Tactical Nuke

Tactical Nuke is a tool that acts as a stealthier alternative to Armitage's hail Mary function. Among other features it can perform reconnaissance, scanning, exploitation, and poist exploitation as a comprehensive component of any penetration testing suite. It is built as a command line application using python and has a modular architecture, to stream line the process. It also interacts with the metasploit database through a python interface (pymetasploit3).

The Purpose of Tactical Nuke was to a design an automated penetration testing suite that could be understood and operated by those that did not have a comprehensive grasp of how to infiltrate a system.

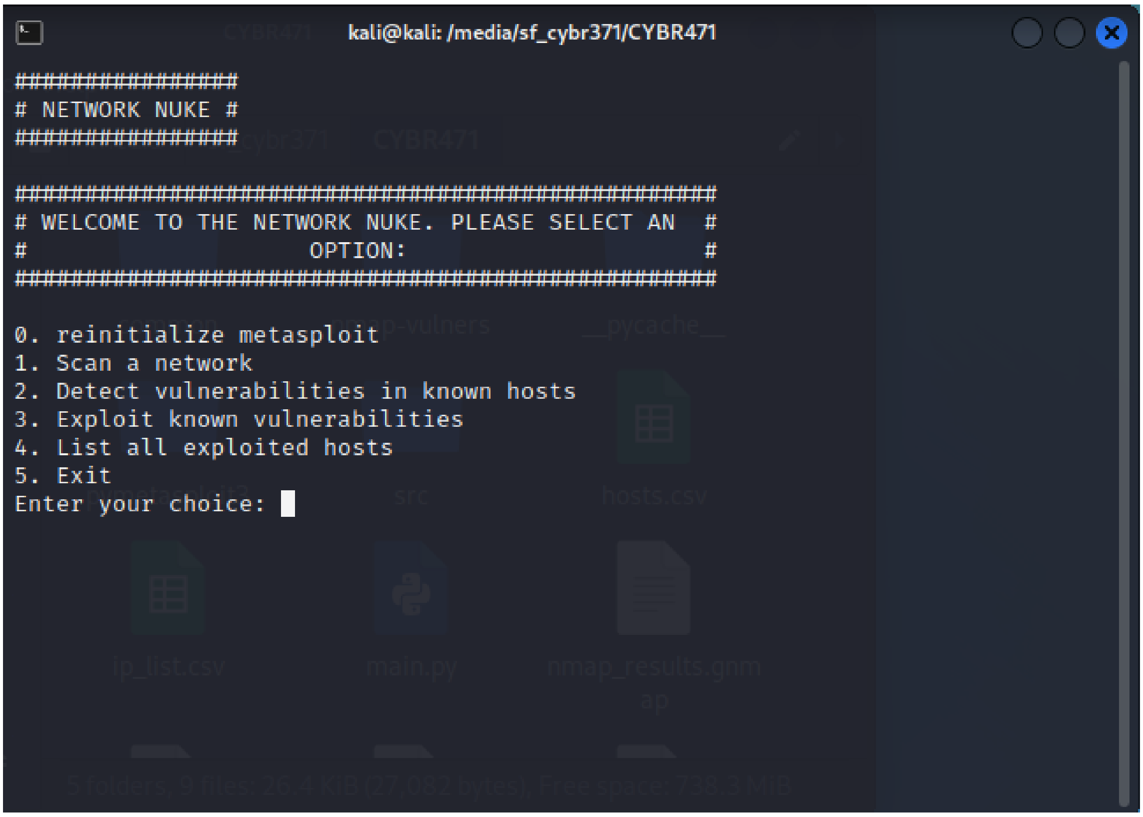
This starts from simply opening the script, where Tactical Nuke comes with most of the neccessary libraries pre installed, and a requirments file for the user to install with pip as follows (provided they have a recent installation of python3 and pip3):

```
cd [Project Dir]
pip install -r requirements.txt
```

After this, all the user needs to do is run the `main.py` script to start the script, which will start all the necessary services for them.

```
python3 main.py
```

Upon launching the Program, the user is met with a pleasant and simplistic interface (11) which only contains the important information such that the user does not become overwhelmed.



```
################
# NETWORK NUKE #
################

###################################################
# WELCOME TO THE NETWORK NUKE. PLEASE SELECT AN   #
#                    OPTION:                      #
###################################################

0. reinitialize metasploit
1. Scan a network
2. Detect vulnerabilities in known hosts
3. Exploit known vulnerabilities
4. List all exploited hosts
5. Exit
Enter your choice: 
```

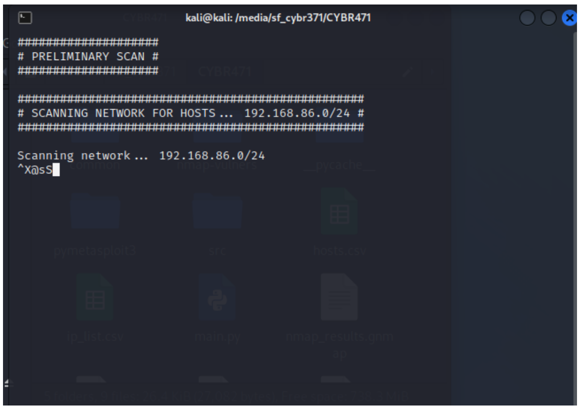
these have been broken down as follows:

Reinitialize metasploit

This is a function for if you wish to refresh your database, or restart the console. Occasionally the metasploit library runs into problems so this is the easiest way to start fresh. Some debugging text will be written on the screen before the user is brought back to to this landing page - which operates as the central hub of the program.

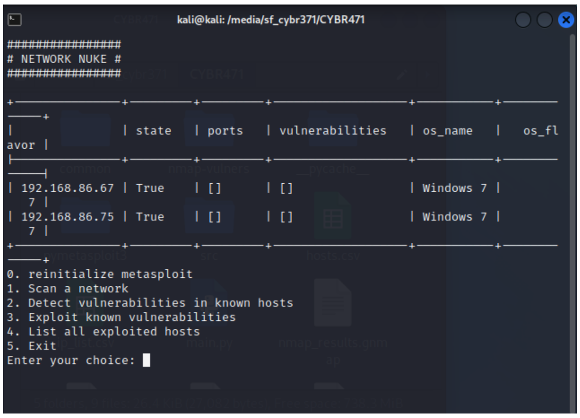
Scan a network

The First task is **Reconnaisence** which can be started by pressing 1 to scan the network. The user will then be asked to give an ip or range of ip's for nmap to scan. This is a simple a precursory nmap scan to identify the windows devices on the network and so runs very quickly.



```
###################
# PRELIMINARY SCAN #
###################

#################################################
# SCANNING NETWORK FOR HOSTS ... 192.168.86.0/24 #
#################################################

Scanning network ... 192.168.86.0/24
^X@sS
```

Running Scan



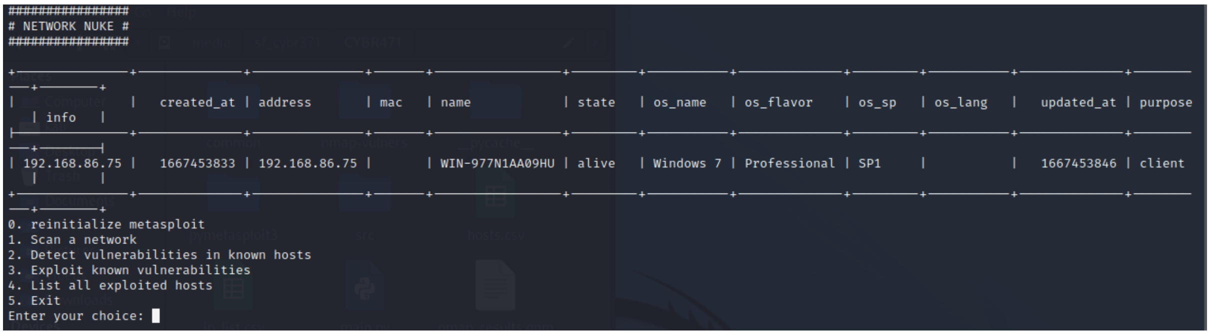
```
################
# NETWORK NUKE #
################

+───────────────+─────────+─────────+──────────────────+────────────+────────
─────+
|               | state   | ports   | vulnerabilities  | os_name    |   os_fl
avor |
├───────────────+─────────+─────────+──────────────────+────────────+────────
─────┤
| 192.168.86.67 | True    | []      | []               | Windows 7 |
   7 |
| 192.168.86.75 | True    | []      | []               | Windows 7 |
   7 |
+───────────────+─────────+─────────+──────────────────+────────────+────────
─────+
0. reinitialize metasploit
1. Scan a network
2. Detect vulnerabilities in known hosts
3. Exploit known vulnerabilities
4. List all exploited hosts
5. Exit
Enter your choice:
```

Scan Results

After the scan has completed, the user will again be returned to the hub where they can see the initial results of the scan. This is simply output to a csv for now.

Vulnerability Detection

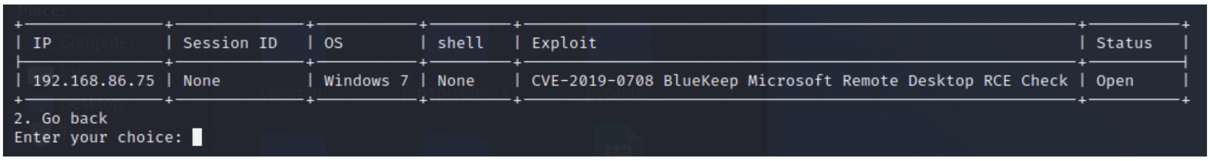
The next step is to do a deaper scan of all the ports on the devices returned by our first scan. This is completed by pressing 2 for “Detect Vulnerabilities in known hosts”. This will find the ports and the services running on them, as well as some vulerabilities we can exploit later. This was also achieved using the **nmap** library and then adding the results to the **metasploit** db.



```
################
# NETWORK NUKE #
################

+───────────────+──────────────+───────────────+───────+──────────────────+─────────+──────────+──────────────+─────────+──────────+──────────────+────────
──+────────+
|               |   created_at | address       | mac   | name             | state   | os_name  | os_flavor    | os_sp   | os_lang  |   updated_at | purpose
  | info   |
├───────────────+──────────────+───────────────+───────+──────────────────+─────────+──────────+──────────────+─────────+──────────+──────────────+────────
──+────────┤
| 192.168.86.75 |   1667453833 | 192.168.86.75 |       | WIN-977N1AA09HU | alive   | Windows 7 | Professional | SP1     |          |   1667453846 | client
  |        |
+───────────────+──────────────+───────────────+───────+──────────────────+─────────+──────────+──────────────+─────────+──────────+──────────────+────────
──+────────+
0. reinitialize metasploit
1. Scan a network
2. Detect vulnerabilities in known hosts
3. Exploit known vulnerabilities
4. List all exploited hosts
5. Exit
Enter your choice:
```

These are also checked against the scripts at this repository - which is automatically installed and placed in the correct folder.

In 4 we can see our hosts with the reccomended exploit based on vulnerabilities identified

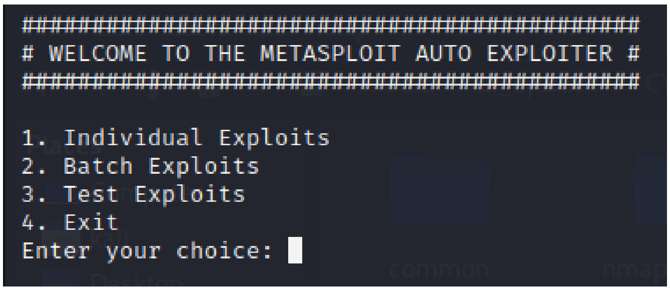


```
+───────────────+────────────+───────────+───────+──────────────────────────────────────────────────────────+────────+
| IP            | Session ID | OS        | shell | Exploit                                                  | Status |
├───────────────+────────────+───────────+───────+──────────────────────────────────────────────────────────+────────┤
| 192.168.86.75 | None       | Windows 7 | None  | CVE-2019-0708 BlueKeep Microsoft Remote Desktop RCE Check | Open   |
+───────────────+────────────+───────────+───────+──────────────────────────────────────────────────────────+────────+
2. Go back
Enter your choice:
```

❓ Aside: I have also implemented this table in Part 4 which is to list the exploited hosts - At this early stage the table looks quite different as It is centered around whether a device is listening on its ports. You may have also noticed we have lost a device. That is because I have a flatmate with a windows 7 computer and didnt want to attack it. I, unfortunatly, do not have the storage space for 3 vm's, nor the memory.

Exploitation

We are now ready to exploit our Targets. All this information is stored within our database and so all we have to is press 3.



```
##############################################
# WELCOME TO THE METASPLOIT AUTO EXPLOITER #
##############################################

1. Individual Exploits
2. Batch Exploits
3. Test Exploits
4. Exit
Enter your choice:
```

At this point we are presented with more options, which I will briefly discuss

Individual Exploits

These consist of:

1. Eternal Romance
2. BlueKeep
3. EternalBlue
4. A Microsoft IIS Script
5. and Eternal Champion (which just points to eternal romance)

These are the core building blocks and combinations that will be used later - but effectively here the use is given the opprtunity to try them individually.

Batch Exploits

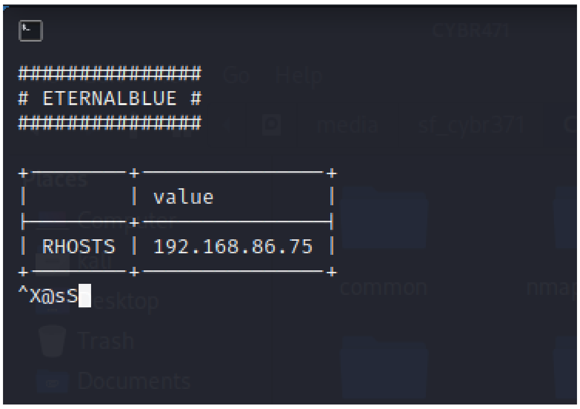
1. All Eternal based scripts
2. All the "Miscelanious" scripts (i.e. the other 2)
3. All Test - which runs all these test scripts
4. And finally, our namesake, the nuclear option, which attempts to run every script in the metasploit database which our target could be vulnerable to (decided by a filtered search based on ports, services, platform, etc)

Test Exploits

1. Broken Script - simply a script that runs without arguments (BlueBook) to test the error handling
2. All the test scripts for testing reasons
3. and finally the true automation which - unfortunately - tends to totally destroy whichever device it is tested against and fails to open and maintain any sessions - it is a WIP.

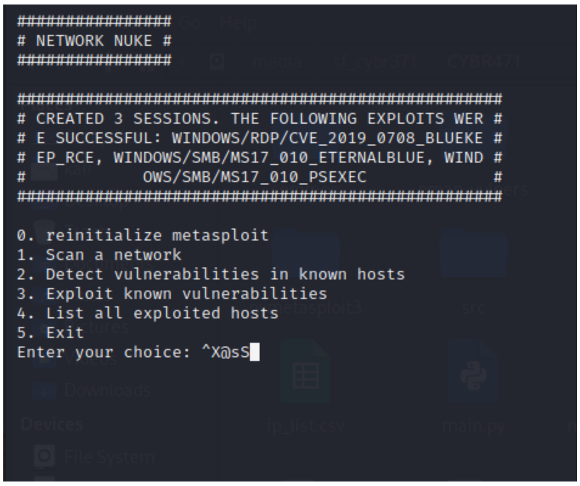
Running Exploits

To run these exploits you follow the menus until 1 or many are selected, and then wait for the process to complete.



```
###############
# ETERNALBLUE #
###############

+---------+--------------+
|         | value        |
+---------+--------------+
| RHOSTS  | 192.168.86.75 |
+---------+--------------+
^X@sS
```

Attempting Exploitation with the parameters listed



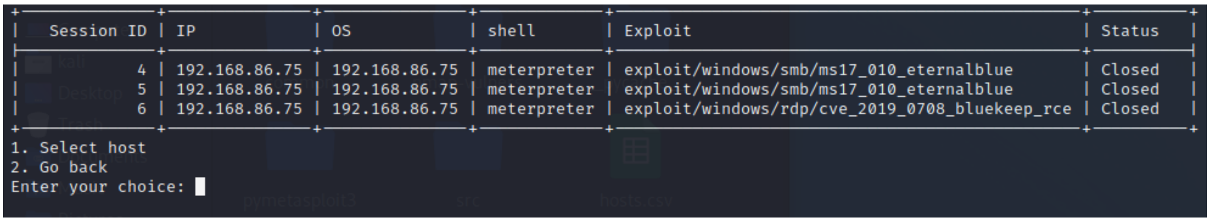
```
################
# NETWORK NUKE #
################

##################################################
# CREATED 3 SESSIONS. THE FOLLOWING EXPLOITS WER #
# E SUCCESSFUL: WINDOWS/RDP/CVE_2019_0708_BLUEKE #
# EP_RCE, WINDOWS/SMB/MS17_010_ETERNALBLUE, WIND #
#          OWS/SMB/MS17_010_PSEXEC               #
##################################################

0. reinitialize metasploit
1. Scan a network
2. Detect vulnerabilities in known hosts
3. Exploit known vulnerabilities
4. List all exploited hosts
5. Exit
Enter your choice: ^X@sS
```

Completed scan detailing the number of sessions created, and which exploits succeeded

The majority of the predefined exploits are looking at port 445 (ie the eternals) and all perform a form of **reverse tcp** (the server - us - making a connection request which is not filtered out by the windows 7 firewall). These are all performed in a serialized manner, waiting for one another to end, on the same console.

> ⚠️ Not all Exploits will succeed, but DONT WORRY! That is totally expected (in fact planned in some cases - its more fun). The program has accounted for this and will just move on to the next one.

All these exploits are instructed to send the resultant meterpreter session to the background and can be accessed from the same menu as where the vulnerabilities were shown - however we now see the exploit used along with the session ID, and the shell. (We do seem to have lost the OS which i'm just noticing which is a shame). We can also see there are all closed, and this is because I made a mistake and forgot to update the table when I added sessions - It is still based on ports and the sum of equalities that gets to a truth value (with 4 factors) has somehow resulted in this.



```
+--------------+----------------+----------------+-------------+---------------------------------------------+----------+
|   Session ID | IP             | OS             | shell       | Exploit                                     | Status   |
|--------------+----------------+----------------+-------------+---------------------------------------------+----------|
|            4 | 192.168.86.75  | 192.168.86.75  | meterpreter | exploit/windows/smb/ms17_010_eternalblue    | Closed   |
|            5 | 192.168.86.75  | 192.168.86.75  | meterpreter | exploit/windows/smb/ms17_010_eternalblue    | Closed   |
|            6 | 192.168.86.75  | 192.168.86.75  | meterpreter | exploit/windows/rdp/cve_2019_0708_bluekeep_rce | Closed   |
+--------------+----------------+----------------+-------------+---------------------------------------------+----------+
1. Select host
2. Go back
Enter your choice: █
```

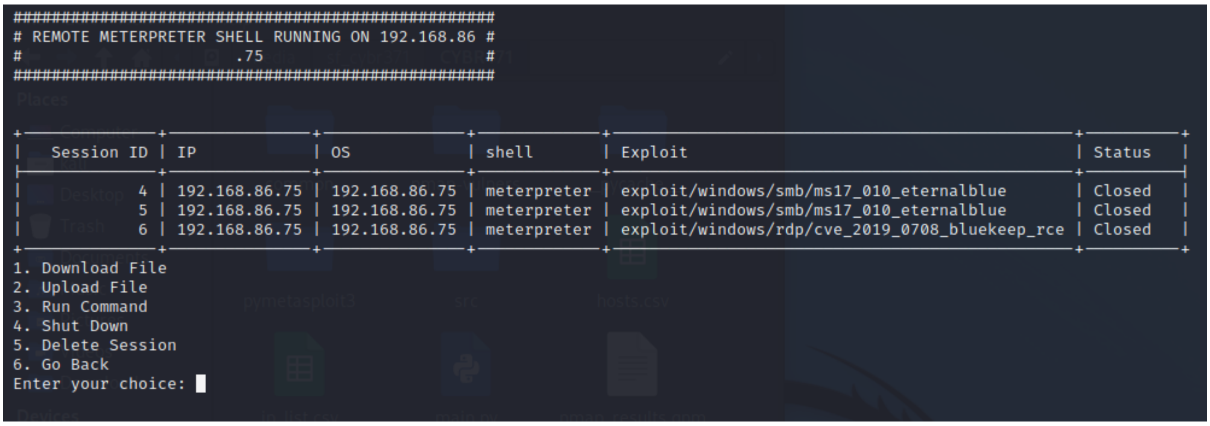
We now have access to these machines and can start messing with them a bit.

> ❓ Another aside - each terminal is limited to one console, which is closed when they exit, however multiple terminals can be opened and act in parallel. Problems only arise here when people are using different versions of the code (I.e. I had one version where one would delete the others console which was problematic for state)

Post Exploitation

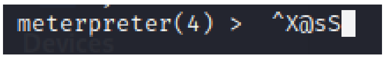
> This is where the ***fun begins - Sywalker, Anakin Mortimer (Darth Vader)***

The first task to to choose to choose a host. Encouraging a proactive outlook from my user base is an important component of my software design process. After this we are taken to a very similar screen Which shows the available operations. I have included shortcuts for shutting down the machine, as well as upload/download of files, and deleting the session as these are common operations I have found users wish to perform. The user also has access to a shell which will give them the full power of meterpreter and powershell.



```
##################################################
# REMOTE METERPRETER SHELL RUNNING ON 192.168.86 #
#                   .75                          #
##################################################

+--------------+----------------+----------------+-------------+---------------------------------------------+----------+
|   Session ID | IP             | OS             | shell       | Exploit                                     | Status   |
|--------------+----------------+----------------+-------------+---------------------------------------------+----------|
|            4 | 192.168.86.75  | 192.168.86.75  | meterpreter | exploit/windows/smb/ms17_010_eternalblue    | Closed   |
|            5 | 192.168.86.75  | 192.168.86.75  | meterpreter | exploit/windows/smb/ms17_010_eternalblue    | Closed   |
|            6 | 192.168.86.75  | 192.168.86.75  | meterpreter | exploit/windows/rdp/cve_2019_0708_bluekeep_rce | Closed   |
+--------------+----------------+----------------+-------------+---------------------------------------------+----------+
1. Download File
2. Upload File
3. Run Command
4. Shut Down
5. Delete Session
6. Go Back
Enter your choice: █
```

Interacting with remote meterpreter session (4)



```
meterpreter(4) >  ^X@sS█
```

command line interface

From here the use can perform any activity their heart desires, but the core ones shown above are detailed below:

Download File

This allows the user to download a file using meterpreter. They must specify the remote path for the download.

Upload File

Similar to Download, this allows the user to upload a file using meterpreter. They must specify both the local path to the file and the remote path where it should be deployed.

Shut Down

This is primarily to ensure persistence of the shell. It will persist itself and then shut down the device - because to be alive is to risk it all. The user should see this target become unavailable during this time.

Delete Session

This will stop the session and drop it from the table.

Run Command

This is how the user interacts with the target devices. the following are good jumping off points in terms of further reading - I found them immensely helpful.

Meterpreter Basic Commands | Offensive Security

Since the Meterpreter provides a whole new environment, we will cover some of the basic Meterpreter commands to get you started and help

https://www.offensive-security.com/metasploit-unleashed/meterpreter-basics/

nsole Commands | Offensive Security

u have finished working with a particular or if you inadvertently select the wrong you can issue the back command to move

s://www.offensive-security.com/metasploit ned/msfconsole-commands/