APT2

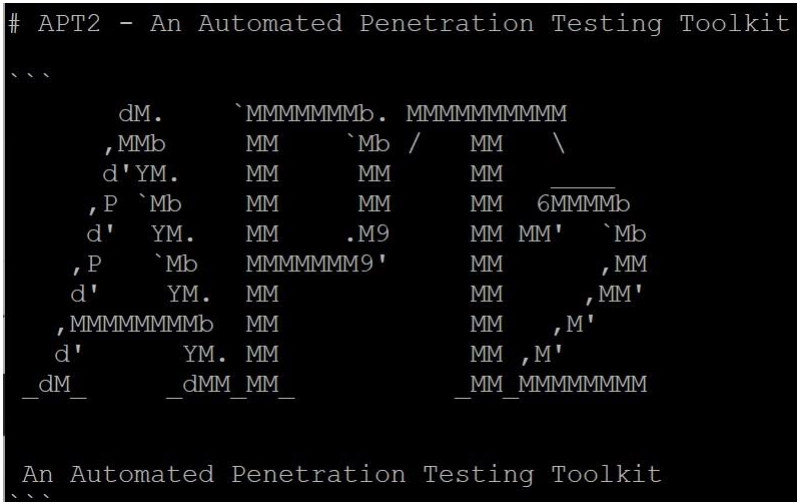
Information Security Inc.

Contents

- About APT2
- Features
- Modules
- Requirements
- Setup
- Installing APT2
- Using APT2
- References

About APT2

- APT2 is an Automated Penetration Testing Toolkit



Features

- This tool will perform an NMap scan, or import the results of a scan from Nexpose, Nessus, or Nmap
- The processed results will be used to launch exploit and enumeration modules according to the configurable Safe Level and enumerated service information

Modules

```
------------------------
LIST OF CURRENT MODULES
------------------------
nmaploadxml                Load NMap XML File
hydrasmbpassword           Attempt to bruteforce SMB passwords
nullsessionrpcclient       Test for NULL Session
msf_snmpenumshares         Enumerate SMB Shares via LanManager OID Values
nmapbasescan               Standard NMap Scan
impacketsecretsdump        Test for NULL Session
msf_dumphashes             Gather hashes from MSF Sessions
msf_smbuserenum            Get List of Users From SMB
anonftp                    Test for Anonymous FTP
searchnfsshare             Search files on NFS Shares
crackPasswordHashJohnTR    Attempt to crack any password hashes
msf_vncnoneauth            Detect VNC Services with the None authentication type
nmapsslscan                NMap SSL Scan
nmapsmbsigning             NMap SMB-Signing Scan
responder                  Run Responder and watch for hashes
msf_openx11                Attempt Login To Open X11 Service
nmapvncbrute               NMap VNC Brute Scan
msf_gathersessioninfo      Get Info about any new sessions
nmapsmbshares              NMap SMB Share Scan
userenumrpcclient          Get List of Users From SMB
httpscreenshot             Get Screen Shot of Web Pages
httpserverversion          Get HTTP Server Version
nullsessionsmbclient       Test for NULL Session
openx11                    Attempt Login To Open X11 Servicei and Get Screenshot
msf_snmplogin              Attempt Login Using Common Community Strings
msf_snmpenumusers          Enumerate Local User Accounts Using LanManager/psProcessUsername OID Values
httpoptions                Get HTTP Options
nmapnfsshares              NMap NFS Share Scan
msf_javarmi                Attempt to Exploit A Java RMI Service
anonldap                   Test for Anonymous LDAP Searches
ssltestsslserver           Determine SSL protocols and ciphers
gethostname                Determine the hostname for each IP
sslsslscan                 Determine SSL protocols and ciphers
nmapms08067scan            NMap MS08-067 Scan
msf_ms08_067               Attempt to exploit MS08-067
```

Requirements

- convert, dirb, hydra, java, john, ldapsearch, msfconsole, nmap, nmblookup, phantomjs, responder, rpcclient, secretsdump.py, smbclient, snmpwalk, sslscan, x11-apps

- Kali Linux users => phantomjs, secretsdump.py (https://github.com/CoreSecurity/impacket/blob/master/examples/secretsdump.py) and x11-apps

Setup

- Kali Linux 2017 "2017.2“ 64 bit (https://www.kali.org/downloads/)

- On Kali Linux install python-nmap library: python setup.py install

Installing APT2

- Installing dependencies

```
root@kali2017:~# apt-get install phantomjs x11-apps
Reading package lists... Done
Building dependency tree
Reading state information... Done
x11-apps is already the newest version (7.7+6+b1).
x11-apps set to manually installed.
The following NEW packages will be installed:
  phantomjs
0 upgraded, 1 newly installed, 0 to remove and 227 not upgraded.
Need to get 302 kB of archives.
After this operation, 943 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ftp.ne.jp/Linux/packages/kali/kali kali-rolling/main amd64 phantomjs amd64 2.1.1+dfsg-2 [3
02 kB]
Fetched 302 kB in 2s (104 kB/s)
Selecting previously unselected package phantomjs.
(Reading database ... 316258 files and directories currently installed.)
Preparing to unpack .../phantomjs_2.1.1+dfsg-2_amd64.deb ...
Unpacking phantomjs (2.1.1+dfsg-2) ...
Setting up phantomjs (2.1.1+dfsg-2) ...
Processing triggers for man-db (2.7.6.1-2) ...
```

Installing APT2

- Installing dependencies (secretsdump.py)

```
root@kali2017:~# git clone https://github.com/CoreSecurity/impacket
Cloning into 'impacket'...
remote: Counting objects: 11734, done.
remote: Compressing objects: 100% (55/55), done.
remote: Total 11734 (delta 43), reused 55 (delta 26), pack-reused 11653
Receiving objects: 100% (11734/11734), 3.99 MiB | 1.65 MiB/s, done.
Resolving deltas: 100% (8872/8872), done.
root@kali2017:~# cd impacket/
root@kali2017:~/impacket# python setup.py install
running install
running build
running build_py
creating build
creating build/lib.linux-x86_64-2.7
```

Installing APT2

- Clone it

```
root@kali2017:~# git clone https://github.com/MooseDojo/apt2
Cloning into 'apt2'...
remote: Counting objects: 818, done.
remote: Total 818 (delta 0), reused 0 (delta 0), pack-reused 818
Receiving objects: 100% (818/818), 211.20 KiB | 350.00 KiB/s, done.
Resolving deltas: 100% (542/542), done.
```

Installing APT2

- Install python-nmap library

```
root@kali2017:~# cd apt2/
root@kali2017:~/apt2# python setup.py install
running install
running bdist_egg
running egg_info
creating apt2.egg-info
writing requirements to apt2.egg-info/requires.txt
writing apt2.egg-info/PKG-INFO
```

Using APT2

- APT2 options

```
root@kali2017:~/apt2# ./apt2.py -h
usage: apt2.py [-h] [-C <config.txt>] [-f [<input file> [<input file> ...]]]
               [--target] [--ip <local IP>] [-v] [-s SAFE_LEVEL]
               [-x EXCLUDE_TYPES] [-b] [--listmodules]

optional arguments:
  -h, --help            show this help message and exit
  -v, --verbosity       increase output verbosity
  -s SAFE_LEVEL, --safelevel SAFE_LEVEL
                        set min safe level for modules. 0 is unsafe and 5 is
                        very safe. Default is 4
  -x EXCLUDE_TYPES, --exclude EXCLUDE_TYPES
                        specify a comma seperatec list of module types to
                        exclude from running
  -b, --bypassmenu      bypass menu and run from command line arguments

inputs:
  -C <config.txt>       config file
  -f [<input file> [<input file> ...]]
                        one of more input files seperated by spaces
  --target              initial scan target(s)

advanced:
  --ip <local IP>       defaults to 192.168.10.12

misc:
  --listmodules         list out all current modules and exit
```

Using APT2

- If getting the following error when starting running APT2

```
[!] Could not connect to Metasploit msgrpc service with the following parameters:
[!]      host     = [127.0.0.1]
[!]      port     = [55552]
[!]      user     = [msf]
[!]      password = [msfpass]
[*] If you wish to make use of Metasploit modules within APT2, please update the config file with the appropiate settings.
[!] Connect by launching msfconsole and then issue the following command:
[!]      load msgrpc User=msf Pass=msfpass ServerPort=55552
```

Using APT2

- Resolve it by configuring Metasploit as below

```
msf > load msgrpc User=msf Pass=msfpass ServerPort=55552
[*] MSGRPC Service:  127.0.0.1:55552
[*] MSGRPC Username: msf
[*] MSGRPC Password: msfpass
[*] Successfully loaded plugin: msgrpc
```

Using APT2

- If getting the following error when starting running APT2

“ Module 'apt2_whois' disabled. Dependency required: 'Missing whois library. To install run: pip install whois‘ “

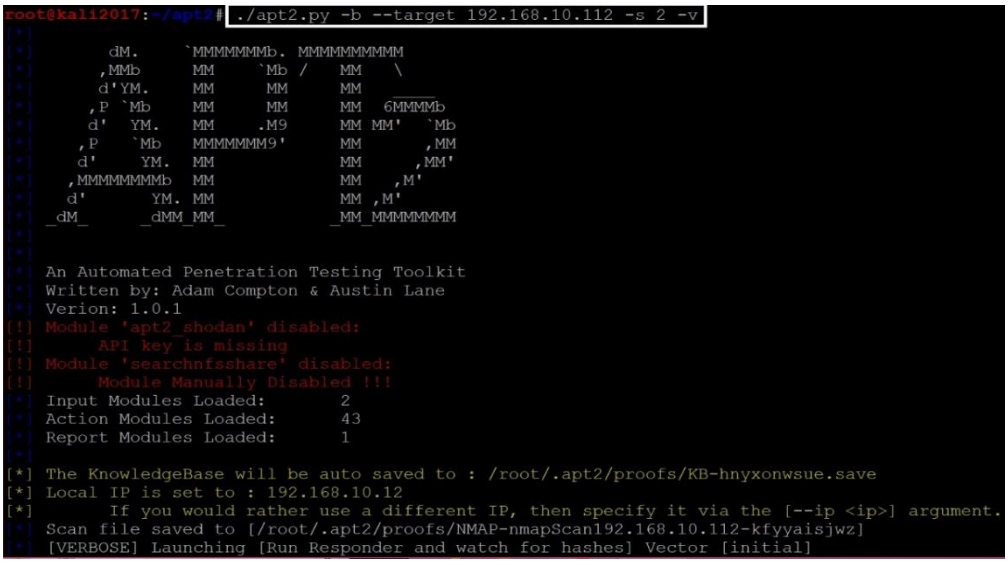
Using APT2

- Resolve it by installing “whois” library

```
root@kali2017:~# pip install whois
Collecting whois
  Downloading whois-0.7.tar.gz
Building wheels for collected packages: whois
  Running setup.py bdist_wheel for whois ... done
  Stored in directory: /root/.cache/pip/wheels/5f/70/36/0e4d6e97c8bebbbe8a725566b9894f4e06074ac5b500890fc2
Successfully built whois
Installing collected packages: whois
Successfully installed whois-0.7
```

Using APT2

- APT2 ran against a target machine (IP 192.168.10.112)

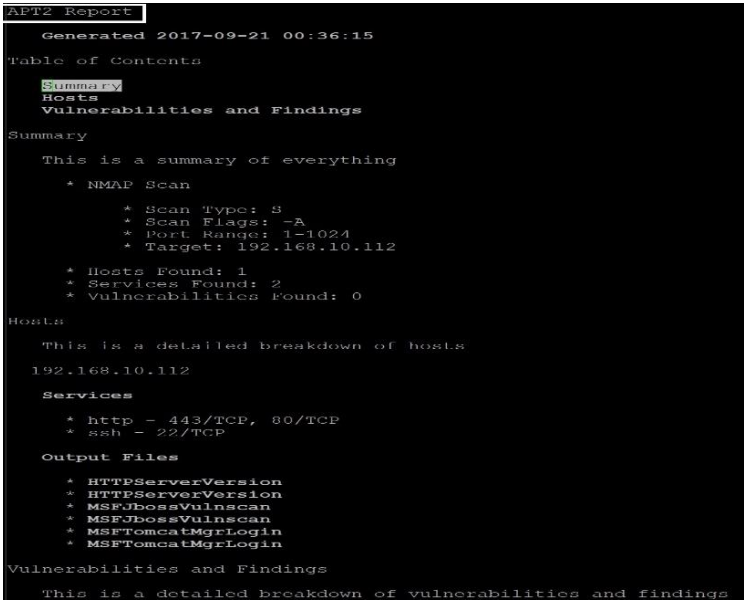


Using APT2

- APT2 ran against a target machine (IP 192.168.10.112)
- Generated reports

Using APT2

- APT2 ran against a target machine (IP 192.168.10.112)
- Generated reports



Using APT2

- APT2 ran against a target machine (IP 192.168.10.112)
- Generated reports

```
Identified Server Version of 192.168.10.112 : Apache/2.4.16 (Fedora) OpenSSL/1.0.2d-fips PHP/5.6.14

Full Headers:
date: Thu, 21 Sep 2017 13:21:15 GMT
content-length: 362
content-type: text/html; charset=iso-8859-1
connection: close
server: Apache/2.4.16 (Fedora) OpenSSL/1.0.2d-fips PHP/5.6.14
```

References

- GitHub
https://github.com/MooseDojo/apt2

- Kali Linux
https://www.kali.org/downloads/

- Secretsdump.py
https://github.com/CoreSecurity/impacket/blob/master/examples/secretsdump.py