ACTIVIDAD 1 - IPTABLES

Cristóbal Suárez Abad

SEGURIDAD Y ALTA DISPONIBILIDAD - 2º ASIR

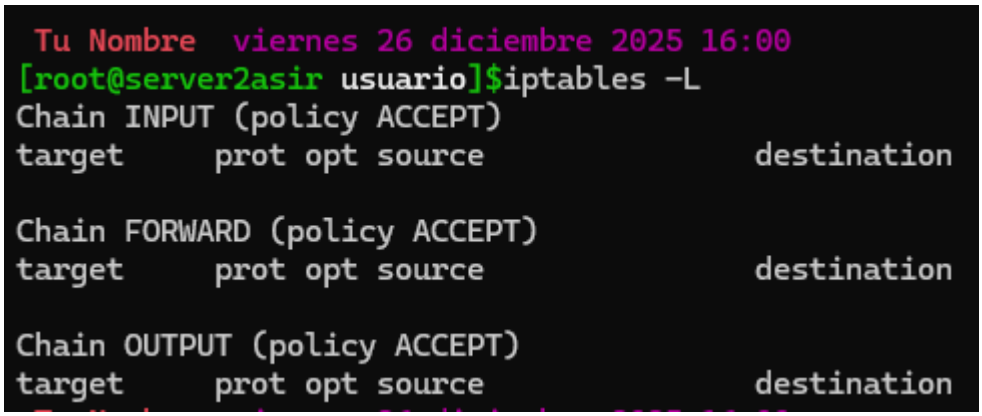
Contenido

IPTABLES

Usando una máquina con cualquier distribución de Linux.

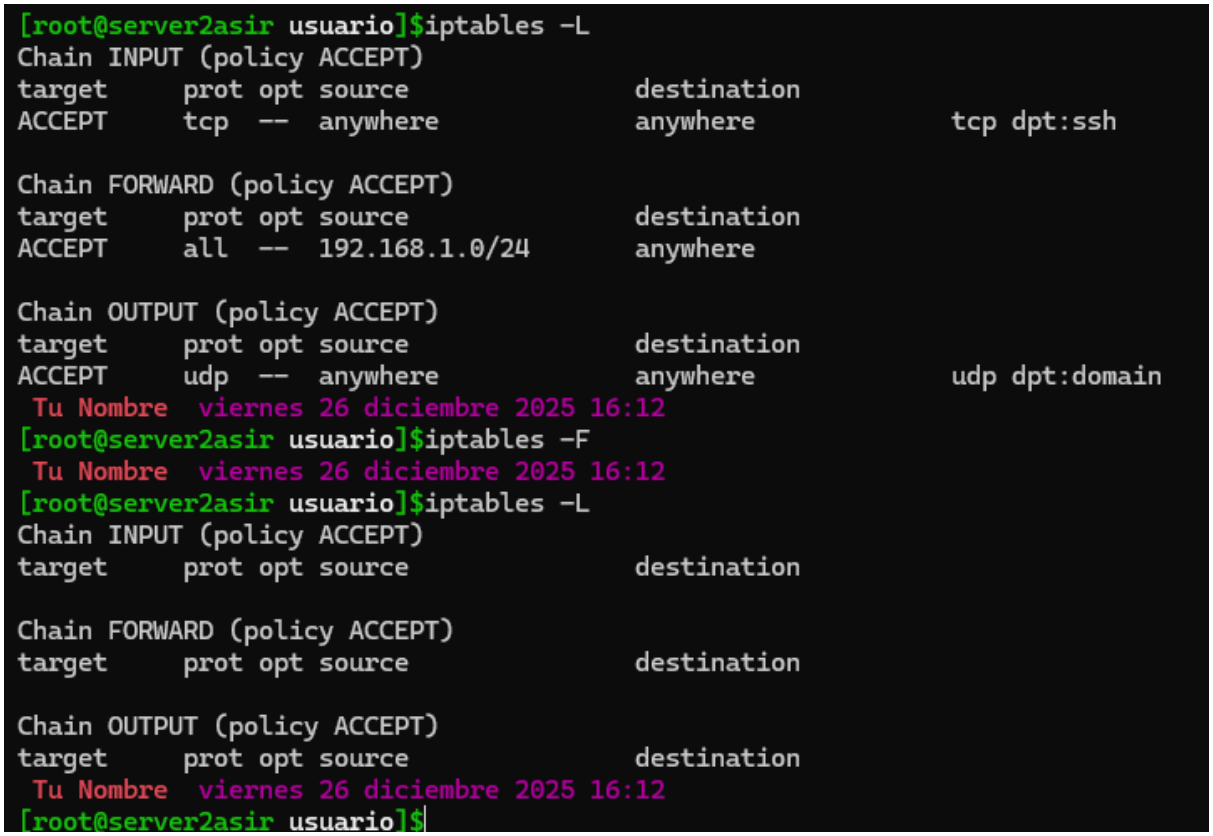
1. Lista todas las reglas o cadenas introducidas en el cortafuegos.

iptables -L



2. Borra todas las reglas introducidas en el cortafuegos.

iptables -F



3. Añade una regla para eliminar todos los paquetes de salida (es decir, que desde tu ordenador no pueda salir ningún paquete) y añade para que se guarde en log estos sucesos

El Log: **iptables -A OUTPUT -j LOG --log-prefix "SALIDA_BLOQUEADA: "**
La restricción: **iptables -A OUTPUT -j DROP**

```
 Tu Nombre  viernes 26 diciembre 2025 16:12
[root@server2asir usuario]$sudo iptables -A OUTPUT -j LOG --log-prefix "SALIDA_BLOQUEADA: "
 Tu Nombre  viernes 26 diciembre 2025 16:21
[root@server2asir usuario]$sudo iptables -A OUTPUT -j DROP
client_loop: send disconnect: Connection reset

C:\Users\Cristobal>
```

- Comprueba después que no tienes acceso a Internet con curl

```
 Tu Nombre  viernes 26 diciembre 2025 16:23
[root@server2asir usuario]$curl -I https://www.google.com
curl: (6) Could not resolve host: www.google.com
 Tu Nombre  viernes 26 diciembre 2025 16:25
[root@server2asir usuario]$ping x.uk
ping: x.uk: Temporary failure in name resolution
 Tu Nombre  viernes 26 diciembre 2025 16:25
[root@server2asir usuario]$
```

- Revisa en logs que se ha registrado el intento de acceder a internet.

sudo dmesg | grep "SALIDA_BLOQUEADA"

```
[ 1459.288182] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=2170 DF PROTO=UDP SPT=40509 DPT=53 LEN=51
[ 1459.288266] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=64787 DF PROTO=UDP SPT=38588 DPT=53 LEN=51
[ 1459.288318] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=30194 DF PROTO=UDP SPT=51797 DPT=53 LEN=51
[ 1459.288352] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=39191 DF PROTO=UDP SPT=51447 DPT=53 LEN=51
[ 1461.531615] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=3380 DF PROTO=UDP SPT=54261 DPT=53 LEN=51
[ 1461.531672] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=64413 DF PROTO=UDP SPT=54342 DPT=53 LEN=51
[ 1461.531719] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=35093 DF PROTO=UDP SPT=32831 DPT=53 LEN=51
[ 1461.531751] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=20725 DF PROTO=UDP SPT=46262 DPT=53 LEN=51
[ 1610.825983] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=31528 DF PROTO=UDP SPT=47317 DPT=53 LEN=51
[ 1610.826035] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=58939 DF PROTO=UDP SPT=42040 DPT=53 LEN=51
[ 1610.826081] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=12807 DF PROTO=UDP SPT=48077 DPT=53 LEN=51
[ 1610.826112] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=53931 DF PROTO=UDP SPT=39379 DPT=53 LEN=51
[ 1615.550473] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=38133 DF PROTO=UDP SPT=35668 DPT=53 LEN=41
[ 1615.550525] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=43493 DF PROTO=UDP SPT=58302 DPT=53 LEN=41
[ 1615.550570] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=33368 DF PROTO=UDP SPT=50790 DPT=53 LEN=41
[ 1615.550601] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=30617 DF PROTO=UDP SPT=51880 DPT=53 LEN=41
Tu Nombre  viernes 26 diciembre 2025 16:25
```

sudo tail -f /var/log/syslog | grep "SALIDA_BLOQUEADA"

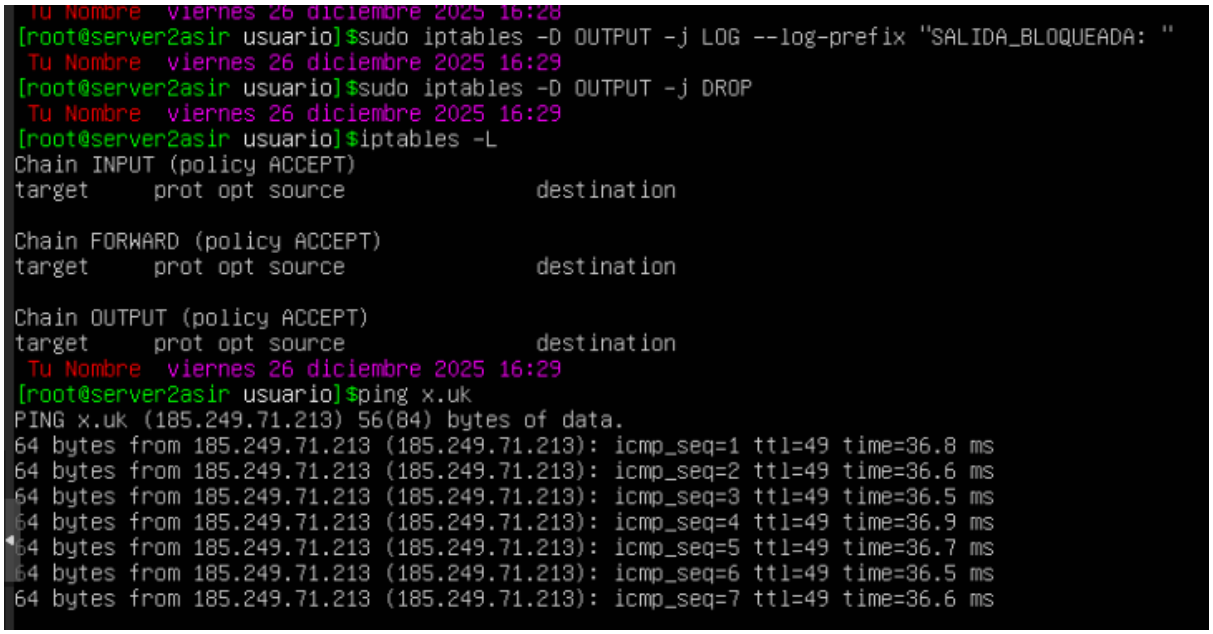
```
[root@server2asir usuario]$sudo tail -f /var/log/syslog | grep "SALIDA_BLOQUEADA"
Dec 26 16:22:55 server2asir kernel: [ 1461.531751] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=20725 DF PROTO=
UDP SPT=46262 DPT=53 LEN=51
Dec 26 16:25:25 server2asir kernel: [ 1610.825983] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=31528 DF PROTO=
UDP SPT=47317 DPT=53 LEN=51
Dec 26 16:25:25 server2asir kernel: [ 1610.826035] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=58939 DF PROTO=
UDP SPT=42040 DPT=53 LEN=51
Dec 26 16:25:25 server2asir kernel: [ 1610.826081] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=12807 DF PROTO=
UDP SPT=48077 DPT=53 LEN=51
Dec 26 16:25:25 server2asir kernel: [ 1610.826112] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=53931 DF PROTO=
UDP SPT=39379 DPT=53 LEN=51
Dec 26 16:25:29 server2asir kernel: [ 1615.550473] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=38133 DF PROTO=
UDP SPT=35668 DPT=53 LEN=41
Dec 26 16:25:29 server2asir kernel: [ 1615.550525] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=43493 DF PROTO=
UDP SPT=58302 DPT=53 LEN=41
Dec 26 16:25:29 server2asir kernel: [ 1615.550570] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=33368 DF PROTO=
UDP SPT=50790 DPT=53 LEN=41
Dec 26 16:25:29 server2asir kernel: [ 1615.550601] SALIDA_BLOQUEADA: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=61 TOS=0x00 PREC=0x00 TTL=64 ID=30617 DF PROTO=
UDP SPT=51880 DPT=53 LEN=41
```

4. Elimina la regla introducida anteriormente y comprueba que de nuevo tienes acceso a Internet.

Como solo hay una podríamos usar: **sudo iptables -F OUTPUT**

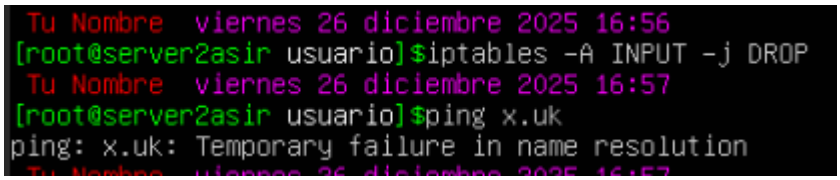
Pero para eliminar solo la especificada, es lo mismo, pero usando “-D” en lugar de “-A”.

sudo iptables -D OUTPUT -j LOG --log-prefix "SALIDA_BLOQUEADA: "
sudo iptables -D OUTPUT -j DROP



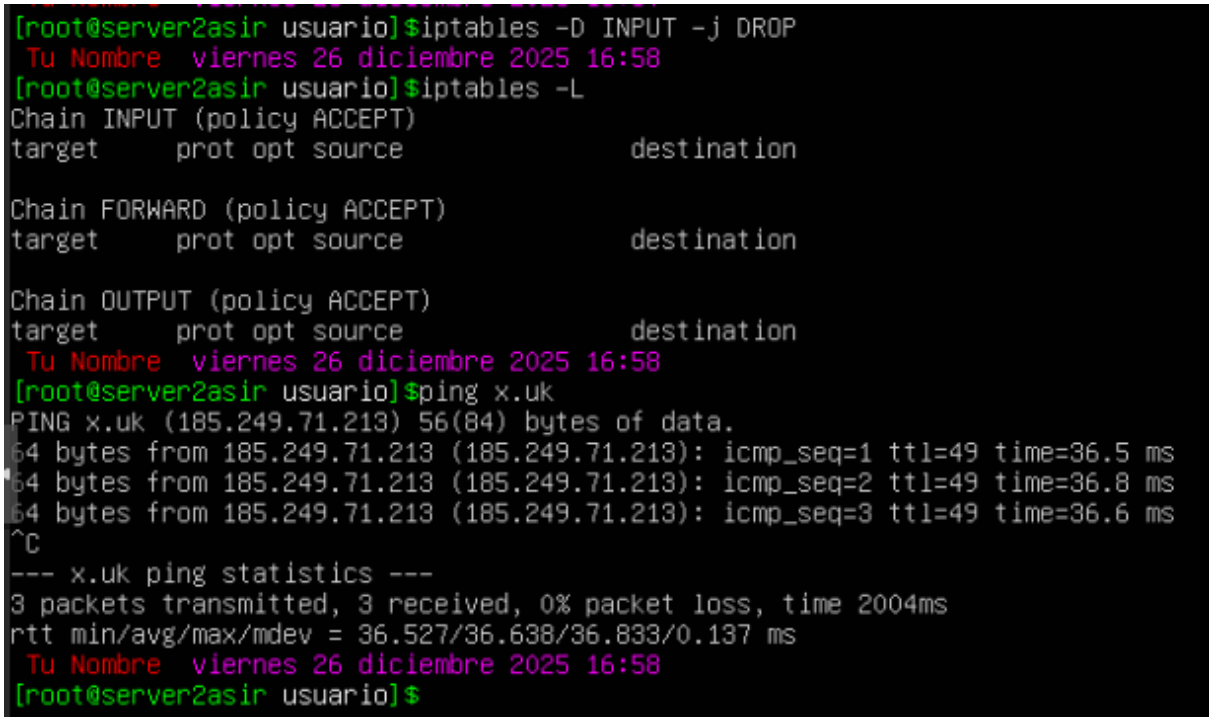
5. Añade una regla para no dejar pasar ningún paquete de entrada. Comprueba en este caso que tampoco tienes conexión a Internet.

iptables -A INPUT -j DROP



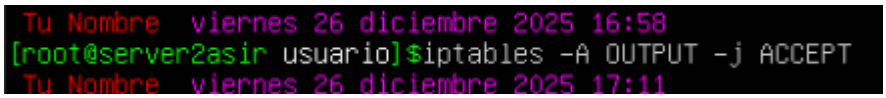
6. Borra la regla introducida en el apartado 5.

iptables -D INPUT -j DROP



7. Añade una regla para dejar pasar todos los paquetes que salgan de tu ordenador.

iptables -A OUTPUT -j ACCEPT

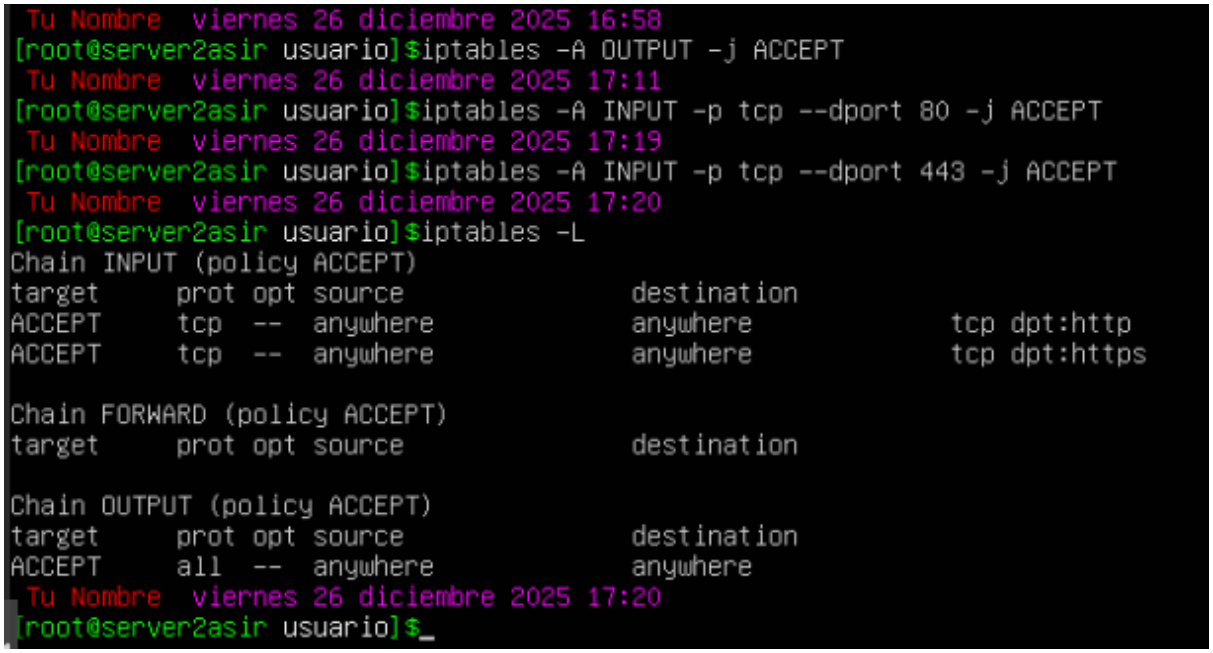


8. Tenemos un servidor web instalado y queremos permitir el acceso desde el exterior. Añade la regla necesaria al cortafuegos.

Los puertos estándar para servicio web son 80 (HTTP) y 443 (HTTPS).

Puerto 80: **sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT**
Puerto 443: **sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT**

“-p” especifica el protocolo de puertos (TCP o UDP).
“--dport" indica el número del puerto.



En el enunciado no se indica, pero si queremos hilar más fino, teniendo en cuenta que la ip privada del servidor es 10.2.17.33, podemos usar:

Puerto 80: sudo iptables -A INPUT -p tcp -d 10.2.17.33 --dport 80 -j ACCEPT
Puerto 443: sudo iptables -A INPUT -p tcp -d 10.2.17.33 --dport 443 -j ACCEPT

“-d 10.2.17.33”: Indica que el “destino” es la IP del servidor.

```
 Tu Nombre  viernes 26 diciembre 2025 17:20
[root@server2asir usuario]$iptables -A INPUT -p tcp -d 10.2.17.33 --dport 80 -j ACCEPT
 Tu Nombre  viernes 26 diciembre 2025 17:27
[root@server2asir usuario]$iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     tcp  --  anywhere             server2asir          tcp dpt:http

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
 Tu Nombre  viernes 26 diciembre 2025 17:27
[root@server2asir usuario]$iptables -A INPUT -p tcp -d 10.2.17.33 --dport 443 -j ACCEPT
 Tu Nombre  viernes 26 diciembre 2025 17:27
[root@server2asir usuario]$iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     tcp  --  anywhere             server2asir          tcp dpt:http
ACCEPT     tcp  --  anywhere             server2asir          tcp dpt:https

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
 Tu Nombre  viernes 26 diciembre 2025 17:27
```

8. Tenemos un servidor ftp funcionando y queremos permitir el acceso ftp desde el exterior.

Para el servicio de FTP se usan varios puertos. El primero que vamos a configurar es el de control.

sudo iptables -A INPUT -p tcp --dport 21 -j ACCEPT
sudo iptables -A INPUT -p tcp -d 10.2.17.33 --dport 21 -j ACCEPT

El problema viene ahora. La mayoría de las conexiones FTP externas utilizan el Modo Pasivo. En este modo, el servidor le indica al cliente un rango de puertos aleatorios para transferir los datos. Este rango de puertos puede variar según el tipo de servidor FTP, por ejemplo, en la guía de ProFTPD[1] se indica:

"The passive FTP connections will use ports from 1024 and up, which means that you must forward all ports 1024-65535 from the NAT to the FTP server!"

Y a continuación se aconseja establecer un puerto específico en el archivo **proftpd.conf**. En este ejercicio no hay que configurar esto, solo abrir un segundo puerto con IPTABLES.

Por ejemplo, podemos usar el 56123:

sudo iptables -A INPUT -p tcp --dport 56123 -j ACCEPT
sudo iptables -A INPUT -p tcp -d 10.2.17.33 --dport 56123 -j ACCEPT

[1] http://www.proftpd.org/docs/howto/NAT.html

```
 Tu Nombre  sábado 27 diciembre 2025 10:50
[root@server2asir usuario]$iptables -A INPUT -p tcp --dport 21 -j ACCEPT
 Tu Nombre  sábado 27 diciembre 2025 10:51
[root@server2asir usuario]$iptables -A INPUT -p tcp -d 10.2.17.33 --dport 21 -j ACCEPT
 Tu Nombre  sábado 27 diciembre 2025 10:51
[root@server2asir usuario]$iptables -A INPUT -p tcp --dport 56123 -j ACCEPT
 Tu Nombre  sábado 27 diciembre 2025 10:55
[root@server2asir usuario]$iptables -A INPUT -p tcp -d 10.2.17.33 --dport 56123 -j ACCEPT
 Tu Nombre  sábado 27 diciembre 2025 10:55
[root@server2asir usuario]$iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     tcp  --  anywhere             server2asir          tcp dpt:https
ACCEPT     tcp  --  anywhere             server2asir          tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ftp
ACCEPT     tcp  --  anywhere             server2asir          tcp dpt:ftp
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:56123
ACCEPT     tcp  --  anywhere             server2asir          tcp dpt:56123

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
 Tu Nombre  sábado 27 diciembre 2025 10:55
[root@server2asir usuario]$_
```

ATENCIÓN: Si queremos utilizar el modo activo para FTP, debemos permitir las conexiones entrantes para el puerto 21 y las salientes para el puerto 20[2][3][4].

sudo iptables -A INPUT -p tcp --dport 21 -j ACCEPT
sudo iptables -A OUTPUT -p tcp --sport 20 -j ACCEPT

ATENCIÓN: En algunos foros[5] se comenta que es mejor usar una regla "**RELATED,ESTABLISHED**" para ligar el puerto 21 con el de descarga, de manera que no se tiene que especificar un puerto (útil si tenemos un rango de puertos).

iptables -A INPUT -p tcp --dport 20:21 --syn -m conntrack --ctstate NEW -j ACCEPT
iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

10. Añade la regla necesaria en IPTABLES.

[2] https://serverfault.com/questions/38398/allowing-ftp-with-iptables
[3] https://slacksite.com/other/ftp.html
[4] https://askubuntu.com/questions/261626/configuration-of-iptables-verfication-actives-services-allow-ftp
[5] https://superuser.com/questions/1535341/how-to-allow-ftp-traffic-in-iptables-without-opening-all-ports-higher-then-1024

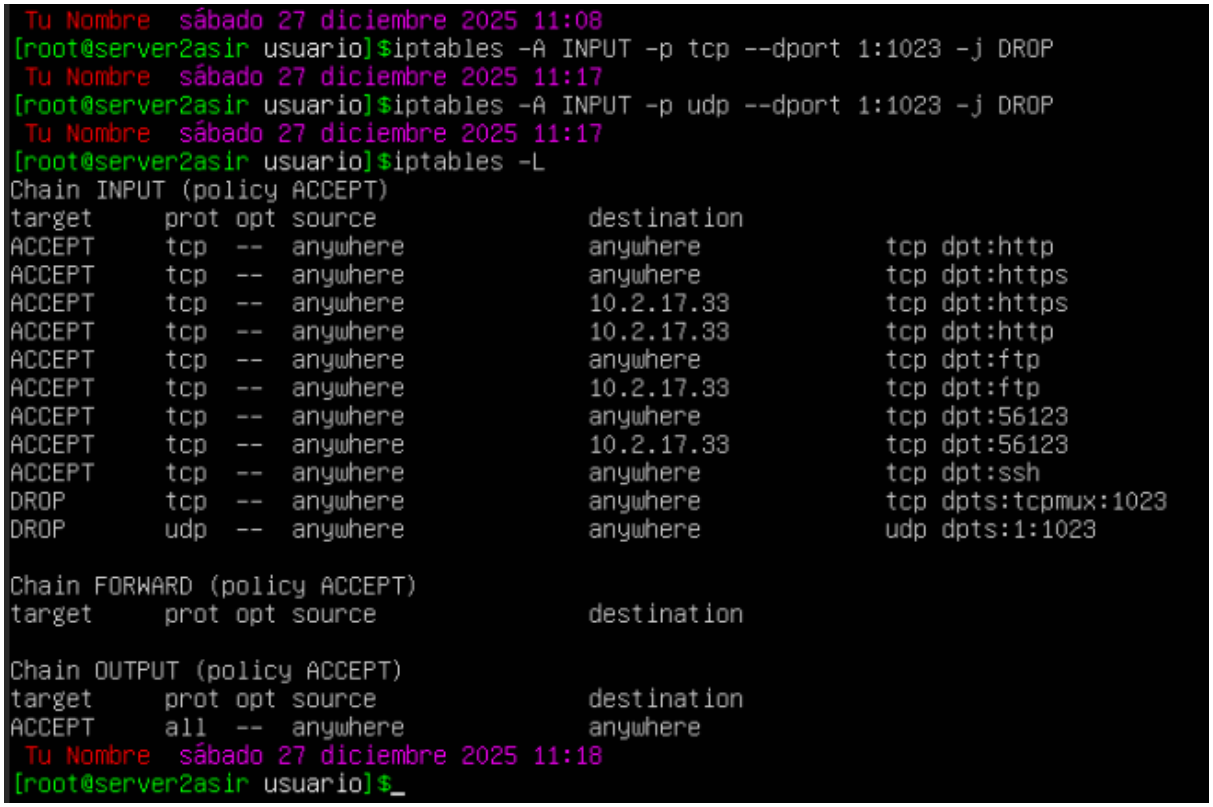
11. Si quieres denegar el acceso por debajo del puerto 1024. ¿Qué reglas debes añadir a IPTABLES?

Si no queremos perder la conexión SSH antes de realizar el ejercicio, debemos usar:

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

Para bloquear del puerto 1024 hacia abajo (1024 no incluido):

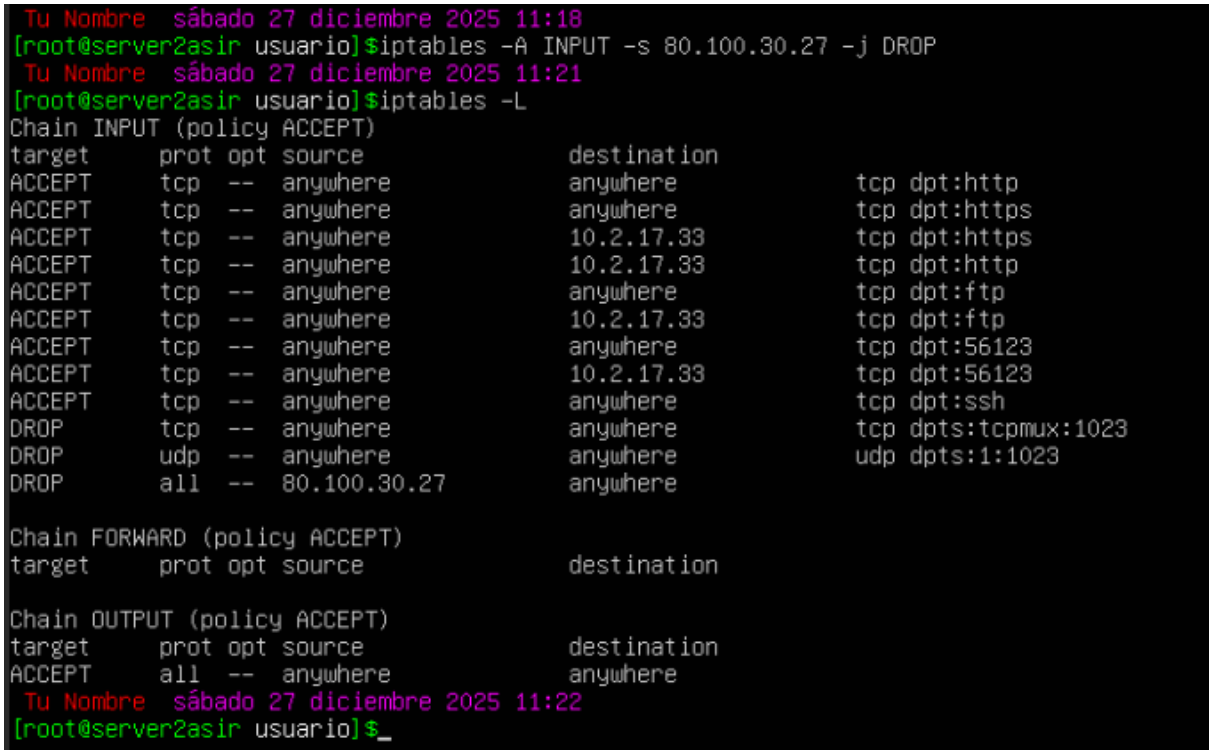
sudo iptables -A INPUT -p tcp --dport 1:1023 -j DROP
sudo iptables -A INPUT -p udp --dport 1:1023 -j DROP



12. Rechaza todos los paquetes que vengan de la dirección 80.100.30.27.

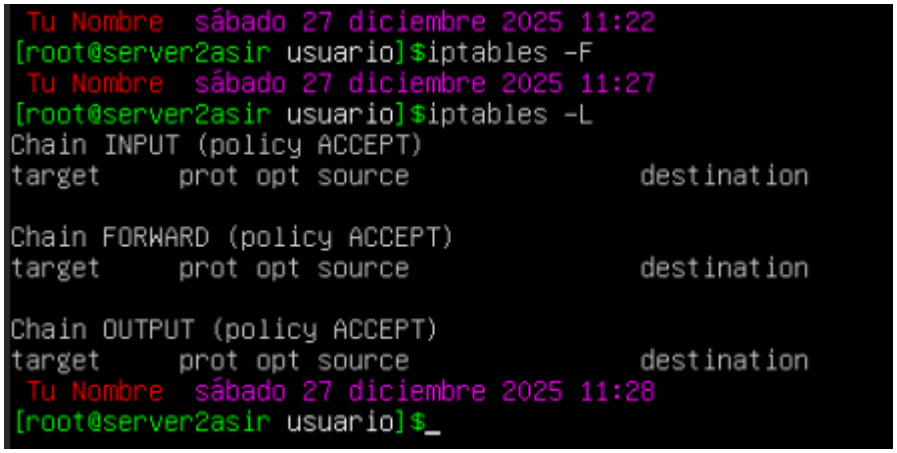
sudo iptables -A INPUT -s 80.100.30.27 -j DROP

"-s" creo que significa "source". El origen de los paquetes.

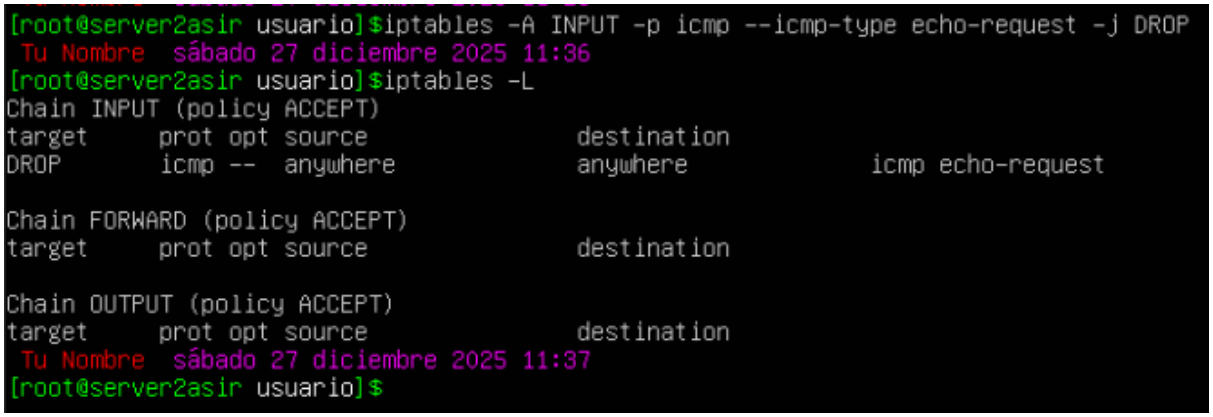


13. Borra todas las reglas introducidas y añade una para que no se pueda hacer ping a nuestro equipo.

iptables -F



sudo iptables -A INPUT -p icmp --icmp-type echo-request -j DROP



Antes:

```
Microsoft Windows [Versión 10.0.26100.1742]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Cristobal>ping 10.2.17.33

Haciendo ping a 10.2.17.33 con 32 bytes de datos:
Respuesta desde 10.2.17.33: bytes=32 tiempo=41ms TTL=62
Respuesta desde 10.2.17.33: bytes=32 tiempo=21ms TTL=62

Estadísticas de ping para 10.2.17.33:
    Paquetes: enviados = 2, recibidos = 2, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 21ms, Máximo = 41ms, Media = 31ms
Control-C
^C
```

Después:

```
C:\Users\Cristobal>ping 10.2.17.33

Haciendo ping a 10.2.17.33 con 32 bytes de datos:
Tiempo de espera agotado para esta solicitud.

Estadísticas de ping para 10.2.17.33:
    Paquetes: enviados = 1, recibidos = 0, perdidos = 1
    (100% perdidos),
Control-C
^C
C:\Users\Cristobal>
```

Para ver si la regla está activa y cuantas veces se ha usado:

sudo iptables -L INPUT -n -v

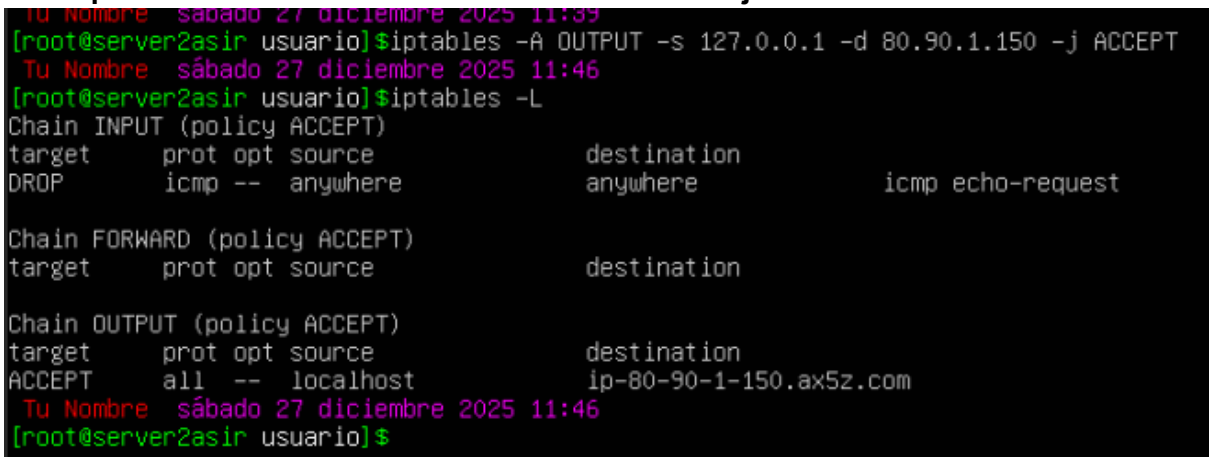
```
[root@server2asir usuario]$iptables -L INPUT -n -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    2   120 DROP       icmp --  *      *       0.0.0.0/0            0.0.0.0/0           icmptype 8
 Tu Nombre  sábado 27 diciembre 2025 11:39
[root@server2asir usuario]$
```

Otros métodos en:

https://askubuntu.com/questions/17548/how-can-i-block-ping-requests-with-iptables

14. Añade una regla para dejar pasar todos los paquetes que procedan de 127.0.0.1 (loopback) y vayan dirigidos al equipo 80.90.1.150

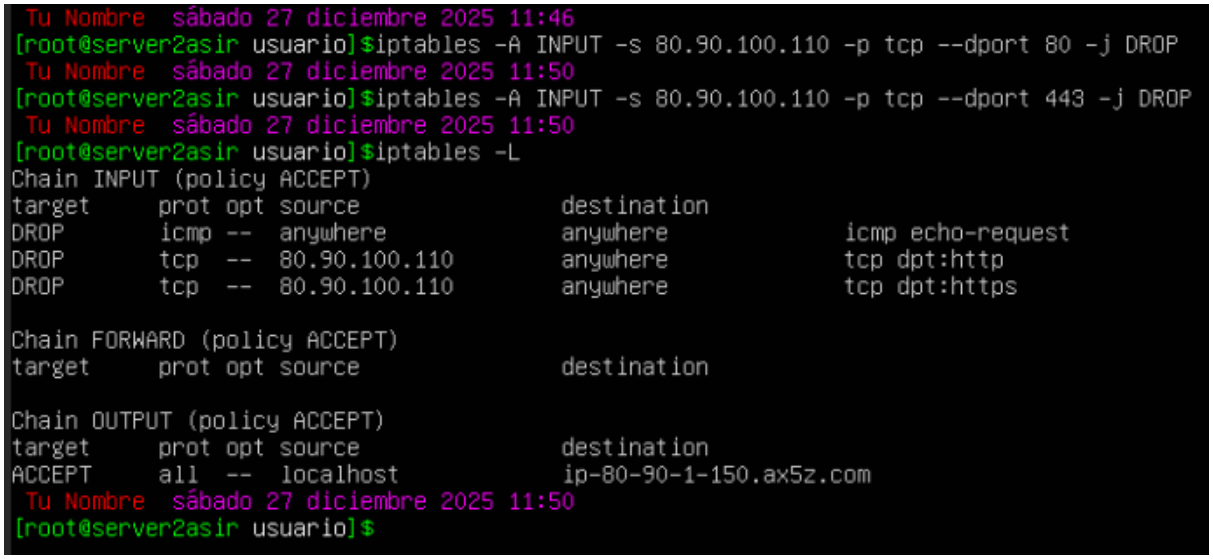
sudo iptables -A OUTPUT -s 127.0.0.1 -d 80.90.1.150 -j ACCEPT



15. No queremos permitir que el equipo con ip 80.90.100.110 se pueda conectar a nuestro servidor web. Añadir una regla para denegar el acceso al servidor web a ese equipo.

Como es servidor Web, especificamos los puertos 80 y 443:

iptables -A INPUT -s 80.90.100.110 -p tcp --dport 80 -j DROP
iptables -A INPUT -s 80.90.100.110 -p tcp --dport 443 -j DROP



16. Añade una regla para permitir el acceso vía ssh (puerto 22) únicamente al equipo con ip 90.80.70.60.

Primero debemos establecer el permiso:
iptables -A INPUT -s 90.80.70.60 -p tcp --dport 22 -j ACCEPT

Y luego bloquear al resto:
iptables -A INPUT -p tcp --dport 22 -j ACCEPT

```
 Tu Nombre  sábado 27 diciembre 2025 11:50
[root@server2asir usuario]$iptables -A INPUT -s 90.80.70.60 -p tcp --dport 22 -j ACCEPT
 Tu Nombre  sábado 27 diciembre 2025 11:56
[root@server2asir usuario]$iptables -A INPUT -p tcp --dport 22 -j DROP
 Tu Nombre  sábado 27 diciembre 2025 11:56
[root@server2asir usuario]$iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       icmp --  anywhere             anywhere             icmp echo-request
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:http
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:https
ACCEPT     tcp  --  60-70.80-90.static-ip.oleane.fr  anywhere             tcp dpt:ssh
DROP       tcp  --  anywhere             anywhere             tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  localhost            ip-80-90-1-150.ax5z.com
 Tu Nombre  sábado 27 diciembre 2025 11:56
[root@server2asir usuario]$
```

- Intenta conectarte a ssh y comprueba que es denegado

```
C:\Users\Cristobal>ssh usuario@10.2.17.33
ssh: connect to host 10.2.17.33 port 22: Connection timed out
```

- Mediante un comando de iptables, muestra de manera detallada todas las reglas que controlan el tráfico entrante a su servidor o máquina. Y visualiza donde aparece el número de paquetes que coincide con la regla anteriormente creada y como puedes aumentar ese número de paquetes.

Ver reglas que controlan tráfico entrante al servidor:

iptables -L INPUT

```
[root@server2asir usuario]$iptables -L INPUT
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       icmp --  anywhere             anywhere             icmp echo-request
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:http
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:https
ACCEPT     tcp  --  60-70.80-90.static-ip.oleane.fr  anywhere             tcp dpt:ssh
DROP       tcp  --  anywhere             anywhere             tcp dpt:ssh
 Tu Nombre  sábado 27 diciembre 2025 14:00
[root@server2asir usuario]$_
```

Si queremos que además muestre el número de paquetes que coinciden con cada regla, usamos:

sudo iptables -L INPUT -n -v --line-numbers

```
[root@server2asir usuario]$iptables -L INPUT -n -v --line-numbers
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source               destination
1        2   120 DROP       icmp --  *      *       0.0.0.0/0            0.0.0.0/0           icmptype 8
2        0     0 DROP       tcp  --  *      *       80.90.100.110        0.0.0.0/0           tcp dpt:80
3        0     0 DROP       tcp  --  *      *       80.90.100.110        0.0.0.0/0           tcp dpt:443
4        0     0 ACCEPT     tcp  --  *      *       90.80.70.60          0.0.0.0/0           tcp dpt:22
5       36  5388 DROP       tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:22
 Tu Nombre  sábado 27 diciembre 2025 14:06
```

“-n” muestra las ips en vez de nombres de direcciones (anywhere, asir2server, tc).
“-v” es de “verbose”: Muestra los contadores de tráfico.
“--line-numbers” Añade un índice numérico a cada regla (“num”), facilitando la gestión.

En la imagen se ve que actualmente tiene 36 paquetes. Si queremos aumentarlo, solo hay que intentar conectarse de nuevo por SSH.

```
[root@server2asir usuario]$iptables -L INPUT -n -v --line-numbers
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source               destination
1        2   120 DROP       icmp --  *      *       0.0.0.0/0            0.0.0.0/0           icmptype 8
2        0     0 DROP       tcp  --  *      *       80.90.100.110        0.0.0.0/0           tcp dpt:80
3        0     0 DROP       tcp  --  *      *       80.90.100.110        0.0.0.0/0           tcp dpt:443
4        0     0 ACCEPT     tcp  --  *      *       90.80.70.60          0.0.0.0/0           tcp dpt:22
5       43  5752 DROP       tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:22
```

17. Añade una regla cuya función sea hacer ping desde nuestra propia máquina (protocolo icmp).

sudo iptables -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT

Recuerda que previamente hemos prohibido responder a los pings. Ahora debemos permitirlo, pero solo de reply.

sudo iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT

De esta manera, si podemos hacer ping al exterior:

```
 Tu Nombre  sábado 27 diciembre 2025 14:23
[root@server2asir usuario]$iptables -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT
 Tu Nombre  sábado 27 diciembre 2025 14:24
[root@server2asir usuario]$iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
 Tu Nombre  sábado 27 diciembre 2025 14:24
[root@server2asir usuario]$iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       icmp --  anywhere             anywhere             icmp echo-request
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:http
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:https
ACCEPT     tcp  --  60-70.80-90.static-ip.oleane.fr  anywhere             tcp dpt:ssh
DROP       tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     icmp --  anywhere             anywhere             icmp echo-reply

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  localhost            ip-80-90-1-150.ax5z.com
ACCEPT     icmp --  anywhere             anywhere             icmp echo-request
 Tu Nombre  sábado 27 diciembre 2025 14:24
[root@server2asir usuario]$ping x.uk
PING x.uk (185.249.71.213) 56(84) bytes of data.
64 bytes from 185.249.71.213 (185.249.71.213): icmp_seq=1 ttl=49 time=36.8 ms
64 bytes from 185.249.71.213 (185.249.71.213): icmp_seq=2 ttl=49 time=36.7 ms
64 bytes from 185.249.71.213 (185.249.71.213): icmp_seq=3 ttl=49 time=36.3 ms
64 bytes from 185.249.71.213 (185.249.71.213): icmp_seq=4 ttl=49 time=36.6 ms
64 bytes from 185.249.71.213 (185.249.71.213): icmp_seq=5 ttl=49 time=36.6 ms
64 bytes from 185.249.71.213 (185.249.71.213): icmp_seq=6 ttl=49 time=36.5 ms
```

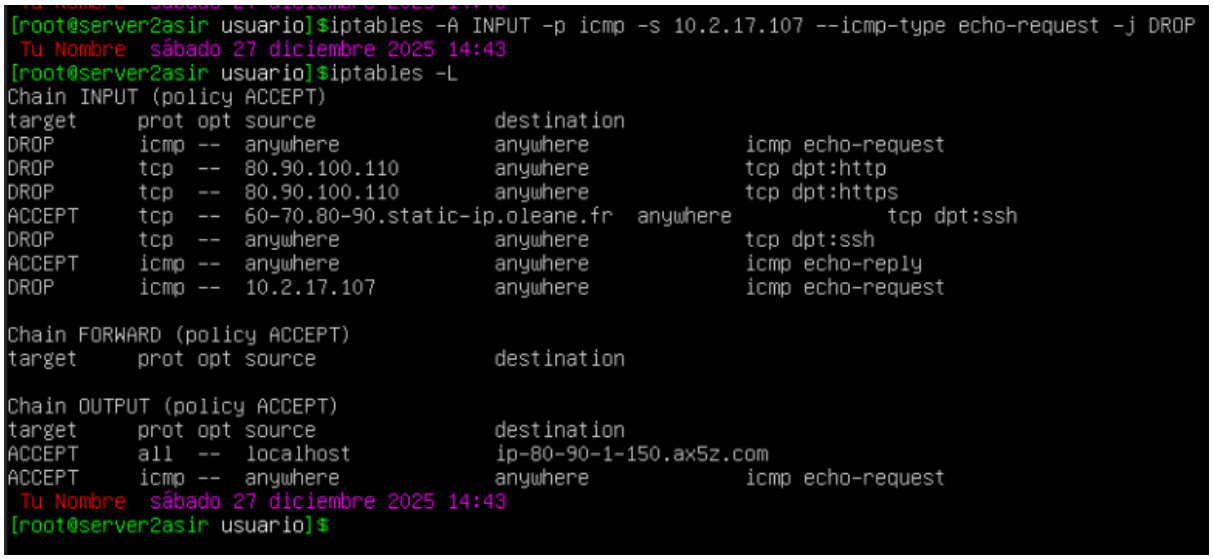
Pero desde fuera no nos pueden hacer ping:

```
C:\Users\Cristobal>ping 10.2.17.33

Haciendo ping a 10.2.17.33 con 32 bytes de datos:
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.

Estadísticas de ping para 10.2.17.33:
    Paquetes: enviados = 4, recibidos = 0, perdidos = 4
    (100% perdidos),
```

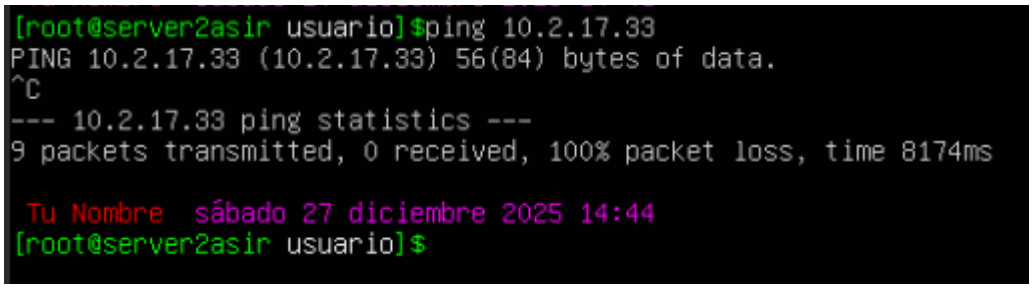
18. Añade una regla para que el ordenador que tienes situado a tu derecha no obtenga respuesta al hacer ping.

Imaginamos que el otro ordenador tiene la ip 10.2.17.107.

sudo iptables -A INPUT -p icmp -s 10.2.17.107 --icmp-type echo-request -j DROP



Y por lo tanto no puede hacer ping a nuestro equipo.



19. Añade una regla para impedir conectarse mediante FTP al ordenador con IP 192.168.1.15.

Usamos[6]:
iptables -A INPUT -p tcp -s 192.168.1.15 --dport 21 -j DROP

```
 Tu Nombre  sábado 27 diciembre 2025 14:51
[root@server2asir usuario]$iptables -A INPUT -p tcp -s 192.168.1.15 --dport 21 -j DROP
 Tu Nombre  sábado 27 diciembre 2025 14:56
[root@server2asir usuario]$iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       icmp --  anywhere             anywhere             icmp echo-request
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:http
DROP       tcp  --  80.90.100.110        anywhere             tcp dpt:https
ACCEPT     tcp  --  60-70.80-90.static-ip.oleane.fr  anywhere             tcp dpt:ssh
DROP       tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     icmp --  anywhere             anywhere             icmp echo-reply
DROP       icmp --  10.2.17.107          anywhere             icmp echo-request
DROP       tcp  --  192.168.1.15         anywhere             tcp dpt:ftp

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  localhost            ip-80-90-1-150.ax5z.com
ACCEPT     icmp --  anywhere             anywhere             icmp echo-request
 Tu Nombre  sábado 27 diciembre 2025 14:57
[root@server2asir usuario]$
```

[6] https://www.24x7serverguard.com/blog/ftp-issues/how-to-block-ftp-access-using-iptables-and-csf-firewall/

20. Crea un pequeño script que contemple los siguientes casos:

- La política por defecto es no dejar entrar ningún paquete al equipo.
- Se permite el acceso al servidor web desde el exterior.
- Se permite acceder al servidor FTP.
- Se permite acceder al servidor SSH desde el exterior.

Creamos el script en "**/etc**":

nano /etc/iptables_custom_rules

Nos basamos en el ejemplo[7]:

 # !/bin/bash

 # Servidor Web
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

 # Servidor FTP
sudo iptables -A INPUT -p tcp --dport 21 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 56123 -j ACCEPT

 # Servicio SSH
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT

 # Se debe poner el último porque lo bloquea todo.
iptables -A INPUT -j DROP

Este script puede ser bastante restrictivo y en algunos sitios se indica que se incluyan estos comandos antes del DROP:

 # Permitir tráfico local (loopback)
iptables -A INPUT -i lo -j ACCEPT

 # Permitir conexiones ya establecidas o relacionadas
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

[7] https://www.researchgate.net/figure/ptables-Example-Shell-Script_fig3_27475149

```
 GNU nano 6.2                                    /etc/iptables_custom_rules *
# !/bin/bash

# Servidor Web
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

# Servidor FTP
sudo iptables -A INPUT -p tcp --dport 21 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 56123 -j ACCEPT

# Servicio SSH
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT

# Permitir tráfico local (loopback)
iptables -A INPUT -i lo -j ACCEPT

# Permitir conexiones ya establecidas o relacionadas
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Se debe poner el último porque lo bloquea todo.
iptables -A INPUT -j DROP
```

Ahora le damos permisos al archivo para que pueda ejecutarse:

chmod +x /etc/iptables_custom_rules

```
[root@server2asir usuario]$chmod +x /etc/iptables_custom_rules
 Tu Nombre  sábado 27 diciembre 2025 15:48
[root@server2asir usuario]$ls -l /etc/iptables_custom_rules
-rwxr-xr-x 1 root root 580 dic 27 15:47 /etc/iptables_custom_rules
```

Y ahora debemos añadir la ruta del script en **/etc/rc.local** para que se ejecute al iniciar el sistema.

nano /etc/rc.local

```
 GNU nano 6.2
#!/bin/sh -e
# rc.local
/etc/iptables_custom_rules
exit 0
```

También debemos darle permisos de ejecución:

chmod +x /etc/rc.local

```
[root@server2asir usuario]$chmod +x /etc/rc.local
 Tu Nombre  sábado 27 diciembre 2025 15:58
[root@server2asir usuario]$ls -l /etc/rc.local
-rwxr-xr-x 1 root root 58 dic 27 15:51 /etc/rc.local
 Tu Nombre  sábado 27 diciembre 2025 15:58
[root@server2asir usuario]$
```

Activar el servicio de compatibilidad (Systemd): Ejecutamos estos comandos para decirle al sistema que use **rc.local** al arrancar:

sudo systemctl enable rc-local

```
 Tu Nombre  sábado 27 diciembre 2025 15:58
[root@server2asir usuario]$sudo systemctl enable rc-local
The unit files have no installation config (WantedBy=, RequiredBy=, Also=,
Alias= settings in the [Install] section, and DefaultInstance= for template
units). This means they are not meant to be enabled using systemctl.

Possible reasons for having this kind of units are:
• A unit may be statically enabled by being symlinked from another unit's
  .wants/ or .requires/ directory.
• A unit's purpose may be to act as a helper for some other unit which has
  a requirement dependency on it.
• A unit may be started when needed via activation (socket, path, timer,
  D-Bus, udev, scripted systemctl call, ...).
• In case of template units, the unit is meant to be enabled with some
  instance name specified.
 Tu Nombre  sábado 27 diciembre 2025 15:59
[root@server2asir usuario]$
```

sudo systemctl start rc-local
sudo systemctl status rc-local

```
 Tu Nombre  sábado 27 diciembre 2025 15:59
[root@server2asir usuario]$sudo systemctl start rc-local
 Tu Nombre  sábado 27 diciembre 2025 15:59
[root@server2asir usuario]$sudo systemctl status rc-local
● rc-local.service - /etc/rc.local Compatibility
     Loaded: loaded (/lib/systemd/system/rc-local.service; enabled-runtime; vendor preset: enabled)
    Drop-In: /usr/lib/systemd/system/rc-local.service.d
             └─debian.conf
     Active: active (exited) since Sat 2025-12-27 15:59:52 UTC; 5s ago
       Docs: man:systemd-rc-local-generator(8)
    Process: 1141 ExecStart=/etc/rc.local start (code=exited, status=0/SUCCESS)
        CPU: 83ms

dic 27 15:59:52 server2asir sudo[1152]:     root : PWD=/ ; USER=root ; COMMAND=/usr/sbin/iptables -A INPUT -p tcp --dport 21 -j ACCEPT
dic 27 15:59:52 server2asir sudo[1152]: pam_unix(sudo:session): session opened for user root(uid=0) by (uid=0)
dic 27 15:59:52 server2asir sudo[1152]: pam_unix(sudo:session): session closed for user root
dic 27 15:59:52 server2asir sudo[1154]:     root : PWD=/ ; USER=root ; COMMAND=/usr/sbin/iptables -A INPUT -p tcp --dport 56123 -j ACCEPT
dic 27 15:59:52 server2asir sudo[1154]: pam_unix(sudo:session): session opened for user root(uid=0) by (uid=0)
dic 27 15:59:52 server2asir sudo[1154]: pam_unix(sudo:session): session closed for user root
dic 27 15:59:52 server2asir sudo[1156]:     root : PWD=/ ; USER=root ; COMMAND=/usr/sbin/iptables -A INPUT -p tcp --dport 22 -j ACCEPT
dic 27 15:59:52 server2asir sudo[1156]: pam_unix(sudo:session): session opened for user root(uid=0) by (uid=0)
dic 27 15:59:52 server2asir sudo[1156]: pam_unix(sudo:session): session closed for user root
dic 27 15:59:52 server2asir systemd[1]: Started /etc/rc.local Compatibility.
 Tu Nombre  sábado 27 diciembre 2025 15:59
[root@server2asir usuario]$
```

Si reiniciamos el servidor y comprobamos las **iptables**, veremos que están ahí:

```
Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»


Last login: Sat Dec 27 15:55:14 2025 from 10.8.3.254
 Tu Nombre  sábado 27 diciembre 2025 16:03
[usuario@server2asir ~]$sudo su
[sudo] password for usuario:
 Tu Nombre  sábado 27 diciembre 2025 16:03
[root@server2asir usuario]$iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ftp
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:56123
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
DROP       all  --  anywhere             anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
 Tu Nombre  sábado 27 diciembre 2025 16:03
[root@server2asir usuario]$
```