



# 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**

**11/09/2023**

**SEGUNDA ENTREGA**

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

**3BH**

## Índice

Usuarios necesarios en el sistema operativo creados de acuerdo al estudio de roles .....	3
Administrador. ....	3
Administrador de base de datos. ....	4
Operador.....	5
Menú operador. ....	6
Menú. ....	6
Menú de alta usuario. ....	7
Alta.....	8
Baja. ....	9
Modificación. ....	10
Menú de alta grupo.....	11
Alta grupo.....	12
Baja grupo. ....	13
Modificación grupo.....	14
Servicios.....	15
Configuración de red en las terminales y servidor. ....	16
Configuración del servicio SSH. ....	16
Archivos crontab con rutinas de backup. ....	17
Configuración del firewall GNU/Linux. ....	18
Filtrado de ips mediante firewall.....	19

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
```

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



## Menú de alta usuario.

```

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

```

```

#!/bin/bash
clear

while [ "$op" != 4 ]; do
echo "

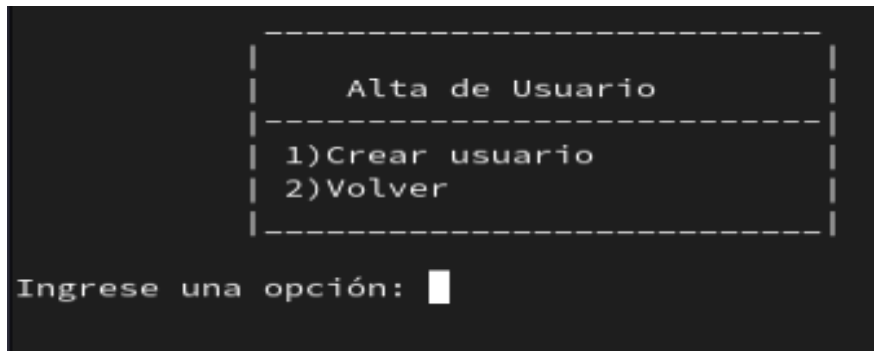
|-----|
|               Configuración de usuario               |
|-----|
| 1) Alta de usuario |
| 2) Baja de usuario |
| 3) Modificar usuario |
| 4) Listar usuarios |
| 5) Volver          |
|-----|

"
read -p "Ingrese una opción: " op
case $op in
1) bash Alta.sh
sleep 0s;;
2) bash Baja.sh
sleep 0s;;
3) bash Modificacion.sh
sleep 0s;;
4) clear
echo "Usuarios Creados:"
cut -d: -f1 /etc/passwd
read -p "Presione Enter para continuar..."
clear
bash ConfigurarU.sh
;;
5) clear
exit;;
*) echo "Opción no valida"
sleep 1s

clear
esac
done

```

Alta.



```
#!/bin/bash
clear
while [ "$op" != 2 ];
do
echo "
| Alta de Usuario |
|-----|
| 1) Crear usuario |
| 2) Volver       |
|                 |
"

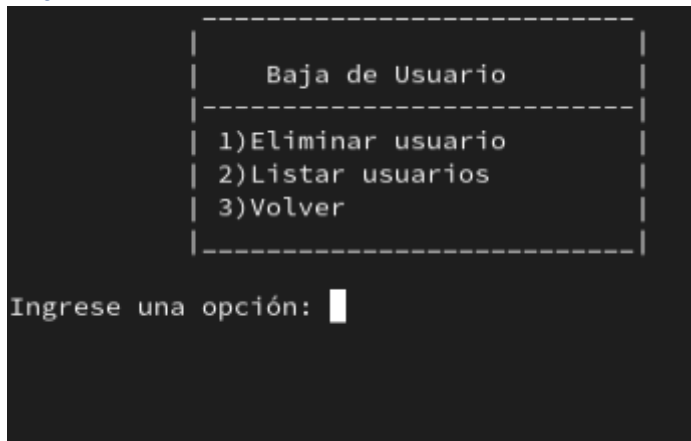
read -p "Ingrese una opción: " opcion

case $opcion in
1)
read -p "Ingrese un usuario: " nom
read -p "Ingrese grupo donde alojarlo: " grp
read -p "Ingrese contraseña: " cont
sudo useradd -g ${grp} -s /bin/bash -d /home/${nom} -p ${cont} -m ${nom}
echo "Creando Usuario"
sleep 2
;;
2)
clear
exit
;;
*) echo "Opción no valida"
sleep 1s

clear;;
esac
done
```



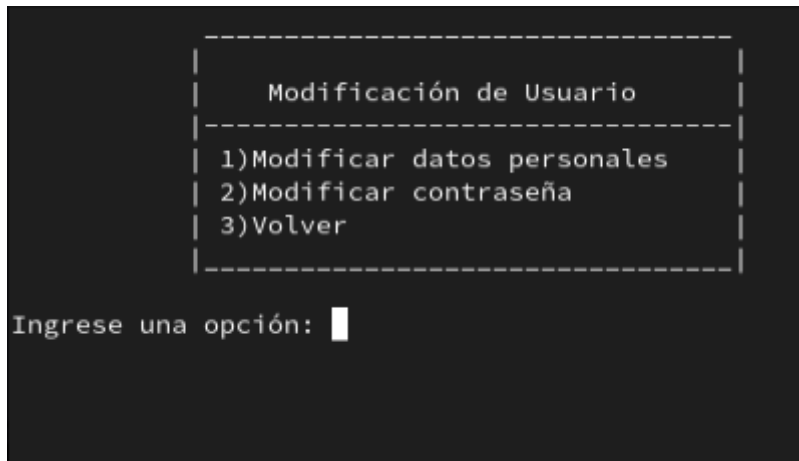
## Baja.



```
#!/bin/bash
while [[ $op != 3 ]]; do
clear
echo "
    Baja de Usuario
    -----
    1)Eliminar usuario
    2)Listar usuarios
    3)Volver
"
read -p "Ingrese una opción: " opcion
case $opcion in
    1) clear
        echo -n "Ingrese el nombre del usuario a borrar:"
        read username
        if [ "$username" = root ]; then
            echo "No tiene los permisos necesarios."
        else
            echo
            echo "¿Desea borrar el directorio de trabajo y todo su contenido [Y/N]?"
            echo -n "Valor por defecto [n]: "
            read delete_home
            fi
            if [ "$delete_home" = "Y" ]; then
                sudo userdel -r $username
            else
                sudo userdel $username
            fi;;
    2) clear
        echo "Usuarios Creados:"
        cut -d: -f1 /etc/passwd
        read -p "Presione Enter para continuar...";;
    3)clear
        exit;;
    *) echo "Opción no valida"
        sleep 1s

clear;;
esac
done
```

## Modificación.



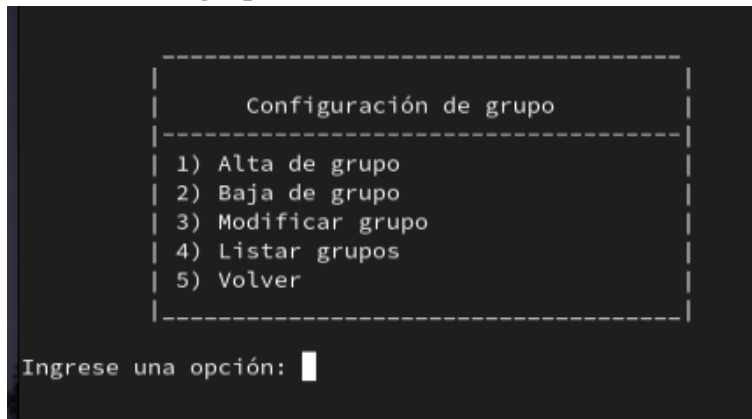
```
#!/bin/bash
clear
while [[ $op != 3 ]]; do

echo "
  Modificación de Usuario
  -----
  1) Modificar datos personales
  2) Modificar contraseña
  3) Volver
"

read -p "Ingrese una opción: " op
case $op in
1) echo
   echo -n "Nombre del usuario al que quiere cambiar su información personal: "
   read username
   echo
   sudo chfn $username
   echo;;
2) echo -n "Ingrese el nombre del usuario que quiere modificar la clave: "
   read username
   echo
   sudo passwd $username
   echo;;
3)
   clear
   exit;;
*) echo "Opción no valida"
   sleep 1s

clear;;
esac
done
```

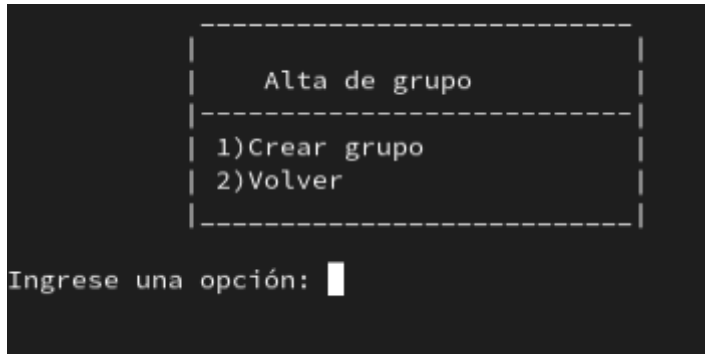
## Menú de alta grupo.



```
#!/bin/bash
clear
while [ "$op" != 5 ] ; do
echo "
    Configuración de grupo
    -----
    1) Alta de grupo
    2) Baja de grupo
    3) Modificar grupo
    4) Listar grupos
    5) Volver
"
read -p "Ingrese una opción: " op
case $op in
    1) bash AltaG.sh
        sleep 0s;;
    2) bash BajaG.sh
        sleep 0s;;
    3) bash ModificarG.sh
        sleep 0s;;
    4) clear
        echo " Grupos creados:"
        less /etc/group
        clear
        bash ConfigurarG.sh
        ;;
    5) clear
        exit;;
    *) echo "Opción no valida"
        sleep 1s

clear
esac
done
```

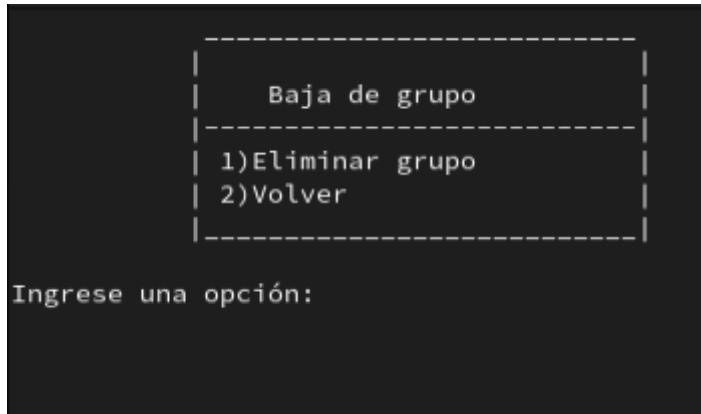
## Alta grupo.



```
#!/bin/bash
clear
while [ "$op" != 2 ];
do
echo "
    Alta de grupo
    -----
    1) Crear grupo
    2) Volver
"
    read -p "Ingrese una opción: " op

    case $op in
        1)
            clear
            read -p "Ingrese el grupo a crear: " grp
            sudo groupadd ${grp}
            echo "Grupo creado satisfactoriamente"
            sleep 2
            clear
            ;;
        2)
            clear
            exit
            ;;
        *)
            echo "Opción no válida."
            sleep 1
            clear
            ;;
    esac
done
```

## Baja grupo.



```

#!/bin/bash

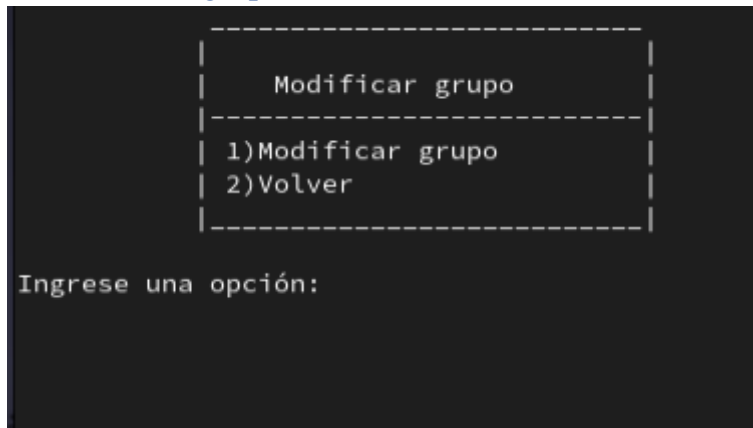
while [[ $op != 2 ]]; do
clear

echo "
      Baja de grupo
    -----
    1) Eliminar grupo
    2) Volver
  -----
"

  read -p "Ingrese una opción: " opcion

  case $opcion in
    1)
      clear
      read -p "Ingrese el nombre del grupo a eliminar: " grp
      sudo groupdel ${grp}
      echo "Grupo eliminado satisfactoriamente"
      sleep 2s
      clear
      ;;
    2)
      clear
      exit
      ;;
    *)
      echo "Opción no válida."
      sleep 1s
      ;;
  esac
done
  
```

## Modificación grupo.



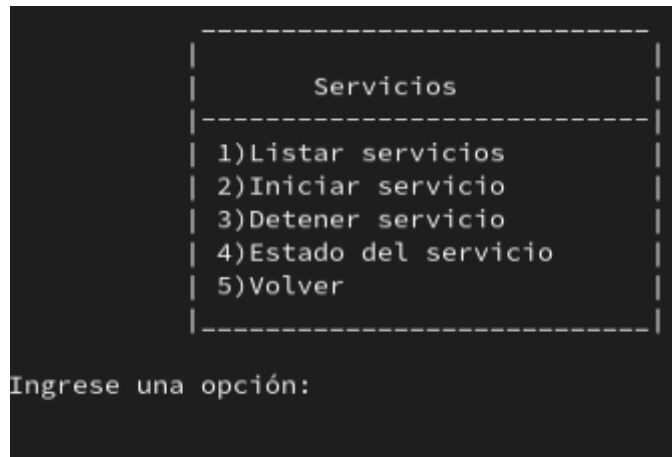
```
#!/bin/bash

while [[ $op != 2 ]]; do
    clear

    echo "
    Modificar grupo
    -----
    1) Modificar grupo
    2) Volver
    "

    read -p "Ingrese una opción: " opcion
    case $opcion in
        1)
            clear
            read -p "Ingrese el nombre del grupo a modificar: " grp
            read -p "Ingrese el nuevo nombre del grupo: " new_grp
            sudo groupmod -n ${new_grp} ${grp}
            echo "Grupo modificado satisfactoriamente"
            sleep 2
            clear
            ;;
        2)
            clear
            exit
            ;;
        *)
            echo "Opción no válida."
            ;;
    esac
done
```

## Servicios.



```
#!/bin/bash
while [[ $op != 5 ]]; do
    clear
    echo "
    "
    read -p "Ingrese una opción: " opcion
    case $opcion in
        1) clear
            sudo systemctl list-unit-files --type service --all;;
        2) clear
            read -p "Ingrese el nombre del servicio a iniciar: " servicio
            sudo systemctl start $servicio
            echo "Servicio $servicio iniciado."
            sleep 2;;
        3) clear
            read -p "Ingrese el nombre del servicio a detener: " servicio
            sudo systemctl stop $servicio
            echo "Servicio $servicio detenido."
            sleep 2;;
        4) clear
            read -p "Ingrese el nombre del servicio a verificar: " servicio
            sudo systemctl status $servicio
            read -p "Presione Enter para continuar...";;
        5) clear
            exit;;
        *) echo "Opción no válida."
            sleep 1;;
    esac
done
```

## 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.

La instalación del servidor SSH es bastante facil, simplemente se debe ejecutar los siguientes comandos en la terminal:

Para instalar usaremos el comando:

`sudo dnf -y install openssh-server openssh-clients`

```
[apptec@fedora ~]$ dnf -y install openssh-server openssh-clients
Error: Este comando debe ejecutarse con privilegios de superusuario (bajo el usuario root en la mayoría de los sistemas).
[apptec@fedora ~]$ sudo dnf -y install openssh-server openssh-clients
Copy repo for PyCharm owned by phracek 1.0 kB/s | 2.1 kB 00:02
Fedora 38 - x86_64 - Updates 56 kB/s | 44 kB 00:00
Fedora 38 - x86_64 - Updates 496 kB/s | 5.3 MB 00:10
Fedora Modular 38 - x86_64 - Updates 52 kB/s | 40 kB 00:00
Fedora Modular 38 - x86_64 - Updates 3.7 kB/s | 1.3 kB 00:00
google-chrome 9.6 kB/s | 3.6 kB 00:00
google-chrome 12 kB/s | 15 kB 00:01
RPM Fusion for Fedora 38 - Nonfree - NVIDIA Driver 6.0 kB/s | 14 kB 00:02
RPM Fusion for Fedora 38 - Nonfree - NVIDIA Driver 17 kB/s | 14 kB 00:00
RPM Fusion for Fedora 38 - Nonfree - Steam
El paquete openssh-server-9.0p1-17.fc38.x86_64 ya está instalado.
El paquete openssh-clients-9.0p1-17.fc38.x86_64 ya está instalado.
Dependencias resueltas.
Nada por hacer.
[!listo!]
```

Crear e iniciar servicio

Con `systemctl enable`, creamos y habilitamos el servicio, para que la siguiente vez que arranquemos Fedora, inicie nuestro servidor SSH.

Comando: `sudo systemctl enable sshd.service`

```
[apptec@fedora ~]$ sudo systemctl enable sshd.service
Created symlink /etc/systemd/system/multi-user.target.wants/sshd.service → /usr/lib/systemd/system/sshd.service.
```

Iniciamos el servicio con el comando: `systemctl start sshd.service`

Con esto ya quedaría el servidor SSH.



## 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
```

## Configuración del firewall GNU/Linux.

Permitir el tráfico SSH (puerto 22):

```
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

Para verificar que funcionara utilizamos el siguiente comando: `sudo iptables -L`

```
[apptec@fedora ~]$ sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
[sudo] contraseña para apptec:
[apptec@fedora ~]$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination          tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh
```

Permitir el tráfico HTTP (puerto 80):

```
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

```
[apptec@fedora ~]$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
[apptec@fedora ~]$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination          tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Permitir el tráfico HTTPS (puerto 443):

```
sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

```
[apptec@fedora ~]$ sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
[apptec@fedora ~]$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination          tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Denegar todo el tráfico entrante por defecto y permitir solo las conexiones establecidas y relacionadas:

```
sudo iptables -P INPUT DROP
sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
[apptec@fedora ~]$ sudo iptables -P INPUT DROP
sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
[apptec@fedora ~]$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source                destination            tcp dpt:ssh
ACCEPT     tcp  --  anywhere               anywhere               tcp dpt:http
ACCEPT     tcp  --  anywhere               anywhere               tcp dpt:https
ACCEPT     all  --  anywhere               anywhere               state RELATED,ESTABLISHED

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

## Filtrado de ips mediante firewall.

Para permitir un rango de direcciones IP

```
[apptec@fedora ~]$ sudo iptables -A INPUT -s 192.168.1.0/24 -j ACCEPT
[apptec@fedora ~]$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source                destination            tcp dpt:ssh
ACCEPT     tcp  --  anywhere               anywhere               tcp dpt:http
ACCEPT     tcp  --  anywhere               anywhere               tcp dpt:https
ACCEPT     all  --  anywhere               anywhere               state RELATED,ESTABLISHED
ACCEPT     all  --  192.168.1.0/24         anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Y con el comando siguiente bloquea las que no estén en el anterior rango:

```
[apptec@fedora ~]$ sudo iptables -A INPUT -j DROP
[apptec@fedora ~]$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source                destination            tcp dpt:ssh
ACCEPT     tcp  --  anywhere               anywhere               tcp dpt:http
ACCEPT     tcp  --  anywhere               anywhere               tcp dpt:https
ACCEPT     all  --  anywhere               anywhere               state RELATED,ESTABLISHED
ACCEPT     all  --  192.168.1.0/24         anywhere
DROP       all  --  anywhere               anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```