🛠️ Herramienta Previa: Netcat (el "Cuchillo Suizo" de TCP/IP)

¿Qué es? `netcat` (comando `nc`) es una utilidad de red que permite leer y escribir datos a través de conexiones de red usando protocolos TCP o UDP.

¿Para qué sirve? * **Depuración:** Comprobar si un puerto está abierto.

- **Transferencia de archivos:** Enviar datos entre equipos.
- **Escaneo de puertos:** De forma básica.
- **Backdoors de prueba:** Ejecutar comandos remotamente (opción `-e`).

Instalación:

- **Ubuntu:** `sudo apt install netcat-openbsd -y`
- **Rocky Linux:** `sudo dnf install nmap-ncat -y`

🔥 Ejercicio 1: UFW en Ubuntu 24.04 (Puerto 4444)

🎯 **OBJETIVO:** Abrir un puerto específico para que Netcat responda con la fecha del sistema.

📋 **PASO A PASO:**

1. **En Ubuntu**, lanza Netcat para que escuche en el puerto `4444` y ejecute `date`:

   ```
   # Nota: En algunas versiones de Ubuntu, por seguridad, -e no está disponible.
   # Usamos un "pipe" como alternativa moderna:
   mkfifo /tmp/f; nc -lk 4444 0</tmp/f | /bin/date >/tmp/f
   ```

2. **Desde Rocky**, intenta conectar: `nc ubuntu-client 4444`. (Fallará porque el firewall está activo).

3. **En Ubuntu**, abre el puerto:

   ```
   sudo ufw allow 4444/tcp
   sudo ufw status
   ```

4. **Prueba final:** Repite la conexión desde Rocky y verás la fecha del servidor Ubuntu.

🛡️ Ejercicio 2: Firewall-cmd en Rocky Linux 10 (Puerto 5555)

🎯 **OBJETIVO:** Usar la sintaxis de zonas de RHEL para abrir un servicio de Netcat.

📋 **PASO A PASO:**

1. **En Rocky**, crea un proceso de escucha:

   ```
   ncat -l 5555 -e /usr/bin/uname -a
   ```

2. **En Rocky**, abre el puerto de forma permanente:

   ```
   sudo firewall-cmd --add-port=5555/tcp --permanent
   sudo firewall-cmd --reload
   ```

3. **Desde Ubuntu**, conecta:

   ```
   nc rocky-server 5555
   ```

💡 **EXPLICACIÓN:** `firewall-cmd` es el cliente para `firewalld`. La opción `--permanent` es vital; si no la usas, la regla desaparecerá al reiniciar el servicio o el equipo.

🚫 Ejercicio 3: Iptables - El dilema de ICMP (REJECT vs DROP)

Aquí bajamos al nivel del núcleo para entender cómo el firewall "rechaza" o "ignora" un paquete.

🎯 **OBJETIVO:** Bloquear el comando `ping` de dos formas distintas.

Caso A: REJECT (El "No" educado)

El firewall detiene el paquete y envía un mensaje de vuelta diciendo que el puerto es inalcanzable.

```
sudo iptables -A INPUT -p icmp --icmp-type echo-request -j REJECT
```

- **Resultado al hacer ping:** Recibirás un mensaje inmediato: *"Destination Port Unreachable"*. El cliente sabe que el servidor está ahí, pero le prohíbe el paso.

Caso B: DROP (El "Visto" o ignorar)

El firewall simplemente tira el paquete a la basura. No responde nada.

```
# Limpiamos la regla anterior primero
sudo iptables -D INPUT -p icmp --icmp-type echo-request -j REJECT

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

- **Resultado al hacer ping:** El comando se quedará colgado hasta que dé un *"Time Out"*. Es más seguro para evitar que escaneen tu red, ya que parece que la IP no existe.

💡 **NOTA:** En Rocky Linux, para que `iptables` sea persistente, necesitas el paquete `iptables-services`. En Ubuntu, `iptables-persistent`.