Server Exploits

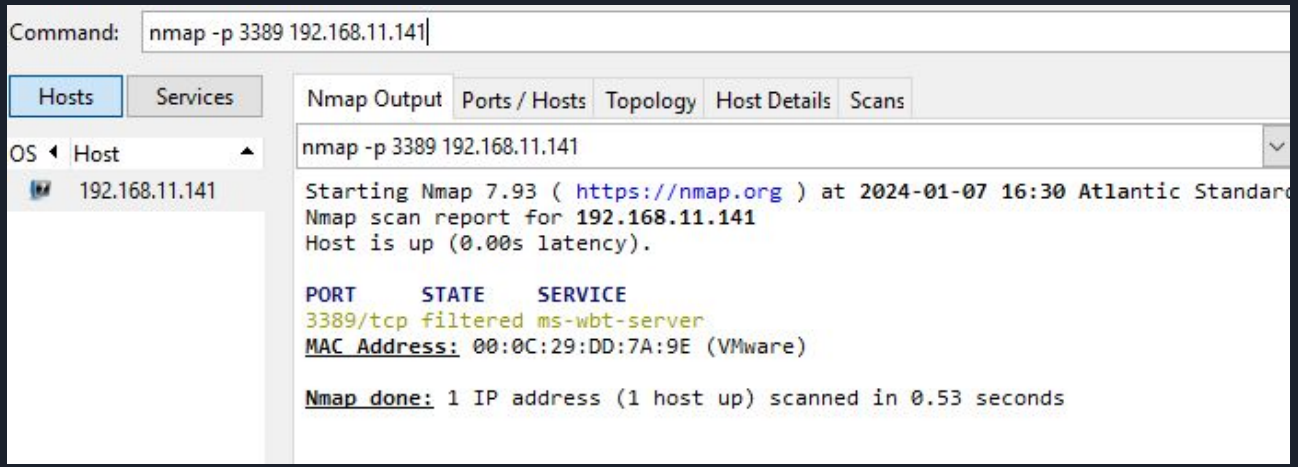
Section 1 - Lab

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

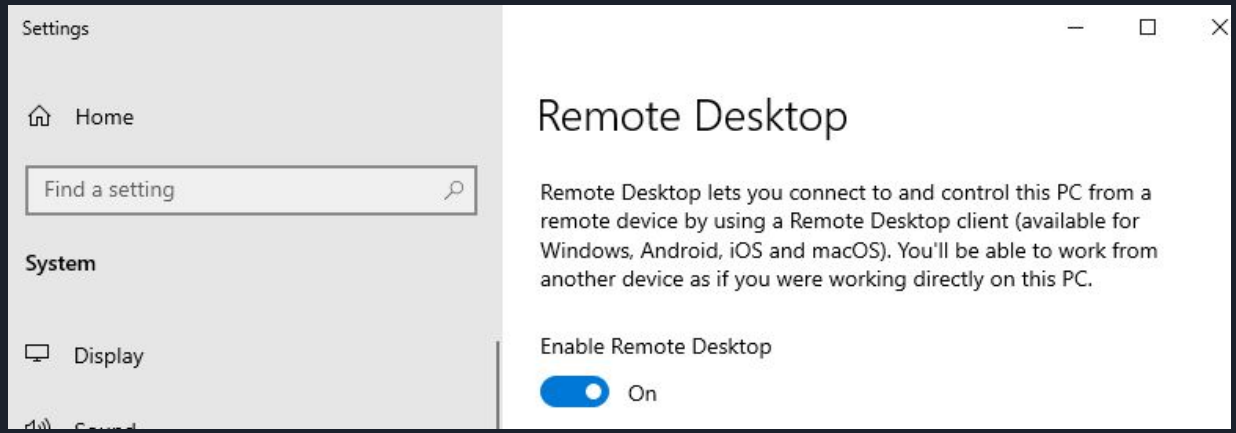
Using Nmap from your kali or host machine on our freshly installed server, we can see that there is no service running on it on port 3389 (default RDP). It appears as filtered.



Enabling Vulnerable RDP

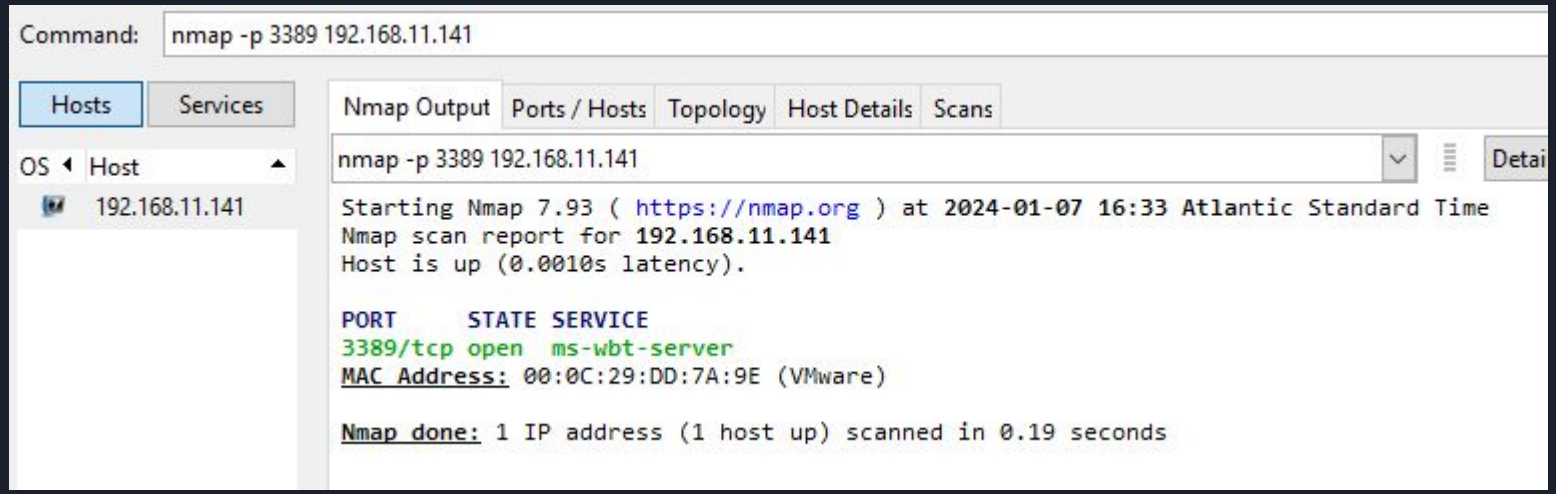
We first need to enable Remote Desktop on your server. This can be found in the settings by searching for Remote Desktop Settings. This will:

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



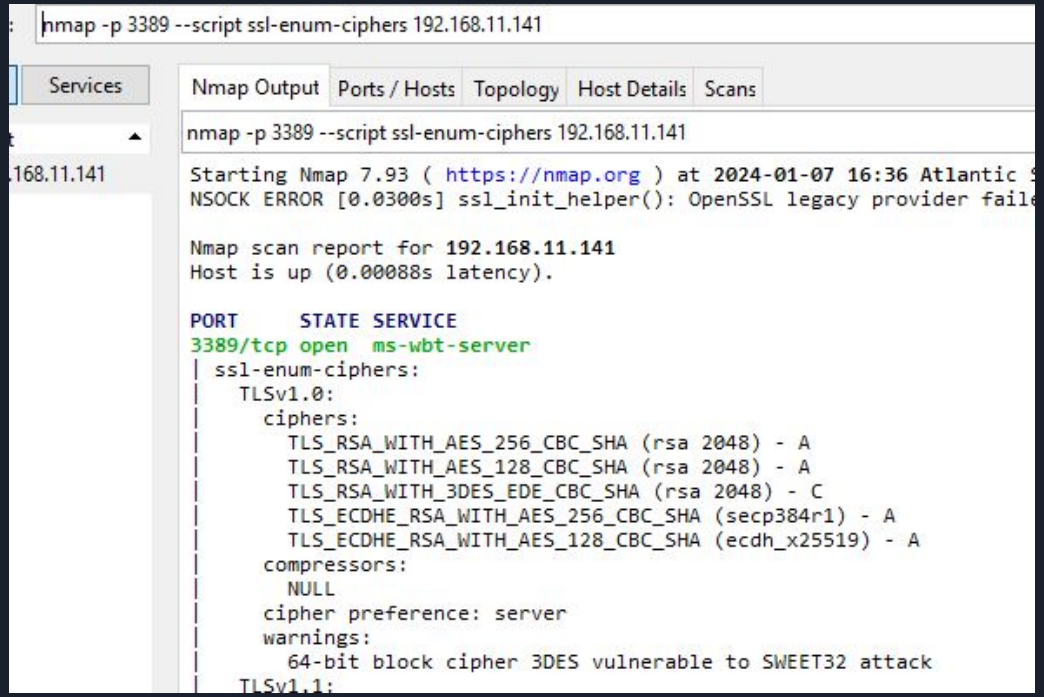
Enabling Vulnerable RDP

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



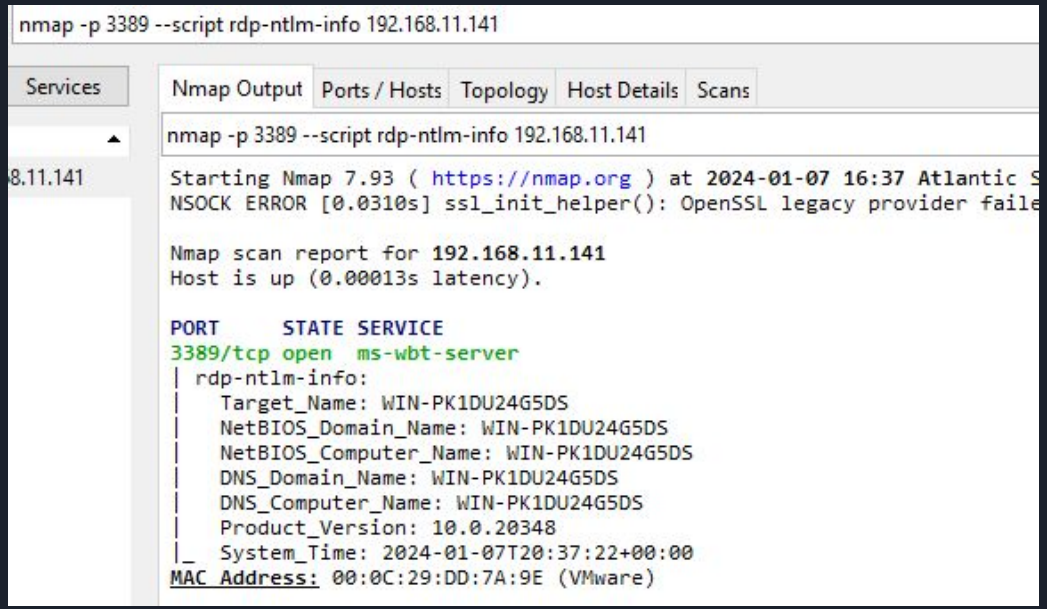
Enabling Vulnerable RDP

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.



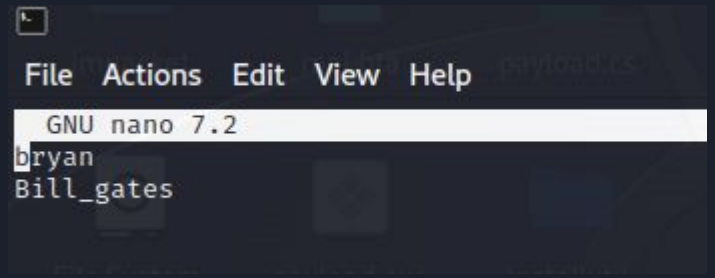
Enabling Vulnerable RDP

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



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.



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 you Windows Server.

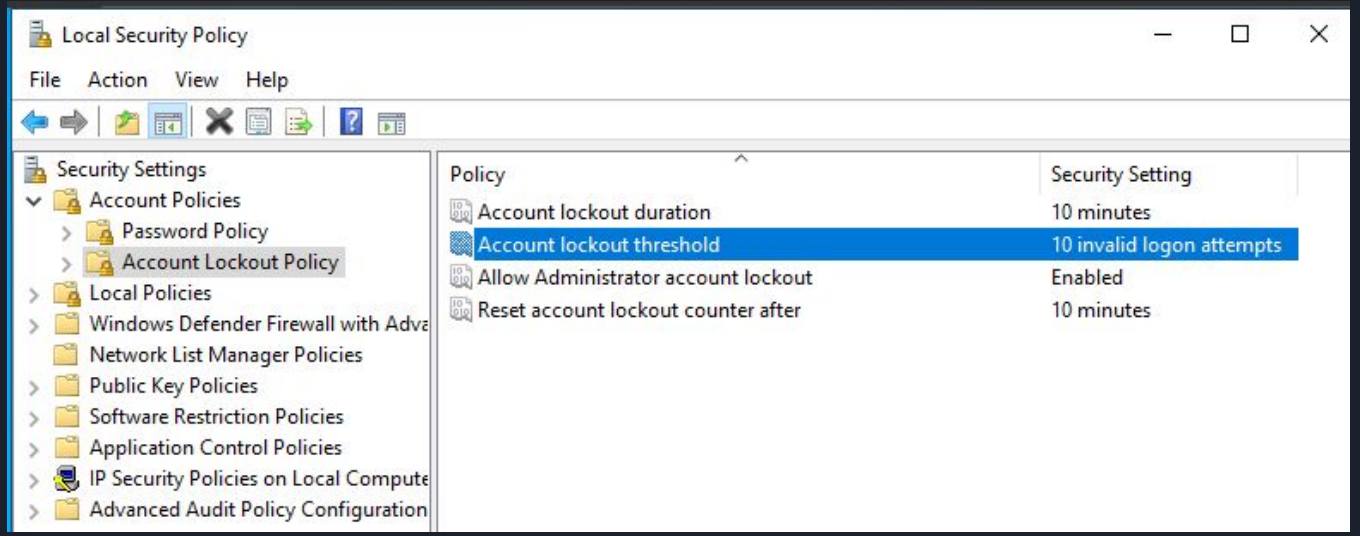
We can now use hydra to try to RDP brute force.

```
┌──(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)
```

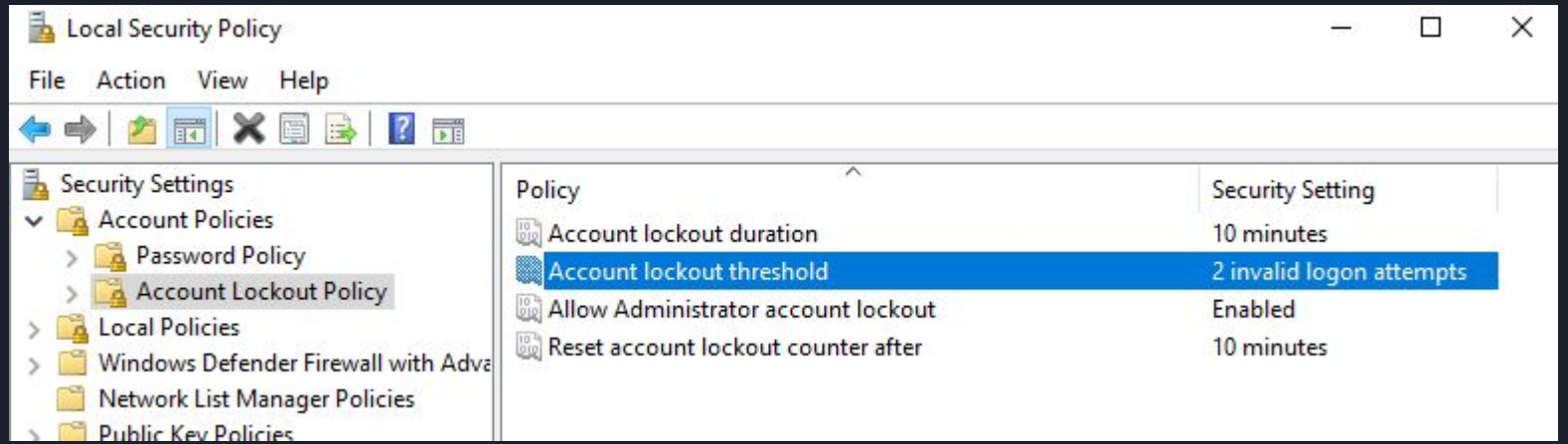
Strengthening RDP

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

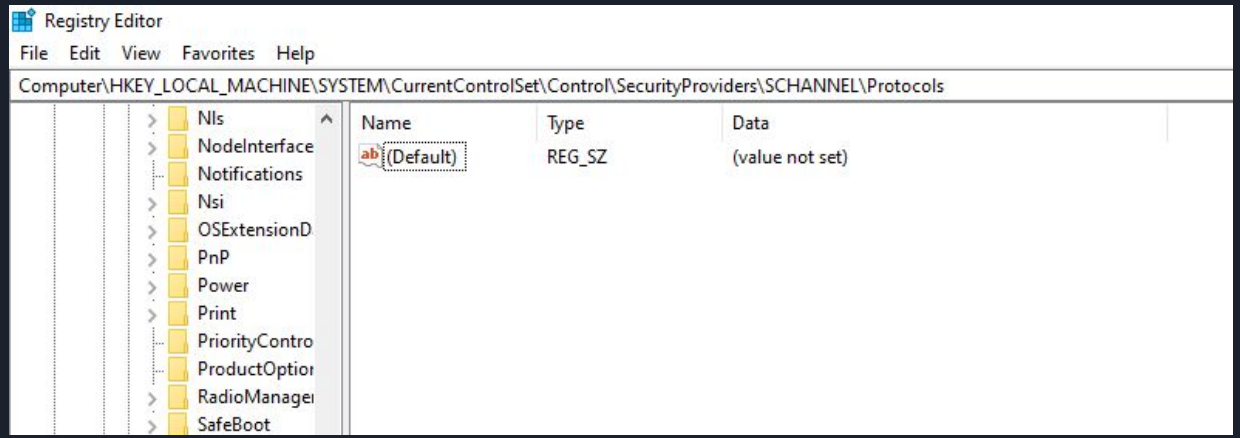
Now when we try our Hydra attack, the account gets locked out after 2 tries.

```
┌──(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 s
ng, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-01-07 15:59:28
[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)
[RE-ATTEMPT] target 192.168.11.141 - login "bryan" - pass "test" - 3 of 10 [child 0] (0/0)
[RE-ATTEMPT] target 192.168.11.141 - login "bryan" - pass "test" - 3 of 10 [child 0] (0/0)
[RE-ATTEMPT] target 192.168.11.141 - login "bryan" - pass "test" - 3 of 10 [child 0] (0/0)
[RE-ATTEMPT] target 192.168.11.141 - login "bryan" - pass "test" - 3 of 10 [child 0] (0/0)
[RE-ATTEMPT] target 192.168.11.141 - login "bryan" - pass "test" - 3 of 10 [child 0] (0/0)
```

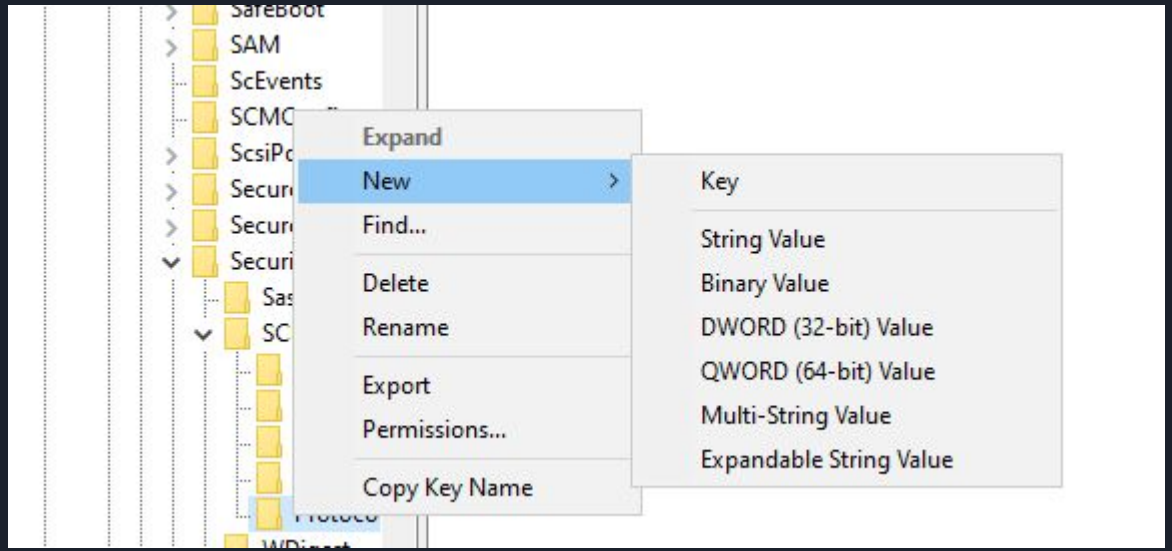
Strengthening RDP

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”



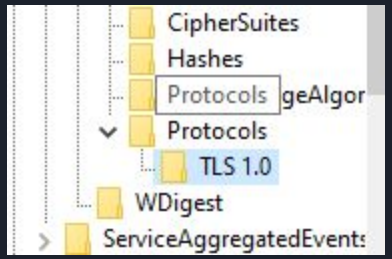
Strengthening RDP

Right click the Protocols Folder and select New -> Key



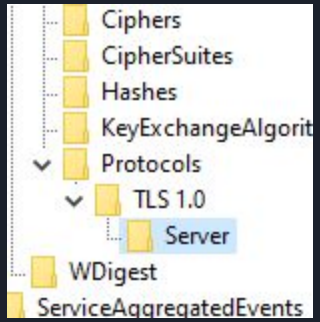
Strengthening RDP

Call the key TLS 1.0



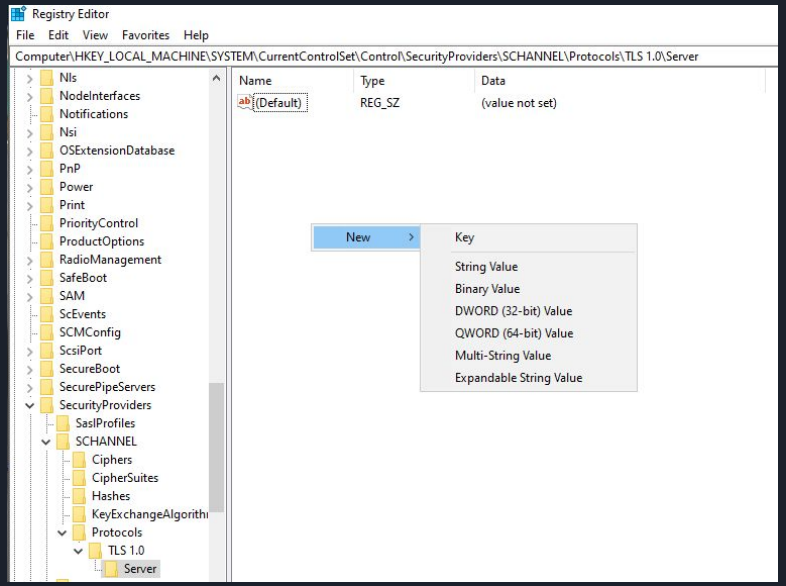
Strengthening RDP

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.

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.