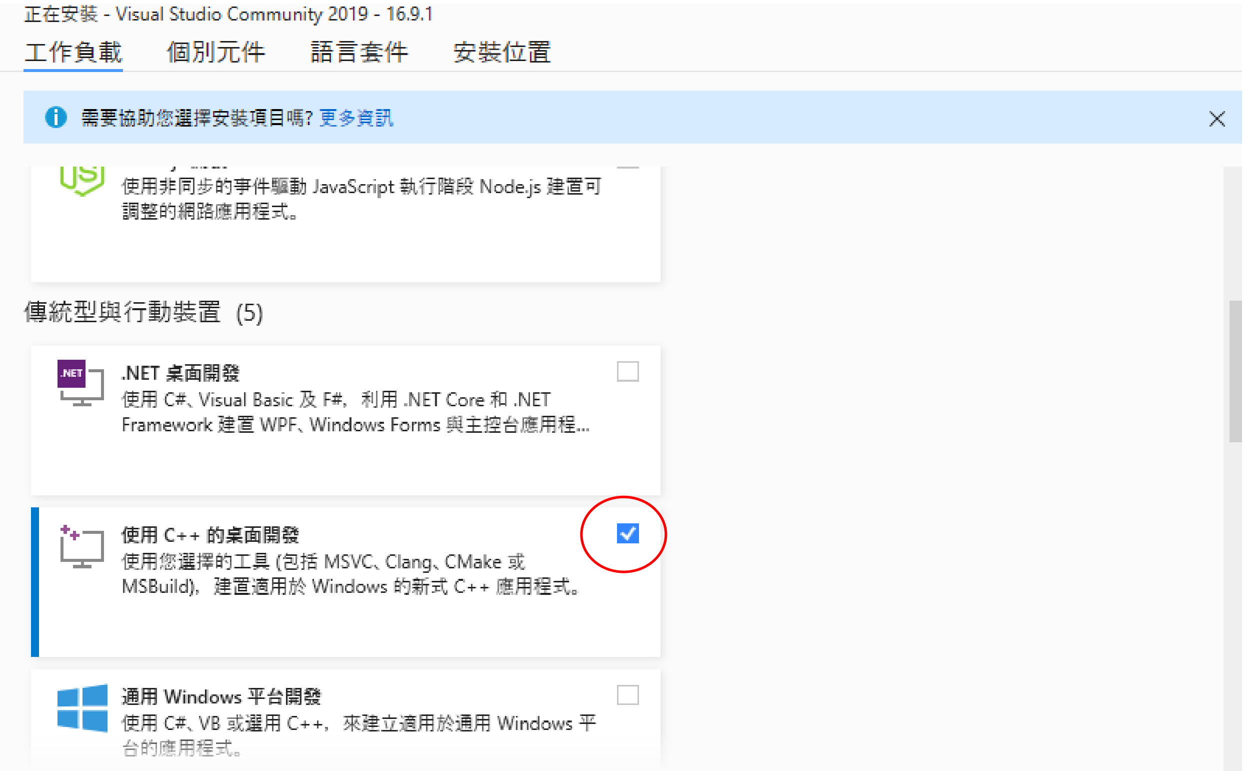
Network Security Practices- Attack and defense HW4

Part1. Installing the kernel rootkit

- In this part, you have to install the windows kernel rootkit, so a Windows environment (Win10 home edition 64bit) is needed. It is recommended to run it in a virtual machine (Virtualbox, Vmware, etc).
- Minimum hardware requirments of a VM:
  - 4GB ram
  - 45GB of hard drive space

Visual Studio

- Download Visual Studio 2019
- Choose the "Desktop development with C++" workload.



正在安裝 - Visual Studio Community 2019 - 16.9.1

需要協助您選擇安裝項目嗎? 更多資訊

使用非同步的事件驅動 JavaScript 執行階段 Node.js 建置可調整的網路應用程式。

傳統型與行動裝置 (5)

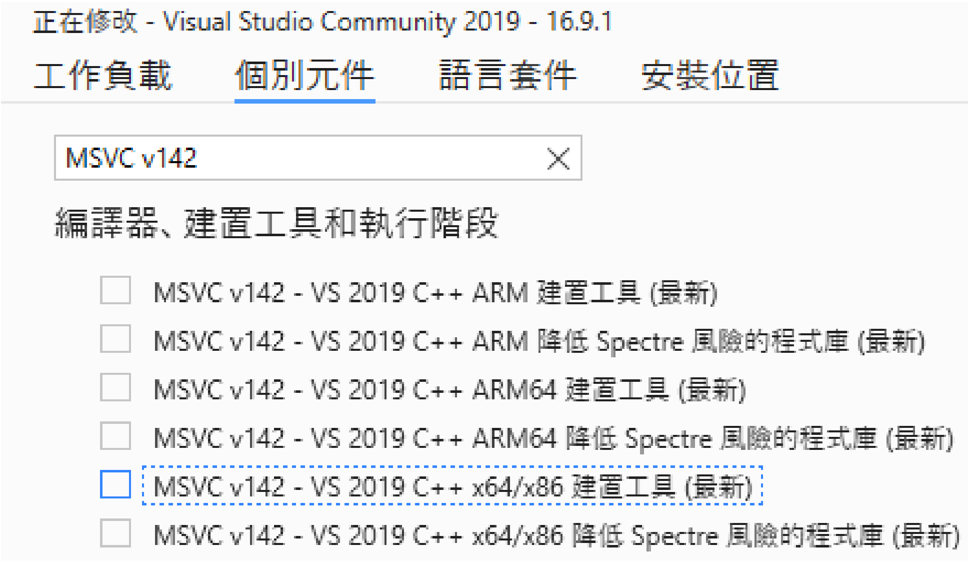
□ .NET 桌面開發
使用 C#、Visual Basic 及 F#，利用 .NET Core 和 .NET Framework 建置 WPF、Windows Forms 與主控台應用程...

☑ 使用 C++ 的桌面開發
使用您選擇的工具 (包括 MSVC、Clang、CMake 或 MSBuild)，建置適用於 Windows 的新式 C++ 應用程式。

□ 通用 Windows 平台開發
使用 C#、VB 或選用 C++，來建立適用於通用 Windows 平台的應用程式。

WDK

- Follow the documentation to install the Windows Driver Kit for Visual Studio 2019.
- For the step 1, If your ***MSVC v142 - VS 2019 C++ x64/x86 build tools*** is the latest version, you can simply install the corresponding architecture of ***Spectre-mitigated libs (Latest)*** and **build tools (Latest).**
- Skip the step 1.5.
- You can find WDK in Visual Studio.

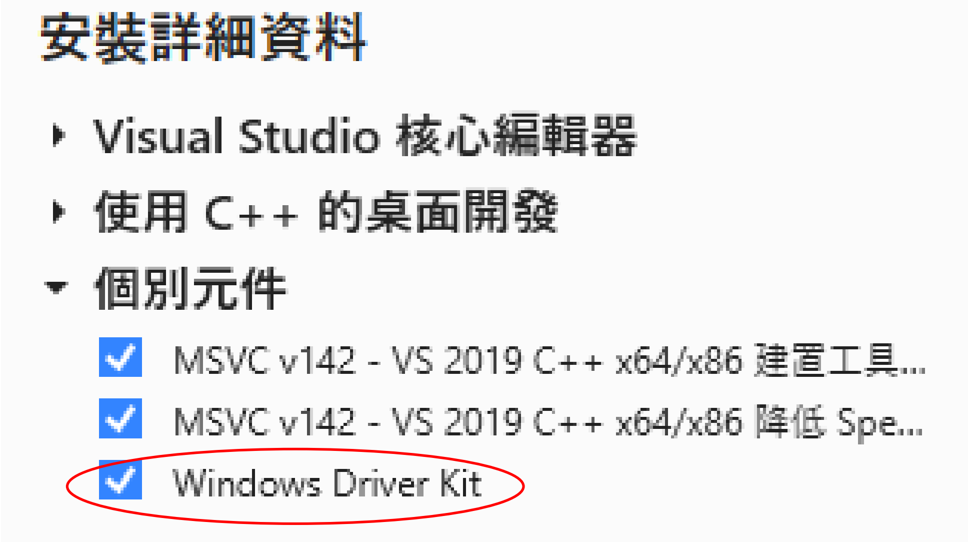


正在修改 - Visual Studio Community 2019 - 16.9.1

MSVC v142

編譯器、建置工具和執行階段

- □ MSVC v142 - VS 2019 C++ ARM 建置工具 (最新)
- □ MSVC v142 - VS 2019 C++ ARM 降低 Spectre 風險的程式庫 (最新)
- □ MSVC v142 - VS 2019 C++ ARM64 建置工具 (最新)
- □ MSVC v142 - VS 2019 C++ ARM64 降低 Spectre 風險的程式庫 (最新)
- □ MSVC v142 - VS 2019 C++ x64/x86 建置工具 (最新)
- □ MSVC v142 - VS 2019 C++ x64/x86 降低 Spectre 風險的程式庫 (最新)

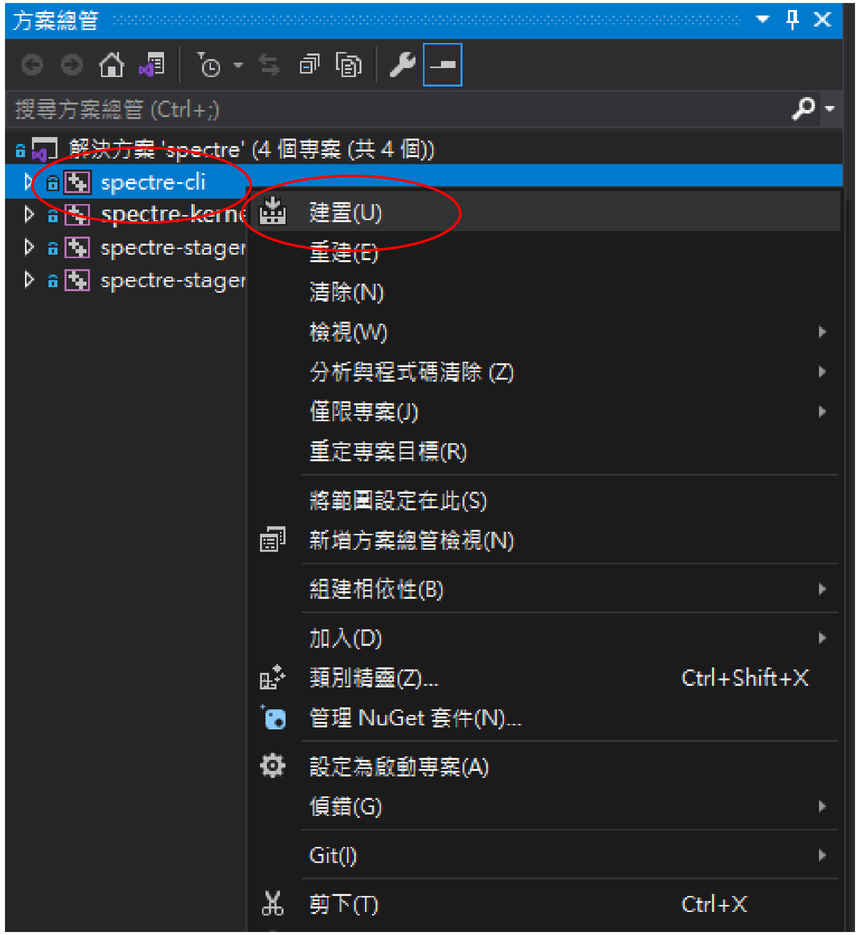


安裝詳細資料

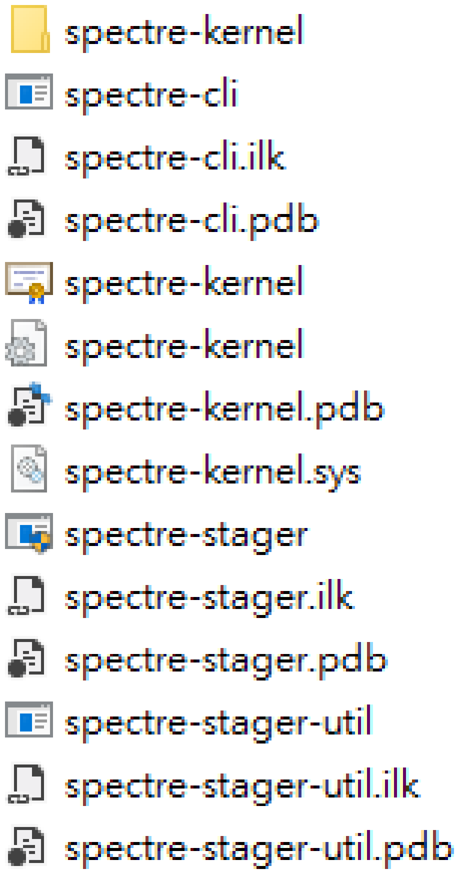
- ▸ Visual Studio 核心編輯器
- ▸ 使用 C++ 的桌面開發
- ▾ 個別元件
  - ☑ MSVC v142 - VS 2019 C++ x64/x86 建置工具...
  - ☑ MSVC v142 - VS 2019 C++ x64/x86 降低 Spe...
  - ☑ Windows Driver Kit

Download and build the rootkit

- Download the rootkit from https://github.com/D4stiny/spectre.
- Enter the folder, open ***spectre.sln*** in the ***spetre*** folder.
- Build the 4 projects inside of the solution in the following order: ***spectre-cli, spectre-stager-util, spectre-stager, spectre-kernel.***



- In the ***spectre/x64/Debug*** folder, there should be the following files:



spectre-kernel
spectre-cli
spectre-cli.ilk
spectre-cli.pdb
spectre-kernel
spectre-kernel
spectre-kernel.pdb
spectre-kernel.sys
spectre-stager
spectre-stager.ilk
spectre-stager.pdb
spectre-stager-util
spectre-stager-util.ilk
spectre-stager-util.pdb

Windows 10 test mode

- The Spectre Rootkit driver is by default built in test-signing mode. To Enable Loading of Test Signed Drivers, open a command line window as admin and run the following command:

```
bcdedit /set testsigning on
```

- To leave test mode, run the command:

```
bcdedit /set testsigning off
```

- Reboot to enter or leave test mode.

Spectre-stage

- Enter the ***spectre/x64/Debug*** folder.
- Spectre-stage can be used to load the rootkit on a victim machine.
- Run the ***spectre-stage.exe*** program to deploy the Spectre Rootkit.

Spetre-cli configure module

- Spetre-cli can be used to control an infected machine.
- Open a command line and run the following command:

```
.\spetre-cli.exe configure test
```

- Enter ***127.0.0.1*** for “Target ip address”, and keep default value for other options.
- It will generate a config file named ***test.cfg***

Spetre-cli ping module

- From the command line run the following command:

```
.\spetre-cli.exe ping test
```

- It uses ***test.cfg*** config file to determine if a host is infected with the Spectre Rootkit.
- If it shows "***failed to scan port on the target machine"*** message, try to reboot and run the above command again.
- Take a screenshot of your result.

Spetre-cli command module

- After using the ping module, you can uses ***test.cfg*** config file to execute a Windows command by typing the following command:

```
.\spetre-cli.exe command test
```

- The module is not robust. Sometimes it will not return the correct result.

Part2. Identifying the kernel rootkit with Volatility

- In this part, you need to dump the memory after installing the rootkit, and use Volatility to analyze the memory dump.

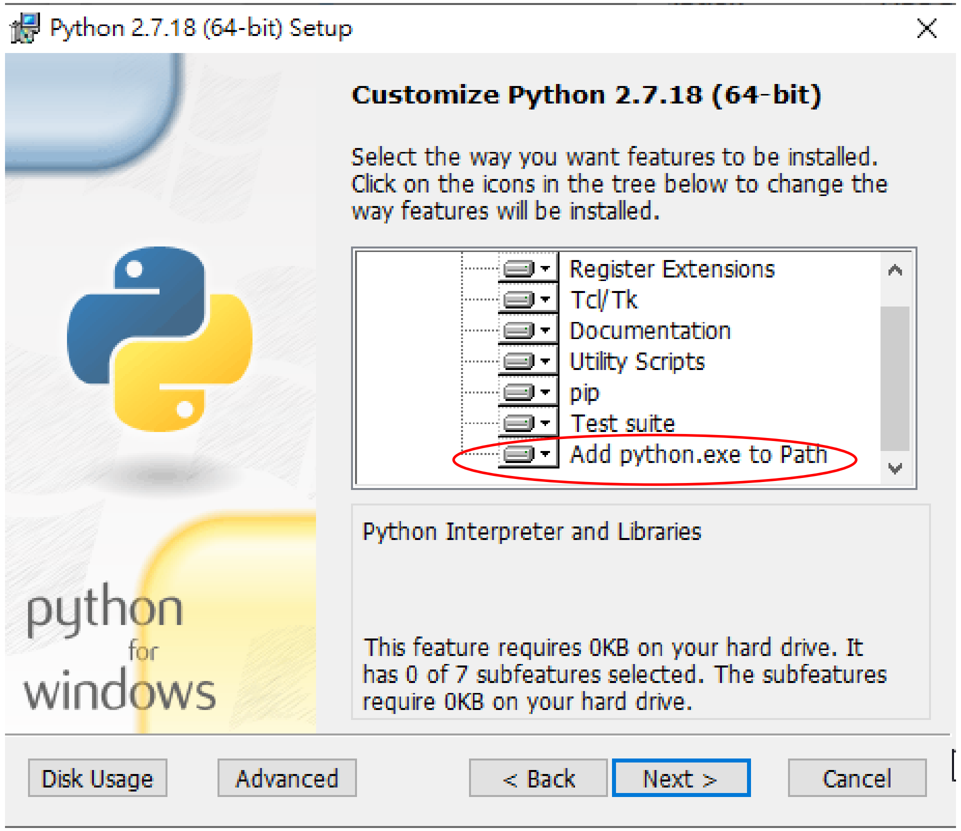
WinpMem

- WinPmem is a physical memory acquisition tool. We can use it to dump the memory.
- Download **winpmem_mini_x64_rc2.exe**
- Run the following command with admin privilege to dump the memory:

```
.\winpmem_mini_x64_rc2.exe [output path]
```

Python 2.7

- Download and install python 2.7
- Enable the option to add Python to Path



Python 2.7.18 (64-bit) Setup

Customize Python 2.7.18 (64-bit)

Select the way you want features to be installed. Click on the icons in the tree below to change the way features will be installed.

- Register Extensions
- Tcl/Tk
- Documentation
- Utility Scripts
- pip
- Test suite
- Add python.exe to Path

Python Interpreter and Libraries

This feature requires 0KB on your hard drive. It has 0 of 7 subfeatures selected. The subfeatures require 0KB on your hard drive.

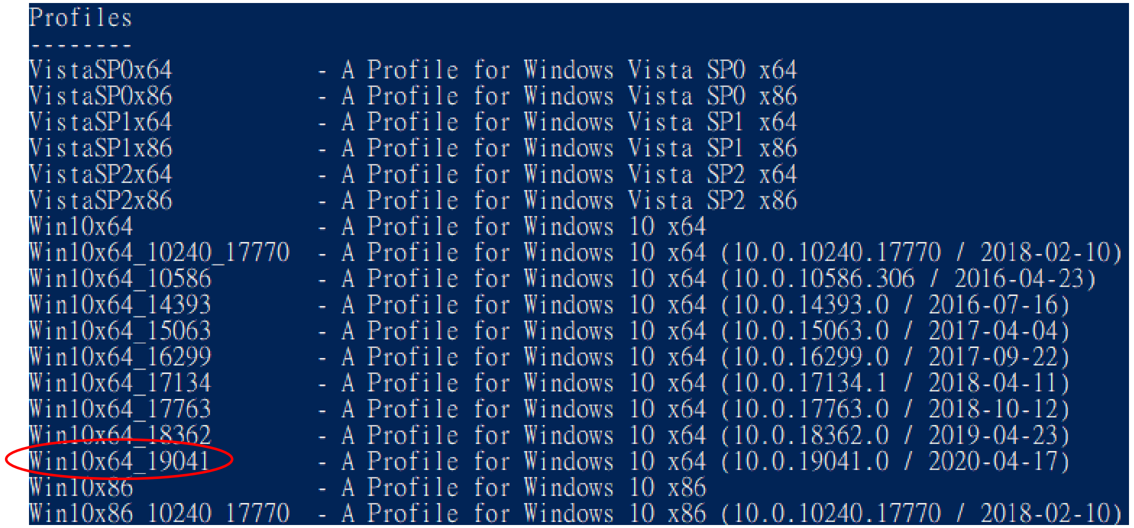
Volatility

- Download and install distorm
- From the command line, type "***pip install pycryptodome***"
- Download Volatility
- Open a command line window, enter the folder and run the following command:

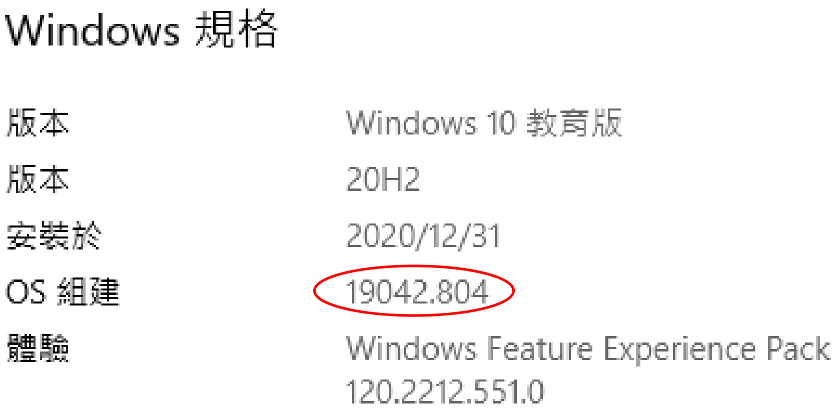
```
python setup.py install
```

Profiles

- Run “***python vol.py --info***” to see the list of available profiles.
- In the ***Settings*** window, select ***System*** > ***About*** to find out the build number.
- Choose the proper profile.



```
Profiles
--------
VistaSP0x64           - A Profile for Windows Vista SP0 x64
VistaSP0x86           - A Profile for Windows Vista SP0 x86
VistaSP1x64           - A Profile for Windows Vista SP1 x64
VistaSP1x86           - A Profile for Windows Vista SP1 x86
VistaSP2x64           - A Profile for Windows Vista SP2 x64
VistaSP2x86           - A Profile for Windows Vista SP2 x86
Win10x64              - A Profile for Windows 10 x64
Win10x64_10240_17770  - A Profile for Windows 10 x64 (10.0.10240.17770 / 2018-02-10)
Win10x64_10586        - A Profile for Windows 10 x64 (10.0.10586.306 / 2016-04-23)
Win10x64_14393        - A Profile for Windows 10 x64 (10.0.14393.0 / 2016-07-16)
Win10x64_15063        - A Profile for Windows 10 x64 (10.0.15063.0 / 2017-04-04)
Win10x64_16299        - A Profile for Windows 10 x64 (10.0.16299.0 / 2017-09-22)
Win10x64_17134        - A Profile for Windows 10 x64 (10.0.17134.1 / 2018-04-11)
Win10x64_17763        - A Profile for Windows 10 x64 (10.0.17763.0 / 2018-10-12)
Win10x64_18362        - A Profile for Windows 10 x64 (10.0.18362.0 / 2019-04-23)
Win10x64_19041        - A Profile for Windows 10 x64 (10.0.19041.0 / 2020-04-17)
Win10x86              - A Profile for Windows 10 x86
Win10x86_10240_17770  - A Profile for Windows 10 x86 (10.0.10240.17770 / 2018-02-10)
```



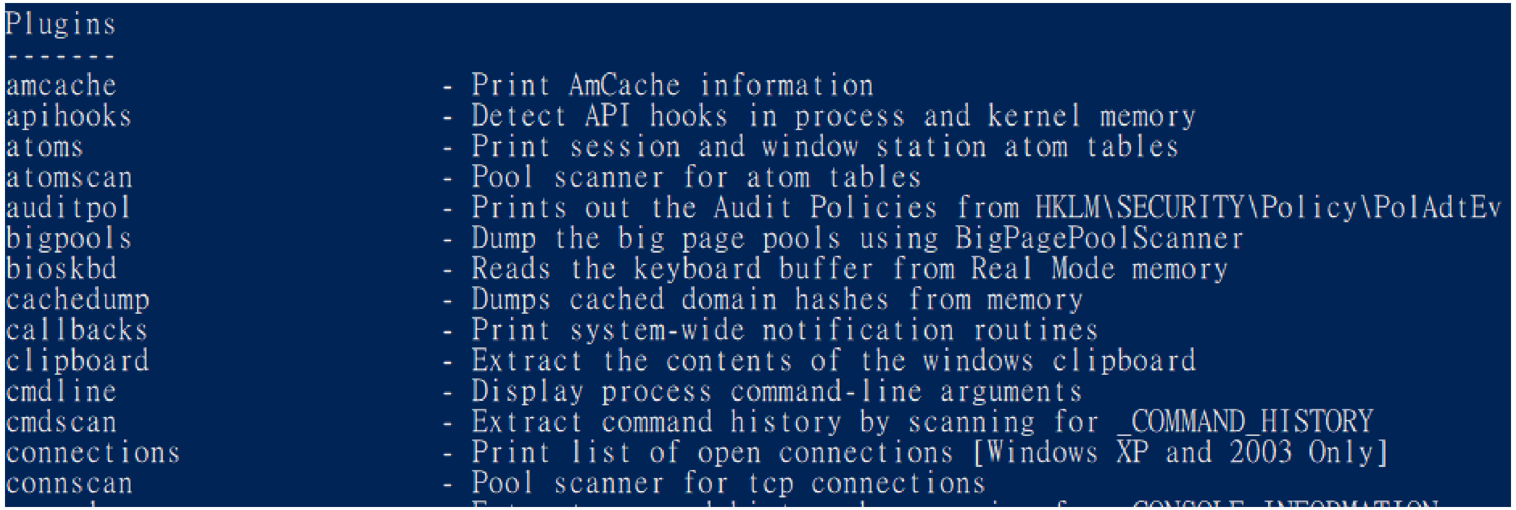
Windows 規格

版本	Windows 10 教育版
版本	20H2
安裝於	2020/12/31
OS 組建	19042.804
體驗	Windows Feature Experience Pack 120.2212.551.0

Basic usage

- Most of the Volatility commands are constructed as follow

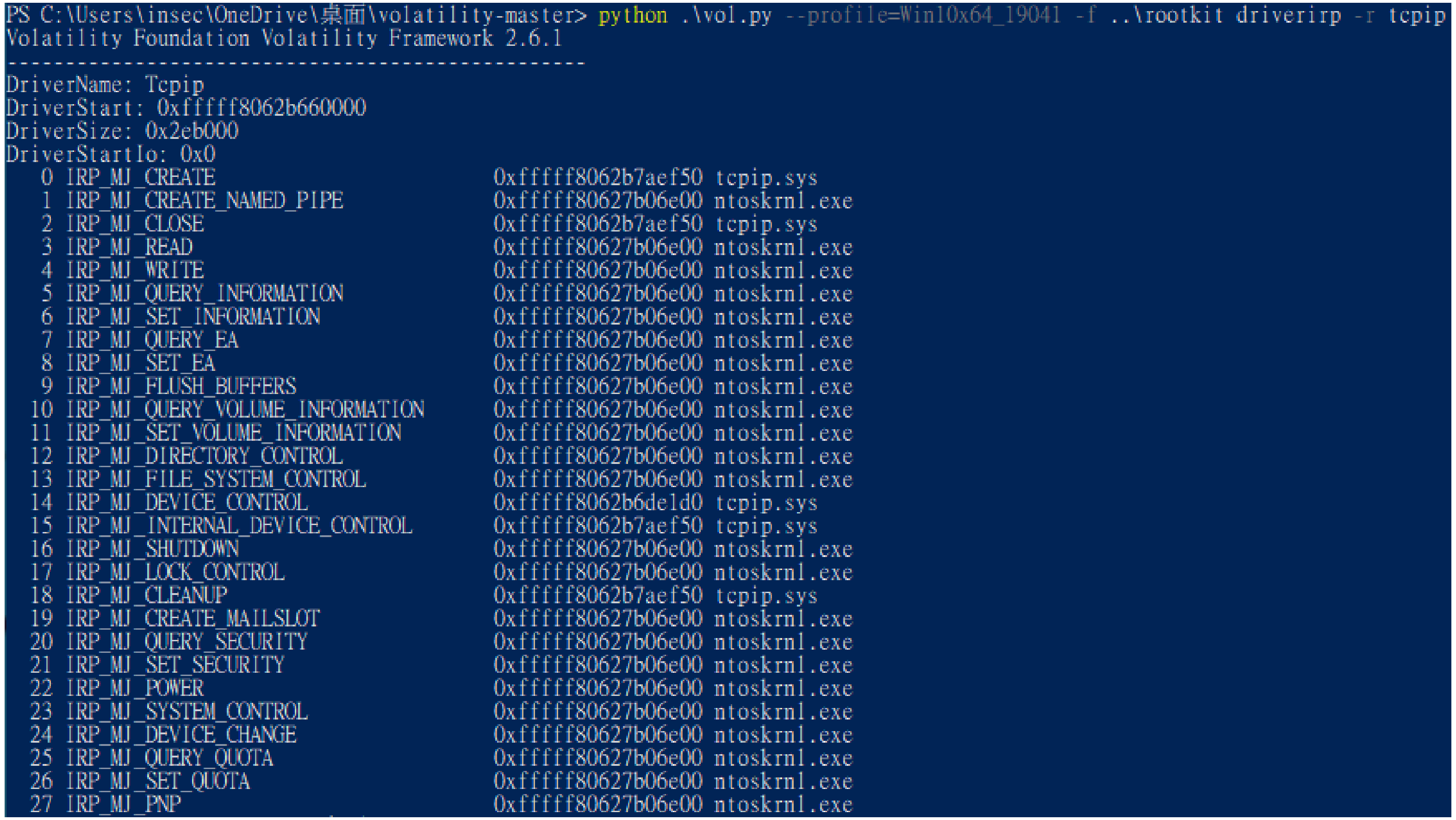
  python vol.py -f [image] --profile=[profile] [plugin]
- Image: the memory dump image
- Profile: the profile of the image (ex: Win10x64_19041)
- Plugin: run “***python vol.py --info***” to see available plugins



```
Plugins
-------
amcache                    - Print AmCache information
apihooks                   - Detect API hooks in process and kernel memory
atoms                      - Print session and window station atom tables
atomscan                   - Pool scanner for atom tables
auditpol                   - Prints out the Audit Policies from HKLM\SECURITY\Policy\PolAdtEv
bigpools                   - Dump the big page pools using BigPagePoolScanner
bioskbd                    - Reads the keyboard buffer from Real Mode memory
cachedump                  - Dumps cached domain hashes from memory
callbacks                  - Print system-wide notification routines
clipboard                  - Extract the contents of the windows clipboard
cmdline                    - Display process command-line arguments
cmdscan                    - Extract command history by scanning for _COMMAND_HISTORY
connections                - Print list of open connections [Windows XP and 2003 Only]
connscan                   - Pool scanner for tcp connections
```

Driverirp

- The ***driverirp*** plugin can list the ***Major function*** array of a ***DRIVER_OBJECT***
- ***-r*** specify the driver name



```
PS C:\Users\insec\OneDrive\桌面\volatility-master> python .\vol.py --profile=Win10x64_19041 -f ..\rootkit driverirp -r tcpip
Volatility Foundation Volatility Framework 2.6.1
--------------------------------------------------
DriverName: Tcpip
DriverStart: 0xfffff8062b660000
DriverSize: 0x2eb000
DriverStartIo: 0x0
   0 IRP_MJ_CREATE                      0xfffff8062b7aef50 tcpip.sys
   1 IRP_MJ_CREATE_NAMED_PIPE           0xfffff80627b06e00 ntoskrnl.exe
   2 IRP_MJ_CLOSE                       0xfffff8062b7aef50 tcpip.sys
   3 IRP_MJ_READ                        0xfffff80627b06e00 ntoskrnl.exe
   4 IRP_MJ_WRITE                       0xfffff80627b06e00 ntoskrnl.exe
   5 IRP_MJ_QUERY_INFORMATION           0xfffff80627b06e00 ntoskrnl.exe
   6 IRP_MJ_SET_INFORMATION             0xfffff80627b06e00 ntoskrnl.exe
   7 IRP_MJ_QUERY_EA                    0xfffff80627b06e00 ntoskrnl.exe
   8 IRP_MJ_SET_EA                      0xfffff80627b06e00 ntoskrnl.exe
   9 IRP_MJ_FLUSH_BUFFERS               0xfffff80627b06e00 ntoskrnl.exe
  10 IRP_MJ_QUERY_VOLUME_INFORMATION    0xfffff80627b06e00 ntoskrnl.exe
  11 IRP_MJ_SET_VOLUME_INFORMATION      0xfffff80627b06e00 ntoskrnl.exe
  12 IRP_MJ_DIRECTORY_CONTROL           0xfffff80627b06e00 ntoskrnl.exe
  13 IRP_MJ_FILE_SYSTEM_CONTROL         0xfffff80627b06e00 ntoskrnl.exe
  14 IRP_MJ_DEVICE_CONTROL              0xfffff8062b6de1d0 tcpip.sys
  15 IRP_MJ_INTERNAL_DEVICE_CONTROL     0xfffff8062b7aef50 tcpip.sys
  16 IRP_MJ_SHUTDOWN                    0xfffff80627b06e00 ntoskrnl.exe
  17 IRP_MJ_LOCK_CONTROL                0xfffff80627b06e00 ntoskrnl.exe
  18 IRP_MJ_CLEANUP                     0xfffff8062b7aef50 tcpip.sys
  19 IRP_MJ_CREATE_MAILSLOT             0xfffff80627b06e00 ntoskrnl.exe
  20 IRP_MJ_QUERY_SECURITY              0xfffff80627b06e00 ntoskrnl.exe
  21 IRP_MJ_SET_SECURITY                0xfffff80627b06e00 ntoskrnl.exe
  22 IRP_MJ_POWER                       0xfffff80627b06e00 ntoskrnl.exe
  23 IRP_MJ_SYSTEM_CONTROL              0xfffff80627b06e00 ntoskrnl.exe
  24 IRP_MJ_DEVICE_CHANGE               0xfffff80627b06e00 ntoskrnl.exe
  25 IRP_MJ_QUERY_QUOTA                 0xfffff80627b06e00 ntoskrnl.exe
  26 IRP_MJ_SET_QUOTA                   0xfffff80627b06e00 ntoskrnl.exe
  27 IRP_MJ_PNP                         0xfffff80627b06e00 ntoskrnl.exe
```

Question

1. Take a screenshot after executing the spectre-cli ping module with at least one infected port.
2. The rootkit creates the fake ***AFD*** driver and replace the ***MajorFunction*** array to point to the hook functions.
   1) Simply Explain “IRP Major Function”.
   2) Generate a memory dump in test mode. Use Volatility to show the MajorFunction array of ***AFD***. Take a screenshot of your result.
   3) Disable test mode and generate another image. Use Volatility to show the MajorFunction array of ***AFD***. Take a screenshot of your result.

Summit the pdf file named ***student_ID.pdf***.

References

- https://github.com/D4stiny/spectre/wiki/Getting-Started
- https://dfironthemountain.wordpress.com/2018/10/29/installing-volatility-on-windows/