10.5.22-1.fc38.x86_64
mariadb-server-utils-3:10.5.22-1.fc38.x86_64  mysql-selinux-1.0.7-2.fc38.noarch  perl-DBD-MariaDB-1.22-4.fc38.x86_64
perl-DBI-1.643-15.fc38.x86_64      perl-File-Copy-2.39-497.fc38.noarch  perl-Math-BigInt-1:1.9998.39-1.fc38.noarch
perl-Math-BigRat-0.2624-3.fc38.noarch  perl-Math-Complex-1.59-497.fc38.noarch  perl-Sys-Hostname-1.24-497.fc38.x86_64
¡Listo!
[AdministradorSistema@192 ~]$
```

Los comandos para gestionar el servicio son muy similares a Apache.

Habilitar:

`sudo systemctl enable mariadb`

Iniciar:

`sudo systemctl start mariadb`

Parar:

`sudo systemctl stop mariadb`

Status:

`sudo systemctl status mariadb`

Reiniciar:

`sudo systemctl restart mariadb`

Como con Apache, ejecutamos los comandos habilitar, iniciar y status.

```
[AdministradorSistema@192 ~]$ sudo systemctl start mariadb
[AdministradorSistema@192 ~]$ sudo systemctl enable mariadb
Created symlink /etc/systemd/system/mysql.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/mysqld.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /usr/lib/systemd/system/mariadb.service.
[AdministradorSistema@192 ~]$ sudo systemctl status mariadb
● mariadb.service - MariaDB 10.5 database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
            └─10-timeout-abort.conf
   Active: active (running) since Sat 2023-11-11 23:57:49 -03; 2min 18s ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
  Main PID: 5270 (mariadb)
    Status: "Taking your SQL requests now..."
     Tasks: 9 (Limit: 6553)
    Memory: 66.2M
       CPU: 1.796s
   CGroup: /system.slice/mariadb.service
            └─5270 /usr/libexec/mariadb --basedir=/usr

nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: The second is mysql@localhost, it has no password either, but
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: you need to be the system 'mysql' user to connect.
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: After connecting you can set the password, if you would need to be
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: able to connect as any of these users with a password and without sudo
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: See the MariaDB Knowledgebase at https://mariadb.com/kb
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: Please report any problems at https://mariadb.org/jira
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: The latest information about MariaDB is available at https://mariadb.org/.
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: Consider joining MariaDB's strong and vibrant community:
nov 11 23:57:48 192.168.1.11 mariadb-prepare-db-dir[5223]: https://mariadb.org/get-involved/
nov 11 23:57:49 192.168.1.11 systemd[1]: Started mariadb.service - MariaDB 10.5 database server.
[AdministradorSistema@192 ~]$
```

## Seguridad MySQL

Por defecto, el gestor de bases de datos viene con el usuario root sin contraseña. Vamos a asignar una contraseña ejecutando: `sudo mysql_secure_installation`

```
[AdministradorSistema@192 ~]$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none): 
Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
[AdministradorSistema@192 ~]$
```

Nota: Nos hará una serie de preguntas sobre si queremos eliminar usuarios anónimos, borrar las tablas de test, etc. Podemos contestar afirmativamente a todo para poder finalizar

## Instalación PHP.

Vamos a instalar los siguientes paquetes para PHP:

```
sudo dnf -y install php php-cli php-php-gettext php-mbstring php-mcrypt php-mysqlnd php-pear php-curl php-gd php-xml php-bcmath php-zip
```



```
#!/bin/bash
clear
while true; do
    echo "
    |-----|
    |               Gestión de Logs               |
    |-----|
    | 1) Mostrar el contenido del log de mensajes |
    | 2) Mostrar el contenido del log de kernel  |
    | 3) Buscar eventos en los logs              |
    | 4) Salir                                   |
    |-----|
    "

    read -p "Ingrese una opción: " opcion

    case "$opcion" in
        1)
            echo -e "\nContenido del log de mensajes:"
            cat /var/log/messages
            ;;
        2)
            echo -e "\nContenido del log de kernel:"
            cat /var/log/kern.log
            ;;
        3)
            read -p "Ingrese el término de búsqueda: " termino_búsqueda
            echo -e "\nResultados de la búsqueda en los logs:"
            grep -i "$termino_búsqueda" /var/log/*
            ;;
        4)
            echo
            exit 0
            ;;
        *)
            echo "Opción no válida. Por favor, ingrese una opción válida."
            sleep 1
            ;;
    esac

    read -rsnl -p "Presiona una tecla para continuar..."
    clear
done
```

## Logs de auditoría creados por el equipo de trabajo.

Primero instalaremos rsyslog y auditd (En este caso ya están instalados), en caso de que no ejecutar el comando `sudo dnf install rsyslog audit` y aceptar cuando lo pida.

```
[root@192 ~]# sudo dnf install rsyslog audit
Docker CE Stable - x86_64
El paquete rsyslog-8.2310.0-1.fc38.x86_64 ya está instalado.
El paquete audit-3.1.2-1.fc38.x86_64 ya está instalado.
Dependencias resueltas.
Nada por hacer.
¡Listo!
[root@192 ~]#
```

Configurar rsyslog para enviar logs de auditoría, para eso hay que editar el archivo de configuración de rsyslog en este caso usando nano.

`sudo nano /etc/rsyslog.conf`

Agregamos la siguiente línea al final: `audit.* /var/log/audit.log`

Guardamos con `ctrl + o` y salimos con `ctrl + x`

```
GNU nano 7.2 /etc/rsyslog.conf
# The authpriv file has restricted access.
authpriv.* /var/log/secure

# Log all the mail messages in one place.
mail.* -/var/log/maillog

# Log cron stuff
cron.* /var/log/cron

# Everybody gets emergency messages
*.emerg :omusrmsg:*

# Save news errors of level crit and higher in a special file.
uucp,news.crit /var/log/spooler

# Save boot messages also to boot.log
local7.* /var/log/boot.log

### sample forwarding rule ###
#action(type="omfwd"
# # An on-disk queue is created for this action. If the remote host is
# # down, messages are spooled to disk and sent when it is up again.
#queue.filename="fwdRule1" # unique name prefix for spool files
#queue.maxdiskspace="1g" # 1gb space limit (use as much as possible)
#queue.saveonshutdown="on" # save messages to disk on shutdown
#queue.type="LinkedList" # run asynchronously
#action.resumeRetryCount="-1" # infinite retries if host is down
# # Remote Logging (we use TCP for reliable delivery)
# # remote_host is: name/ip, e.g. 192.168.0.1, port optional e.g. 10514
#Target="remote_host" Port="XXX" Protocol="tcp")
audit.* /var/log/audit.log
```

Ejecutamos `sudo systemctl restart rsyslog` para reiniciar el servicio

Configurar `auditd`, iniciar y habilitar el servicio de `auditd`:

`sudo systemctl start auditd`

`sudo systemctl enable auditd`

Para revisar los logs podemos usar: `cat /var/log/audit/audit.log`

## Estudio de los diferentes roles de los usuarios del servidor.

En el servidor tendremos 3 usuarios.

El usuario Administrador, adminbd y el operador.

### **Administrador:**

Encargado general del sistema y base de datos.

Configura, optimiza y asegura el sistema.

Gestiona usuarios, permisos y realiza tareas críticas.

### **AdminBD (Administrador de Base de Datos):**

Enfocado en la administración exclusiva de la base de datos.

Configura, optimiza y asegura la base de datos.

Gestiona usuarios y permisos a nivel de base de datos.

### **Operador:**

Realiza tareas operativas diarias y atiende solicitudes de usuarios.

Colabora en el monitoreo y mantiene la operatividad del sistema.

No realiza cambios significativos en la configuración del sistema o la base de datos.

## Usuarios necesarios en el sistema operativo creados de acuerdo al estudio de roles.

### Administrador.

```
Alta de Usuario
1) Crear usuario
2) Volver

Ingrese una opción: 1
Ingrese un usuario: administrador
Ingrese grupo donde alojarlo: admin
Ingrese contraseña: admin
Creando Usuario
```



```
-----
|               |
|   Alta de Usuario   |
|-----|
| 1) Crear usuario    |
| 2) Volver           |
|-----|
|
Ingrese una opción: 1
Ingrese un usuario: administrador
Ingrese grupo donde alojarlo: admin
Ingrese contraseña: admin
Creando Usuario

apptec
administrador
Presione Enter para continuar...
```

### Administrador de base de datos.

```
-----
|               |
|   Alta de Usuario   |
|-----|
| 1) Crear usuario    |
| 2) Volver           |
|-----|
|
Ingrese una opción: 1
Ingrese un usuario: adminBD
Ingrese grupo donde alojarlo: adminbd
Ingrese contraseña: adminbd

-----
|               |
| Configuración de usuario |
|-----|
| 1) Alta de usuario    |
| 2) Baja de usuario    |
| 3) Modificar usuario  |
| 4) Listar usuarios    |
| 5) Volver             |
|-----|
|
Ingrese una opción: 4

apptec
administrador
adminBD
Presione Enter para continuar...
```



## Operador.

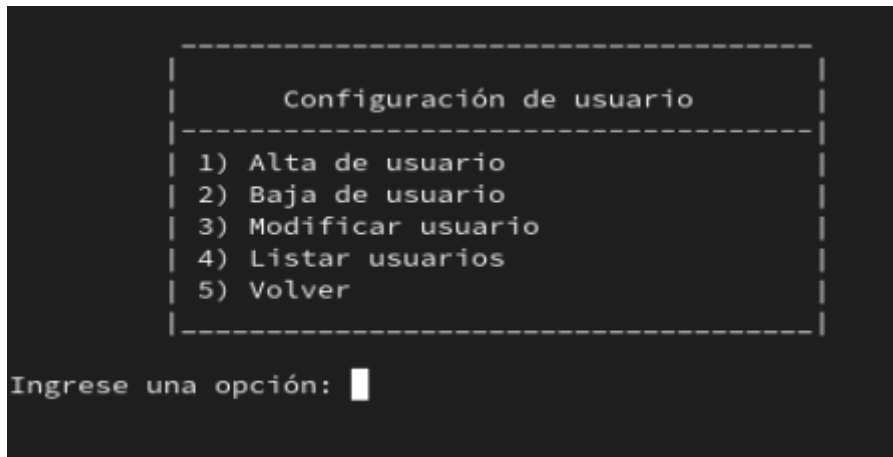
```
-----
|           Alta de Usuario           |
|-----|
| 1) Crear usuario                    |
| 2) Volver                          |
|-----|
Ingrese una opción: 1
Ingrese un usuario: operador
Ingrese grupo donde alojarlo: op
Ingrese contraseña: operador
```

```
-----
| Configuración de usuario            |
|-----|
| 1) Alta de usuario                  |
| 2) Baja de usuario                  |
| 3) Modificar usuario                |
| 4) Listar usuarios                  |
| 5) Volver                          |
|-----|
Ingrese una opción: 4
```

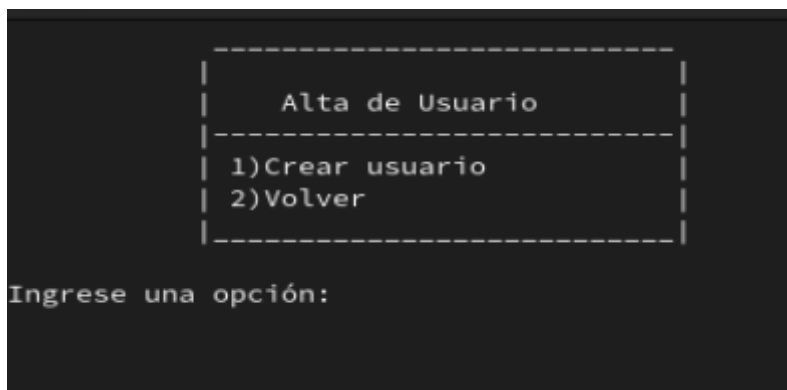
## Menú operador.

```
-----
|           Menú principal            |
|-----|
| 1) Configuración de usuario         |
| 2) Configuración de grupo           |
| 3) Servicios                        |
| 4) Respaldos                        |
| 5) Red                              |
| 6) Firewall                         |
| 7) Logs                             |
| 8) Salir                            |
|-----|
Ingrese una opción:
```

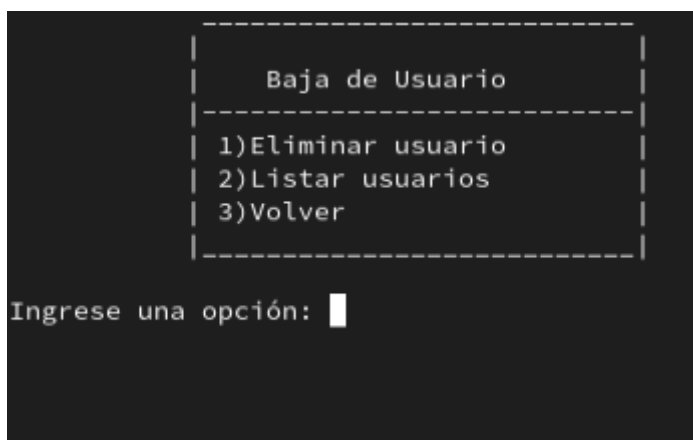
## Menú de alta usuario.



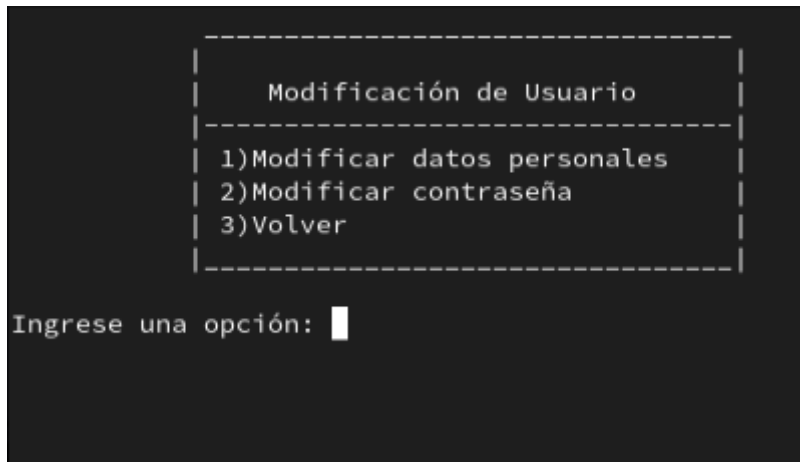
## Alta.



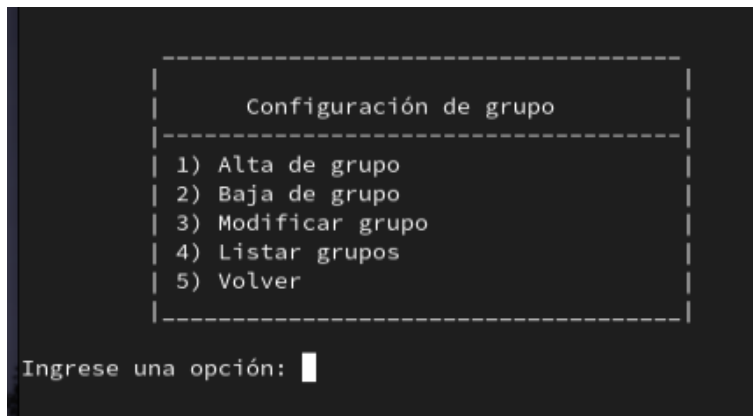
## Baja.



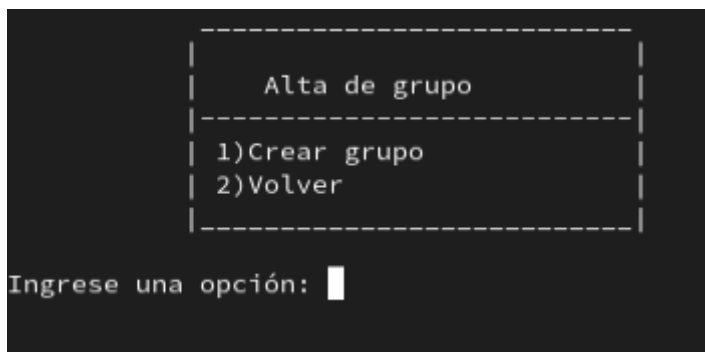
## Modificación.



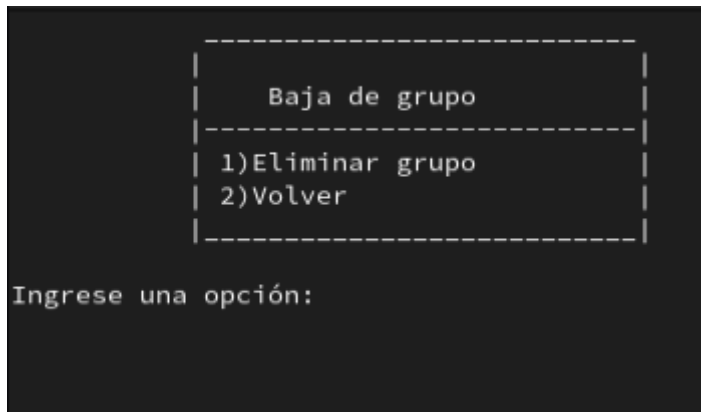
## Menú de alta grupo.



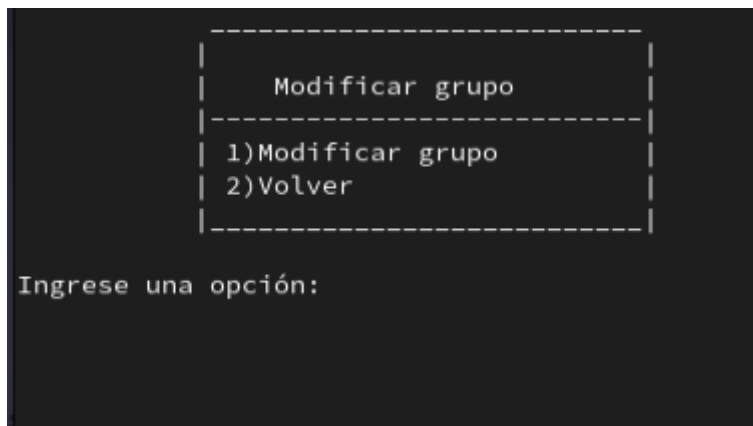
## Alta grupo.



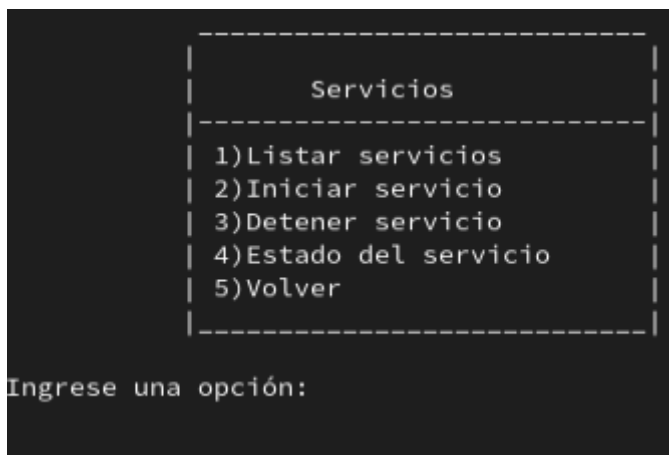
### Baja grupo.



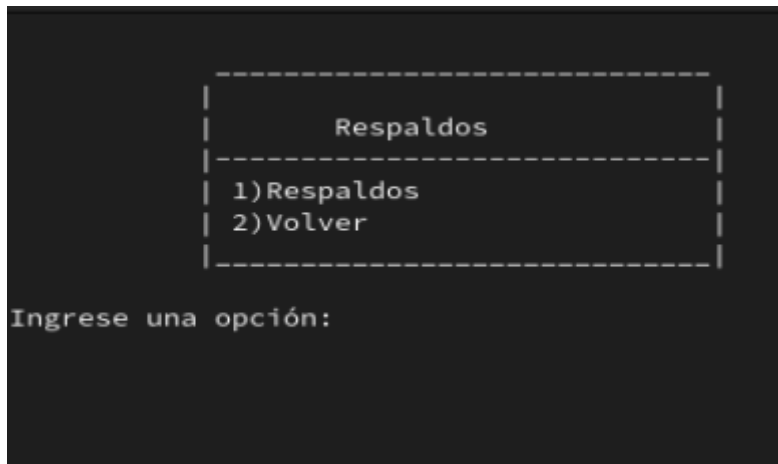
### Modificación grupo.



### Servicios.



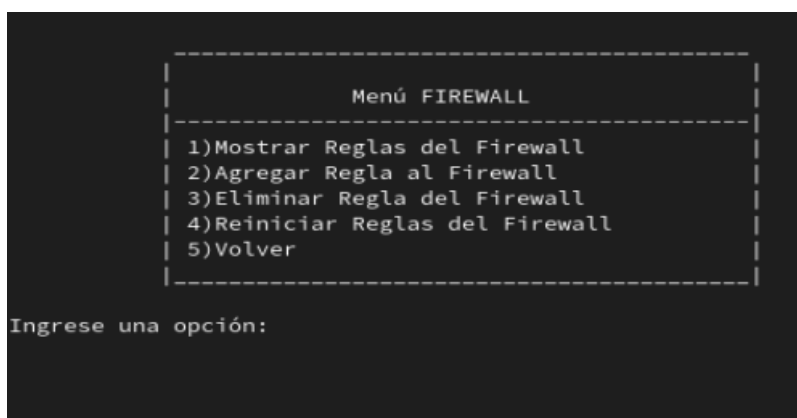
## Respaldos.



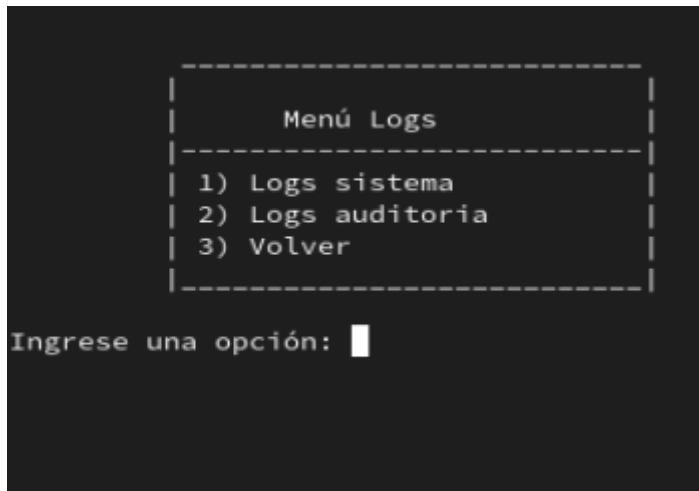
## RED.



## Firewall.



## Logs.



## Configuración de red en las terminales y servidor.

La asignación de la IP del servidor será una IP fija ya que daría bastantes problemas ponerlo por DHCP.

En las tablets de los jueces si usaremos DHCP, ya que no nos cambia mucho que cambien de IP.

## Configuración del servicio SSH.

Instalar el Servicio SSH:

```
sudo dnf install openssh-server
```

```
[AdministradorSistema@192 ~]$ sudo dnf install openssh-server
[sudo] contraseña para AdministradorSistema:
Última comprobación de caducidad de metadatos hecha hace 2:16:42, el dom 12 nov 2023 10:43:34.
El paquete openssh-server-9.0p1-17.fc38.x86_64 ya está instalado.
Dependencias resueltas.
Nada por hacer.
¡Listo!
[AdministradorSistema@192 ~]$
```

En este caso el paquete ya está instalado. Iniciar y Habilitar el Servicio SSH:

Después de instalar el servidor SSH, hay que iniciarlo y habilitarlo para que se ejecute automáticamente al iniciar el sistema:

```
sudo systemctl start sshd
```

```
sudo systemctl enable sshd
```

Con el comando `sudo systemctl status sshd` veremos en qué estado se encuentra el servicio.

```
[AdministradorSistema@192 ~]$ sudo systemctl start sshd
[sudo] contraseña para AdministradorSistema:
[AdministradorSistema@192 ~]$ sudo systemctl enable sshd
[AdministradorSistema@192 ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
   Drop-In: /usr/lib/systemd/system/service.d
            └─10-timeout-abort.conf
   Active: active (running) since Sun 2023-11-12 09:50:07 -03; 3h 16min ago
     Docs: man:sshd(8)
            man:sshd_config(5)
   Main PID: 929 (sshd)
     Tasks: 1 (limit: 6872)
    Memory: 2.2M
       CPU: 32ms
    CGroup: /system.slice/sshd.service
            └─929 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

nov 12 09:50:06 localhost systemd[1]: Starting sshd.service - OpenSSH server daemon...
nov 12 09:50:07 localhost sshd[929]: Server listening on 0.0.0.0 port 22.
nov 12 09:50:07 localhost sshd[929]: Server listening on :: port 22.
nov 12 09:50:07 localhost systemd[1]: Started sshd.service - OpenSSH server daemon.
[AdministradorSistema@192 ~]$
```

Configurar el Firewall para SSH:

```
sudo firewall-cmd --add-service=ssh --permanent
```

```
sudo firewall-cmd --reload
```

```
[AdministradorSistema@192 ~]$ sudo firewall-cmd --add-service=ssh --permanent
Warning: ALREADY_ENABLED: ssh
success
```

```
[AdministradorSistema@192 ~]$ sudo firewall-cmd --reload
success
```

Conectar al Servidor SSH:

Ahora, desde otro sistema utilizando el siguiente comando nos conectaremos:

```
ssh -L 3307:localhost:3307 AdministradorSistema@192.168.1.11
```

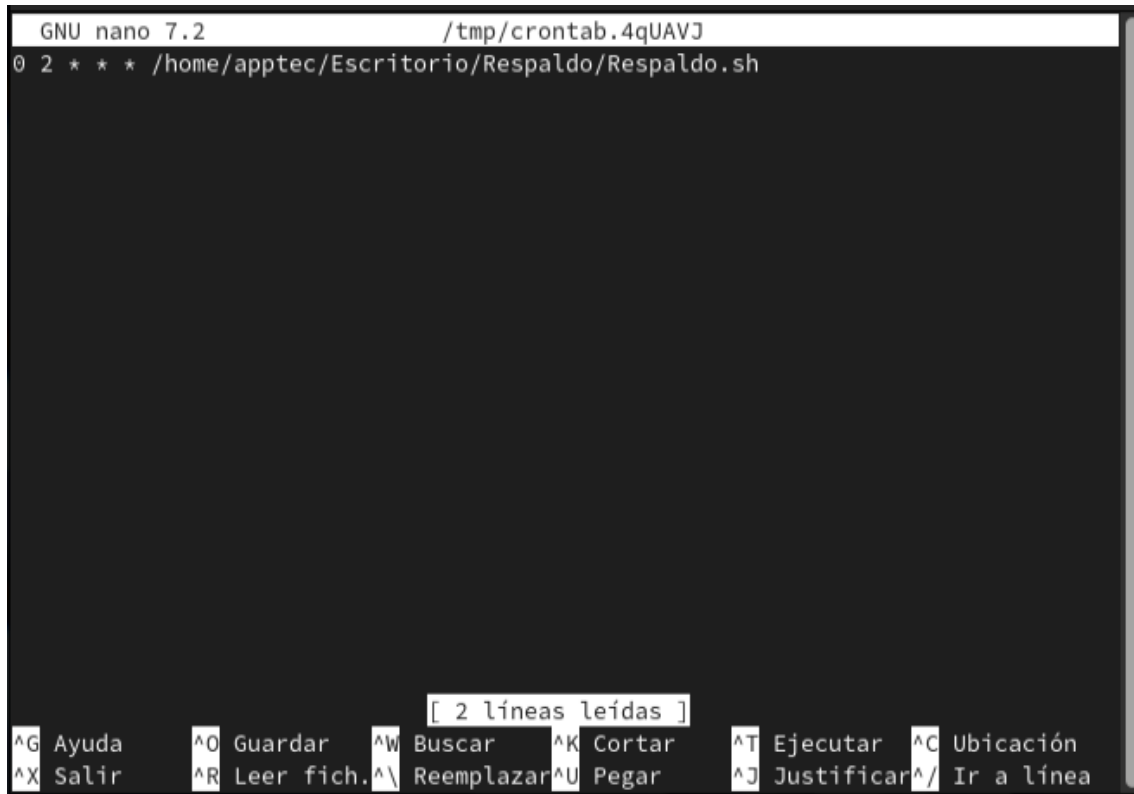
```
C:\Users\Mauro>ssh -L 3307:localhost:3307 AdministradorSistema@192.168.1.11
AdministradorSistema@192.168.1.11's password:
Web console: https://192.168.1.11:9090/ or https://192.168.1.11:9090/

Last failed login: Sun Nov 12 13:36:10 -03 2023 from 192.168.1.9 on ssh:notty
There were 2 failed login attempts since the last successful login.
Last login: Sun Nov 12 09:52:34 2023
[AdministradorSistema@192 ~]$
```

## Archivos crontab con rutinas de backup.

Configuración del archivo crontab para realizar una copia de seguridad diariamente a las 2:00 de la mañana.

```
[apptec@fedora ~]$ crontab -e
crontab: installing new crontab
```



```
GNU nano 7.2 /tmp/crontab.4qUAVJ
0 2 * * * /home/apptec/Escritorio/Respaldo/Respaldo.sh
```

[ 2 líneas leídas ]

^G Ayuda   ^O Guardar   ^W Buscar   ^K Cortar   ^T Ejecutar   ^C Ubicación  
 ^X Salir   ^R Leer fich.   ^\ Reemplazar   ^U Pegar   ^J Justificar   ^/ Ir a línea

```
#!/bin/bash
2
3 DIRECTORIO="/home/apptec/Documentos"
4
5 RUTA_RESPALDO="admin:///media/sf_D_DRIVE/respaldo"
6
7 FECHA=$(date +"%Y%m%d%H%M%S")
8 NOMBRE_RESPALDO="respaldo_$(FECHA).tar.gz"
9
10 # Crear el archivo de respaldo
11 tar -czvf "$RUTA_RESPALDO/$NOMBRE_RESPALDO" "$DIRECTORIO"
12
13 # Registrar el evento en un archivo de registro
14 echo "Se realizó una copia de seguridad de $DIRECTORIO el $FECHA" >> admin:///media/sf_D_DRIVE/respaldo/
  respaldos.log
15
```