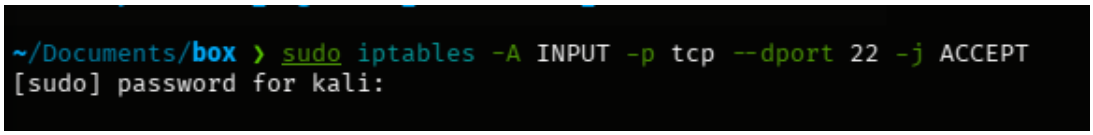
Ejercicios sencillos de iptables en Linux

Ejercicio 1 – Permitir SSH en el puerto 22

🎯 Objetivo: Asegurar que puedes conectarte por SSH desde tu equipo Windows o desde Kali a sí misma.

🔧 Comando:

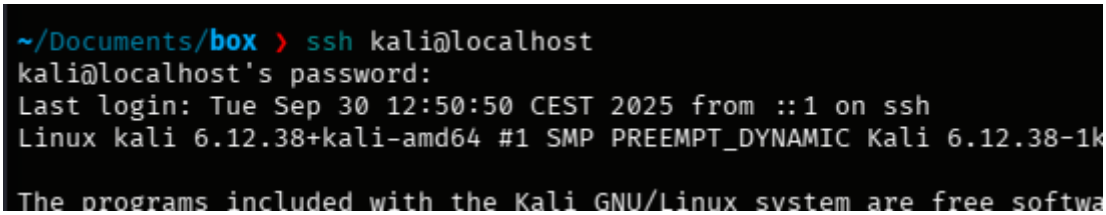
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT



🧪 Cómo probarlo:

Desde Kali:

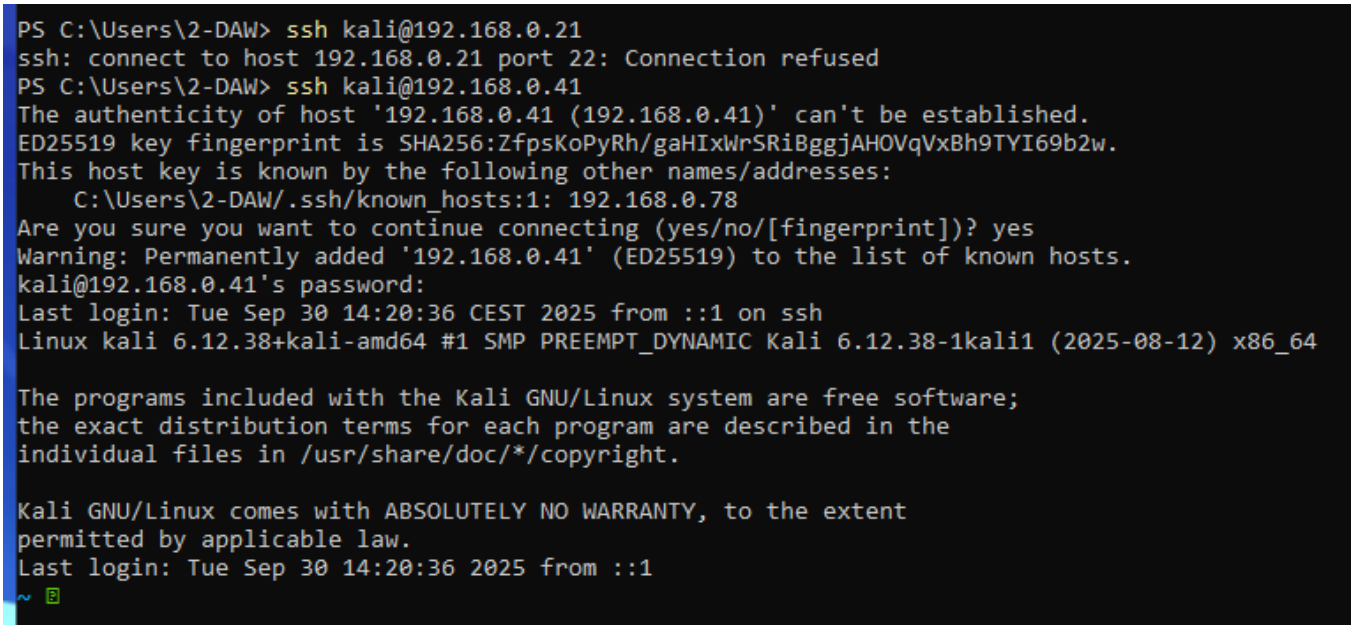
ssh kali@localhost



Desde Windows (PowerShell o CMD):

ssh kali@192.168.1.55

✅ Si la conexión es exitosa, la regla funciona.



🔹 Ejercicio 2 – Bloquear el puerto 23 (Telnet)

🎯 Objetivo: Simular bloqueo de servicios no utilizados como Telnet.

🔧 Comando:

sudo iptables -A INPUT -p tcp --dport 23 -j DROP

🧪 Cómo probarlo:

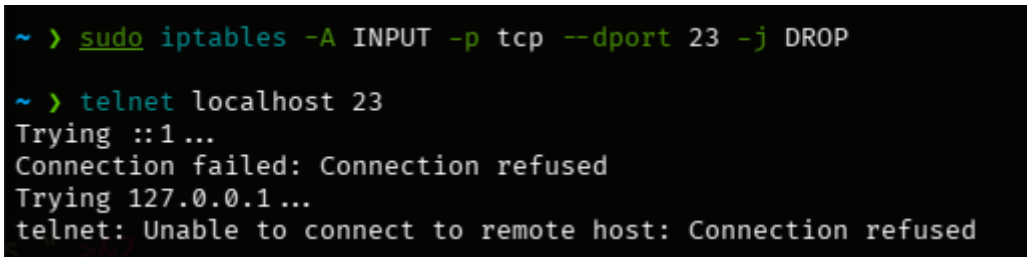
En Kali, install Telnet:

sudo apt install telnet -y

Ejecuta:

telnet localhost 23

❌ Verás que no responde (se cuelga), porque el firewall lo está bloqueando.



🔹 Ejercicio 3 – Permitir tráfico HTTP (puerto 80)

🎯 Objetivo: Habilitar el acceso web desde el navegador de Windows al servidor Apache de Kali.

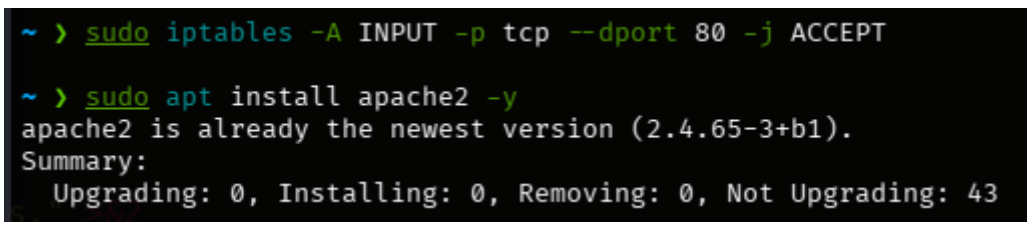
🔧 Comando:

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

🧪 Cómo probarlo:

Instala Apache en Kali:

sudo apt install apache2 -y



Desde Windows, abre el navegador y visita:

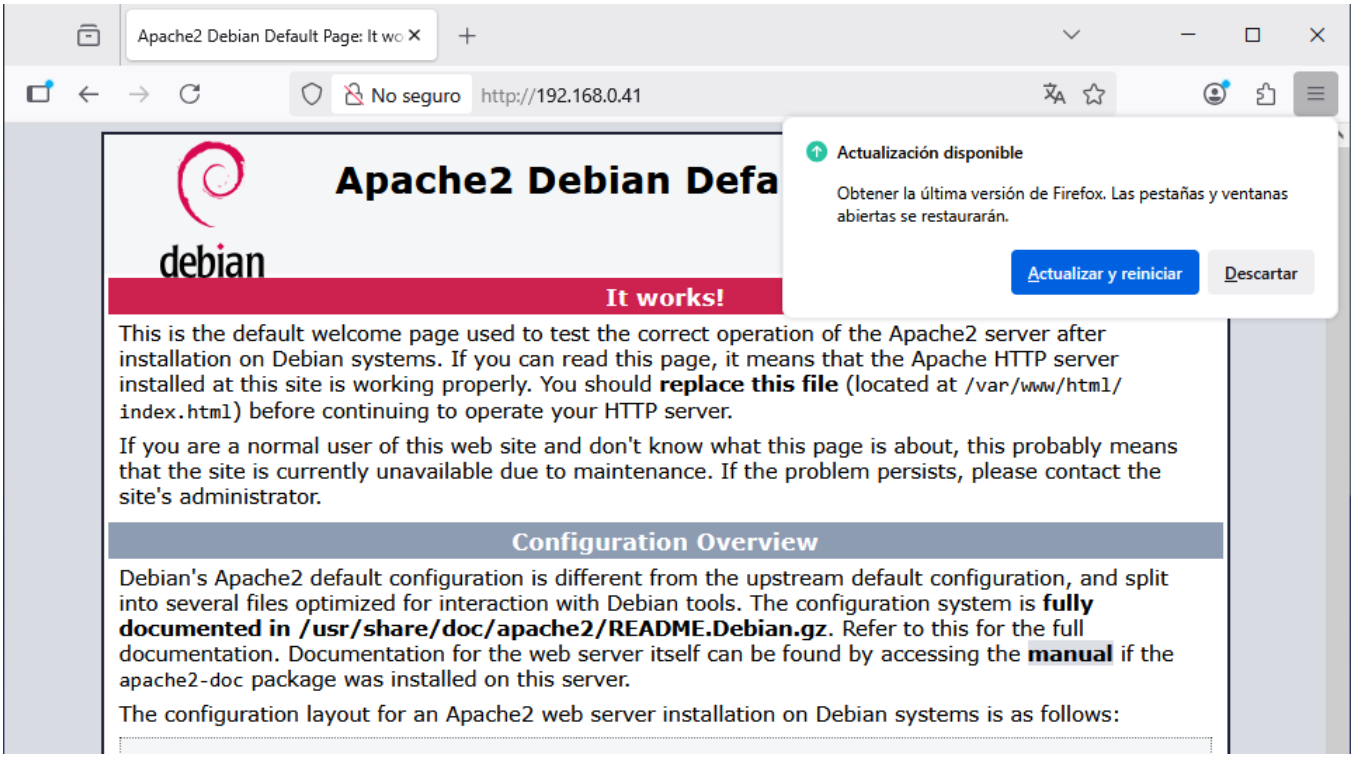
http://192.168.1.55

✅ Si se carga la página, el puerto está abierto.

```
~ ❯ curl localhost:80

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//
dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; cha
    <title>Apache2 Debian Default Page: It works</title>
    <style type="text/css" media="screen">
  * {
    margin: 0px 0px 0px 0px;
    padding: 0px 0px 0px 0px;
  }

  body, html {
    padding: 3px 3px 3px 3px;
```

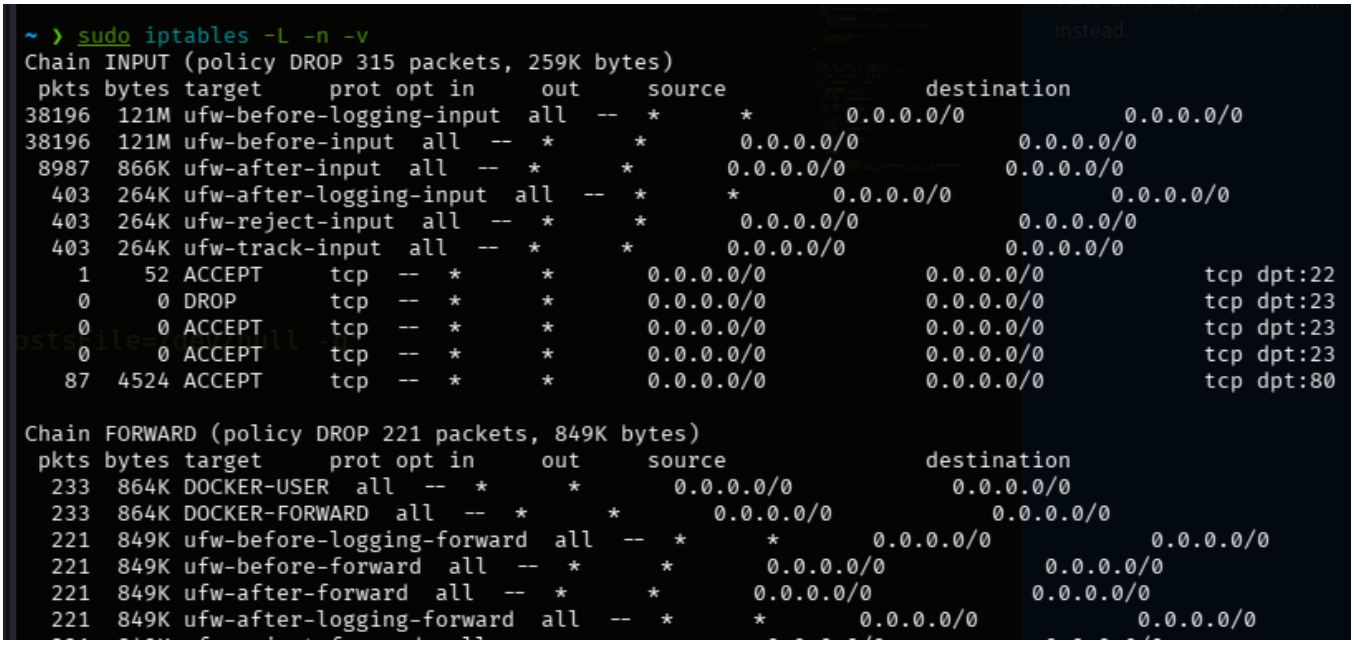


🔹 Ejercicio 4 – Ver todas las reglas activas

🎯 Objetivo: Ver qué reglas están en uso y cuántas veces se han aplicado.

🔧 Comando:

sudo iptables -L -n -v



🔎 Verás columnas como:

pkts → paquetes que han coincidido con esa regla

bytes → tráfico acumulado

target → lo que hace (ACCEPT, DROP…)

dpt: → puerto destino (como 22 o 80)

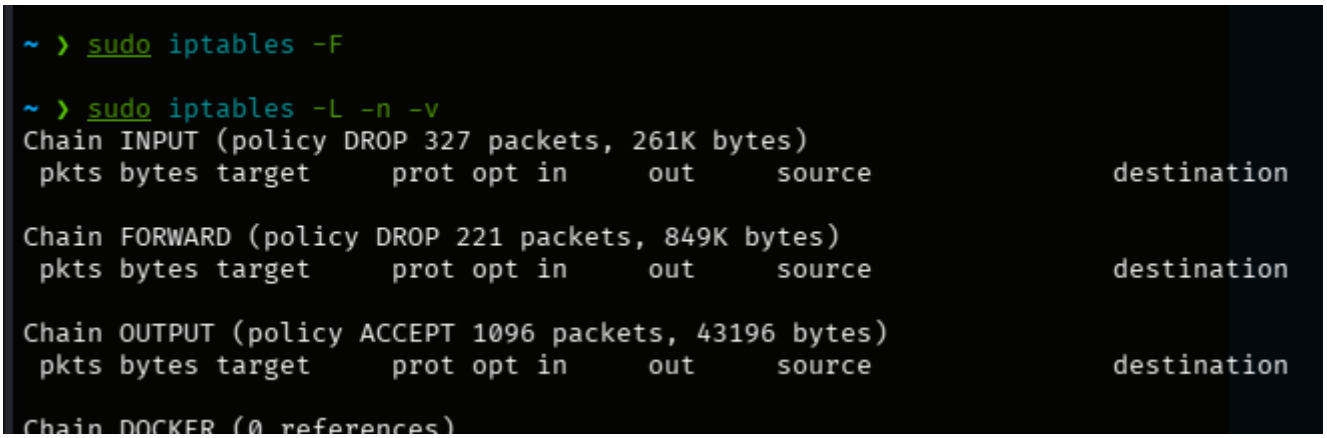
🔹 Ejercicio 5 – Borrar todas las reglas

🎯 Objetivo: Dejar el sistema sin restricciones para reiniciar las pruebas.

🔧 Comando:

sudo iptables -F

✅ Después de esto, todo el tráfico está permitido (por defecto en Kali). Puedes verificarlo intentando Telnet o SSH sin reglas previas.



🔹 Ejercicio 6 – Crear y usar una cadena personalizada

🎯 Objetivo: Organizar las reglas en una cadena propia (por ejemplo, "MI_CADENA").

🔧 Comandos:

sudo iptables -N MI_CADENA

sudo iptables -A INPUT -j MI_CADENA

sudo iptables -A MI_CADENA -p tcp --dport 443 -j ACCEPT

```
~ ❯ sudo iptables -N MI_CADENA
~ ❯ sudo iptables -A INPUT -j MI_CADENA
```

```
~ ❯ sudo iptables -A MI_CADENA -p tcp --dport 443 -j ACCEPT
```

🧪 Cómo probarlo:

Activa un servicio en el puerto 443 (HTTPS) o prueba con curl si tienes configurado algo:

curl -v https://localhost

✅ Verás que el tráfico pasa por tu cadena personalizada.

```
~ ❯ curl -v https://localhost
* Host localhost:443 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
*   Trying [::1]:443...
* connect to ::1 port 443 from ::1 port 52210 failed: Connection refused
*   Trying 127.0.0.1:443...
^C

~ ❯ curl -v http://localhost
* Host localhost:80 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
*   Trying [::1]:80...
* Connected to localhost (::1) port 80
* using HTTP/1.x
> GET / HTTP/1.1
> Host: localhost
> User-Agent: curl/8.15.0
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200 OK
< Date: Tue, 30 Sep 2025 12:41:54 GMT
< Server: Apache/2.4.65 (Debian)
< Last-Modified: Thu, 29 May 2025 19:23:17 GMT
< ETag: "29cf-6364b3ad32860"
< Accept-Ranges: bytes
< Content-Length: 10703
< Vary: Accept-Encoding
< Content-Type: text/html
<
```

🔹 Ejercicio 7 – Registrar intentos de conexión a SSH en logs

🎯 Objetivo: Ver en los logs del sistema cada vez que alguien intenta usar SSH.

🔧 Comando:

sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "Intento SSH: "

🧪 Cómo probarlo:

Conéctate a Kali desde Windows o desde otra terminal:

ssh kali@localhost

En otra ventana de terminal, ejecuta:

sudo journalctl -f

✅ Verás una línea con el prefijo Intento SSH: indicando que se ha detectado un intento de conexión.