



Server Exploits

Section 1 - Lab

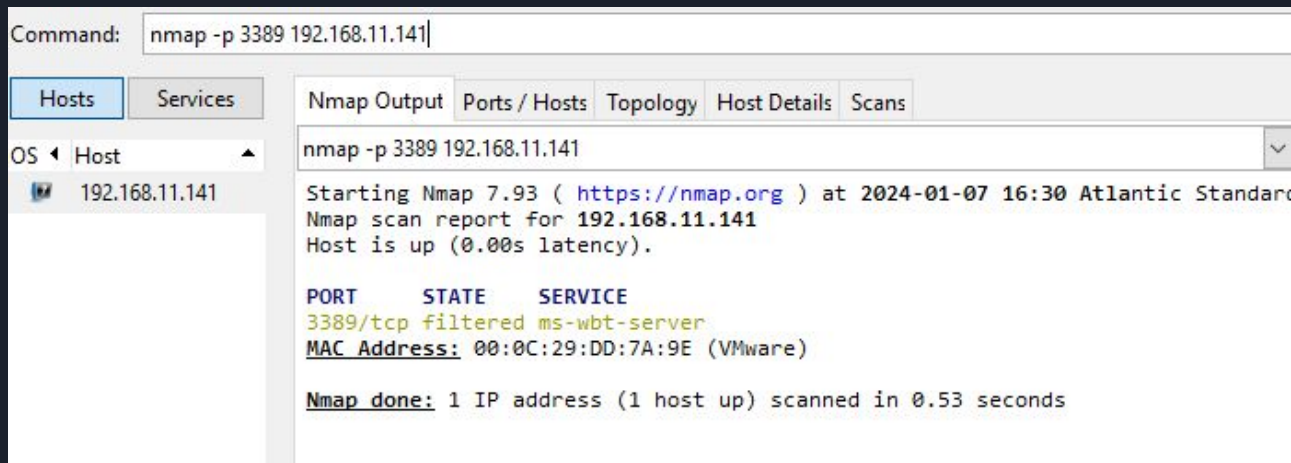


Server Setup

- Setup Windows Server 2019
 - 4 cores, 8GB RAM, 60GB storage
 - Install the Desktop Experience
- Local Admin
 - 12 character password
- Local User
 - Do not setup with a microsoft online account, keep it local
 - You can set up a local account through VMWare when you set up your machine
- Take a snapshot when you are done (will be helpful for the assignment)

Enabling Vulnerable RDP

Using Nmap from your kali or host machine on our freshly installed server, we can see that there is no service running on it on port 3389 (default RDP). It appears as filtered.



Command: `nmap -p 3389 192.168.11.141`

Hosts Services

OS Host

192.168.11.141

Nmap Output Ports / Hosts Topology Host Details Scans

`nmap -p 3389 192.168.11.141`

Starting Nmap 7.93 (<https://nmap.org>) at 2024-01-07 16:30 Atlantic Standard
Nmap scan report for 192.168.11.141
Host is up (0.00s latency).

PORT	STATE	SERVICE
3389/tcp	filtered	ms-wbt-server

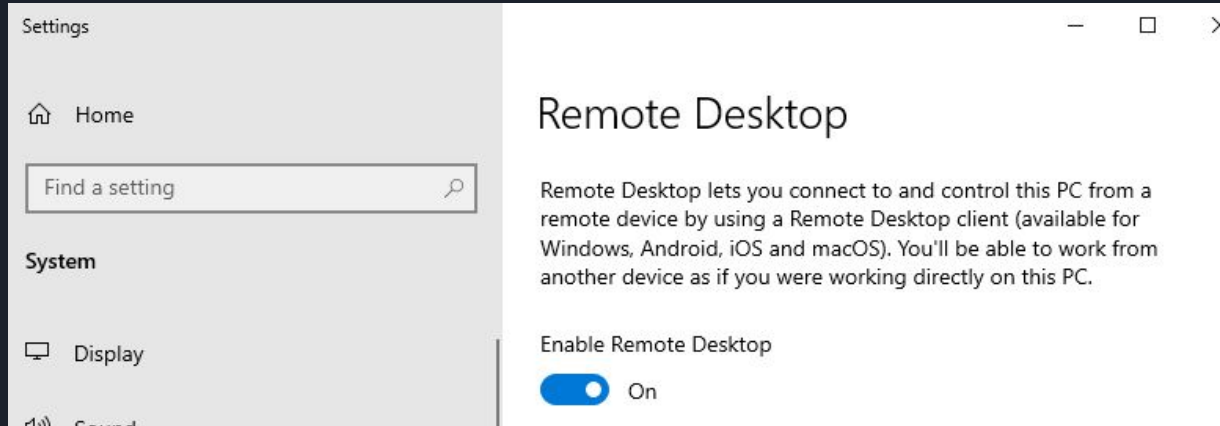
MAC Address: 00:0C:29:DD:7A:9E (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.53 seconds

Enabling Vulnerable RDP

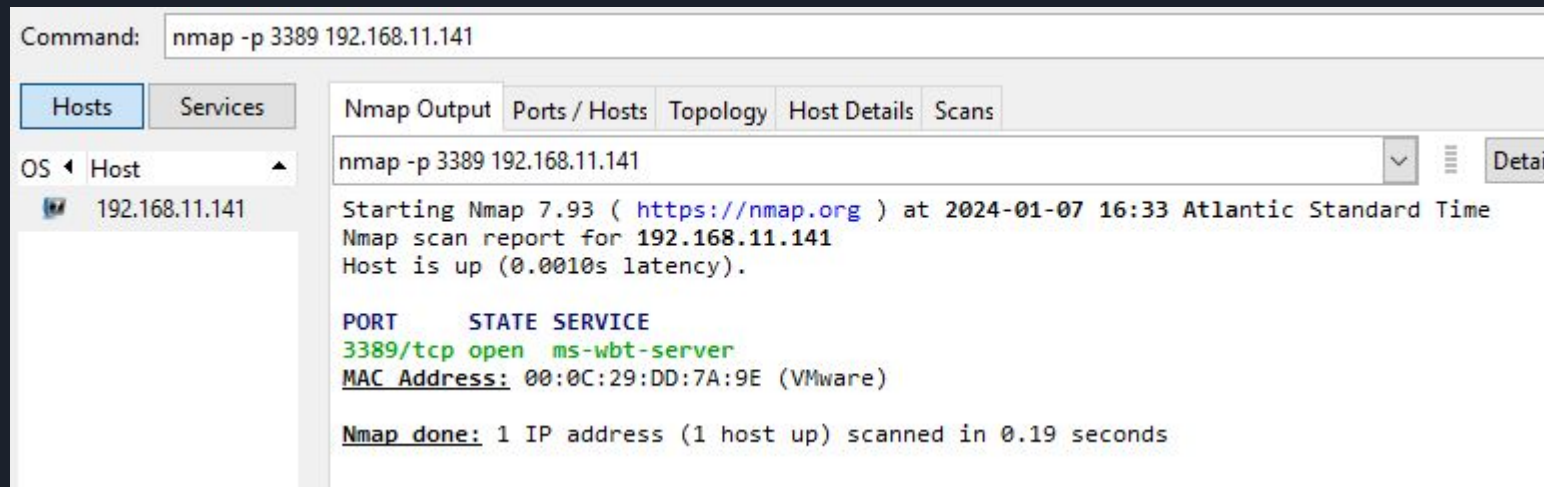
We first need to enable Remote Desktop on your server. This can be found in the settings by searching for Remote Desktop Settings. This will:

1. Enable the RDP Service
2. Configure the Windows Firewall to allow incoming connections on port 3389



Enabling Vulnerable RDP

We should now see on nmap that the service is open. You should not be able to connect via RDP to your host machine.



Command: `nmap -p 3389 192.168.11.141`

Hosts Services

OS Host

192.168.11.141

Nmap Output Ports / Hosts Topology Host Details Scans

`nmap -p 3389 192.168.11.141`

Starting Nmap 7.93 (<https://nmap.org>) at 2024-01-07 16:33 Atlantic Standard Time
Nmap scan report for 192.168.11.141
Host is up (0.0010s latency).

PORT	STATE	SERVICE
3389/tcp	open	ms-wbt-server

MAC Address: 00:0C:29:DD:7A:9E (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds

Enabling Vulnerable RDP

We can also take a look at what SSL ciphers/protocols the server accepts. Note TLS1.0 and the 3DES cipher. These are considered weak.

```
hmap -p 3389 --script ssl-enum-ciphers 192.168.11.141
```

Services

Nmap Output

Ports / Hosts

Topology

Host Details

Scans

nmap -p 3389 --script ssl-enum-ciphers 192.168.11.141

Starting Nmap 7.93 (<https://nmap.org>) at 2024-01-07 16:36 Atlantic S
NSOCK ERROR [0.0300s] ssl_init_helper(): OpenSSL legacy provider failed

Nmap scan report for 192.168.11.141
Host is up (0.00088s latency).

PORT	STATE	SERVICE
3389/tcp	open	ms-wbt-server

ssl-enum-ciphers:

TLSv1.0:

ciphers:

- TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
- TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
- TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 2048) - C
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp384r1) - A
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A

compressors:

NULL

cipher preference: server

warnings:

- 64-bit block cipher 3DES vulnerable to SWEET32 attack

TLSv1.1:

Enabling Vulnerable RDP

The rdp-ntlm-info script also can tell us a lot of information about the server that can help an attacker.

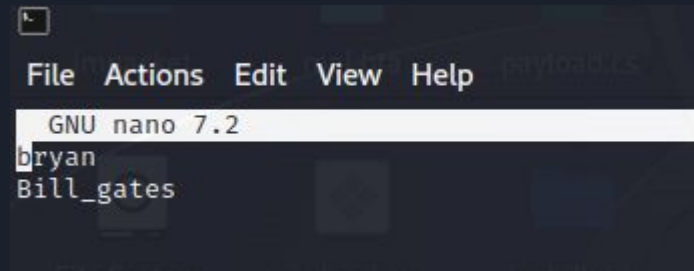
```
nmap -p 3389 --script rdp-ntlm-info 192.168.11.141
```

Services	Nmap Output	Ports / Hosts	Topology	Host Details	Scans
192.168.11.141	<pre>nmap -p 3389 --script rdp-ntlm-info 192.168.11.141 Starting Nmap 7.93 (https://nmap.org) at 2024-01-07 16:37 Atlantic S NSOCK ERROR [0.0310s] ssl_init_helper(): OpenSSL legacy provider failed Nmap scan report for 192.168.11.141 Host is up (0.00013s latency). PORT STATE SERVICE 3389/tcp open ms-wbt-server rdp-ntlm-info: Target_Name: WIN-PK1DU24G5DS NetBIOS_Domain_Name: WIN-PK1DU24G5DS NetBIOS_Computer_Name: WIN-PK1DU24G5DS DNS_Domain_Name: WIN-PK1DU24G5DS DNS_Computer_Name: WIN-PK1DU24G5DS Product_Version: 10.0.20348 _ System_Time: 2024-01-07T20:37:22+00:00 MAC Address: 00:0C:29:DD:7A:9E (VMware)</pre>				



Brute Force Attack

By default the RDP service will allow for a brute force attack. We can use a tool on Kali called Hydra to launch a brute force attack. We will first need a list of usernames to try. You can create a list of usernames on kali with “nano user.txt” and write out a couple usernames, including the actual username you've created for your Windows server. Use Ctrl+x then Y+Enter to save the file.



```
File Actions Edit View Help
GNU nano 7.2
bryan
Bill_gates
```




Fixing RDP

1. Lockout Policy
2. Encryption
3. Disable Copy/paste
4. Disable restarts

Brute Force Attack

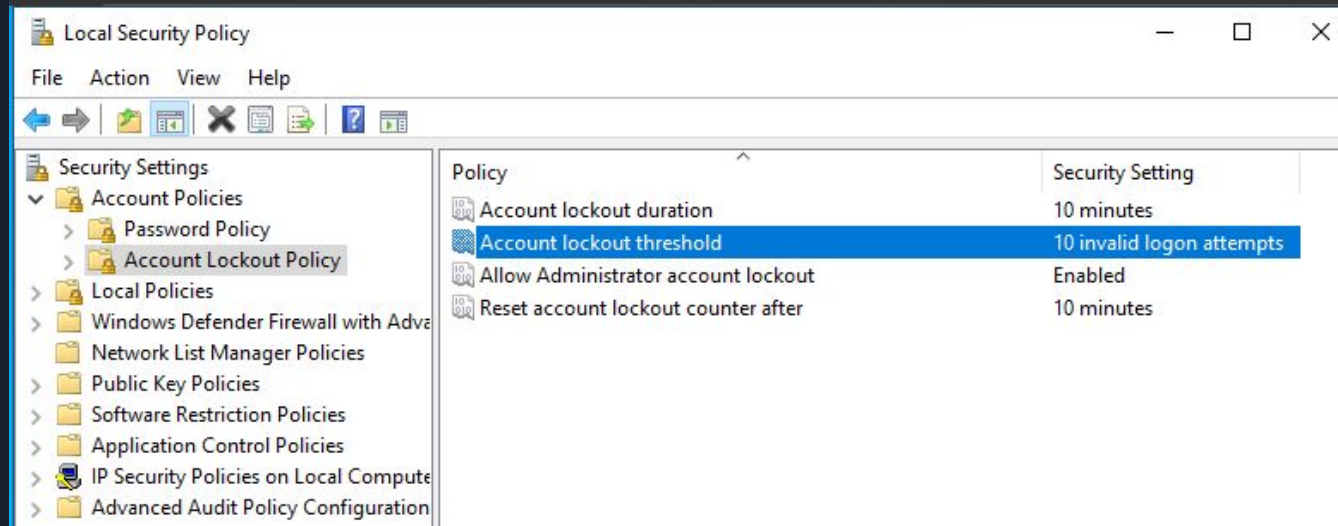
Do the same to make a list of passwords (about 5) including the password you used to set up your Windows Server.

We can now use hydra to try to RDP brute force.

```
(bryan@kali)-[~]  
$ hydra -L user.txt -P pass.txt rdp://192.168.11.141 -V -t 1  
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret se  
ng, these *** ignore laws and ethics anyway).  
  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-01-07 15:53:55  
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.  
[DATA] max 1 task per 1 server, overall 1 task, 10 login tries (l:2/p:5), ~10 tries per task  
[DATA] attacking rdp://192.168.11.141:3389/  
[ATTEMPT] target 192.168.11.141 - login "bryan" - pass "password" - 1 of 10 [child 0] (0/0)  
[ATTEMPT] target 192.168.11.141 - login "bryan" - pass "pass" - 2 of 10 [child 0] (0/0)  
[ATTEMPT] target 192.168.11.141 - login "bryan" - pass "test" - 3 of 10 [child 0] (0/0)  
[ATTEMPT] target 192.168.11.141 - login "bryan" - pass "try" - 4 of 10 [child 0] (0/0)  
[ATTEMPT] target 192.168.11.141 - login "bryan" - pass "#Crafty123" - 5 of 10 [child 0] (0/0)  
[3389][rdp] host: 192.168.11.141 login: bryan password: #Crafty123  
[ATTEMPT] target 192.168.11.141 - login "Bill_gates" - pass "password" - 6 of 10 [child 0] (0/0)
```

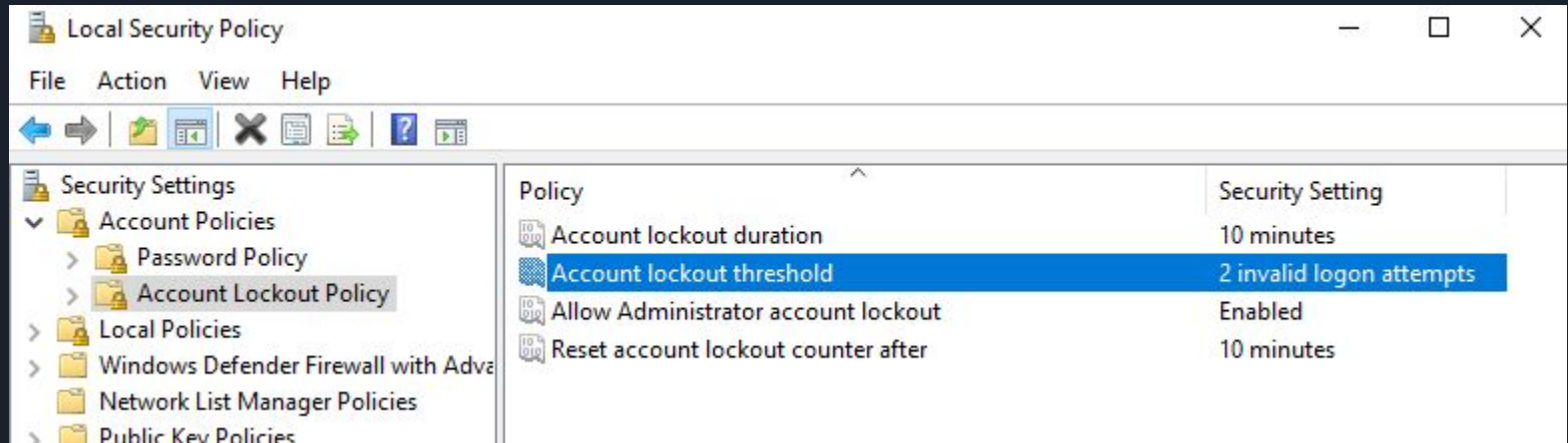
Strengthening RDP

Let's first enable a lockout policy that will prevent a brute-force attack. We can open the Local Security Policy in Windows via the search bar and navigate to Account Policies -> Account Lockout Policy.



Strengthening RDP

By default, our account lockout threshold is set to 10, but we can change this to a better number like 2. Double click the Account Lockout threshold value and set it to 10 then hit Apply.



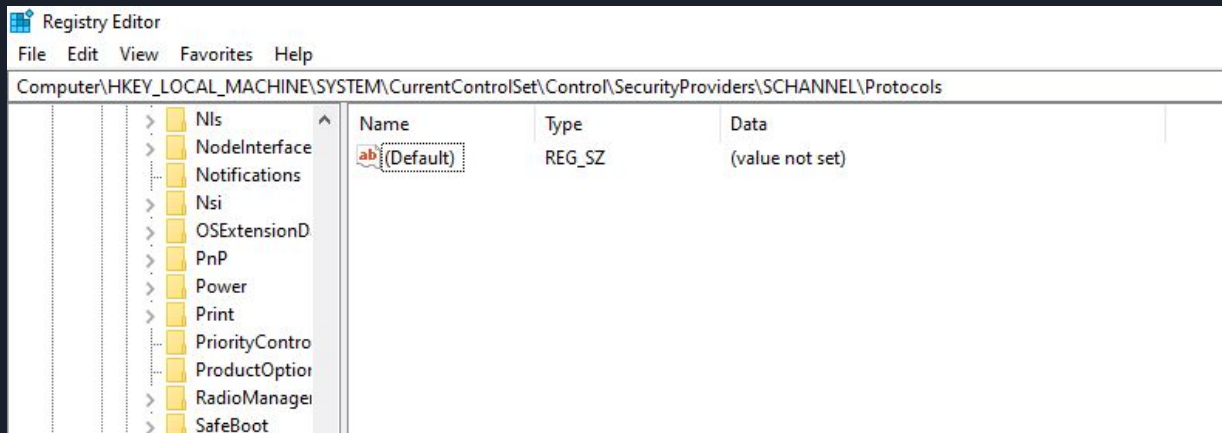
Strengthening RDP

Now when we try our Hydra attack, the account gets locked out after 2 tries.

[illegible]

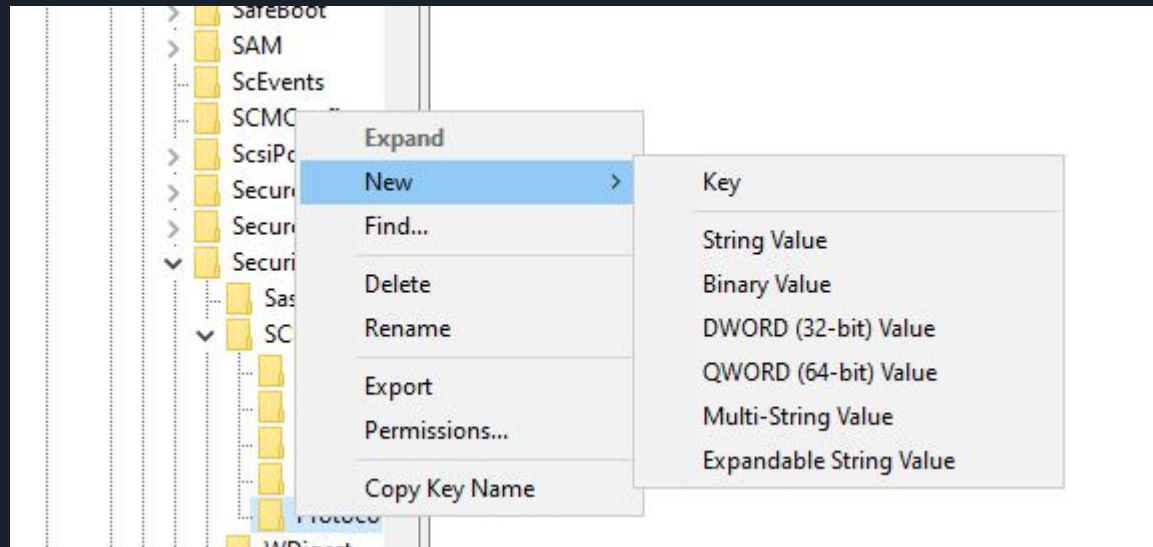
Strengthening RDP

We can also disable the use of TLS 1.0 through the use of registry keys. Registry keys are essentially settings in Windows, and most of the detailed configurations in Windows are done through editing registry keys. Open up Regedit (Registry Editor) and navigate to “HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols”



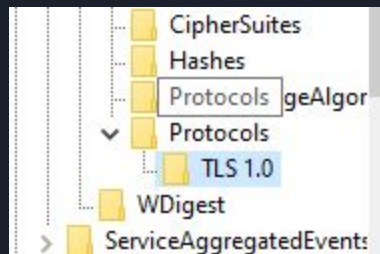
Strengthening RDP

Right click the Protocols Folder and select New -> Key



Strengthening RDP

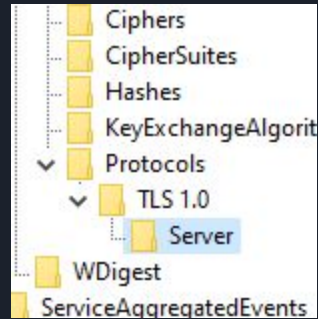
Call the key TLS 1.0





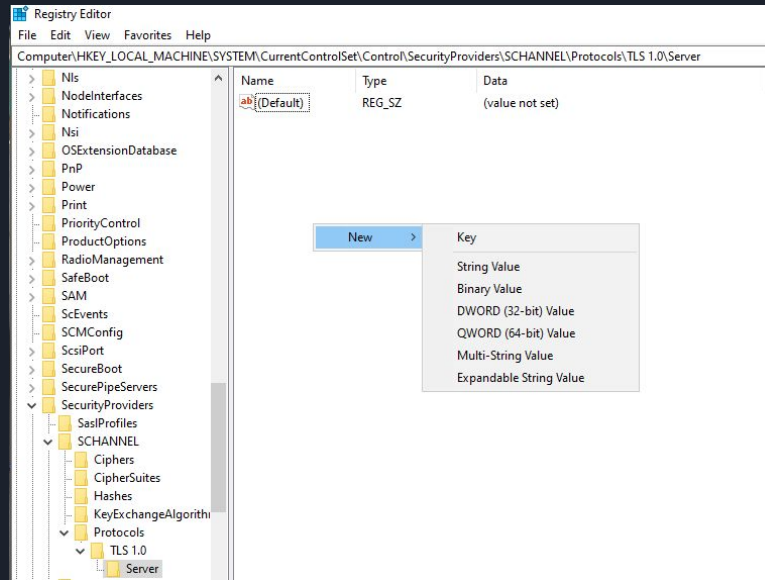
Strengthening RDP

Make a new key under TLS 1.0 called Server



Strengthening RDP



Select the Server key and right click the value window (on the right side) and click New -> DWORD Value





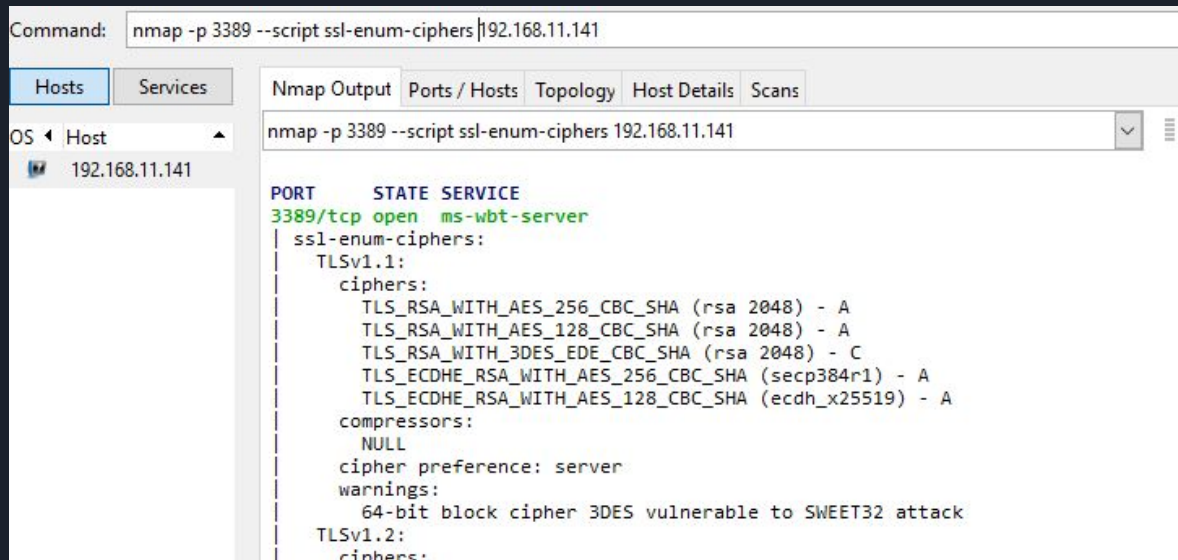
Strengthening RDP

Call the value Enabled and ensure the Data is set to 0. This means that Enabled = False, or in other words, TLS 1.0 is disabled. Restart your server after this change.

STEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0\Server		
Name	Type	Data
 (Default)	REG_SZ	(value not set)
 Enabled	REG_DWORD	0x00000000 (0)

Strengthening RDP

You should now see with Nmap that TLS 1.0 has been removed. Try the same with a Client key rather than a server. This would make it so any outgoing RDP connections wouldn't accept TLS 1.0.



```
Command: nmap -p 3389 --script ssl-enum-ciphers 192.168.11.141

Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans
OS Host
192.168.11.141

nmap -p 3389 --script ssl-enum-ciphers 192.168.11.141

PORT      STATE SERVICE
3389/tcp  open  ms-wbt-server
| ssl-enum-ciphers:
|   TLSv1.1:
|     ciphers:
|       TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
|       TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
|       TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 2048) - C
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp384r1) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A
|     compressors:
|       NULL
|     cipher preference: server
|     warnings:
|       64-bit block cipher 3DES vulnerable to SWEET32 attack
|   TLSv1.2:
|     ciphers:
```