1. What is SMB Protocol?

SMB (Server Message Block) is a **network file sharing protocol** primarily used in **Windows environments**. It allows users to:

- Share **files and folders**
- Use **shared printers**
- Perform **remote administration**

🧠 **2. How SMB Works (Internally)**

- Runs **over TCP port 445** (modern systems)
- Operates on a **client-server** model:
  - **Client** requests a file/resource
  - **Server** responds with access (or denial)

📦 **3. What Services Does SMB Provide?**

Service	Description
File Sharing	Access to shared files/folders over a network
Printer Sharing	Shared printers within LAN
Named Pipes	For inter-process communication
Remote Administration	Including registry access and managing services
Authentication & Session Control	Uses NTLM/Kerberos for login

---

🛠️ **4. Tools to Exploit or Use SMB in Pentesting**

Tool	Purpose
smbclient	Access SMB shares like an FTP client
smbmap	Lists all shares and permissions
enum4linux	Enumerates users/shares/domains (Linux)
smbclient.py (Impacket)	SMB client using Python
crackmapexec	All-in-one SMB enumeration, exploitation
nmap	With SMB scripts: version detection, vuln scan

Tool	Purpose
Metasploit	With modules like ms08_067_netapi, psexec
rpcclient	Query services via RPC over SMB
smbget	Download files via SMB (like wget for SMB)

📜 5. Common SMB Commands for Pentesting

- **smbclient (Interactive SMB shell):**

bash

CopyEdit

```
smbclient //IP/share -U username
```

Inside the shell:

bash

CopyEdit

```
ls          # list files
get <file>     # download
put <file>     # upload
cd <dir>       # change dir
```

- **smbmap:**

bash

CopyEdit

```
smbmap -H <IP>
smbmap -H <IP> -u <user> -p <pass>
```

- **enum4linux:**

bash

CopyEdit

```
enum4linux -a <IP>
```

- **crackmapexec (Powerful for Active Directory):**

bash

CopyEdit

```
crackmapexec smb <IP> -u <user> -p <pass> --shares
```

crackmapexec smb <IP> --users

🔹 **Nmap NSE scripts:**

bash

CopyEdit

nmap --script smb-enum-shares,smb-enum-users -p445 <IP>

🚨 6. SMB Vulnerabilities You Can Exploit

Vulnerability	Description
MS08-067	Remote code execution (RCE) via SMBv1 (old Win)
EternalBlue (MS17-010)	Buffer overflow in SMBv1
SMBGhost (CVE-2020-0796)	RCE in SMBv3
Null Sessions	Anonymous access to SMB info
Weak Permissions	Upload files or overwrite scripts
Password Reuse	NTLM hash reuse via SMB relay
IPC$ share	May leak sensitive info

⚠️ 7. Common Issues in SMB Pentesting

Problem	Solution
SMB version mismatch	Use tools that support SMB1, 2, or 3 (e.g., --option=client min protocol=NT1)
Authentication fails	Try anonymous login or brute-force with Hydra/Medusa
Access denied on shares	Check for misconfigured permissions with smbmap
Blocked port 445	Use **pivoting** or **proxychains**
AV/EDR detection	Obfuscate payloads and use encryption (SMB over SSH tunnel)

✅ How to Use SMB Effectively in Pentesting

1. **Enumerate first:**
    - Run nmap -p445 --script smb* <IP>

    - Use enum4linux, smbmap, smbclient
2. **Check for anonymous access**:
    - Try smbclient //IP/share -N
3. **Look for misconfigured shares**:
    - Upload payloads to accessible shares
4. **Check for RCE vulnerabilities**:
    - ms08_067_netapi, eternalblue, etc.
5. **Dump credentials if access is gained**:
    - Use secretsdump.py from Impacket
6. **Use Relay Attacks if in AD**:
    - Example: NTLM Relay with ntlmrelayx.py

What is smbclient?

Think of it like ftp but for Windows shares — it lets you connect to shared folders on Windows (and Samba) machines using the SMB protocol.

🛠️ **Basic Syntax:**

bash

CopyEdit

smbclient //<IP>/<SHARE> -U <username>

Example:

bash

CopyEdit

smbclient //10.10.10.10/public -U anonymous

📜 **All Commonly Used smbclient Commands (inside smb prompt):**

Once you're connected to the SMB share and inside the interactive smb: prompt, you can use the following commands:

Command	Description
ls	List files and directories in current SMB share

Command	Description
cd <dir>	Change directory
lcd <dir>	Change **local** directory (on your machine)
get <file>	Download file from SMB share
put <file>	Upload file to SMB share
mget *	Get multiple files (wildcard)
mput *	Upload multiple files
del <file>	Delete file on SMB share
rm <file>	Alias for del
mkdir <dir>	Create directory on SMB share
rmdir <dir>	Remove directory
prompt	Toggle prompting for mget/mput
recurse	Toggle recursive directory listing
exit / quit	Exit smbclient
help	Show available commands

🔐 Useful Options When Connecting

Option	Meaning
-U <user>	Username (e.g., anonymous, guest, user)
-N	No password prompt
-p <port>	Specify port (default SMB is 445)
-L <host>	List available shares (no connection needed)
-I <ip>	Specify IP address explicitly

Examples:

bash

CopyEdit

```
# List all available shares on a host
smbclient -L //10.10.10.10 -N
```

Connect to a share as anonymous

smbclient //10.10.10.10/public -U anonymous

🔍 **How Pentesters Use It:**

1. Enumerate Shares

bash

CopyEdit

smbclient -L //<target_ip> -N

2. Access Share

bash

CopyEdit

smbclient //<target_ip>/share_name -U anonymous

3. Browse and Download

bash

CopyEdit

ls

get interesting_file.txt

4. Upload Malicious File (e.g., reverse shell)

bash

CopyEdit

put shell.php

Then try to access it in the browser, like:
http://<target_ip>/shell.php

⚠️ **Common Problems Faced and Fixes:**

Problem	Solution
Share requires credentials	Try default creds, brute force, or enum4linux
Permission denied when uploading	Try another share, or privilege escalation
File doesn't show up on web server	Maybe wrong directory; try find via RCE or shell

Problem	Solution
Access denied when connecting	Use proper flags like -N, -U guest, or different username

---

📦 Services Provided by SMB

Service	Description
File Sharing	Share files/folders across a network
Printer Sharing	Access network printers
Authentication	NTLM/LDAP-based user login
Inter-process Communication (IPC$)	Named pipes and SMB messages for system info

Why Are There Different Tools for SMB?

SMB is used for file sharing, printer access, and inter-process communication on Windows/Linux systems. Because **enumeration**, **authentication**, **file access**, and **exploitation** are separate phases, one tool can't do everything efficiently or stealthily.

Think of it like this:

🔧 Some tools are screwdrivers, some are hammers — **smbclient is not a jack-of-all-trades**.

🧰 Key SMB Tools and When to Use Them

Tool	Use Case	Description
smbclient	Manual file browsing, upload/download	Like FTP for SMB; great for direct file share interaction after access
enum4linux	Enumeration (usernames, shares, OS, etc.)	Legacy tool similar to Windows enum; useful for pre-auth data gathering
smbmap	Share permissions mapping	Lists read/write/execute perms per share and user
crackmapexec	Brute force, user enumeration, exploit	Swiss army knife for SMB; works with creds, automates many tasks
rpcclient	User, group, policy enumeration	Command-line tool for low-level RPC interactions

Tool	Use Case	Description
smbclient.py (Impacket)	More advanced file interaction	Python version with more scripting/custom capabilities
nmap --script smb-*	Scanning and vulnerability detection	Great for automating recon or checking for specific vulns (MS17-010 etc.)
Metasploit	Exploitation of SMB vulns (EternalBlue)	Exploiting vulnerabilities like MS08-067, MS17-010

✅ Which to Use When?

Goal	Recommended Tool(s)
Check if SMB port open	nmap -p 139,445 <target>
Get users, shares, policies	enum4linux, smbmap, rpcclient
Check permissions for shares	smbmap, smbclient, crackmapexec
Log into a share & download files	smbclient, smbclient.py
Brute-force SMB login	hydra, crackmapexec, medusa
Exploit MS17-010 (EternalBlue)	Metasploit, nmap, impacket
Execute reverse shell or RCE	psexec.py (Impacket), Metasploit

---

🔥 Common SMB Problems in Pentesting

Problem	Cause / Fix
Access Denied / No share access	Need valid credentials or anonymous access isn't allowed
Can't upload file	Share is read-only — use smbmap to find writable shares
Exploit fails (e.g., EternalBlue)	Target is patched or incompatible — verify OS and patch version
Connection timeout or reset	Firewall or intrusion detection — use proxychains/VPN/obfuscation
Credentials not working	Wrong hash format or account locked — try spraying or guessing

---

🧠 How to Make SMB Usage Effective

1. **Start with Enumeration**: Use nmap, enum4linux, smbmap to find what's open.

2. **Check Share Permissions**: Use smbmap -H <IP> to find read/write/executable shares.
3. **Try Logging in**: Use smbclient //IP/share -U user or crackmapexec.
4. **Upload/Download Payloads**: If write access is allowed, upload web shells/rev shells.
5. **Exploit**: Use Metasploit or psexec.py if creds are available or exploit possible.