

# A LITTLE GUIDE TO



## SMB ENUMERATION

[blacksideseecurity.site](http://blacksideseecurity.site)

Project by  
YOUNESS BOUSSEDARI  
Cyber security Engineer

# Contents

Introduction .....	3
What is SMB? .....	3
SMB Working.....	3
SMB Versions.....	3
SMB Security .....	4
SMB Enumeration: Hostname.....	4
nmblookup.....	4
nbtscan.....	5
nbstat NSE Script.....	5
nbtstat.....	6
Ping .....	7
smb-os-discovery NSE Script.....	7
SMB Enumeration: Share and Null Session.....	8
SMBMap.....	8
smbclient.....	9
smb-enum-shares NSE Script.....	10
Net view .....	11
Metasploit: smb_enumshares .....	12
CrackMapExec.....	13
rpcclient .....	13
SMB Enumeration: Vulnerability Scanning .....	14
smb-vuln NSE Script.....	14
SMB Enumeration: Users.....	15
Metasploit: smb_lookupsid.....	15
Impacket: Lookupsid .....	16
SMB Enumeration: Enum4Linux .....	17
Conclusion.....	20

## Introduction 😊 :

We will shine the light on the process or methodology for enumerating SMB services on the Target System/Server in this article. There are numerous tools and methods to perform enumeration, we will be finding different types of information on SMB throughout the article.

## What is SMB? 🤔 :

SMB or Server Message Block is the modernized concept of what was used to known as Common Internet File System. It works as an Application Layer Network Protocol. It is designed to be used as a File Sharing Protocol. Different Applications can on a system can read and write simultaneously to the files and request the server for services inside a network. One of the interesting functionalities of SMB is that it allows users to run it top of its TCP/IP protocol or other network protocols. Additionally, with the help of SMB, authorized users, applications, or software can access files or other resources on a remote server. Users can perform actions that include reading data, creating data, and updating data. Clients and servers communicate using something called SMB client request.

## SMB Working 😊 :

The SMB Protocol delegates the client to communicate with other participants in the same network, allowing it to access files or services open to it in the network. In order for it to function the other device also requires the implemented network protocol and receive and process the respective client request using an SMB server application. Therefore, client computers using SMB connect to a supporting server using NetBIOS over TCP/IP, IPX/SPX, or NetBEUI. The initial establishment of the connection is required for exchanging information. Subsequent data transport is regulated by the provisions of the TCP protocol. SMB functions as a request-response or client-server protocol. Once the connection is established, the client's computer or program can then open, read/write, and access files similar to the file system on a local computer.

## SMB Versions 😊 :

**CIFS:** The old version of SMB, which was included in Microsoft Windows NT 4.0 in 1996.

**SMB 1.0 / SMB1:** The version used in Windows 2000, Windows XP, Windows Server 2003 and Windows Server 2003 R2.

**SMB 2.0 / SMB2:** This version is used in Windows Vista and Windows Server 2008.

**SMB 2.1 / SMB2.1:** This version is used in Windows 7 and Windows Server 2008 R2.

**SMB 3.0 / SMB3:** This version used in Windows 8 and Windows Server 2012.

**SMB 3.02 / SMB3:** This version is used in Windows 8.1 and Windows Server 2012 R2.

**SMB 3.1: This version is used in Windows Server 2016 and Windows 10.**

**Currently, the latest version of SMB is SMB 3.1.1, which was introduced with Windows 10 and Windows Server 2016. This version supports AES 128 GCM encryption in addition to AES 128 CCM encryption added in SMB3 and implements pre-authentication integrity check using SHA-512 hash.**

**SMB 3.1.1 also makes secure negotiation mandatory when connecting to clients using SMB 2.x and higher.**

## SMB Security :

The SMB protocol supports two levels of security. The first is the share level. The server is protected at this level and each share has a password. The client computer or user has to enter the password to access data or files saved under the specific share. This is the only security model available in the Core and Core plus SMG protocol definitions. User level protection was later added to the SMB protocol. It is applied to individual files and each share is based on specific user access rights. Once a server authenticates the client, he/she is given a unique identification (UID) that is presented upon access to the server. The SMB protocol has supported individual security since LAN Manager 1.0 was implemented.

## SMB Enumeration: Hostname :

Firstly, we will start the enumeration of the SMB by finding the hostname of the target machine. This can be done by various tools.

### nmblookup :

We started with nmblookup tool. It is designed to make use of queries for the NetBIOS names and then map them to their subsequent IP addresses in a network. The options allow the name queries to be directed at a particular IP broadcast area or to a particular machine. All queries are done over UDP.

### For unique names:

00: Workstation Service (workstation name)

03: Windows Messenger service

06: Remote Access Service

20: File Service (also called Host Record)

21: Remote Access Service client

1B: Domain Master Browser – Primary Domain Controller for a domain

1D: Master Browser

### For group names:

00: Workstation Service (workgroup/domain name)

1C: Domain Controllers for a domain

1E: Browser Service Election

## nmblookup -A 192.168.11.113

```
[root@kali]# nmblookup -A 192.168.11.113
Looking up status of 192.168.11.113
WIN-11-SERVER <00> - B <ACTIVE>
WORKGROUP <00> - <GROUP> B <ACTIVE>
WIN-11-SERVER <20> - B <ACTIVE>
WORKGROUP <1e> - <GROUP> B <ACTIVE>

MAC Address = 08-00-27-DE-1E-0E
```

Here, we can see that we have enumerated the hostname to be DESKTOP-ATNONJ9.

## **nbtscan :**

Moving Forward we used nbtscan tool. NBTscan is a program for scanning IP networks for NetBIOS name information. It sends NetBIOS status query to each address in supplied range and lists received information in human-readable form. For each responded host it lists IP address, NetBIOS computer name, logged-in user name and MAC address (such as Ethernet)

## nbtscan 192.168.11.113

```
[root@kali]# nbtscan 192.168.11.113
Doing NBT name scan for addresses from 192.168.11.113
IP address      NetBIOS Name      Server      User      MAC address
192.168.11.113  WIN-11-SERVER    <server>   <unknown>  08:00:27:de:1e:0e
```

Finally, we can see that we have enumerated the hostname to be DESKTOP-ATNONJ9.

## **nbstat NSE Script :**

This nmap script attempts to retrieve the target's NetBIOS names and MAC address. By default, the script displays the name of the computer and the logged-in user; if the verbosity is turned up, it displays all names the system thinks it owns. It also shows the flags that we studied in nmblookup tool.

## nmap --script smb-os-discovery.nse 192.168.11.113

```

└─[root@kali]~[/home/kali]
# nmap --script smb-os-discovery.nse 192.168.11.113
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-02 20:04 EST
Nmap scan report for 192.168.11.113 (192.168.11.113)
Host is up (0.0016s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
1433/tcp   open  ms-sql-s
5985/tcp   open  wsman
MAC Address: 08:00:27:DE:1E:0E (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Host script results:
| smb-os-discovery:
| OS: Windows Server 2025 Standard Evaluation 26100 (Windows Server 2025 Standard Evaluation 6.3)
| Computer name: win-11-server
| NetBIOS computer name: WIN-11-SERVER\x00
| Workgroup: WORKGROUP\x00
|_ System time: 2026-02-02T17:04:35-08:00

Nmap done: 1 IP address (1 host up) scanned in 4.96 seconds

```

Finally, we can see that we have enumerated the hostname to be DESKTOP-ATNONJ9.

## nbtstat :

This Windows command displays the NetBIOS over TCP/IP (NetBT) protocol statistics. It can read the NetBIOS name tables for both the local computer and remote computers. It can also read the NetBIOS name cache. This command allows a refresh of the NetBIOS name cache and the names registered with Windows Internet Name Service (WINS). When used without any parameters, this command displays Help Information. This command is available only if the Internet Protocol (TCP/IP) protocol is installed as a component in the properties of a network adapter in Network Connections

## nbtstat -A 192.168.11.113

```

C:\Users\PC>nbtstat -A 192.168.11.113
vEthernet (Default Switch):
NodeIpAddress: [172.27.192.1] Scope Id: []
    Host not found.

Ethernet 6:
NodeIpAddress: [192.168.55.1] Scope Id: []
    Host not found.

OpenVPN Connect DCO Adapter:
NodeIpAddress: [0.0.0.0] Scope Id: []
    Host not found.

Connexion au réseau local:
NodeIpAddress: [0.0.0.0] Scope Id: []
    Host not found.

Ethernet:
NodeIpAddress: [0.0.0.0] Scope Id: []
    Host not found.

Wi-Fi:
NodeIpAddress: [192.168.11.111] Scope Id: []

        NetBIOS Remote Machine Name Table

        Name          Type       Status
WIN-11-SERVER <00>  UNIQUE    Registered
WORKGROUP     <00>  GROUP     Registered
WIN-11-SERVER <20>  UNIQUE    Registered
WORKGROUP     <1E>  GROUP     Registered

        MAC Address = 08-00-27-DE-1E-0E

```

Finally, we can see that we have enumerated the hostname to be DESKTOP-ATNONJ9

### mb-os-discovery NSE Script

This NSE script attempts to determine the operating system, computer name, domain, workgroup, and current time over the SMB protocol (ports 445 or 139). You achieve it by initiating a session with the anonymous account (or with a proper user account, if one is given; it likely doesn't make a difference); in response to a session starting, the server will send back all this information.

The following fields may be included in the output, depending on the circumstances (e.g., the workgroup name is mutually exclusive with domain and forest names) and the information available:

- OS
- Computer name
- Domain name
- Forest name
- FQDN
- NetBIOS computer name
- NetBIOS domain name
- Workgroup
- System time

### nmap --script nbstat.nse 192.168.11.113

```
[root@kali]# nmap --script nbstat.nse 192.168.11.113
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-02 20:36 EST
Nmap scan report for 192.168.11.113 (192.168.11.113)
Host is up (0.0010s latency).

Not shown: 995 filtered tcp ports (no-response)

PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
1433/tcp   open  ms-sql-s
5985/tcp   open  wsman
MAC Address: 08:00:27:DE:1E:0E (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Host script results:
| nbstat: NetBIOS name: WIN-11-SERVER, NetBIOS user: <unknown>, Flags: <unique><active>
|_ WORKGROUP<00>          Flags: <group><active>
| WIN-11-SERVER<20>        Flags: <unique><active>
|_ WORKGROUP<1e>          Flags: <group><active>

Nmap done: 1 IP address (1 host up) scanned in 4.77 seconds
```

## SMB Enumeration: Share and Null Session 😊:

As we discussed earlier that SMB works on sharing files and resources. In order to transfer these files or resources, there are data streams that are called shares. There are public shares that are accessible to everyone on the network and then there are the user-specific shares. Let's enumerate these shares.

# SMBMap 😊:

**SMBMap** allows users to enumerate samba share drives across an entire domain. List share drives, drive permissions, share contents, upload/download functionality, file name auto-download pattern matching, and even execute remote commands. This tool was designed with pen testing in mind and is intended to simplify searching for potentially sensitive data across large networks.

**smbmap -H 192.168.11.113**

```
[root@kali]# smbmap -H 192.168.11.113
[!] SMBMap - Samba Share Enumerator v1.10.7 | Shawn Evans - ShawnDEvans@gmail.com
[!] https://github.com/ShawnDEvans/smbmap
[!] How to Use smbmap(1)
[!] Checking for open ports...
[*] Detected 1 hosts serving SMB
[*] Established 1 SMB connections(s) and 0 authenticated session(s)
[!] Something weird happened on (192.168.11.113) Error occurs while reading from remote(104) or
[*] Closed 1 connections
```

```
smbmap -H 192.168.11.113 -u Administrator -p  
Anonymous_1212
```

```
[root@kali) [/home/kali]
# smbmap -H 192.168.11.113 -u Administrator -p Anonymous_1212

[+] IP: 192.168.11.113:445      Name: 192.168.11.113      Status: ADMIN!!!
Disk          Permissions      Comment
ADMIN$        READ, WRITE     Remote Admin
C$           READ, WRITE     Default share
IPC$          READ ONLY      Remote IPC
partage       READ, WRITE
Public        If not present, it can be downloaded directly from the official repository, such as
SQL2025       from the GitHub script db file
Users         READ, WRITE
[*] Closed 1 connections documentation:
```

## smbclient :

**smbclient** is samba client with an "ftp like" interface. It is a useful tool to test connectivity to a Windows share. It can be used to transfer files, or to look at share names. In addition, it has a nifty ability to 'tar' (backup) and restore files from a server to a client and vice versa. We enumerated the target machine and found the guest share using the **smbclient** directly. Then we connect to the guest share and see that there is a text file by the name of **file.txt**. We can download it using the **get command**

**smbclient -L 192.168.11.113**

## smbclient //192.168.11.113/

# get file.txt

```

└─(root㉿kali)-[~/home/kali]
# smbclient -L 192.168.11.113
Password for [WORKGROUP\root]:

```

Sharename	Type	Comment
ADMIN\$	Disk	Remote Admin
C\$	Disk	Default share
IPC\$	IPC	Remote IPC
partage	Disk	
Public	Disk	
SQL2025	Disk	The <code>smb-nbstat.nse</code> script is a pre-installed Nmap Scripting Engine (NSE) script used to retrieve network statistics (netstat) via SNMP. It is located by default in <code>/usr/share/nmap/scripts/</code> on Windows.
Users	Disk	

Reconnecting with SMB1 for workgroup listing.  
do\_connect: Connection to 192.168.11.113 failed (Error NT\_STATUS\_RESOURCE\_NAME\_NO  
Unable to connect with SMB1 -- no workgroup available

## smbclient //192.168.11.113/Users/

```

└─(root㉿kali)-[~/home/kali]
# smbclient //192.168.11.113/Users/
Password for [WORKGROUP\root]:
Try "help" to get a list of possible commands.
smb: \> ls
.
..
Administrator
All Users
Default
Default User
desktop.ini
haker
Public

```

File	Type	Date	Time	Size
Administrator	D	0	Mon Feb 2	20:46:48 2026
All Users	DHSrn	0	Mon Apr 1	03:26:42 2024
Default	DHR	0	Thu Jan 15	13:07:54 2026
Default User	DHSrn	0	Mon Apr 1	03:26:42 2024
desktop.ini	AHS	174	Mon Apr 1	03:01:26 2024
haker	D	0	Sat Jan 17	09:55:26 2026
Public	DR	0	Thu Jan 15	13:09:50 2026

12908287 blocks of size 4096. 5580128 blocks available

## get File.txt

```

└─(root㉿kali)-[~/home/kali]
# smbclient //192.168.11.113/Users/
Password for [WORKGROUP\root]:
Try "help" to get a list of possible commands.
smb: \> ls
.
..
Administrator
All Users
Default
Default User
desktop.ini
File.txt.txt
haker
Public

```

File	Type	Date	Time	Size
Administrator	DR	0	Mon Feb 2	21:00:52 2026
All Users	DHS	0	Mon Feb 2	20:46:48 2026
Default	D	0	Wed Jan 21	19:09:46 2026
Default User	DHSrn	0	Mon Apr 1	03:26:42 2024
desktop.ini	DHR	0	Thu Jan 15	13:07:54 2026
File.txt.txt	AHS	174	Mon Apr 1	03:01:26 2024
haker	D	16	Mon Feb 2	21:01:13 2026
Public	DR	0	Sat Jan 17	09:55:26 2026

12908287 blocks of size 4096. 5579938 blocks available

smb: \> get File.txt.txt

getting file \File.txt.txt of size 16 as File.txt.txt (1.7 Kilobytes/sec) (average 1.7 Kilobytes/sec)

smb: \>

Then we enumerate the user-specific share. We connect to the SMB as user raj and find a share by the name of 'share'. We reconfigured the smbclient command to access the share and we see that we find a file named raj.txt. Again, we can download this file as well as using the get command.

### Smb-enum-shares NSE Script 😊:

This NSE script attempts to list shares using the svsvc.NetShareEnumAll MSRPC function and retrieve more information about them using svsvc.NetShareGetInfo. If access to those functions is denied, the system checks a list of common share names. An administrator account is required to call NetShareGetInfo on all versions of Windows up to 2003, as well as Windows Vista, Windows 7, and Windows 10, if UAC is turned down. Even if NetShareEnumAll is restricted, users attempting to connect to a share will always reveal its existence. So, if NetShareEnumAll fails, the script uses a pre-generated list of shares, based on a large test network. If any of those succeed, the script records them. After finding a list of shares, the script attempts to connect to each of them anonymously, which divides them into "anonymous" for shares that the NULL user can connect to, or "restricted" for shares that require a user account.

[nmap --script smb-enum-shares -p139,445 192.168.11.113](#)

```
(root㉿kali)-[~/home/kali]
# nmap --script smb-enum-shares.nse -p 139,445 192.168.11.113
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-02 21:09 EST
Nmap scan report for 192.168.11.113 (192.168.11.113)
Host is up (0.00069s latency).

PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds

MAC Address: 08:00:27:DE:1E:0E (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Host script results:
| smb-enum-shares:
|   account_used: guest
|   \\\192.168.11.113\ADMIN$:
|     Type: STYPE_DISK_TREE_HIDDEN
|     Comment: Remote Admin
|     Anonymous access: <none>
|     Current user access: <none>
|   \\\192.168.11.113\C$:
|     Type: STYPE_DISK_TREE_HIDDEN
|     Comment: Default share
|     Anonymous access: <none>
|     Current user access: <none>
```

```
Comment: Remote IPC
Anonymous access: READ
Current user access: READ/WRITE
\\192.168.11.113\Public:
    Type: STYPE_DISKTREE
    Comment:
    Anonymous access: <none>
    Current user access: READ
\\192.168.11.113\SQL2025:
    Type: STYPE_DISKTREE
    Comment:
    Anonymous access: <none>
    Current user access: READ
\\192.168.11.113\Users:
    Type: STYPE_DISKTREE
    Comment:
    Anonymous access: <none>
    Current user access: READ
\\192.168.11.113\partage:
    Type: STYPE_DISKTREE
    Comment:
    Anonymous access: <none>
    Current user access: READ
Nmap done: 1 IP address (1 host up) scanned in 1.30 seconds
```

Then, we can see that we have the shares listed although the Access is Denied the existence of the share is confirmed.

## Net view 😊:

Displays a list of domains, computers, or resources that are being shared by the specified computer.

Used without parameters, net view displays a list of computers in your current domain. This time we are on the Windows machine. We used the net view with the /All parameter to list all the shares on the target machine.

**net view \\192.168.11.113 /All**

```
C:\Users\Administrator>net view \\192.168.11.113 /All ←  
Shared resources at \\192.168.11.113
```

Share name	Type	Used as	Comment
------------	------	---------	---------

ADMIN\$	Disk	Remote Admin
C\$	Disk	Default share
IPC\$	IPC	Remote IPC
partage	Disk	
Public	Disk	
SQL2025	Disk	
Users	Disk (UNC)	

The command completed successfully.

```
C:\Users\Administrator>net use \\192.168.11.113\Users
```

Local name	\\192.168.11.113\Users
Remote name	
Resource type	Disk
Status	OK
# Opens	0
# Connections	1

The command completed successfully.

```
C:\Users\Administrator>net use  
New connections will be remembered.
```

Status	Local	Remote	Network
--------	-------	--------	---------

OK	\\192.168.11.113\Users	Microsoft Windows Network
----	------------------------	---------------------------

The command completed successfully.

hen we changed the command by adding the share, and we can read the contents of that share.

Now using the copy command, we can download the file from share.

## Metasploit: smb\_enumshares 😊:

The `smb_enumshares` module enumerates any SMB shares that are available on a remote system.

You need the IP Address of the target server or machine followed by the set of credentials that can

be used to access the share.

[use auxiliary/scanner/smb/smb\\_enumshares](#)

[set rhosts 192.168.11.113](#)

[set smbuser Administrator](#)

[set smbpass Anonymous\\_1212](#)

[exploit](#)

```
View the full module info with the info, or info -d command.

msf auxiliary(scanner/smb/smb_enumshares) > set RHOST 192.168.11.113
RHOST => 192.168.11.113
msf auxiliary(scanner/smb/smb_enumshares) > set smbuser Administrator
smbuser => Administrator
msf auxiliary(scanner/smb/smb_enumshares) > set smbpass Anonymous_1212
smbpass => Anonymous_1212
msf auxiliary(scanner/smb/smb_enumshares) > exploit
[-] 192.168.11.113:139 - Login Failed: Unable to negotiate SMB1 with the remote host: Not a valid SMB packet
[!] 192.168.11.113:139 - peer_native_os is only available with SMB1 (current version: SMB3)
[!] 192.168.11.113:139 - peer_native_lm is only available with SMB1 (current version: SMB3)
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.23/lib/recog/fingerprint/regexp_factory.rb:34:
g: nested repeat operator '+' and '?' was replaced with '*' in regular expression
[+] 192.168.11.113:139 - ADMIN$ - (DISK|SPECIAL) Remote Admin
[+] 192.168.11.113:139 - C$ - (DISK|SPECIAL) Default share
[+] 192.168.11.113:139 - IPC$ - (IPC|SPECIAL) Remote IPC
[+] 192.168.11.113:139 - partage - (DISK) total_target>
[+] 192.168.11.113:139 - Public - (DISK)
[+] 192.168.11.113:139 - SQL2025 - (DISK)
[+] 192.168.11.113:139 - Users - (DISK) directly from the official Nmap repository, such as
[*] 192.168.11.113: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/smb_enumshares) >
```

## CrackMapExec 😊:

CrackMapExec (a.k.a CME) is a post-exploitation tool that helps automate assessing the security of large Active Directory networks. Built with stealth in mind, CME follows the concept of "Living off the Land": abusing built-in Active Directory features/protocols to achieve its functionality and allowing it to evade most endpoint protection/IDS/IPS solutions. Additionally, CrackMapExec can Map the network hosts, Generate Relay List, enumerate shares and access, enumerate active sessions, enumerate disks, enumerate logged on users, enumerate domain users, Enumerate Users by bruteforcing RID, enumerate domain groups, enumerate local groups etc.

### crackmapexec smb 192.168.11.113 -u 'Administrator' -p 'Anonymous\_1212' --Users

```
(root㉿kali)-[~/home/kali]$ crackmapexec smb 192.168.11.113 -u 'Administrator' -p 'Anonymous_1212' --Users
SMB      192.168.11.113 445   WIN-11-SERVER  [*] Windows Server 2025 Standard Evaluation 26100 x64 (name:WIN-11-SERVER)
(signing=False) (SMBv1=True)
SMB      192.168.11.113 445   WIN-11-SERVER  [+] win-11-server\Administrator:Anonymous_1212 (Pwn3d!)
```

Then, we can see different shares and the permissions that are allowed on that particular share.

## rpcclient 😊:

rpcclient is a utility initially developed to test MS-RPC functionality in Samba itself. It has undergone several stages of development and stability. Many system administrators have now written scripts around it to manage Windows NT clients from their UNIX workstation. We will be using it to enumerate the users on the SMB shares using the option of netshareenum as shown in the image below.

### rpcclient -U "" -N 192.168.11.113

## netshareenum



## netshareenum

```
[root@kali]~[/home/kali]
# rpcclient -U "Administrator%Anonymous_1212" 192.168.11.113
rpcclient $> netshareenum
netname: partage
    remark: (null)
    path:   C:\partage
    password: scripts nmap (null)
netname: Users
    remark: download nmap nse scripts
    path:   C:\Users
    password: (null)
netname: Public
    remark: (null) retrieve network statistics (netsstat) via SNMP. It is located by default in /usr/
    path:   C:\Users\Public on Linux or c:\Program Files\Nmap\scripts\ on Windows.
    password: No manual (null) is required for standard Nmap installations.
netname: SQL2025
    remark: How to Use snmp-netstat.nse:
    path:   C:\SQL2025\ipt
    password: (null) --script snmp-netstat <target>
rpcclient $> netshareenumall
netname: ADMIN$ * verify installation!
    remark: Remote Admin
    path:   C:\WINDOWS\sub script db file
    password: (null)
netname: C$           Detailed usage instructions are available at the Nmap snmp-netstat
```

## netshareenumall

```
password: (null)
rpcclient $> netshareenumall
netname: ADMIN$           remark: Remote Admin
          path: C:\WINDOWS
          password: (null)
netname: C$                remark: Default share
          path: C:\
          password: (null)
netname: IPC$              remark: Remote IPC
          path: used to retrieve network statistics (netstat) via SNMP. It is
          password: (null)
netname: partage           No manual download is required for standard Nmap insta
          remark: How to Use snmp-netstat.nse:
          path: C:\partage
          password: + Run the script
netname: Public             nmap -sU -p 161 --script snmp-netstat <target>
          remark: Verify installation
          path: C:\Users\Public
          password: (null)
netname: SQL2025            If not present, it can be downloaded directly from the o
          remark: Documentation:
          path: C:\SQL2025   from the GitHub script.db file.
                               Page instructions are available at the Nmap
                               documentation page.
```

# SMB Enumeration: Vulnerability Scanning 😊

We enumerate a SMB server to compromise it, as we need to enumerate and find possible vulnerabilities that we can use to exploit the server. In order to do this in an optimized method, we can perform Vulnerability Scanning. There might be multiple tools to perform this kind of Scanning but here we will be focusing on this NSE script.

## smb-vuln NSE Script 😊

Nmap in past used to have a script by the name of smb-check-vulns. It used to scan the target server for the various vulnerabilities such as:

- conficker
- cve2009-3103
- ms06-025
- ms07-029
- regsvc-dos
- ms08-067

Then the script was divided into single vulnerability checks that can run individually, such as smb-vuln-ms08-067. Hence to check all SMB vulnerabilities available in the Nmap Scripting Engine we use the \* with the script.

**nmap --script smb-vuln\* 192.168.11.113**

```
(root㉿kali)-[~/home/kali]
# nmap --script smb-vuln* 192.168.11.113 bstat.nse
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-02 21:52 EST
Nmap scan report for 192.168.11.113 (192.168.11.113)
Host is up (0.00097s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
1433/tcp   open  ms-sql-s
5985/tcp   open  wsman
MAC Address: 08:00:27:DE:1E:0E (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: NT_STATUS_OBJECT_NAME_NOT_FOUND
|smb-vuln-ms08-067:
|  VULNERABLE: ← How to Use snmp-netstat.nse:
|  Microsoft Windows system vulnerable to remote code execution (MS08-067)
|  State: LIKELY VULNERABLE ←
|  IDs: CVE:CVE-2008-4250
|    The Server service in Microsoft Windows 2000 SP4, XP SP2 and SP3, Server 2003 SP1 and SP2, Vista Gold and SP1, Server 2008, and 7 Pre-Beta allows remote attackers to execute arbitrary code via a crafted RPC request that triggers the overflow during path canonicalization.
|    from the GitHub script db file.
|  Disclosure date: 2008-10-23
|  References: ← Documentation:
|    https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-4250
Report incomplete predictions
snmp-netstat NSE script - Nmap
Script Arguments Example Usage S
netstat, Script types: portrule, Gate
nmap →
nmap/scripts/script.db at master
Entry (filename = http-litespeed-s
Categories... Entry
Building custom NSE and Disc
Scripts - CyberScope - NetAlly
Aug 10, 2023 - A wide selection of
```

## SMB Enumeration: Users 😊:

In a Windows environment, the system assigns each user a unique identifier called Security ID or SID, which controls access to various resources like Files, Registry keys, network shares, etc. Hence, users shouldn't compromise their SID.

### Metasploit: smb\_lookupsid 🤖

The `smb_lookupsid` module brute-forces SID lookups on a range of targets to determine what local users exist in the system. Knowing what users exist on a system can greatly speed up any further brute-force logon attempts later on.

```
use auxiliary/scanner/smb/smb_lookupsid
```

```
set rhosts 192.168.11.113
```

```
set smbuser Administrator
```

```
set smbpass Anonymous_1212
```

```
exploit
```

```
set Rhosts      set Rhostname
msf auxiliary(scanner/smb/smb_lookupsid) > set Rhosts 192.168.11.113
Rhosts ⇒ 192.168.11.113  load script nmap/smb-nbstat.nse
msf auxiliary(scanner/smb/smb_lookupsid) > set smbpass Anonymous_1212
smbpass ⇒ Anonymous_1212  load nmap for windows10
msf auxiliary(scanner/smb/smb_lookupsid) > set smbuser Administrator
smbuser ⇒ Administrator
msf auxiliary(scanner/smb/smb_lookupsid) > ex
exit    exploit
msf auxiliary(scanner/smb/smb_lookupsid) > exploit
[*] 192.168.11.113:445 - PIPE(lsarpc) LOCAL(WORKGROUP - null) DOMAIN(WIN-11-SERVER - S-1-5-21-2836125315
[*] Trying RID 4000 / 4000
SMB Lookup SIDs Output
The 'snmp-netstat.nse' script is a pre-installed Nmap Scripting Engine (NSE) script used to retrieve network statistics (netstat) via SNMP. It is located by default in /usr/share/nmap/scripts/ on Linux or C:\Program Files\Nmap\scripts\ on Windows.
Type   Name          share/nmap/scripts/ on Linux or C:\Program Files\Nmap\scripts\ on Windows.  RID
_____
No manual download is required for standard Nmap installations. ↴
User   Administrator      500
User   Guest            501
User   DefaultAccount   503
User   WDAGUtilityAccount 504
Group  None             513
User   test              1004
User   hacker            1006
Alias  SQLServer2005SQLBrowserUser$WIN-11-SERVER 1007
[*] 192.168.11.113: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Here, we can see that through enumerating SMB we have extracted two users: Administrator and hacker

## mpacket: Lookupsid 😊

A Security Identifier (SID) is a unique value of variable length that is used to identify a user account.

Through a SID User Enumeration, we can extract information about what users exist and their data.

Lookupsid script can enumerate both local and domain users. There is a Metasploit module too for this attack. If you are planning on injecting a target server with a golden or a silver ticket then one of the things that are required is the SID of the 500 user. Lookupsid.py can be used in that scenario.

When we provide the following parameters to the Lookupsid in such a format as shown below.

### Requirements:

- Domain
- Username
- Password/Password Hash
- Target IP Address.

`python3 lookupsid.py win-11-srver/Administrator:Anonymous_1212@192.168.11.113`

```
(root㉿kali)-[~/home/kali/Downloads] # python3 lookupsid.py win-11-srver/Administrator:Anonymous_1212@192.168.11.113
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Brute forcing SIDs at 192.168.11.113 Blame: Executable File - 286 lines (166 loc) ~ 7.95 KB
[*] StringBinding ncacn_np:192.168.11.113[\pipe\lsarpc]
[*] Domain SID is: S-1-5-21-2836125315-1793590764-2615395141
500: WIN-11-SERVER\Administrator (SidTypeUser) ← Collection of Python classes for working with network protocols
501: WIN-11-SERVER\Guest (SidTypeUser)
503: WIN-11-SERVER\DefaultAccount (SidTypeUser)
504: WIN-11-SERVER\WDAGUtilityAccount (SidTypeUser)
513: WIN-11-SERVER\None (SidTypeGroup)
1004: WIN-11-SERVER\test (SidTypeUser) ← All rights reserved.
1006: WIN-11-SERVER\haker (SidTypeUser) ← This software is provided under a slightly modified version of the GNU General Public License. See the LICENSE file.
1007: WIN-11-SERVER\SQLServer2005SQLBrowserUser$WIN-11-SERVER (SidTypeAlias)

(root㉿kali)-[~/home/kali/Downloads]
```

## SMB Enumeration: Enum4Linux 😊:

Enum4linux is a tool that developers designed to detect and extract data or enumerate from Windows and Linux operating systems, including SMB hosts on a network. Enum4linux can discover the following:

- Domain and group membership
- User listings
- Shares on a device (drives and folders)
- Password policies on a target
- The operating system of a remote target
- To begin with, we start a normal scan using enum4linux. It then extracts the RID Range, Usernames, Workgroup, Nbtstat Information, Sessions, SID Information, and OS Information

## enum4linux 192.168.11.113

```
(root㉿kali)-[~/home/kali/Downloads] Open Source ▾ Enterprise ▾ Pricing ▾ Search or jump to
# enum4linux 192.168.11.113
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Mon Feb 2 22:15:51 2026
  [ fortra / impacket ] ( Target Information )
  Target ip... Issues 192.168.11.113 tests 347 ▾ Discussions ▾ Actions ▾ Projects ▾ Security ▾ Insights
  RID Range ..... 500-550,1000-1050
  Username ..... ''
  Password ..... ''
  Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none
  [ master ] gabrielg5 Techdebt examples bootstrapping v2 (#1926) ▾
  [ 2026 ] ( Enumerating Workgroup/Domain on 192.168.11.113 )
  [ 2026 ] Code Blame Executable File 296 lines (166 loc) 7.96 KB
  [+] Got domain/workgroup name: WORKGROUP /usr/bin/python
    karmaSMB.py
  [+] keylistattack.py
  [+] kintercept.py
  Looking up status of 192.168.11.113
    WIN-11-SERVER <00> - B <ACTIVE> Workstation Service
    WORKGROUP <00> - <GROUP> B <ACTIVE> Domain/Workgroup Name
    WIN-11-SERVER <20> - B <ACTIVE> File Server Service
    WORKGROUP <1e> - <GROUP> B <ACTIVE> Browser Service Elections
  MAC Address = 08-00-27-DE-1E-0E
```

## enum4linux -a -u Administrator -p Anonymous\_1212 192.168.11.113

```
( Share Enumeration on 192.168.11.113 )
  Platform ▾ Solutions ▾ Resources ▾ Open Source ▾ Enterprise ▾ Pricing ▾ Search or jump to
do_connect: Connection to 192.168.11.113 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
  [ 2026 ] ( Share Enumeration on 192.168.11.113 )
  [ 2026 ] Platform Solutions Resources Open Source Enterprise Pricing ▾ Search or jump to
  [+] Sharename Type Comment
    ADMIN$ Disk Remote Admin
    C$ Disk Default share
    IPC$ IPC Remote IPC
    partage Disk Impacket / examples / lookupsid.py
    Public Disk
    SQL2025 Disk
    Users Disk
  Reconnecting with SMB1 for workgroup listing.
  Unable to connect with SMB1 -- no workgroup available
  [ 2026 ] ( Share Enumeration on 192.168.11.113 )
  [ 2026 ] Code Blame Executable File 296 lines (166 loc) 7.96 KB
  [+] Attempting to map shares on 192.168.11.113
  //192.168.11.113/ADMIN$ Mapping: OK Listing: OK Writing: N/A Python classes for working with network protocols
  [ E ] Can't understand response:
  [+] kintercept.py
    $Recycle.Bin DHS 0 Sat Jan 17 10:08:35 2026
    $WINDOWS.~BT DHn 0 Sun Jan 18 10:27:06 2026
    Documents and Settings DHSrn 0 Thu Jan 15 13:07:54 2026
    DumpStack.log.tmp AHS 12288 Tue Feb 3 04:41:24 2026
    pagefile.sys AHS 1476395008 Tue Feb 3 04:41:24 2026
    partage check.py D 0 Mon Feb 2 20:46:48 2026
    PerfLogs D 0 Mon Apr 1 03:02:26 2024
```

```
( Password Policy Information for 192.168.11.113 )  
forrta / impacket Public  
[+] Attaching to 192.168.11.113 using Administrator:Anonymous_1212  
[+] Trying protocol 139/SMB ...  
[!] Protocol failed: Cannot request session (Called Name:192.168.11.113)  
[+] Trying protocol 445/SMB ...  
[+] Found domain(s):  
[+] WIN-11-SERVER  
[+] Builtin  
[+] KarmaSMB.py  
[+] Password Info for Domain: WIN-11-SERVER  
[+] Minimum password length: None  
[+] Password history length: None  
[+] Maximum password age: 41 days 23 hours 53 minutes  
[+] Password Complexity Flags: 001001  
[+] Domain Refuse Password Change: 0  
[+] Domain Password Store Cleartext: 0  
[+] Domain Password Lockout Admins: 1
```

Finally, we have the Share Enumeration, which had the guest share that we enumerated earlier. Then we see that it attempted to enumerate inside the print share and IPC but faced restrictions. Then we have the Password Policy Information regarding the users on the system. It indicates whether the password was changed recently or if it has never been changed. It also tells us the complexity and other stuff regarding users and the operating system of the target system.

## Conclusion 😊

In this article, we discuss the various scripts and tools that can enumerate with the SMB/MSRPC services on a target system. Enumeration is the key step in order to compromise and in order to defend your system and network. Be sure to safeguard your SMB service.