**Configurar reglas básicas para SSH, HTTP y HTTPS**

Para esta practica usaremos iptables que lo tenemos instalado de otros proyectos en nuestra maquina Debian, para configurar y agregar reglas básicas aceptando tráfico específico esencial para el funcionamiento del servidor web (HTTP, HTTPS) y para la administración remota segura (SSH), mientras bloqueas todo el tráfico no autorizado, incrementando así la seguridad del servidor.

* Bloquea todas las conexiones entrantes no autorizadas y permite las conexiones salientes.

*# Bloquea todas las conexiones entrantes no autorizadas*



*# Permite conexiones salientes*



* Añade una regla para permitir tráfico SSH (puerto 22)



* Añade una regla para permitir tráfico HTTP (puerto 80)



* Añade una regla para permitir tráfico HTTPS (puerto 443)



* Añade una regla para bloquear todo el tráfico que no coincide con las reglas anteriores.



* Asegurate de guardar la configuración que acabas de ingresar.



**Configuración de Reglas Avanzadas con iptables**

Las reglas de iptables también pueden determinar cómo manejar diferentes tipos de tráfico de red basándose en criterios como dirección IP, puerto, protocolo, y estado de la conexión y se utiliza principalmente para la seguridad de la red, controlando el tráfico que entra y sale de un sistema.

Para lograr resultados similares a los que se logran con las listas de control de acceso (ACLs) en dispositivos Cisco (aunque la sintaxis es diferente), podemos definir en nuestro servidor con wordpress, reglas que permiten o bloquean tráfico basadas en direcciones IP y máscaras de red con iptables.

Por ejemplo para permitir tráfico desde una red específica 192.168.1.0/24 (equivalente a una máscara de 255.255.255.0) y denegar el resto, de manera similar a lo que harías con una ACL estándar de Cisco podríamos hacer lo siguiente:

*# permitir el tráfico desde la red 192.168.1.0/24*



*# denegar todo el tráfico que no coincida con las reglas anteriores*



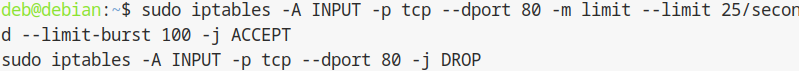
* Añade una regla para limitar nuevas conexiones al puerto 80 para proteger contra ataques DDoS.

**1.-Configura la regla con iptables**

**Establece un límite de conexiones por IP**: La siguiente regla permite un máximo de 10 conexiones por minuto desde una misma IP. Si la IP excede este límite, las conexiones adicionales serán descartadas.



**2.-Configura un límite de conexiones por segundo**: Para evitar un alto volumen de conexiones por segundo, puedes establecer una regla adicional que permita hasta 25 conexiones por segundo, con un burst de hasta 100.

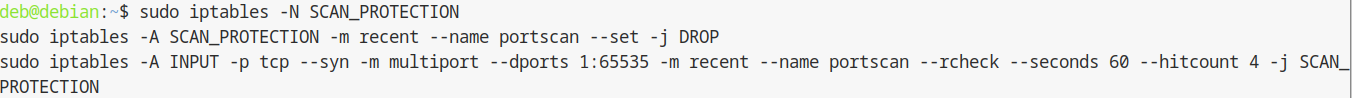


**3.- Guardar las Reglas**

****

* **Añade una regla para protección contra escaneo de puertos y ataques SYN Flood.**

**1.-Protección contra escaneo de puertos**

Esta regla detecta intentos de escaneo y bloquea múltiples conexiones entrantes en poco tiempo:

**2.-Protección contra ataques SYN Flood**

****

**3.-Guardar las Reglas**

****

**Validando trabajo**

**A screenshot of a computer

Description automatically generated**

**1. Ejecuta el script**

Una vez pegado el script [check-rules.sh](https://github.com/breatheco-de/configuring-firewall-and-acl-exercise-tutorial/blob/main/.learn/assets/check-rules.sh) en tu maquina Debian, abre la terminal y navega al directorio donde se encuentra el script, en nuestro caso ./Escritorio y haz que el script sea ejecutable (si aún no lo es). Esto se puede hacer usando el comando chmod:

****

**A screenshot of a computer program

Description automatically generatedEjecuta el script especificando su nombre. También pueden proporcionar cualquier argumento necesario. Asumiendo que no se necesitan argumentos adicionales para este ejemplo, deben ejecutar:**