CUCKOO SANDBOX 安裝教學

CS4003701

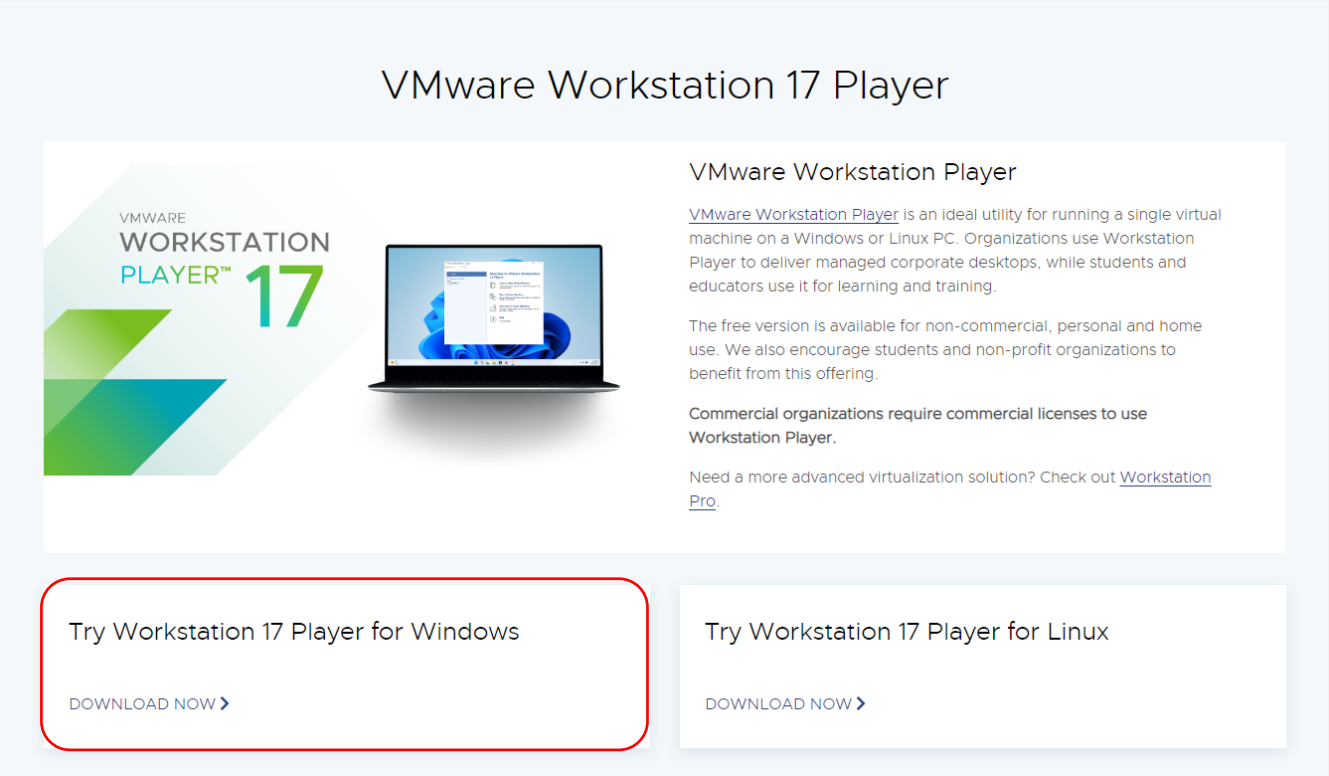
資訊安全導論

CUCKOO SANDBOX 安裝教學

- 1.下載VMWARE
- 2. 創立UBUNTU虛擬機
- 3. 安裝CUCKOO SANDBOX
- 4.安裝CUCKOO所需WIN7虛擬機
- 5.初始化設定CUCKOO SANDBOX
- 6.啟用CUCKOO SANDBOX
- 7.備份與重啟CUCKOO SANDBOX
- 8.使用CUCKOO SANDBOX
- 9.繪製CALL API SEQUENCE
- 10.撰寫CUCKOO SANDBOX腳本(參考用)
- 11.虛擬機硬碟空間擴充(參考用)
- 12.PYCHARM+ANACONDA安裝教學(參考用)

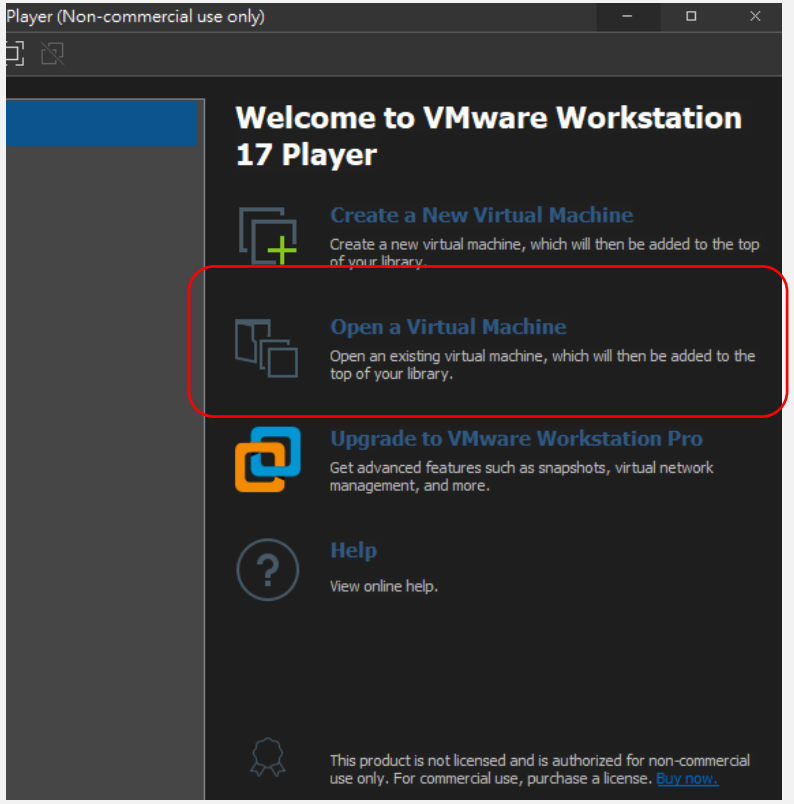
下載VMWARE

https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html

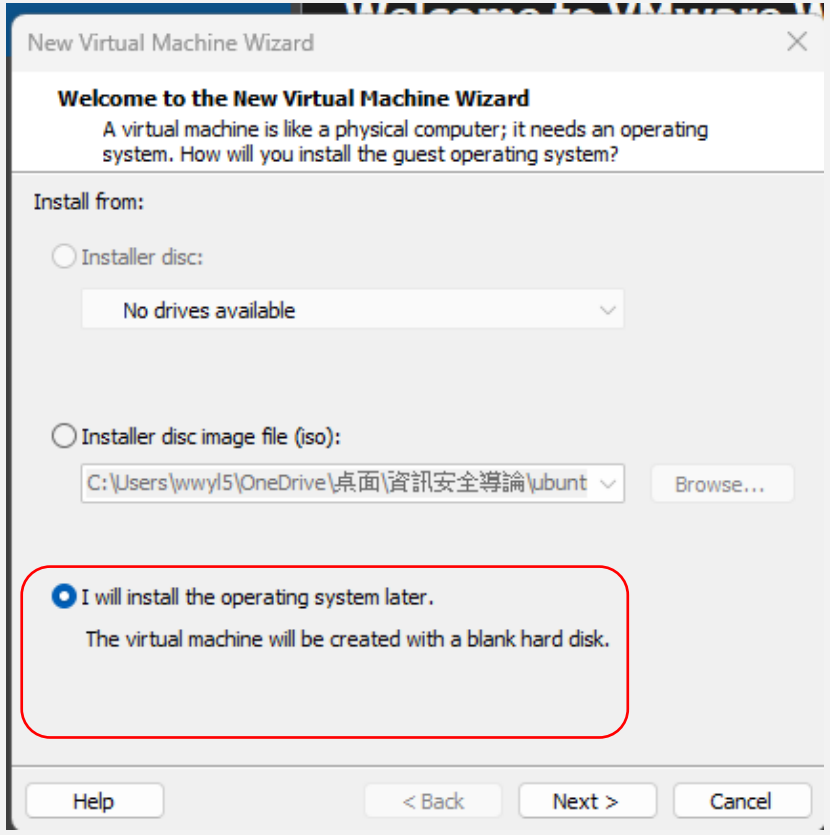


創立UBUNTU虛擬機

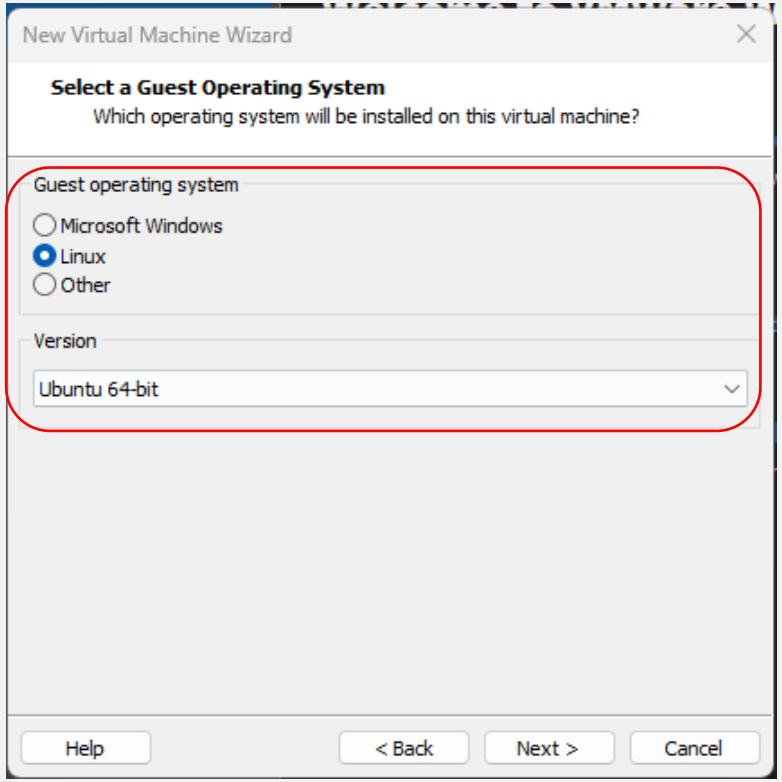
1.點選創建虛擬機



2.點選等等設定映像檔

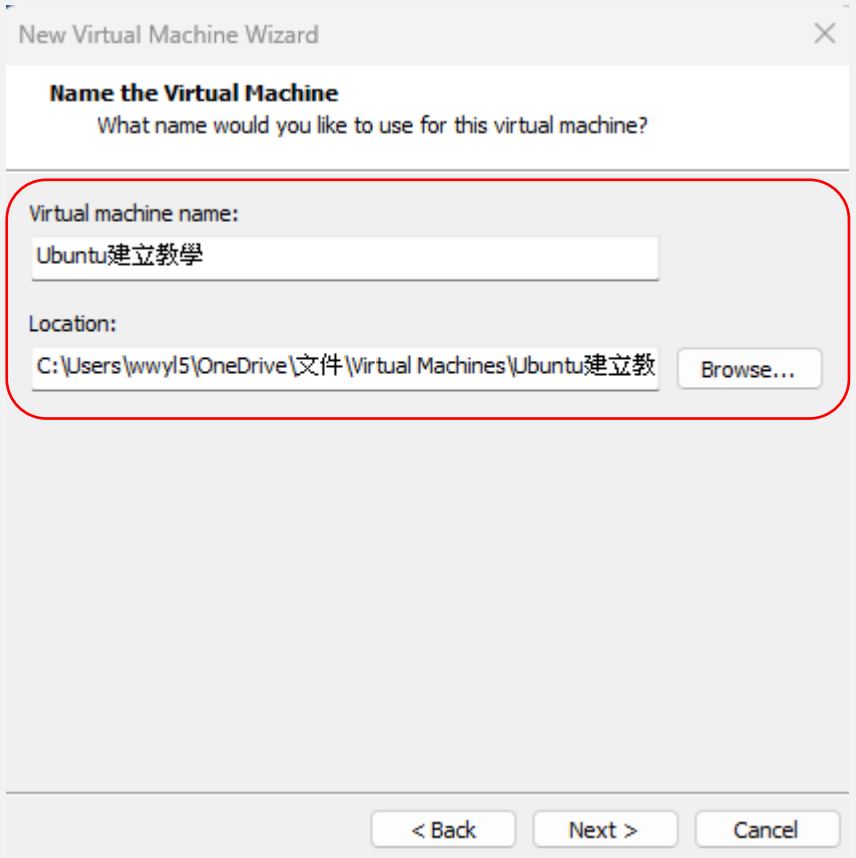


3.選擇Linux+Ubuntu64bit

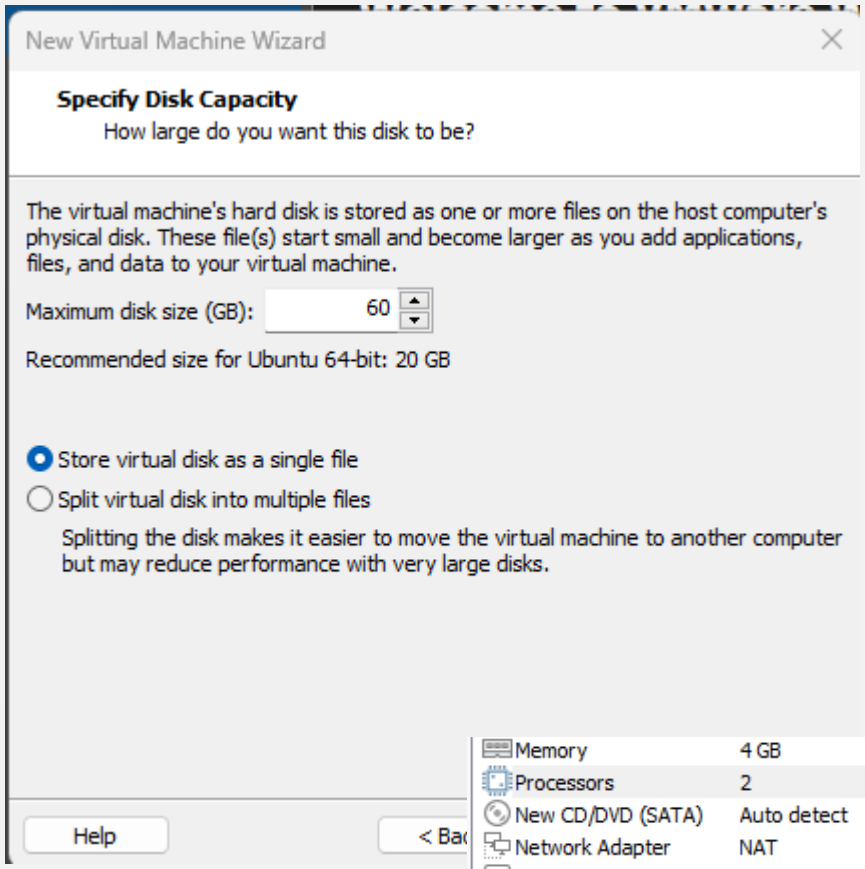


創立UBUNTU虛擬機

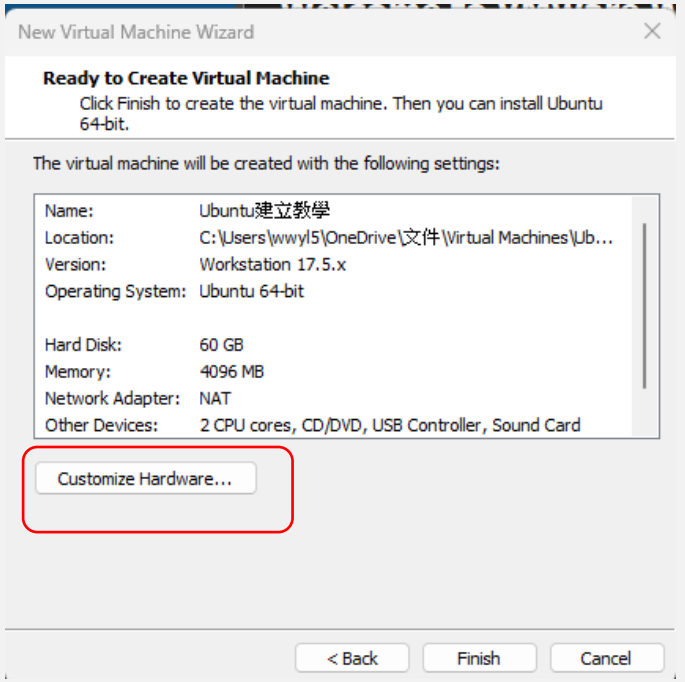
4.選擇儲存位置
(要稍微記一下之後會用到)

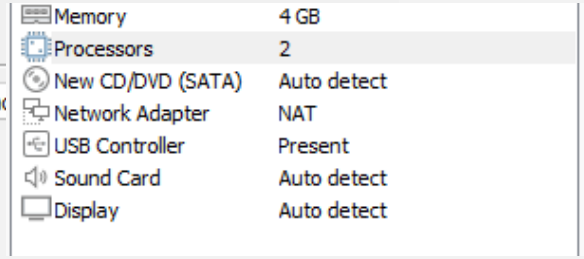


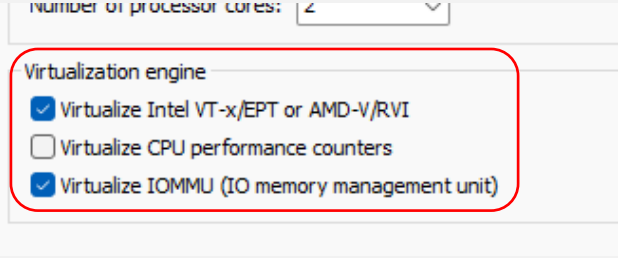
5. 設定虛擬機硬碟大小，並設置儲存成一個檔案
(建議要>70GB，不過以後還可以擴充)



6.處理器可視化設定



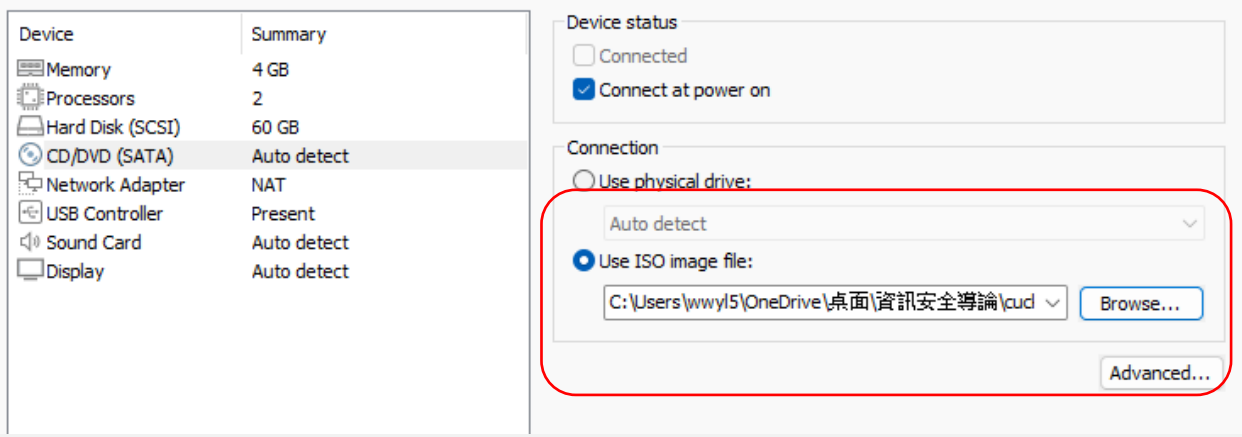




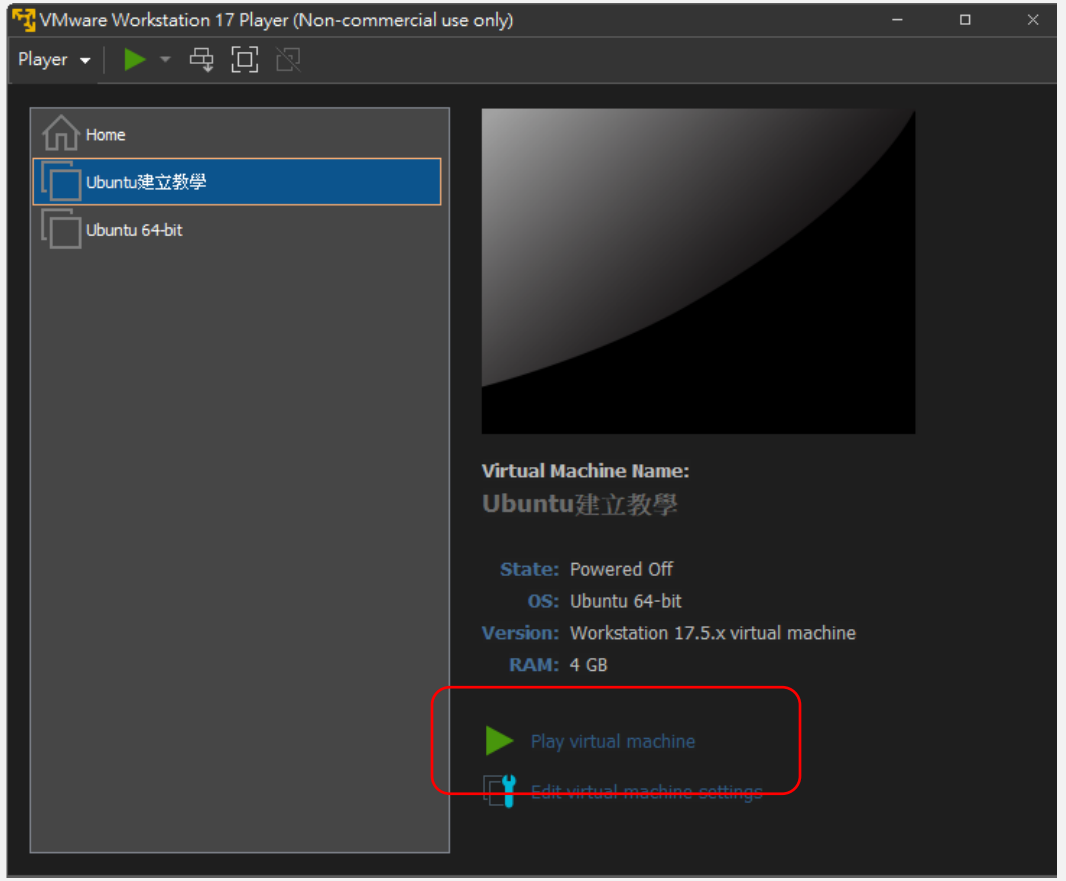
創立UBUNTU虛擬機

7.設定映像檔路徑(於MOODLE上下載)

(UBUNTU-18.04.ISO)

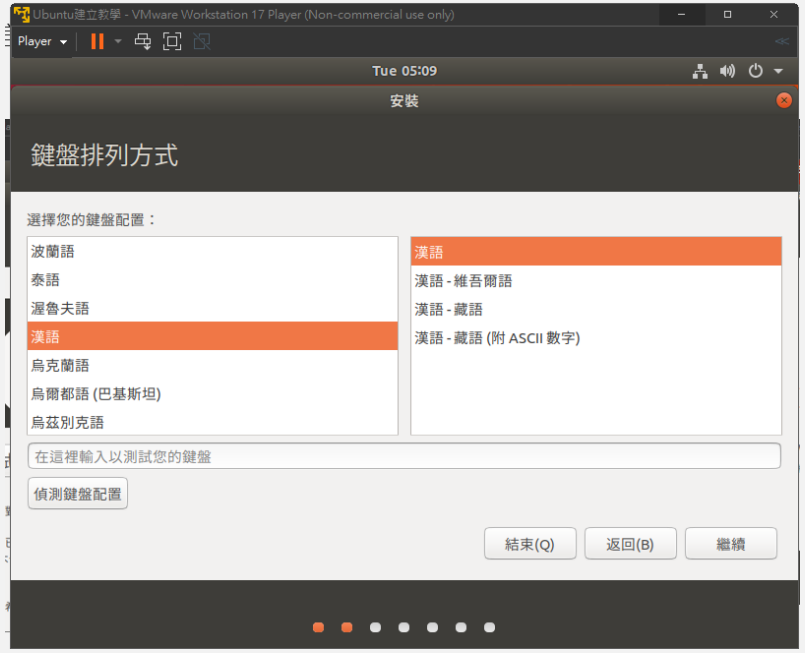


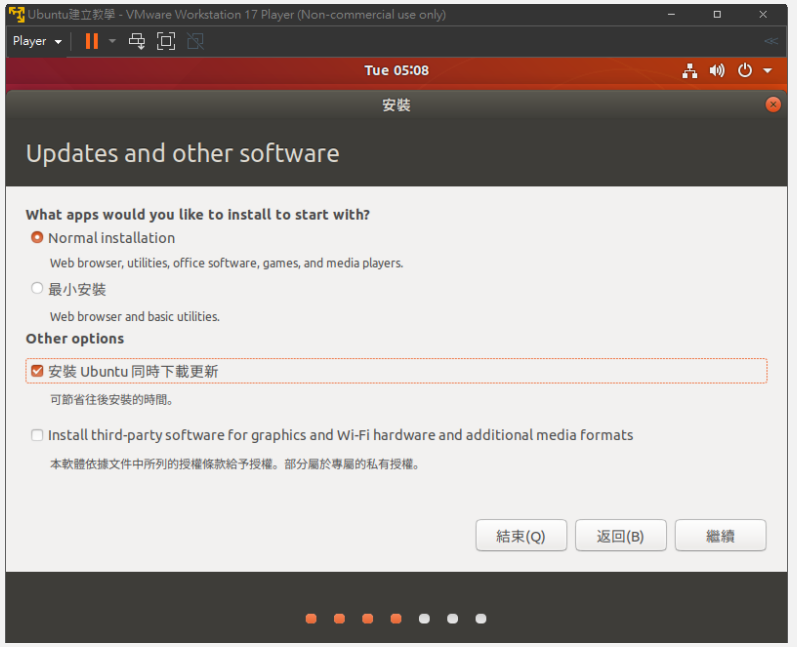
8.開啟虛擬機進行作業系統安裝

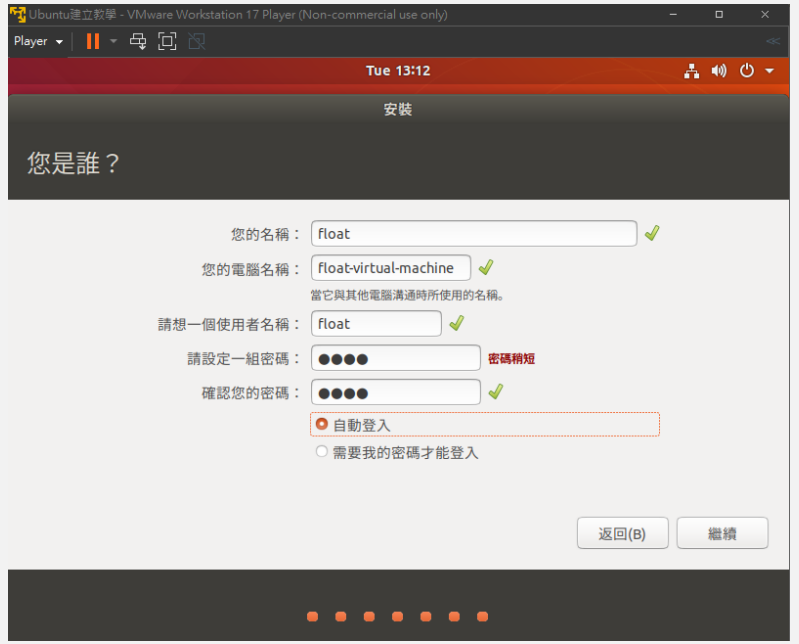


創立UBUNTU虛擬機

9. 點擊安裝程序

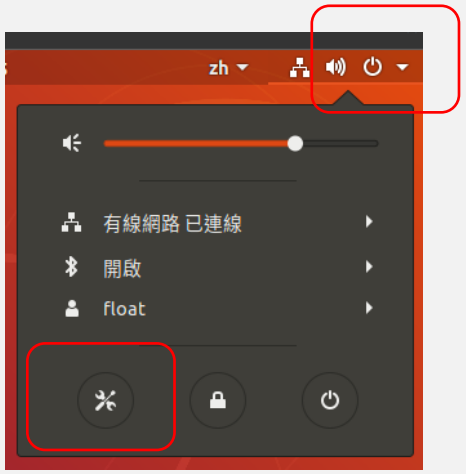






創立UBUNTU虛擬機

10. 調整螢幕大小，可以先將”設定”視窗調小再設定(記得別升級到UBUNTU20.04)



安裝CUCKOO SANDBOX

1.更新套件庫:

- sudo apt-get update

2.下載相關套件:

- sudo apt-get -y install python python-pip python-dev libffi-dev libssl-dev
- sudo apt-get -y install python-virtualenv python-setuptools
- sudo apt-get -y install libjpeg-dev zlib1g-dev swig

3.下載MongoDB:

- sudo apt-get -y install mongodb

4.下載PostgreSQL:

- sudo apt-get -y install postgresql libpq-dev

```
Success. You can now start the database server using:

    /usr/lib/postgresql/10/bin/pg_ctl -D /var/lib/postgresql/10/main -l logfile
start

Ver Cluster Port Status Owner    Data directory              Log file
10  main    5432 down   postgres /var/lib/postgresql/10/main /var/log/postgresql
/postgresql-10-main.log
update-alternatives: using /usr/share/postgresql/10/man/man1/postmaster.1.gz to
provide /usr/share/man/man1/postmaster.1.gz (postmaster.1.gz) in auto mode
設定 postgresql (10+190ubuntu0.1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
Processing triggers for libc-bin (2.27-3ubuntu1.4) ...
Processing triggers for systemd (237-3ubuntu10.52) ...
float@float-virtual-machine:~$
```

5.下載VirtualBox:

- sudo apt-get -y install virtualbox

6.下載tcpdump AppArmor:

- sudo apt-get -y install tcpdump apparmor-utils

安裝CUCKOO SANDBOX

8.設定tcpdump AppArmor解除限制:

- sudo aa-disable /usr/sbin/tcpdump

```
float@float-virtual-machine:~$ sudo aa-disable /usr/sbin/tcpdump
Disabling /usr/sbin/tcpdump.
float@float-virtual-machine:~$
```

9.讓tcpdump獲得後台運行權限:

- sudo setcap cap_net_raw,cap_net_admin=eip /usr/sbin/tcpdump

10.驗證設定是否成功:

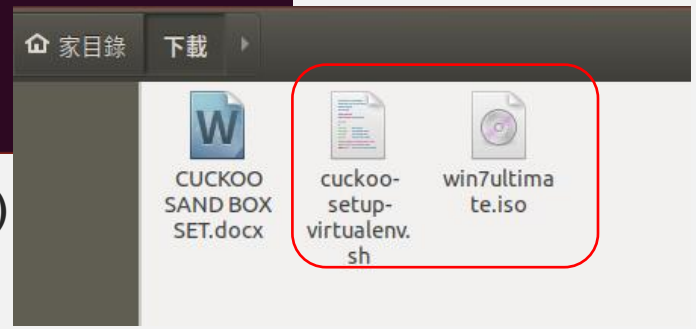
- getcap /usr/sbin/tcpdump

```
float@float-virtual-machine:~$ sudo setcap cap_net_raw,cap_net_admin=eip /usr/sb
in/tcpdump
float@float-virtual-machine:~$ getcap /usr/sbin/tcpdump
/usr/sbin/tcpdump = cap_net_admin,cap_net_raw+eip
float@float-virtual-machine:~$
```

11.下載M2Crypto:

- pip install 'm2crypto<0.36'

```
  Stored in directory: /home/float/.cache/pip/wheels/6d/b3/6e/4c057fb6aafc93b1a8
d5607d46da439260ae21dbca3e9ceba8
Successfully built m2crypto
Installing collected packages: typing, m2crypto
Successfully installed m2crypto-0.35.2 typing-3.10.0.0
float@float-virtual-machine:~$
```



12.將cuckoo-setup-virtualenv.sh+win7ultimate.iso檔下載到虛擬機(從Moodle上下載)

13.進入cuckoo-setup-virtualenv.sh所在資料夾，賦予執行腳本權限:

- sudo chmod +x cuckoo-setup-virtualenv.sh

```
float@float-virtual-machine:~$ cd 下載
float@float-virtual-machine:~/下載$ sudo chmod +x cuckoo-setup-virtualenv.sh
float@float-virtual-machine:~/下載$
```

14.安裝沙盒環境套件:

- sudo -u username ./cuckoo-setup-virtualenv.sh

```
float@float-virtual-machine:~/下載$ sudo -u float ./cuckoo-setup-virtualenv.sh
已有:1 https://dl.google.com/linux/chrome/deb stable InRelease
已有:2 http://tw.archive.ubuntu.com/ubuntu bionic InRelease
已有:3 http://tw.archive.ubuntu.com/ubuntu bionic-updates InRelease
已有:4 http://tw.archive.ubuntu.com/ubuntu bionic-backports InRelease
已有:5 http://security.ubuntu.com/ubuntu bionic-security InRelease
```

安裝CUCKOO SANDBOX

15.再次確認沙盒套件已下載成功:

- sudo apt-get update && sudo apt-get -y install virtualenv
- sudo apt-get -y install virtualenvwrapper
- sudo apt-get -y install python3-pip

16.再次確認環境變數是否生效:

- source ~/.bashrc

17.創建沙盒:

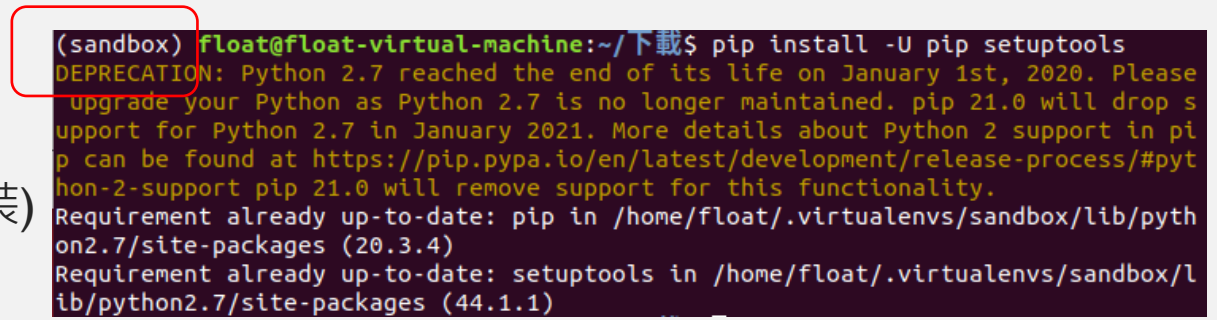
- mkvirtualenv -p python2.7 sandbox

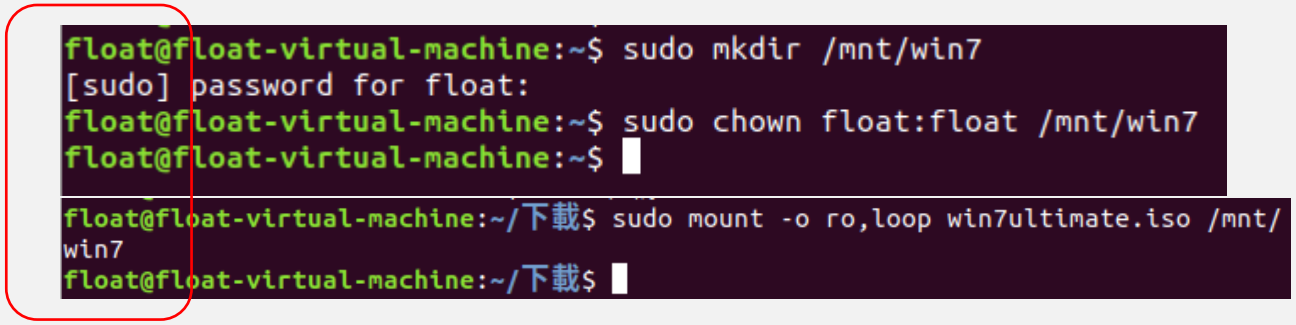
18.安裝Cuckoo Sandbox:(記得要確定你在剛創的沙盒裡面安裝)

- pip install -U pip setuptools
- pip install -U cuckoo

19.掛載win7映像檔到虛擬機裡:(在沙盒外面進行)

- sudo mkdir /mnt/win7
- sudo chown username:username /mnt/win7
- sudo mount -o ro,loop win7ultimate.iso /mnt/win7





安裝CUCKOO所需WIN7虛擬機

20.再次確認相關套件是否安裝成功:(在沙盒外面確認)

- sudo apt-get -y install build-essential libssl-dev libffi-dev python-dev genisoimage
- sudo apt-get -y install zlib1g-dev libjpeg-dev
- sudo apt-get -y install python-pip python-virtualenv python-setuptools swig

21.下載vmcloak虛擬機:(在沙盒裡面下載)

- pip install -U vmcloak
- vmcloak

```
(sandbox) float@float-virtual-machine:~/下載$ vmcloak
/home/float/.virtualenvs/sandbox/local/lib/python2.7/site-packages/OpenSSL/crypt
o.py:14: CryptographyDeprecationWarning: Python 2 is no longer supported by the
Python core team. Support for it is now deprecated in cryptography, and will be
removed in the next release.
  from cryptography import utils, x509
(sandbox) float@float-virtual-machine:~/下載$
```

22.用vmclock設定HOST-ONLY network:

- vmcloak-vboxnet0

```
(sandbox) float@float-virtual-machine:~/下載$ vmcloak-vboxnet0
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
Interface 'vboxnet0' was successfully created
(sandbox) float@float-virtual-machine:~/下載$
```

23.開始在沙盒中安裝win7虛擬機:

- vmcloak init --verbose --win7x64 win7x64base --cpus 2 --ramsize 2048

```
(sandbox) float@float-virtual-machine:~/下載$ vmcloak init --verbose --win7x64 w
in7x64base --cpus 2 --ramsize 2048
/home/float/.virtualenvs/sandbox/local/lib/python2.7/site-packages/OpenSSL/crypt
o.py:14: CryptographyDeprecationWarning: Python 2 is no longer supported by the
Python core team. Support for it is now deprecated in cryptography, and will be
removed in the next release.
  from cryptography import utils, x509
INFO:vmcloak.abstract:Got file 'python-2.7.6.msi' from 'https://www.python.org/f
tp/python/2.7.6/python-2.7.6.msi', with matching checksum.
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
INFO:vmcloak:Starting the Virtual Machine u'win7x64base' to install Windows.
```

```
(sandbox) float@float-virtual-machine:~/下載$ vmcloak init --verbose --win7x64 w
in7x64base --cpus 2 --ramsize 2048
/home/float/.virtualenvs/sandbox/local/lib/python2.7/site-packages/OpenSSL/crypt
o.py:14: CryptographyDeprecationWarning: Python 2 is no longer supported by the
Python core team. Support for it is now deprecated in cryptography, and will be
removed in the next release.
  from cryptography import utils, x509
INFO:vmcloak.abstract:Got file 'python-2.7.6.msi' from 'https://www.python.org/f
tp/python/2.7.6/python-2.7.6.msi', with matching checksum.
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
INFO:vmcloak:Starting the Virtual Machine u'win7x64base' to install Windows.
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
INFO:vmcloak:Added image u'win7x64base' to the repository.
(sandbox) float@float-virtual-machine:~/下載$
```

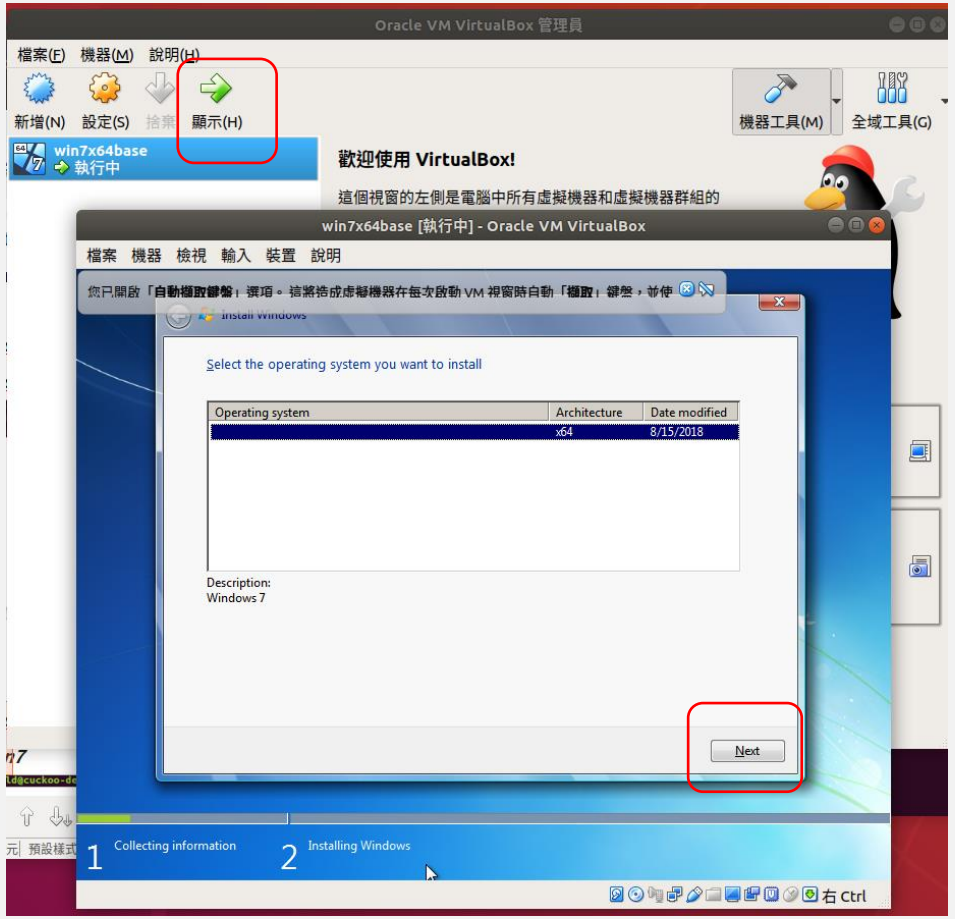
安裝CUCKOO所需WIN7虛擬機

24.去開啟Virtualbox點擊”顯示”進行windows安裝程序:

25.複製剛剛安裝的windows虛擬機:

- vmcloak clone win7x64base win7x64cuckoo

```
(sandbox) float@float-virtual-machine:~/下載$ vmcloak clone win7x64base win7x64c
uckoo
/home/float/.virtualenvs/sandbox/local/lib/python2.7/site-packages/OpenSSL/crypt
o.py:14: CryptographyDeprecationWarning: Python 2 is no longer supported by the
Python core team. Support for it is now deprecated in cryptography, and will be
removed in the next release.
  from cryptography import utils, x509
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
```



26.創建4個相同的windows虛擬機:(提供cuckoo sandbox多工執行)

- vmcloak snapshot --count 4 win7x64cuckoo 192.168.56.101

```
(sandbox) float@float-virtual-machine:~/下載$ vmcloak snapshot --count 4 win7x64
cuckoo 192.168.56.101
/home/float/.virtualenvs/sandbox/local/lib/python2.7/site-packages/OpenSSL/crypt
o.py:14: CryptographyDeprecationWarning: Python 2 is no longer supported by the
Python core team. Support for it is now deprecated in cryptography, and will be
removed in the next release.
  from cryptography import utils, x509
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
(sandbox) float@float-virtual-machine:~/下載$
```

安裝CUCKOO所需WIN7虛擬機

27.列出目前安裝好的windows虛擬機:

- vmcloak list vms

```
(sandbox) float@float-virtual-machine:~/下載$ vmcloak list vms
/home/float/.virtualenvs/sandbox/local/lib/python2.7/site-packages/OpenSSL/crypt
o.py:14: CryptographyDeprecationWarning: Python 2 is no longer supported by the
Python core team. Support for it is now deprecated in cryptography, and will be
removed in the next release.
  from cryptography import utils, x509
192.168.56.1011 192.168.56.101
192.168.56.1012 192.168.56.102
192.168.56.1013 192.168.56.103
192.168.56.1014 192.168.56.104
(sandbox) float@float-virtual-machine:~/下載$
```

28.初始化並進入cuckoo的config dictionary:

- cuckoo init
- cd .cuckoo/conf
- cuckoo community --force

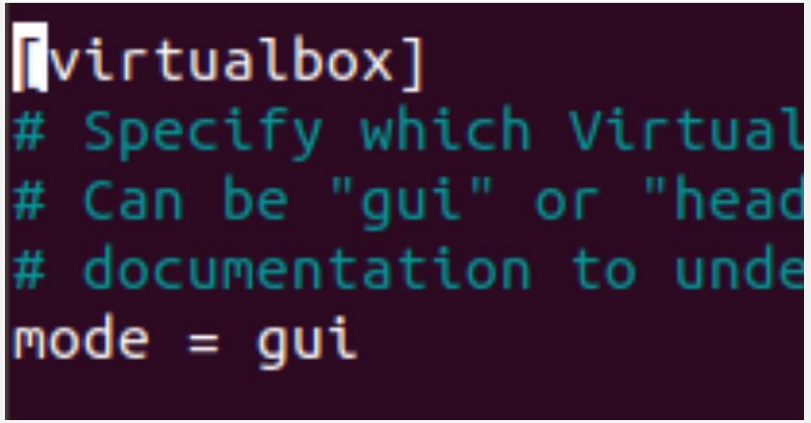
```
(sandbox) float@float-virtual-machine:~$ cd .cuckoo/conf
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ cuckoo community --force
2024-03-05 15:21:22,911 [cuckoo.apps.apps] INFO: Downloading.. https://github.co
m/cuckoosandbox/community/archive/master.tar.gz
2024-03-05 15:21:27,865 [cuckoo] INFO: Finished fetching & extracting the commun
ity files!
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$
```

```
(sandbox) float@float-virtual-machine:~/下載$ cuckoo init

  eeee e   e eeee e   e  eeeee eeeee
  8  8 8   8 8  8 8   8  8  88 8  88
  8e   8e  8 8e   8eee8e 8   8 8   8
  88   88  8 88   88   8 8   8 8   8
  88e8 88ee8 88e8 88   8 8eee8 8eee8

 Cuckoo Sandbox 2.0.7
 www.cuckoosandbox.org
 Copyright (c) 2010-2018

=====================================================================
    Welcome to Cuckoo Sandbox, this appears to be your first run!
    We will now set you up with our default configuration.
    You will be able to see and modify the Cuckoo configuration,
    Yara rules, Cuckoo Signatures, and much more to your likings
    by exploring the /home/float/.cuckoo directory.

    Among other configurable items of most interest is the
    new location for your Cuckoo configuration:
                /home/float/.cuckoo/conf
=====================================================================

Cuckoo has finished setting up the default configuration.
Please modify the default settings where required and
start Cuckoo again (by running `cuckoo` or `cuckoo -d`).
(sandbox) float@float-virtual-machine:~/下載$
```

初始化設定CUCKOO SANDBOX

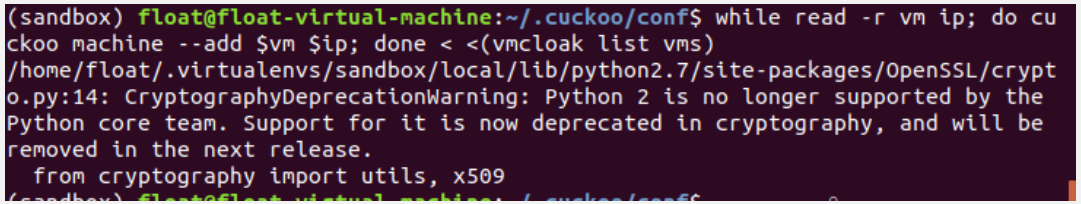
29.開啟.cuckoo/conf/virtualbox.conf並把mode的headless改成gui:

- nano virtualbox.conf
- mode = gui
- ^x+y+enter(儲存)



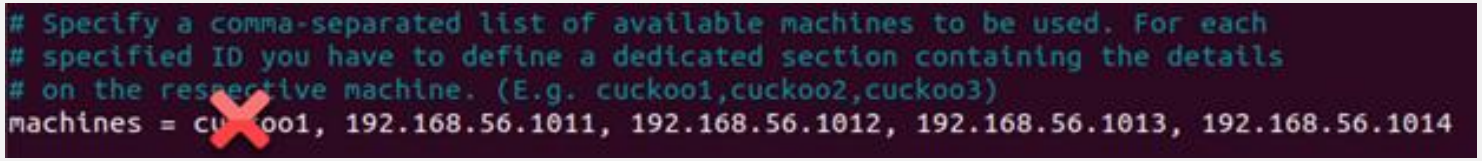
30.把變更應用在其他3個虛擬機身上:

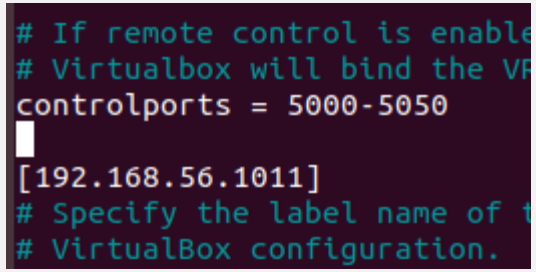
- while read -r vm ip; do cuckoo machine --add $vm $ip; done < <(vmcloak list vms)



31.再打開virtualbox.conf，把machines的cuckoo1刪掉，然後把controlports=5000-5050的下面都刪除直到[192.168.56.1011]:

(記得儲存^x+y+enter)

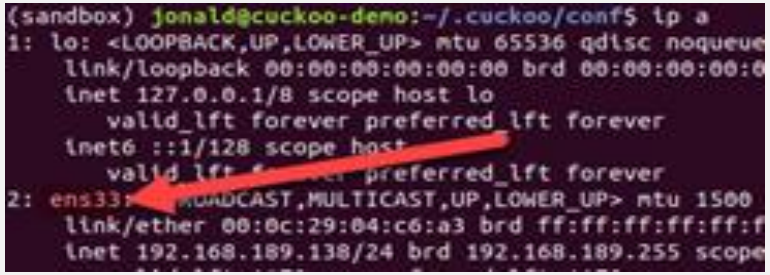




初始化設定CUCKOO SANDBOX

32.檢查設定:

- ip a



33.設定參數:(在沙盒裡設置)

- sudo sysctl -w net.ipv4.conf.vboxnet0.forwarding=1
- sudo sysctl -w net.ipv4.conf.ens33.forwarding=1
- sudo iptables -t nat -A POSTROUTING -o ens33 -s 192.168.56.0/24 -j MASQUERADE
- sudo iptables -P FORWARD DROP
- sudo iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
- sudo iptables -A FORWARD -s 192.168.56.0/24 -j ACCEPT

```
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo sysctl -w net.ipv4.conf.vboxnet0.forwarding=1
[sudo] password for float:
net.ipv4.conf.vboxnet0.forwarding = 1
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo sysctl -w net.ipv4.conf.ens33.forwarding=1
net.ipv4.conf.ens33.forwarding = 1
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -t nat -A POSTROUTING -o ens33 -s 192.168.56.0/24 -j MASQUERADE
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -P FORWARD DROP
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -A FORWARD -s 192.168.56.0/24 -j ACCEPT
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$
```

啟用CUCKOO SANDBOX(1)

34.下載terminator:(在沙盒外面)

- sudo apt-get install terminator

```
float@float-virtual-machine:~/下載$ sudo apt-get install terminator
正在讀取套件清單... 完成
正在重建相依關係
正在讀取狀態資料... 完成
下列的額外套件將被安裝：
```

35.開一個新終端，我們叫它T1，進入沙盒:

- workon sandbox

36. 以root執行cuckoo，讓T1運行不要動它:

- cuckoo rooter --sudo --group username

```
float@float-virtual-machine:~$ workon sandbox
(sandbox) float@float-virtual-machine:~$ cuckoo rooter --sudo --group float
[sudo] password for float:
2024-03-05 15:33:22,123 [cuckoo] INFO: Starting Cuckoo Rooter (group=float)!
```

初始化設定CUCKOO SANDBOX

37.回到剛剛那個終端，開啟routing.conf，把Internet後面的none改成ens33:

- nano routing.conf
- ^x+y+enter(儲存)

```
# Network interface that allows a VM to connect to the entire internet, the
# "dirty line" so to say. Note that, just like with the VPNs, this will allow
# malicious traffic through your network. So think twice before enabling it.
# (For example, to use eth0 as dirty line: "internet = eth0").
internet = ens33
```

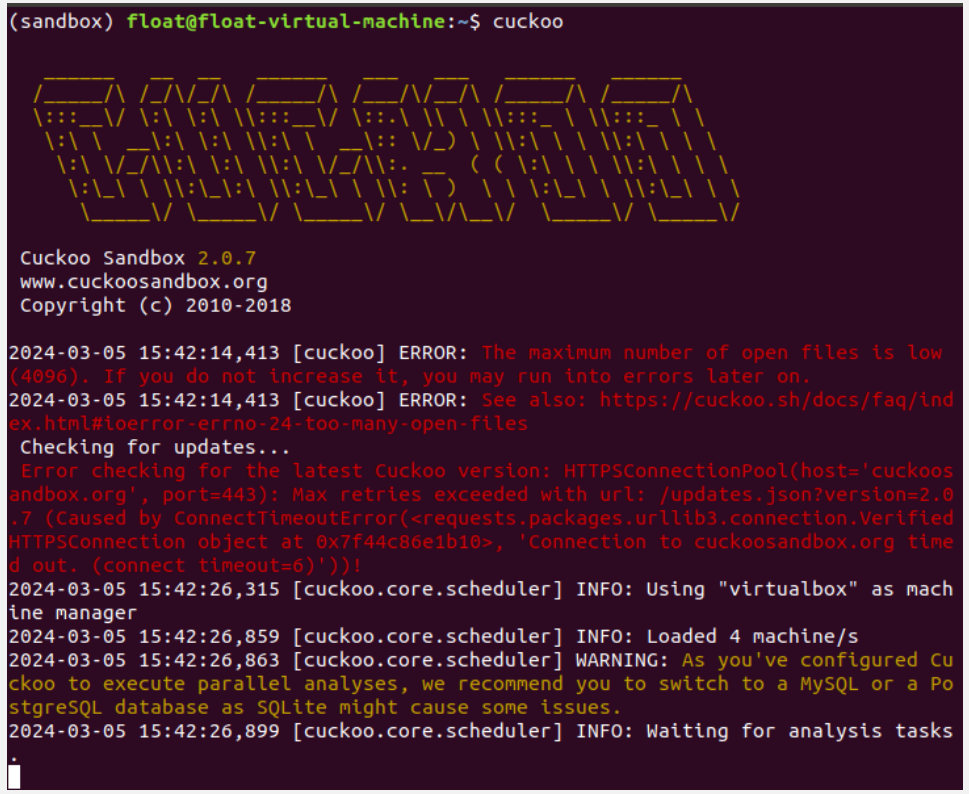
38.開啟reporting.conf，把mongodb欄位的enabled的no改成yes:(記得儲存)

```
[mongodb]
enabled = yes
host = 127.0.0.1
port = 27017
db = cuckoo
store_memdump = yes
paginate = 100
# MongoDB authentication (optional).
```

啟用CUCKOO SANDBOX(2)

39.再開一個新的終端，我們叫它T2，進入沙盒，執行cuckoo，他會持續運行，放著不要動:

- workon sandbox
- cuckoo

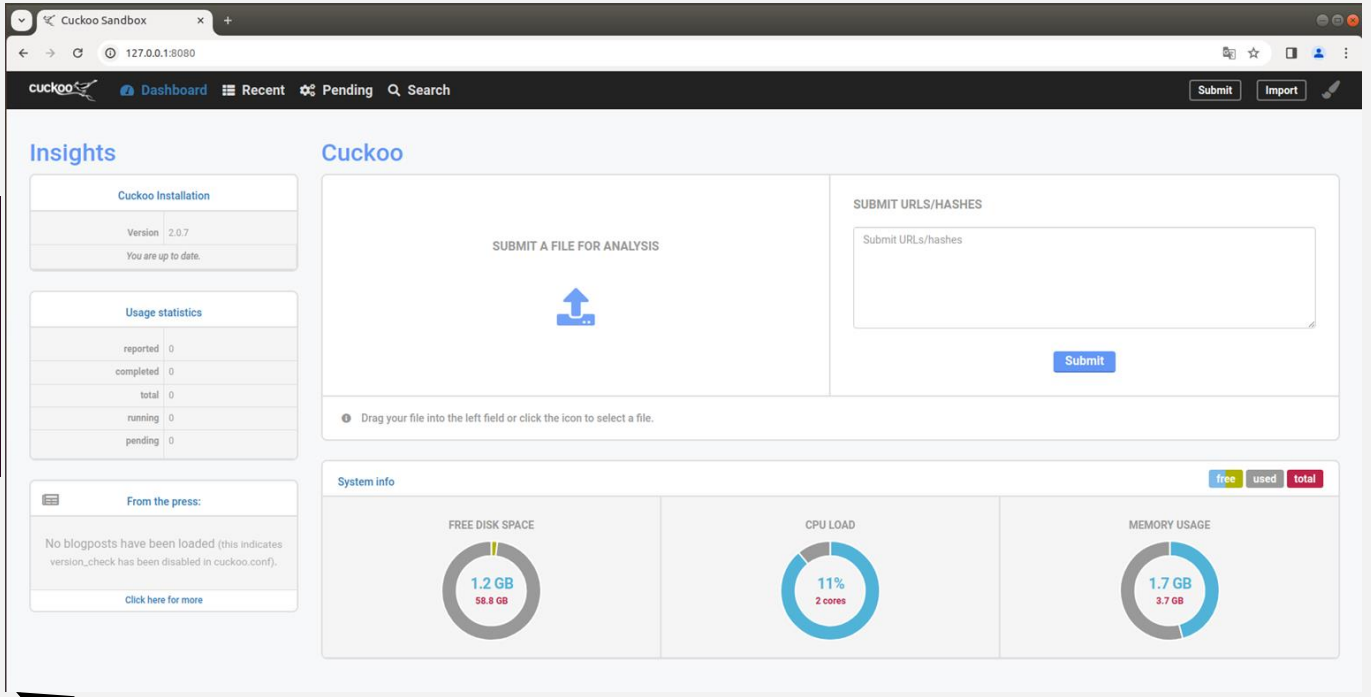


啟用CUCKOO SANDBOX(3)

40.開第三個新終端T3，進入沙盒，連線到網頁gui server，讓它持續運行:

- workon sandbox
- cuckoo web --host 127.0.0.1 --port 8080

```
float@float-virtual-machine:~$ workon sandbox
(sandbox) float@float-virtual-machine:~$ cuckoo web --host 127.0.0.1 --port 8080
Performing system checks...

System check identified no issues (0 silenced).
March 05, 2024 - 15:45:01
Django version 1.8.4, using settings 'cuckoo.web.web.settings'
Starting development server at http://127.0.0.1:8080/
Quit the server with CONTROL-C.
```



41.開啟網站，這就是cuckoo的gui介面，記得T1、T2、T3都要放著讓它運行，不然CUCKOO會停止運作:

- ***http://127.0.0.1:8080***

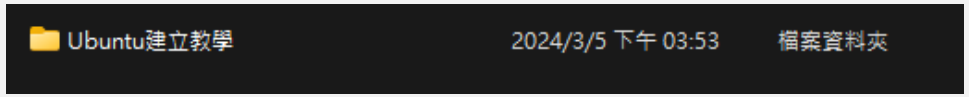
備份與重啟CUCKOO SANDBOX

- 當你成功開啟T1,T2,T3之後，請注意，盡量以暫停虛擬機的方式取代關閉虛擬機，因為一關機的話設定就會跑掉，需要重新設置，否則cuckoo sandbox會無法使用，下面是兩種備份重啟的方法。

(1)虛擬機備份:缺點是整個vm的狀態會回復到備份之前，所有檔案與程式都要重新下載。

1.接下來先把虛擬機暫停，這樣他就會停止在運行的狀態，下次開啟還會繼續運行，不用重開:

2.找到當初存虛擬機的檔案位置，把整個虛擬機資料夾備份，若之後發現壞檔，則覆蓋掉壞掉的檔案，進行回檔:



(2)重啟Cuckoo sandbox:

- 若遇到vm當機,或是需要擴充硬碟空間等不得不關機的狀況時，使用下列步驟重新設定:

1. 用vmclock設定HOST-ONLY network:

- workon sandbox
- vmcloak-vboxnet0

備份與重啟CUCKOO SANDBOX

```
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo sysctl -w net.ipv4.conf.vboxnet0.forwarding=1
[sudo] password for float:
net.ipv4.conf.vboxnet0.forwarding = 1
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo sysctl -w net.ipv4.conf.ens33.forwarding=1
net.ipv4.conf.ens33.forwarding = 1
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -t nat -A POSTROUTING -o ens33 -s 192.168.56.0/24 -j MASQUERADE
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -P FORWARD DROP
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$ sudo iptables -A FORWARD -s 192.168.56.0/24 -j ACCEPT
(sandbox) float@float-virtual-machine:~/.cuckoo/conf$
```

2. 重新設定參數:(在沙盒裡設置)

- sudo sysctl -w net.ipv4.conf.vboxnet0.forwarding=1
- sudo sysctl -w net.ipv4.conf.ens33.forwarding=1
- sudo iptables -t nat -A POSTROUTING -o ens33 -s 192.168.56.0/24 -j MASQUERADE
- sudo iptables -P FORWARD DROP
- sudo iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
- sudo iptables -A FORWARD -s 192.168.56.0/24 -j ACCEPT

3.重新開啟T1 T2 T3:

T1:

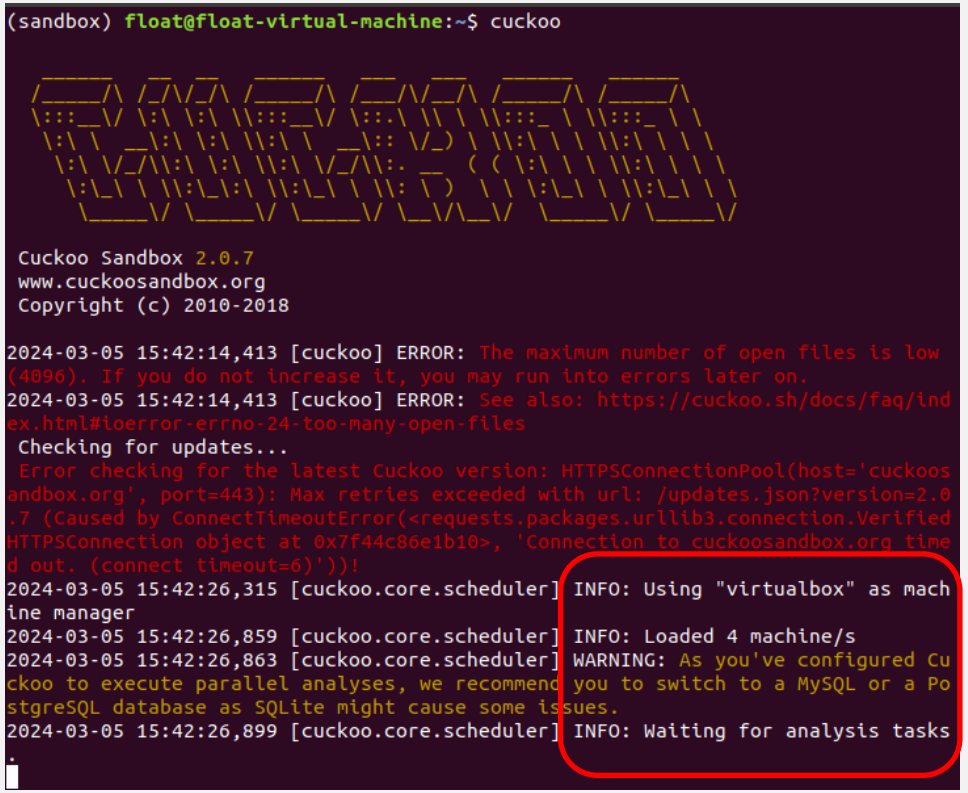
- workon sandbox
- cuckoo rooter --sudo --group username

T2:

- workon sandbox
- Cuckoo

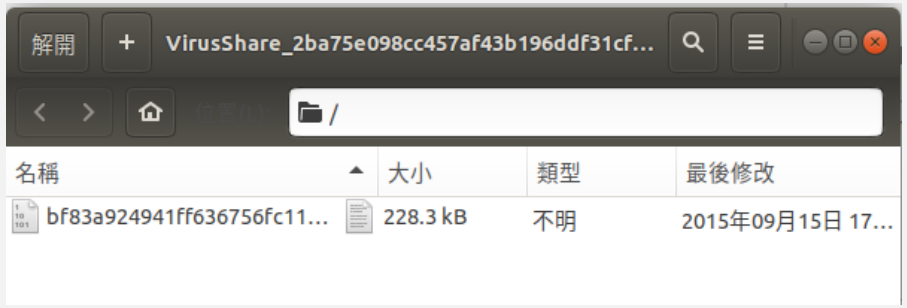
T3:

- workon sandbox
- cuckoo web --host 127.0.0.1 --port 8080

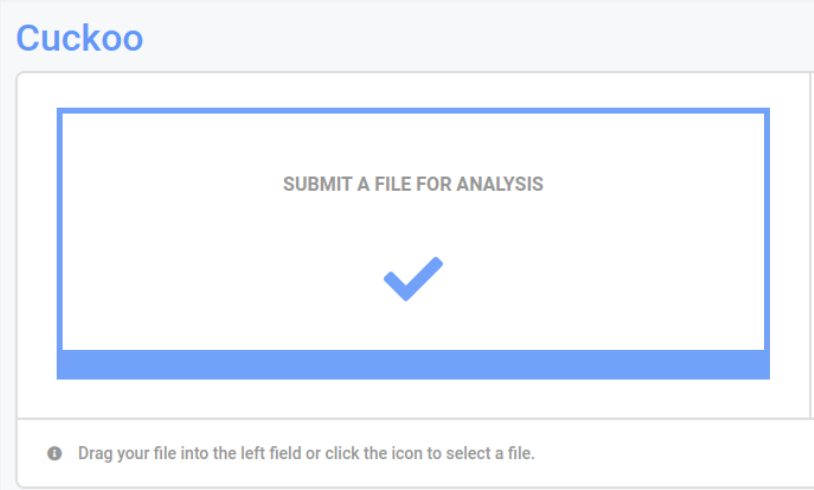


使用CUCKOO SANDBOX

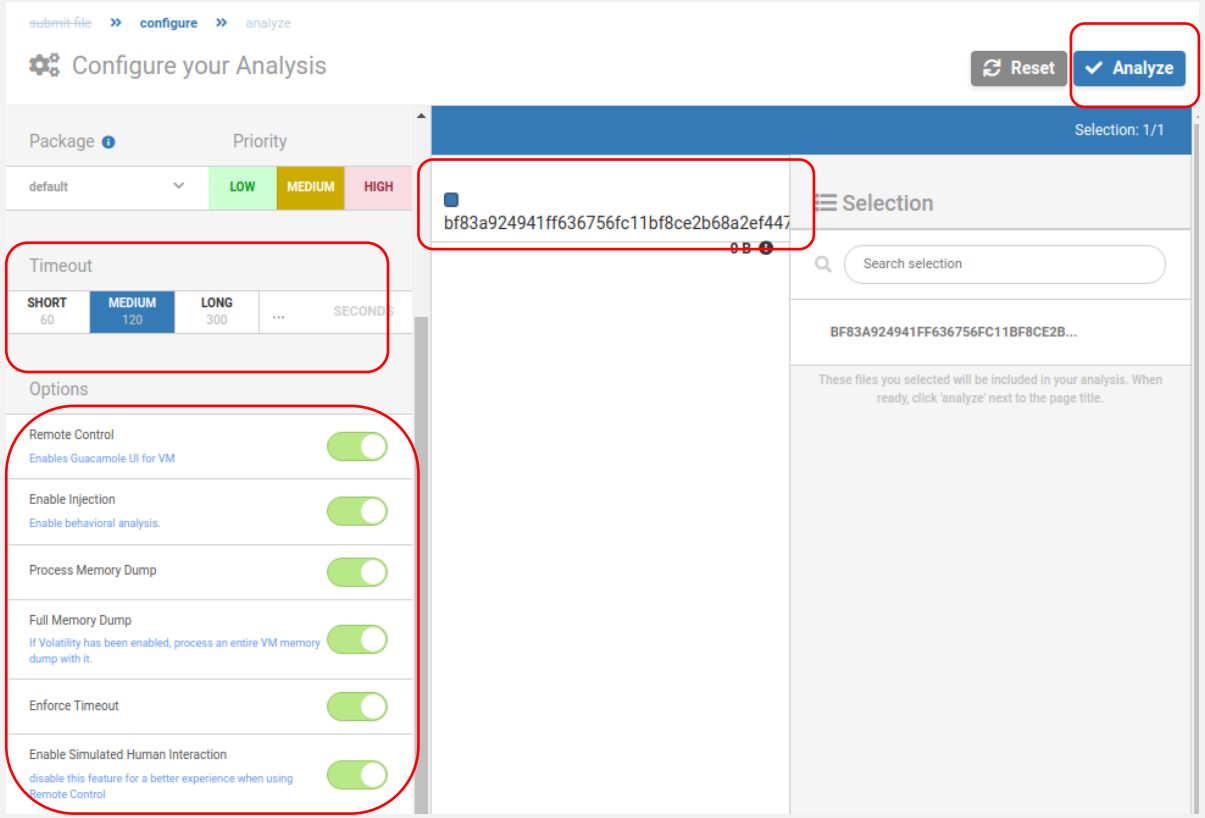
1.將病毒檔解壓縮:



2.丟入中間的提交區:

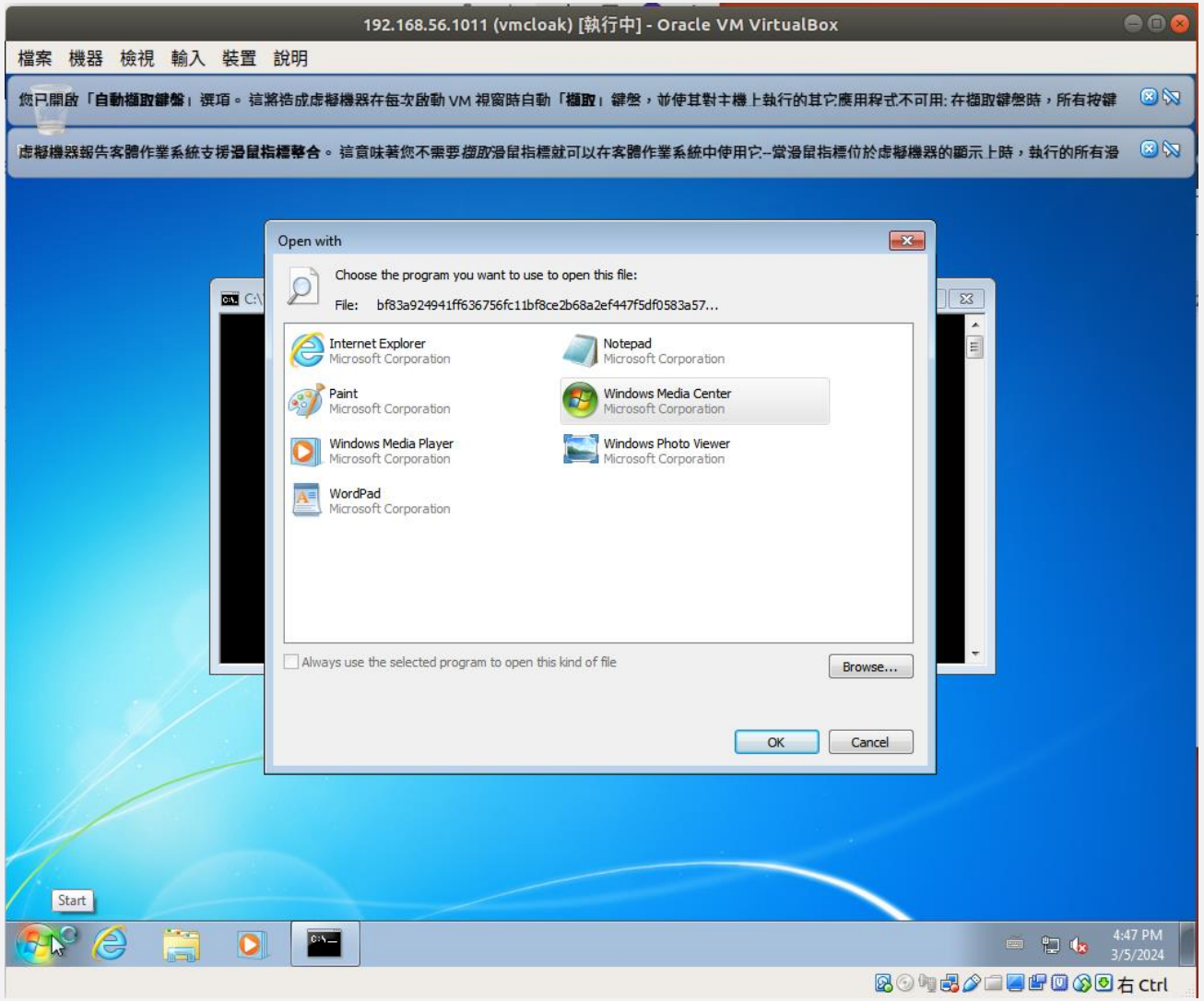


3.設定執行時間等設定，勾選該病毒檔，開始分析:

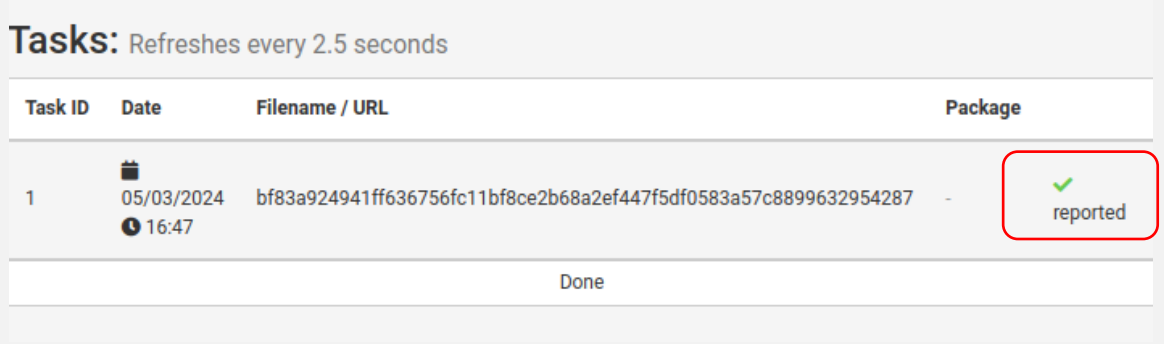


使用CUCKOO SANDBOX

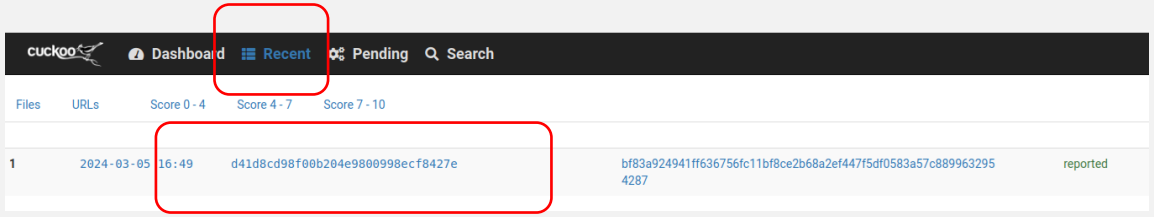
4.它會自動執行win7虛擬機進行分析:



5.結束之後它會打勾:

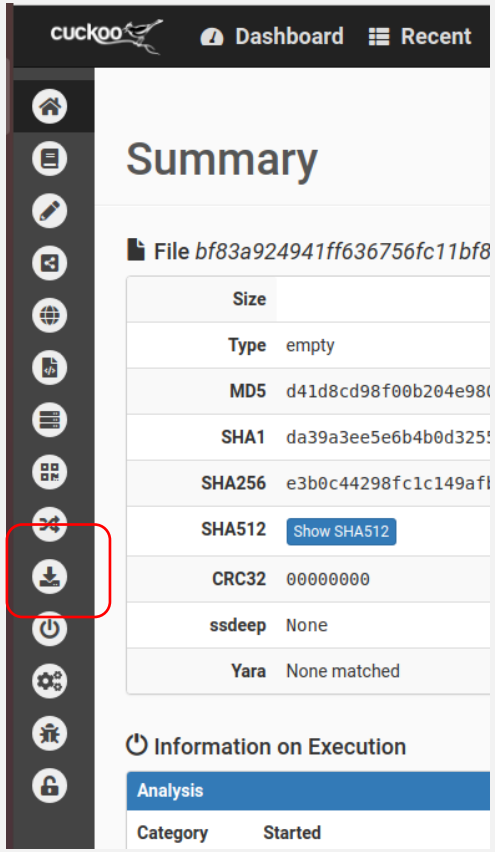


6.按recent，選擇你要下載的分析報告:

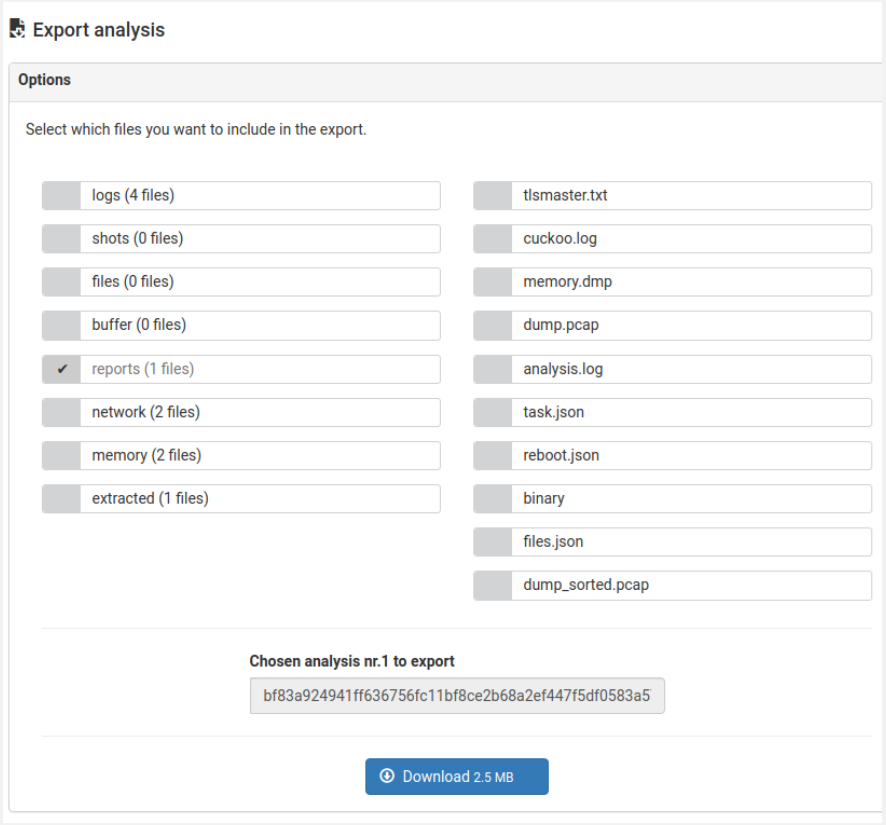


使用CUCKOO SANDBOX

7.點擊輸出分析報告:

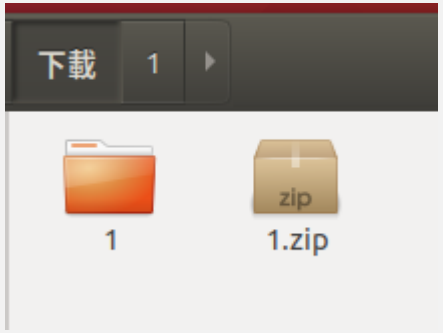


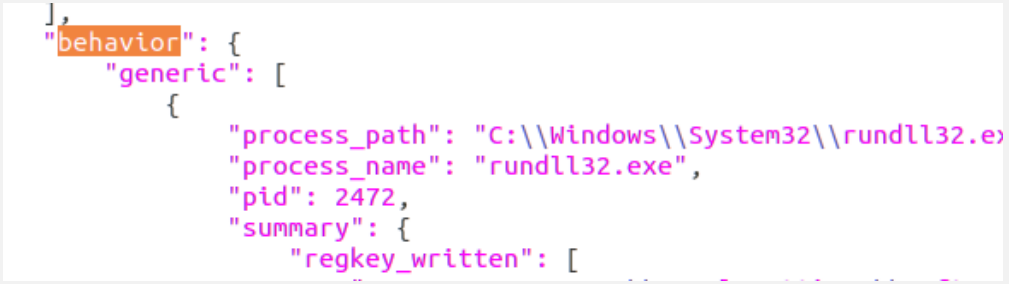
8.只要勾選reports並下載:



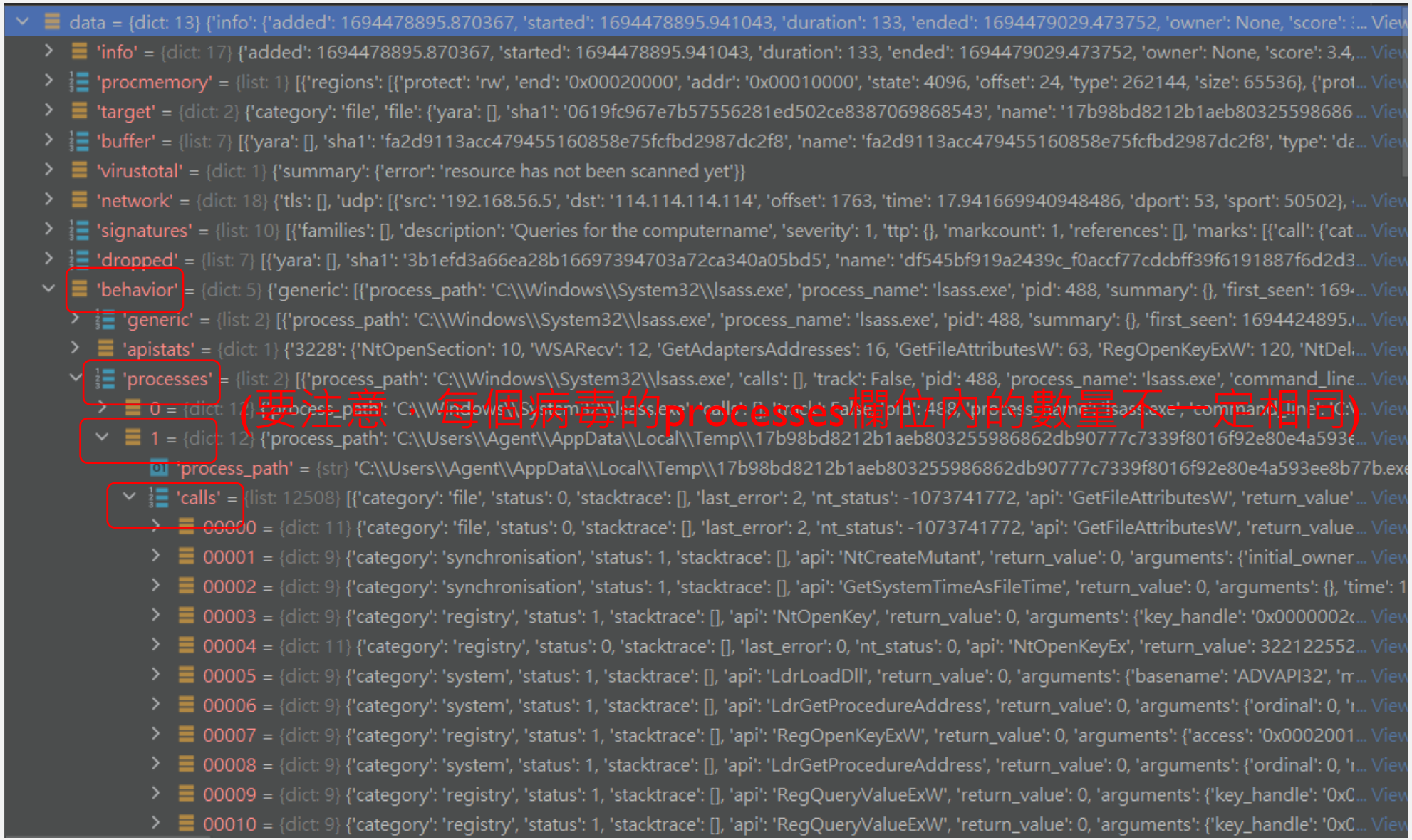
9.會得到一個壓縮檔，點進去找report.json:

(其中的behavior裡面的processes裡的calls欄位就是我們要的結果)

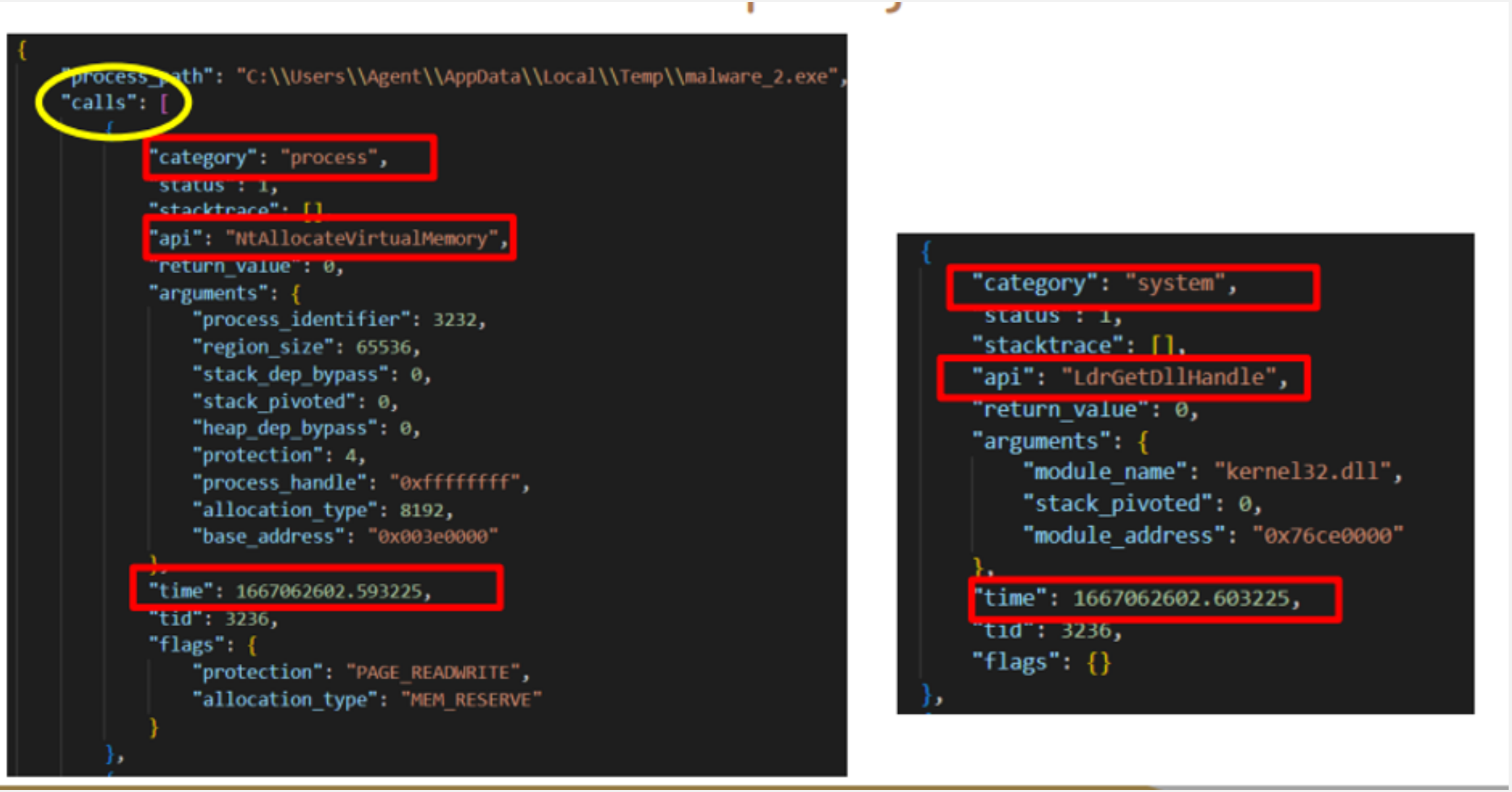




繪製CALL API SEQUENCE



繪製CALL API SEQUENCE



繪製CALL API SEQUENCE

- 表格記錄了call的種類、功能、時間

	A	B	C	D
1	process_number	call_category	call_api	call_time
2	1	file	NtWriteFile	1694408112
3	1	file	NtWriteFile	1694408112
4	1	file	NtWriteFile	1694408112
5	1	file	NtWriteFile	1694408112
6	1	file	NtWriteFile	1694408112
7	1	file	NtWriteFile	1694408112
8	1	file	NtWriteFile	1694408112
9	1	file	NtWriteFile	1694408112
10	1	file	NtWriteFile	1694408112
11	1	file	NtWriteFile	1694408112

繪製CALL API SEQUENCE

API call sequence 轉成影像

根據paper上的規則轉成影像

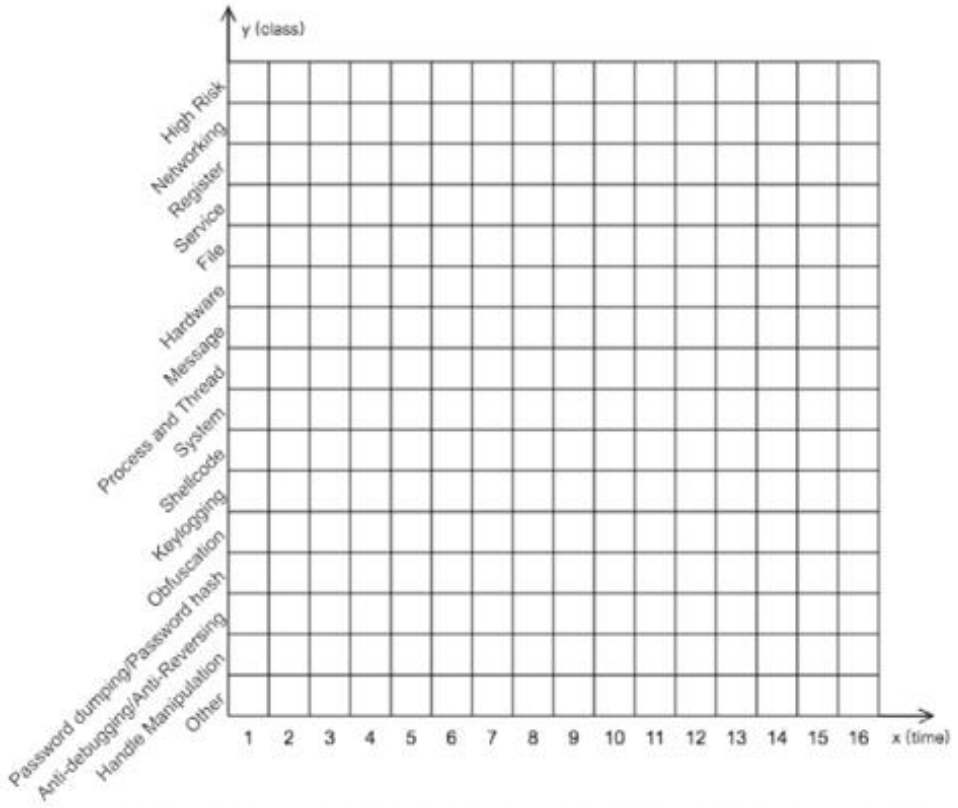
其中X軸為時間軸，切成16等分

Y軸為API call 的種類及功能，也是16等分

X軸的部分，取最早的API call當 start time

最晚的API call當 end time

Y軸的項目為自定義不一定要照著paper 的分類。



繪製CALL API SEQUENCE

color_mapping_num.csv

range_f	range_e	networkin	registry	service	file	system	message	process	LdrGetPrc	synchronis	NtQueryV	LdrUnloac	NtOpenDi	ReadProce	CreatePro	NtAllocate	NtReadFil
0	0	0	11	22	33	44	55	66	77	88	99	110	121	132	143	154	165
0	3	1	12	23	34	45	56	67	78	89	100	111	122	133	144	155	166
3	7	2	13	24	35	46	57	68	79	90	101	112	123	134	145	156	167
7	12	3	14	25	36	47	58	69	80	91	102	113	124	135	146	157	168
12	18	4	15	26	37	48	59	70	81	92	103	114	125	136	147	158	169
18	25	5	16	27	38	49	60	71	82	93	104	115	126	137	148	159	170
25	33	6	17	28	39	50	61	72	83	94	105	116	127	138	149	160	171
33	42	7	18	29	40	51	62	73	84	95	106	117	128	139	150	161	172
42	100	8															
100	200	9															
200	1000	10															

(參考論文中有定義)

color_mapping.csv

range_f	range_e	networkin	registry	service	file	system	message	process	LdrGetPrc	synchronis	NtQueryV	LdrUnloac	NtOpenDi	ReadProce	CreatePro	NtAllocate	NtReadFil
0	0	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF	#FFFFFF
0	3	#FFC1E0	#FFBFFF	#FFDAC8	#D3FF93	#CAFFFF	#C1FFE4	#D6D6AI	#DCB5FF	#FFFF6F	#FFE66F	#D8D8EE	#97CBFF	#B9B9FF	#FFD1A4	#FF9797	#93FF93
3	7	#FFAAD5	#FFA6FF	#FFCBB3	#CCFF80	#BBFFFF	#ADFED(	#CDCD9/	#D3A4FF	#FFFF37	#FFE153	#C7C7E2	#84C1FF	#AAAAFF	#FFC78E	#FF7575	#79FF79
7	12	#FF95CA	#FF8EFF	#FFBD9D	#B7FF4A	#A6FFFF	#96FED1	#C2C287	#CA8EFF	#F9F900	#FFDC35	#B8B8DC	#66B3FF	#9393FF	#FFBB77	#FF5151	#53FF53
12	18	#FF79BC	#FF77FF	#FFAD86	#A8FF24	#4DFFFF	#4EFEB3	#B9B973	#BE77FF	#E1E100	#FFD306	#A6A6D2	#46A3FF	#7D7DFF	#FFAF60	#FF2D2D	#28FF28
18	25	#FF60AF	#FF44FF	#FF9D6F	#9AFF02	#00FFFF	#1AFD9C	#AFAF61	#B15BFF	#C4C400	#EAC100	#9999CC	#2894FF	#6A6AFF	#FFA042	#FF0000	#00EC00
25	33	#FF359A	#FF00FF	#FF8F59	#8CEA00	#00E3E3	#02F78E	#A5A552	#9F35FF	#A6A600	#D9B300	#8080C0	#0080FF	#4A4AFF	#FF9224	#EA0000	#00DB00
33	42	#FF0080	#E800E8	#FF8040	#82D900	#00CACA	#02DF82	#949449	#921AFF	#8C8C00	#C6A300	#7373B9	#0072E3	#2828FF	#FF8000	#CE0000	#00BB00
42	100	#F00078	#D200D2	#FF5809	#73BF00	#00AEAE	#01B468	#808040	#8600FF	#737300	#737300	#5A5AAI	#0066CC	#0000E3	#EA7500	#AE0000	#00A600
100	200	#D9006C	#AE00AE	#F75000	#64A600	#009393	#019858	#707038	#6E00FF	#5B5B00	#5B5B00	#5151A2	#005AB5	#0000C6	#D26900	#930000	#009100
200	1000	#BF0060	#930093	#D94600	#548C00	#005757	#01814A	#616130	#5B00AE	#5B5B00	#5B5B00	#484891	#004B97	#0000C6	#BB5E00	#750000	#007500

繪製CALL API SEQUENCE

撰寫CUCKOO SANDBOX腳本(參考用)

若希望批量製作分析檔，可以使用cuckoo提供的輕量化API server，以python撰寫腳本，將整個資料集逐個上傳分析。

參考網址:https://cuckoo.readthedocs.io/en/latest/usage/api/

REST API

As mentioned in Submit an Analysis, Cuckoo provides a simple and lightweight REST API server that is under the hood implemented using Flask.

Starting the API server

In order to start the API server you can simply do:

```
$ cuckoo api
```

By default it will bind the service on **localhost:8090**. If you want to change those values, you can use the following syntax:

```
$ cuckoo api --host 0.0.0.0 --port 1337
$ cuckoo api -H 0.0.0.0 -p 1337
```

To allow only authenticated access to the API, the `api_token` in `cuckoo.conf` must be set to a secret value. In new Cuckoo installations, a random token is automatically generated for you. To access the API, you must send the `Authorization: Bearer <token>` header with all your requests using the token defined in the configuration. Note that if you want to access the API over an insecure network such as the Internet, you should run the API server behind nginx described in the next section and enable HTTPS.

1.必需套件curl

```
(base) float@float-virtual-machine:~$ sudo apt-get install curl
正在讀取套件清單... 完成
正在重建相依關係
正在讀取狀態資料... 完成
下列的額外套件將被安裝：
  libcurl4
下列【新】套件將會被安裝：
```

2.找到這個檔案，刪除is_xhr這個參數，
才能成功使用api server的command

json.py
~/.virtualenvs/sandbox/lib/python2.7/site-packages/flask

```
if current_app.config['JSONIFY_PRETTYPRINT_REGULAR'] and not request.is_xhr:
    indent = 2
    separators = (', ', ': ')
```

```
if current_app.config['JSONIFY_PRETTYPRINT_REGULAR']:
    indent = 2
    separators = (', ', ': ')
```

撰寫CUCKOO SANDBOX 腳本(參考用)

3.使用cuckoo api開啟官方host:8090的api server:(與web server:8080不衝突)

Starting the API server

In order to start the API server you can simply do:

```
$ cuckoo api
```

4.每個vm用戶會有一組專屬的憑證，用於在api server中識別你的project，而不會與其他人的project搞混:

To allow only authenticated access to the API, the `api_token` in `cuckoo.conf` must be set to a secret value. In new Cuckoo installations, a random token is automatically generated for you. To access the API, you must send the `Authorization: Bearer <token>` header with all your requests using the token defined in the configuration. Note that if you want to access the API over an insecure network such as the Internet, you should run the API server behind nginx described in the next section and enable HTTPS.

撰寫CUCKOO SANDBOX腳本(參考用)

5.使用/tasks/create/file來上傳病毒檔並分析，若上傳成功，你會獲得一個任務序號 (可於web server ui查看進度)

POST /tasks/create/file	Adds a file to the list of pending tasks to be processed and analyzed.

```python
import requests

REST_URL = "http://localhost:8090/tasks/create/file"
SAMPLE_FILE = "/path/to/malwr.exe"
HEADERS = {"Authorization": "Bearer S4MPL3"}

with open(SAMPLE_FILE, "rb") as sample:
    files = {"file": ("temp_file_name", sample)}
    r = requests.post(REST_URL, headers=HEADERS, files=files)

# Add your code to error checking for r.status_code.

task_id = r.json()["task_id"]

# Add your code for error checking if task_id is None.
```

撰寫CUCKOO SANDBOX腳本(參考用)

6. 在分析完成後，你就可以使用剛剛獲得的task_id，以指令的形式下載report

/tasks/report

GET /tasks/report/ *(int: id) / (str: format)*

Returns the report associated with the specified task ID.

Example request.

```
curl -H "Authorization: Bearer S4MPL3" http://localhost:8090/tasks/report/1
```

Parameters:

- `id` *(required) (int)* - ID of the task to get the report for
- `format` *(optional)* - format of the report to retrieve [json/html/all/dropped/package_files]. If none is specified the JSON report will be returned. `all` returns all the result files as tar.bz2, `dropped` the dropped files as tar.bz2, `package_files` files uploaded to host by analysis packages.

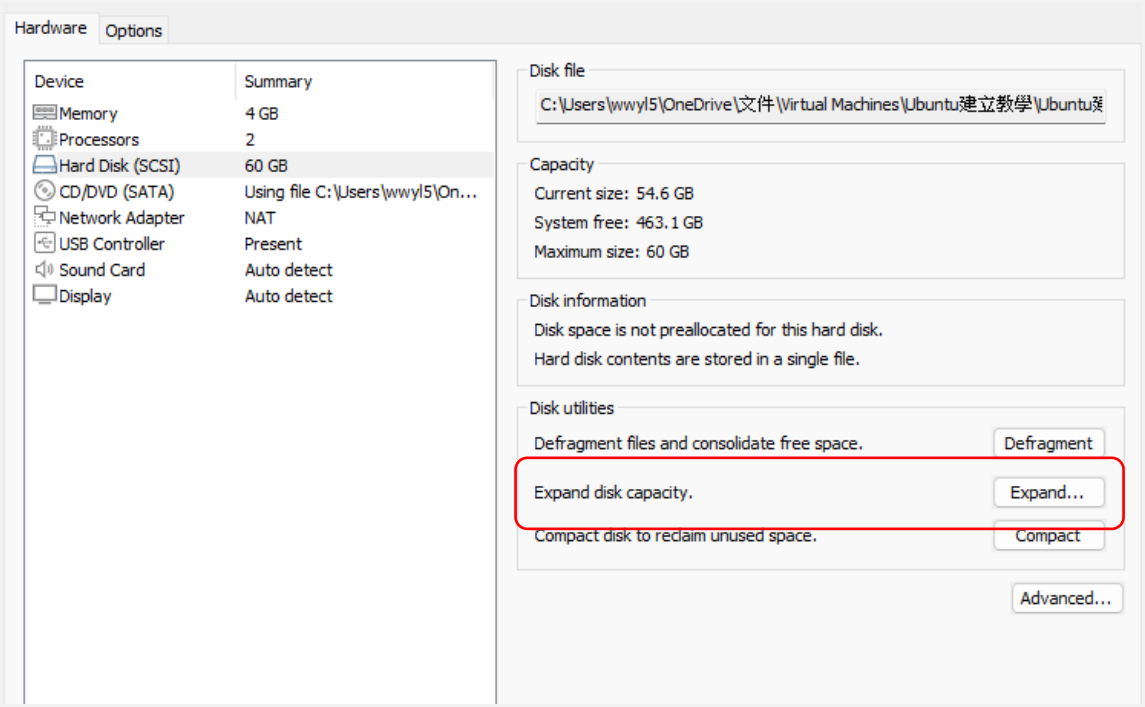
Status codes:

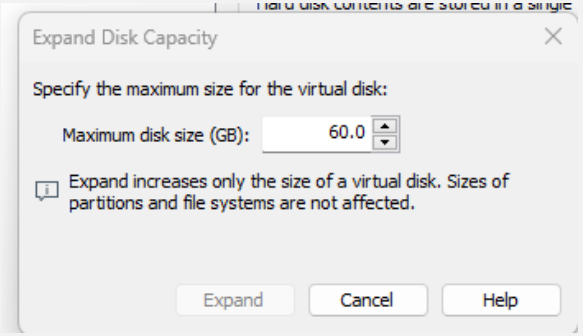
- `200` - no error
- `400` - invalid report format
- `404` - report not found

虛擬機硬碟空間擴充(參考用)

- 如果你發現你的容量耗盡，可以透過擴充空間來增加容量，但注意必須要關機才可以操作。

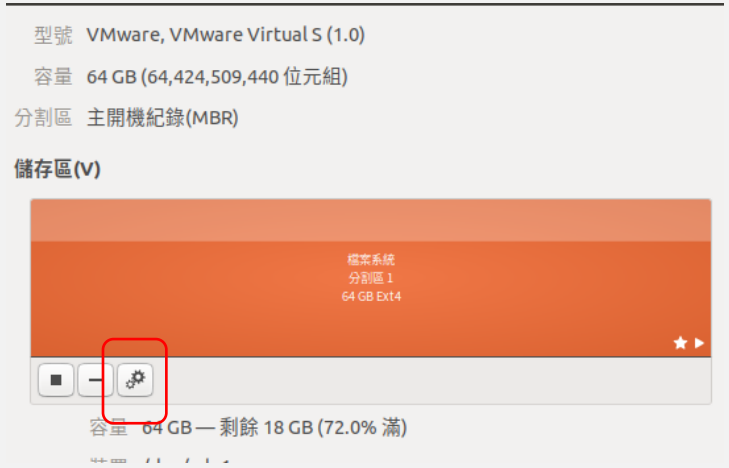
1.點擊擴充，提高最大容量





2.開啟”磁碟”:

3.點齒輪選擇”調整大小”，把它拉滿才會應用在虛擬機裡:



PYCHARM+ANACONDA安裝教學

(PYTHON 使用環境不限，這裡提供UBUNTU虛擬機的PYCHARM環境設置使用教學)

CS4003701

資訊安全導論

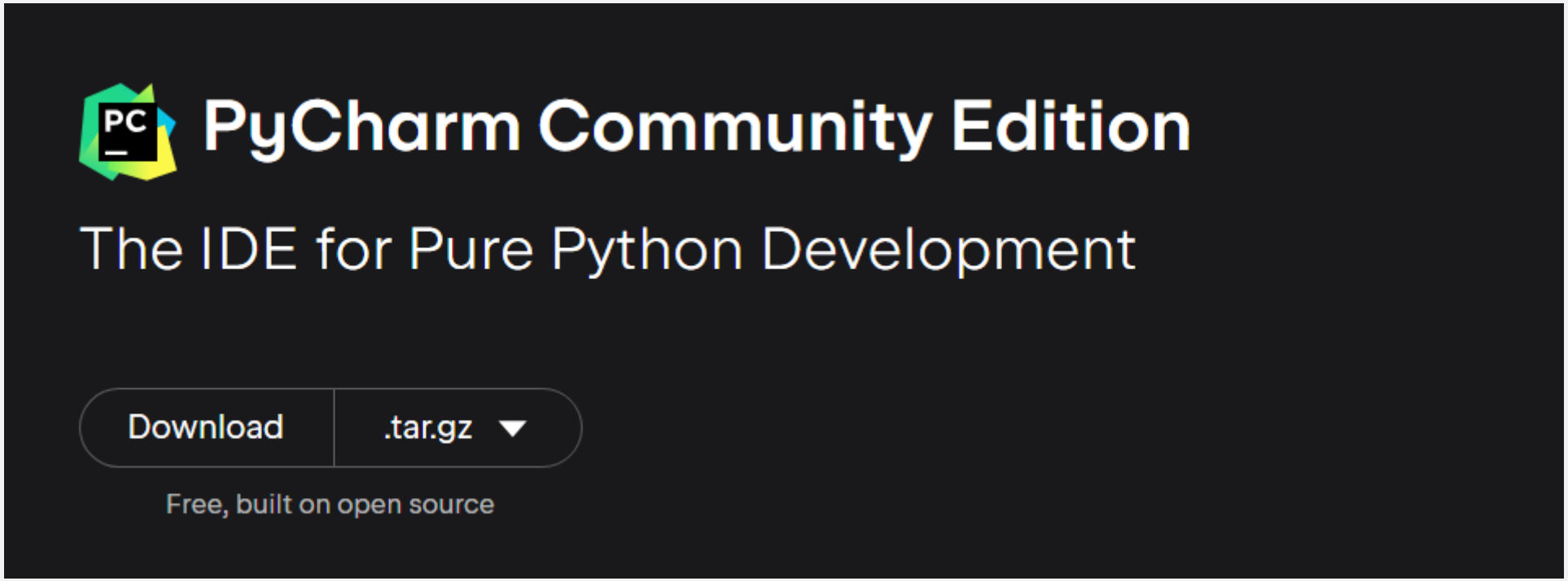
使用PYCHARM +ANACONDA

- 可以建立獨立的Python環境來使用，以避免不同套件的版本互相感擾
- 未來可建立不同環境來區分不同類型的作業
- Anaconda是建立環境與下載套件的工具
- Pycharm是開啟程式碼的GUI環境

安裝PYCHARM +ANACONDA

- https://www.jetbrains.com/pycharm/download/?section=linux

記得要下載community版

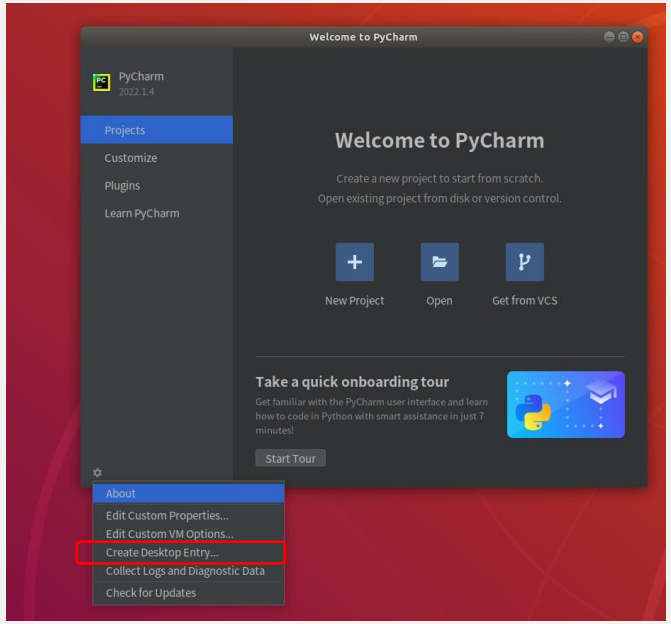


安裝PYCHARM +ANACONDA

1.解壓縮

```
float@float-virtual-machine:~$ cd 下載
float@float-virtual-machine:~/下載$ cd pycharm-community-[redacted]
float@float-virtual-machine:~/下載/pycharm-community-2022.1.4$ cd bin
float@float-virtual-machine:~/下載/pycharm-community-2022.1.4/bin$ sh ./pycharm.sh
2024-03-05 00:16:13,033 [   2397]   WARN - #c.i.o.a.i.ActionManagerImpl - keymap "Visual Studio" not found PluginDescriptor(name=IDEA CORE, id=com.intellij, descriptorPath=plugin.xml, path=~/下載/pycharm-community-[redacted]lib, version=221.6008.17, package=com.intellij.feedback, isBundled=true)
```

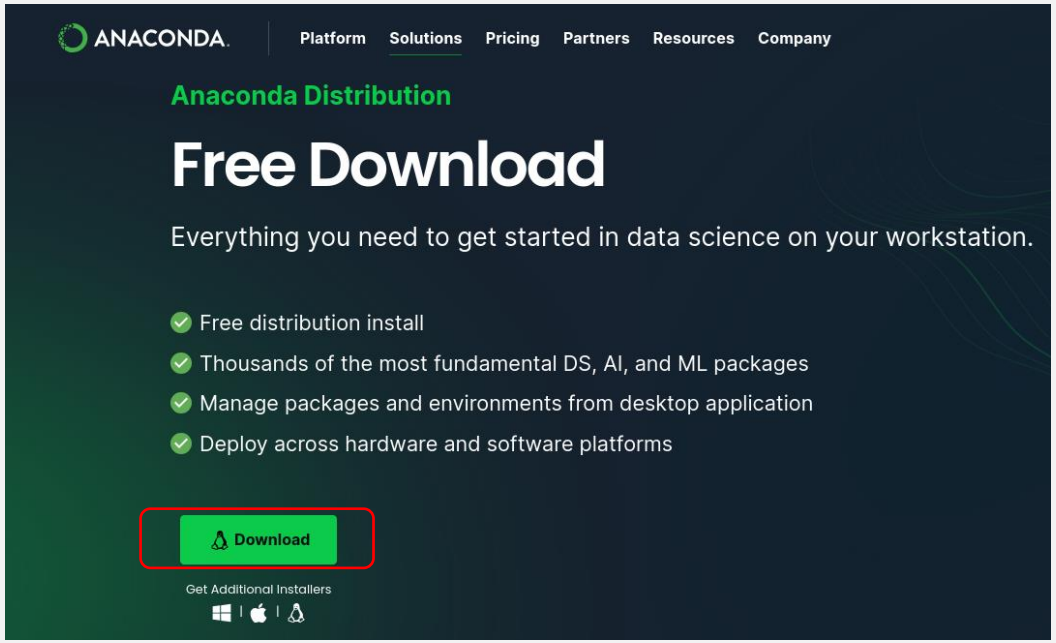
2.用終端機進入bin資料夾並執行



3.點左下角齒輪新增桌面捷徑

4.就可在本機軟體搜尋找到它

安裝PYCHARM +ANACONDA



- https://www.anaconda.com/download/#linux

1.點選連結進去，點選Linux安裝包
2.進到下載資料夾，執行腳本檔
3.要按Enter然後慢慢把用戶協議看完
4.最後要輸入yes同意，然後就確認安裝路徑

```
float@float-virtual-machine:~$ cd 下載
float@float-virtual-machine:~/下載$ bash Anaconda3-2024.02-1-Linux-x86_64.sh

Welcome to Anaconda3 2024.02-1

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>>
END USER LICENSE AGREEMENT
```

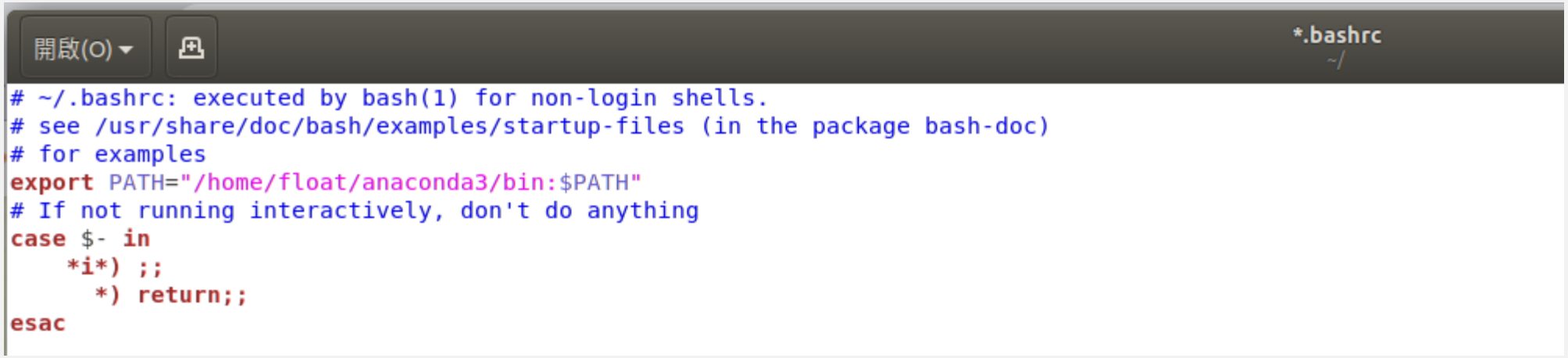
```
Please answer 'yes' or 'no':'
>>> yes

Anaconda3 will now be installed into this location:
/home/float/anaconda3

  - Press ENTER to confirm the location
  - Press CTRL-C to abort the installation
  - Or specify a different location below

[/home/float/anaconda3] >>>
PREFIX=/home/float/anaconda3

Unpacking payload ...

Installing base environment...
```

設定PYTHON環境變數

1.進入環境變數設定檔

```
float@float-virtual-machine:~$ sudo gedit ~/.bashrc
```

2.在隨意地方新增export PATH="/home/username/anaconda3/bin:$PATH"，並儲存



3.讓環境變數生效

```
float@float-virtual-machine:~$ source ~/.bashrc
```

4.檢查是否成功，成功會顯示下面的路徑

```
float@float-virtual-machine:~$ which python
/home/float/anaconda3/bin/python
```

建立測試環境

1.建立一個名為test_env，python直譯器版本為3.7的環境

```
float@float-virtual-machine:~$ conda create --name test_env python=3.7
```

2.啟動環境

```
float@float-virtual-machine:~$ conda activate test_env
```

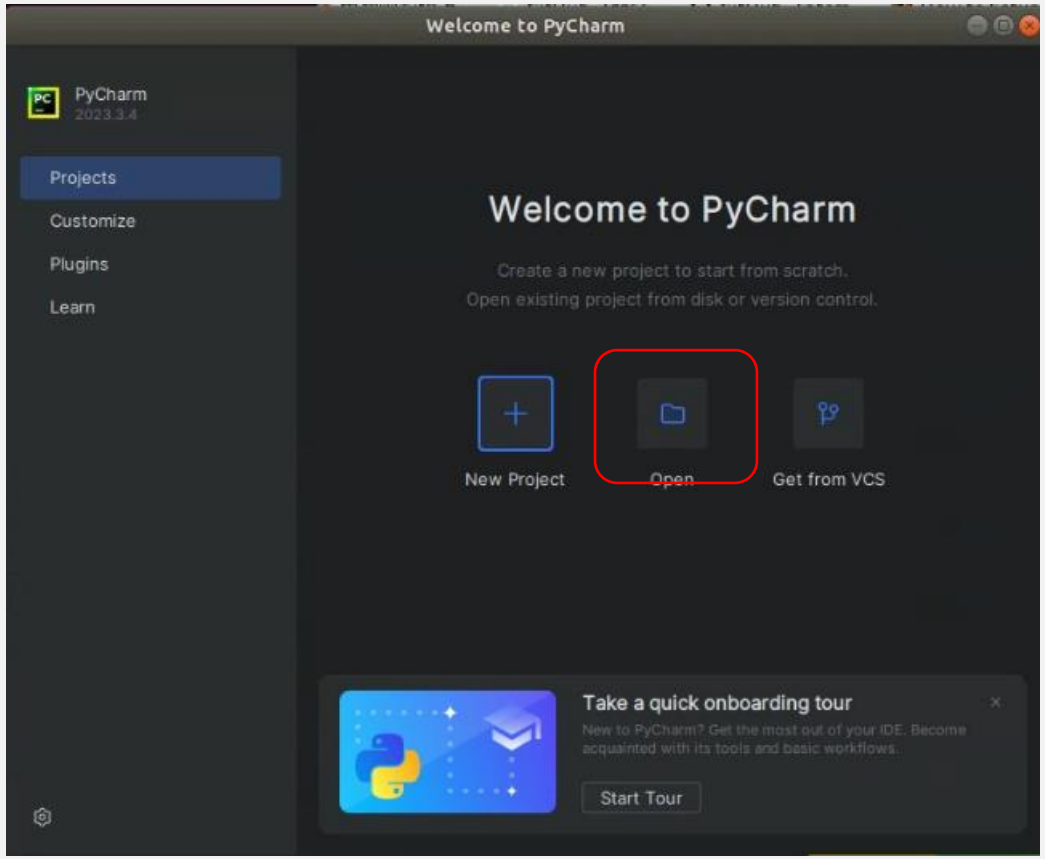
3.使用 ”conda install 套件名稱” or “pip install 套件名稱” 來下載執行所需套件庫

```
(test_env) float@float-virtual-machine:~$ pip install numpy
Collecting numpy
  Downloading numpy-1.21.6-cp37-cp37m-manylinux_2_12_x86_64.manylinux2010_x86_64
.whl (15.7 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 15.7/15.7 MB 8.3 MB/s eta 0:00:00
Installing collected packages: numpy
Successfully installed numpy-1.21.6
(test_env) float@float-virtual-machine:~$
```

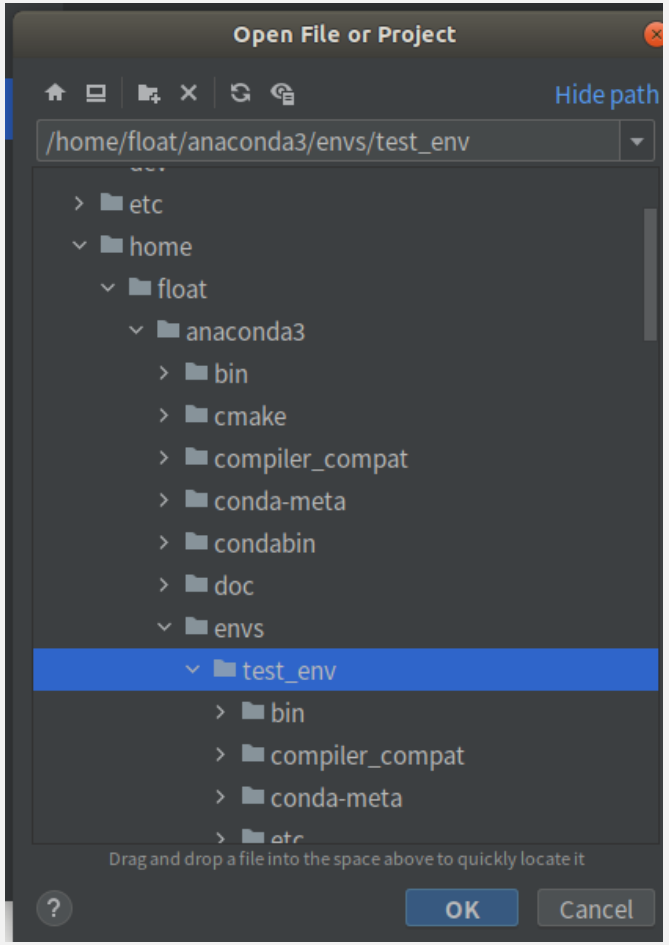
PS.若出現”conda init before conda activate”的報錯，請再輸入一次”source ~/.bashrc”，讓環境變數重新生效，並關掉終端機再開一個。

建立測試環境

4.下載好套件後，打開Pycharm，點選open。

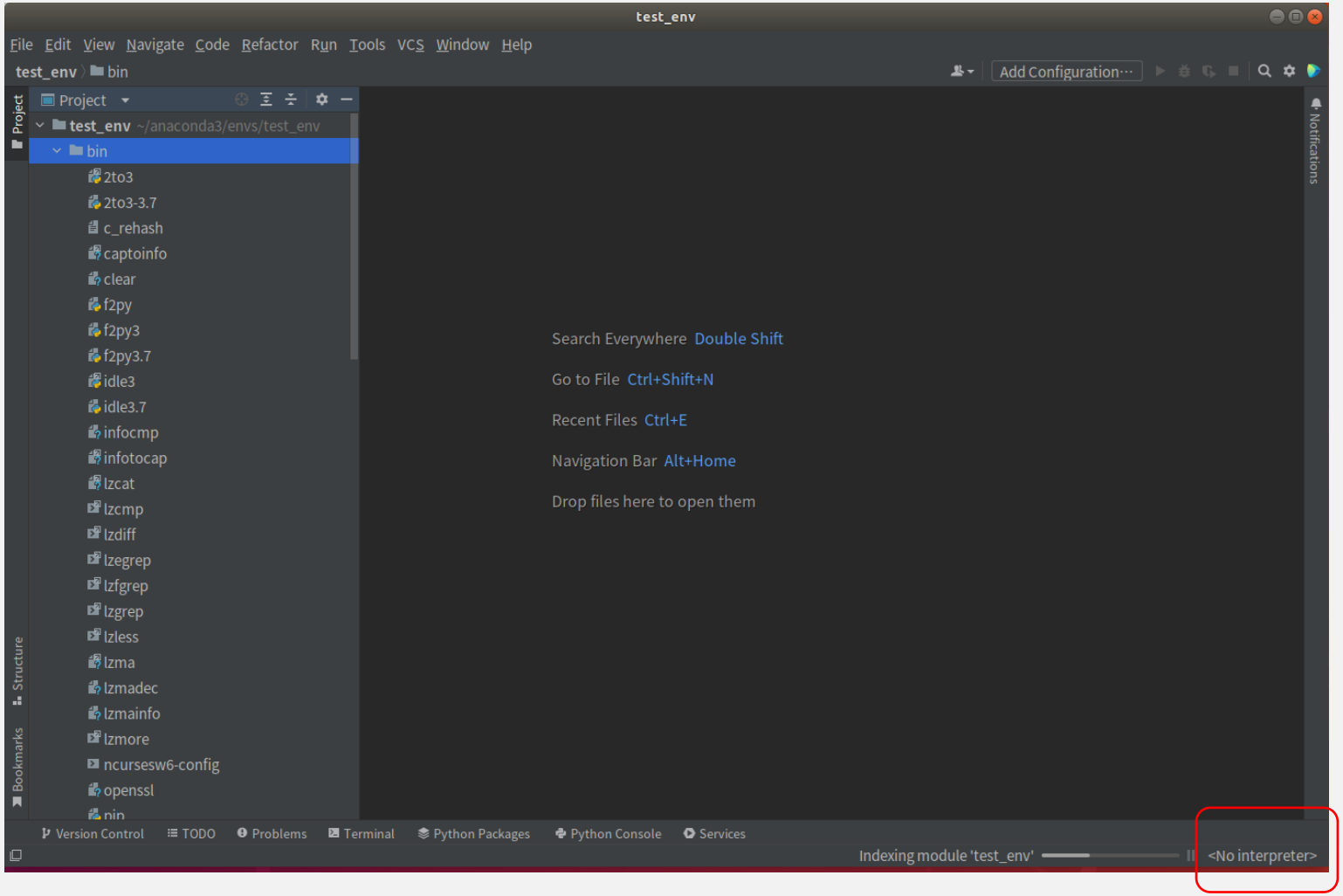


5.找到envs資料夾，點選你想要開的環境的資料夾

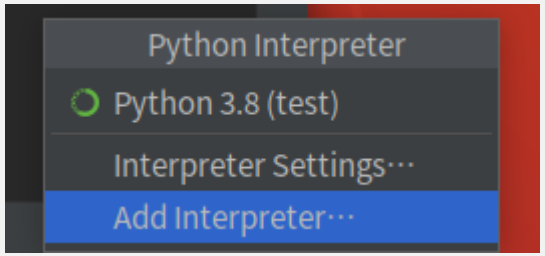


建立測試環境

6.點擊右下角設定直譯器

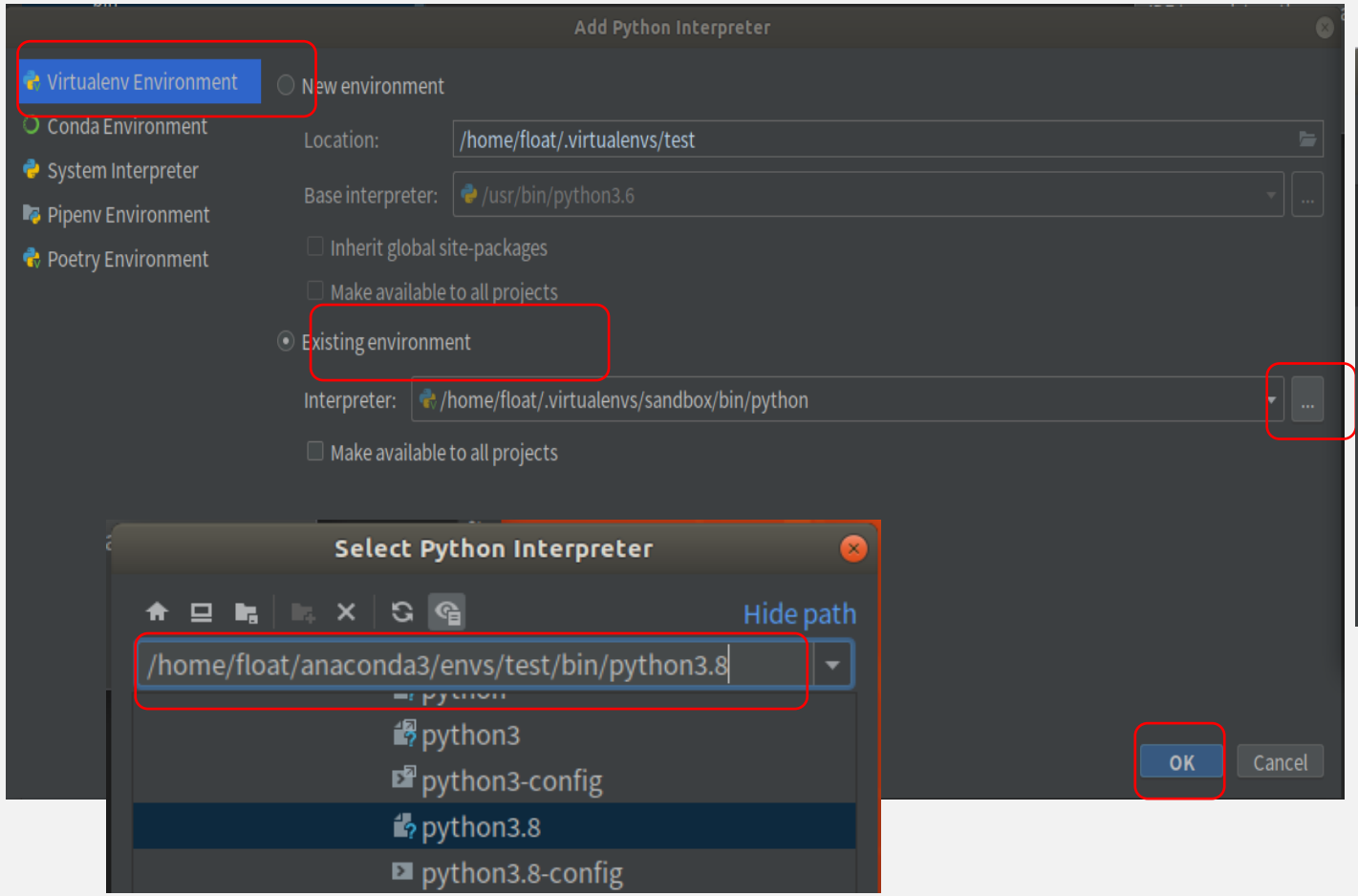


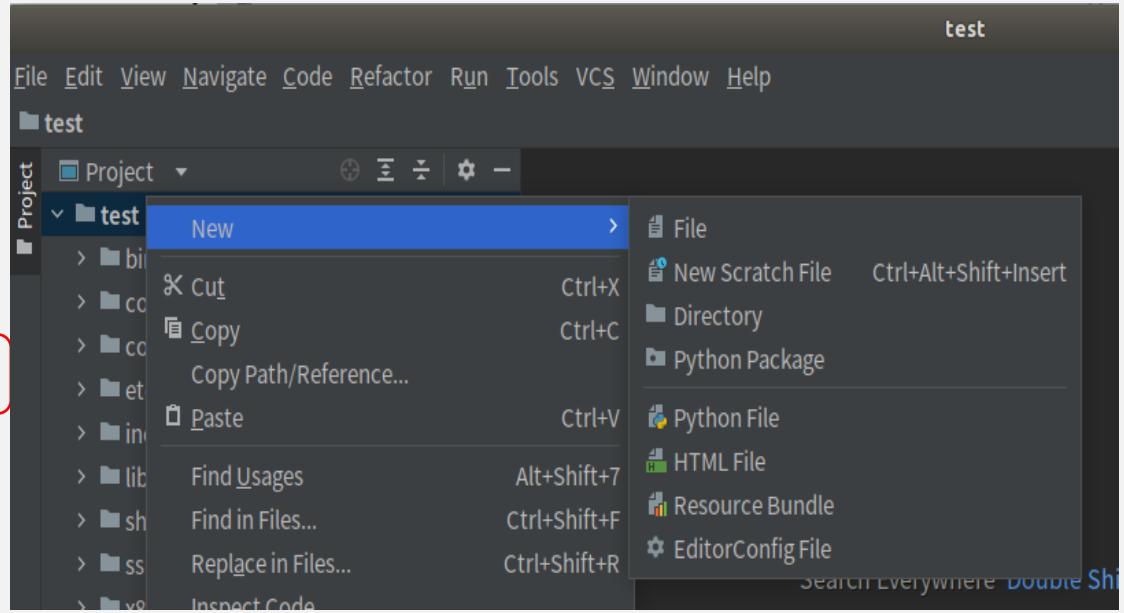
7.如果有你的環境檔名的直譯器可以直接點選就好



建立測試環境

8.如果沒有，則點選Virtualenv Environment
9.點選Exist Enviroment
10.找到環境資料夾裡的python3.x檔案(有些版本在bin資料夾裡面，有些版本直接在環境資料夾裡面)
11.最後按ok啟動直譯器
12.啟動後就可以在左邊的檔案區，建立或複製python檔或其他檔案開始開發





建立測試環境

13.點擊Configuration，選擇Current file，即可按旁邊的綠色按鈕執行python檔案