

安裝 HELK

下載 Ubuntu ISO

ubuntu.com/download/desktop

sagishahar/lpeworkshop: Windows / Linux Local Privilege Escalation Workshop

CANONICAL

ubuntu

Enterprise ▾ Developer ▾ Community ▾ Download ▾

Search Sign in

Downloads Overview Cloud IoT Raspberry Pi Server Desktop Alternative downloads Ubuntu flavours

Download Ubuntu Desktop

下載 Ubuntu Desktop 20.04

Ubuntu 20.04.2.0 LTS

Download the latest LTS version of Ubuntu, for desktop PCs and laptops. LTS stands for long-term support — which means five years, until April 2025, of free security and maintenance updates, guaranteed.

[Ubuntu 20.04 LTS release notes](#)

Recommended system requirements:

- 2 GHz dual core processor or better
- 4 GB system memory
- 25 GB of free hard drive space
- Internet access is helpful
- Either a DVD drive or a USB port for the installer media

For other versions of Ubuntu Desktop including torrents, the network installer, a list of local mirrors, and past releases [see our alternative downloads](#).

[Download](#)

Ubuntu 21.04

The latest version of the Ubuntu operating system for desktop PCs and laptops, Ubuntu 21.04 comes with nine months, until January 2022, of security and maintenance updates.

[Download](#)

<https://ubuntu.com/download/desktop>

新增 VM

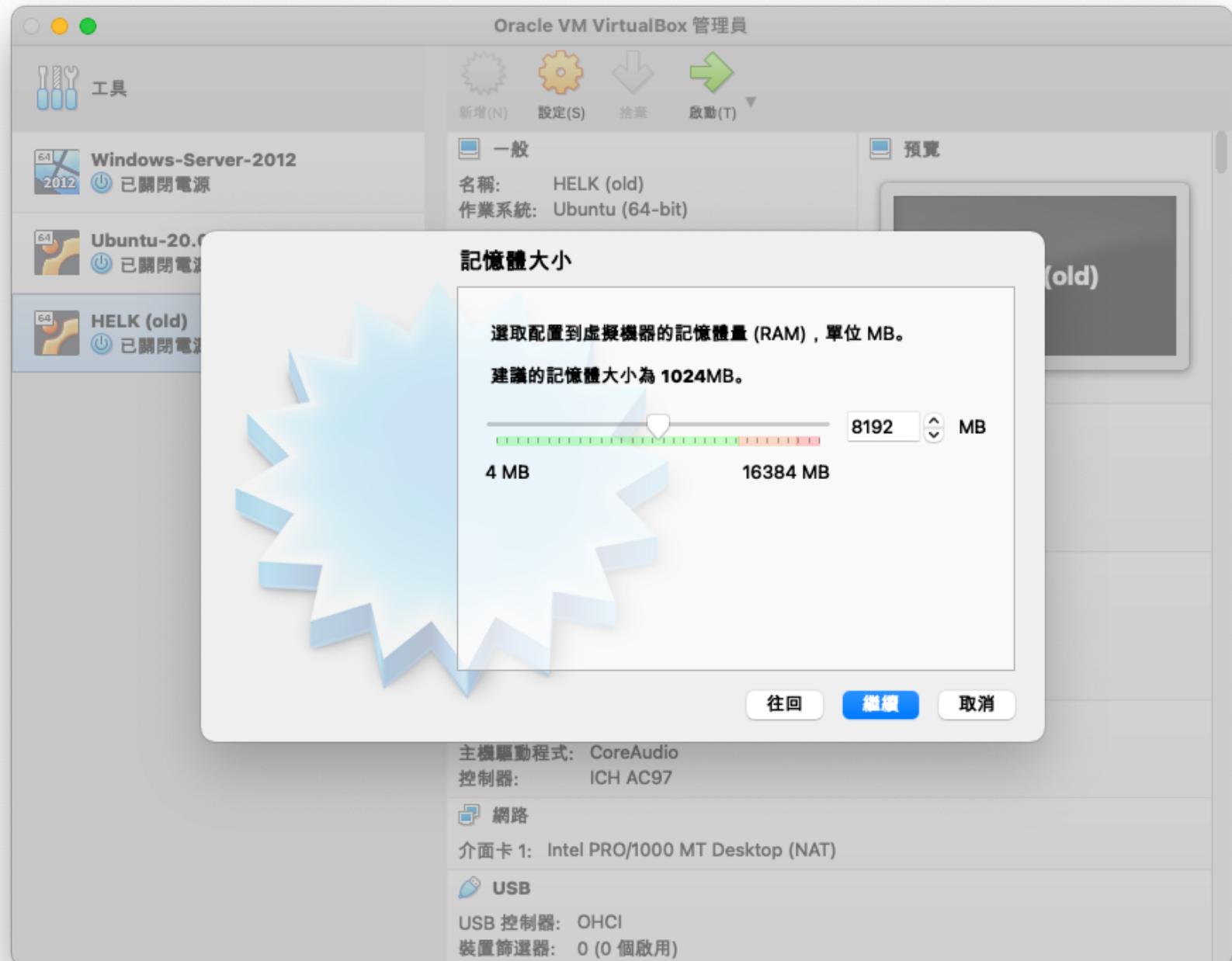
新增 VM

- 名稱：HELK
- 類型：Linux
- 版本：Ubuntu (64 bit)



設定 RAM

- 實測至少需大於 6400 MB
- 建議開到 8192 MB 以上較能順暢執行



設定 Disk

- 選擇「立即建立虛擬硬碟」



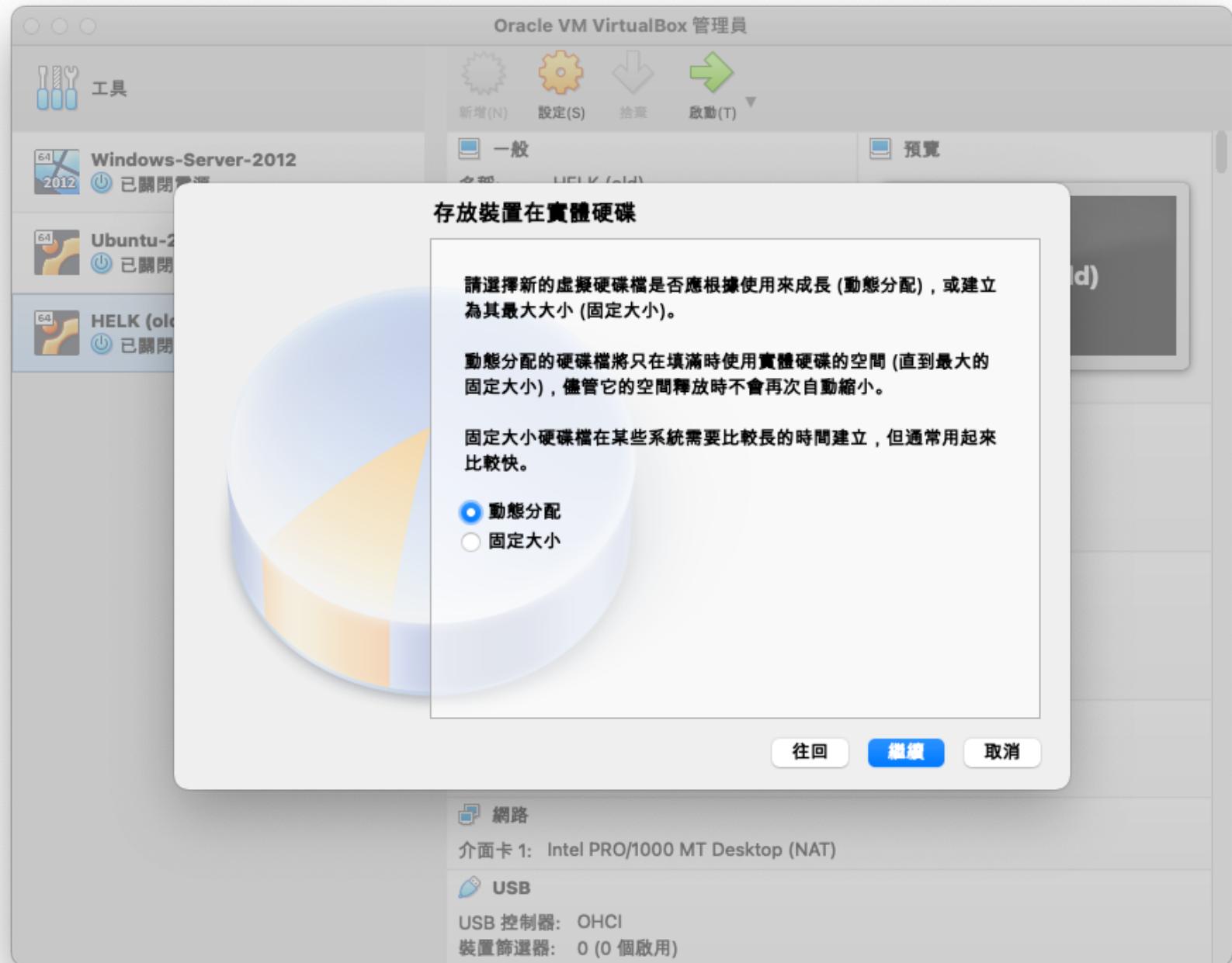
設定 Disk

- 選擇「立即建立虛擬硬碟」
- 選擇「VDI」



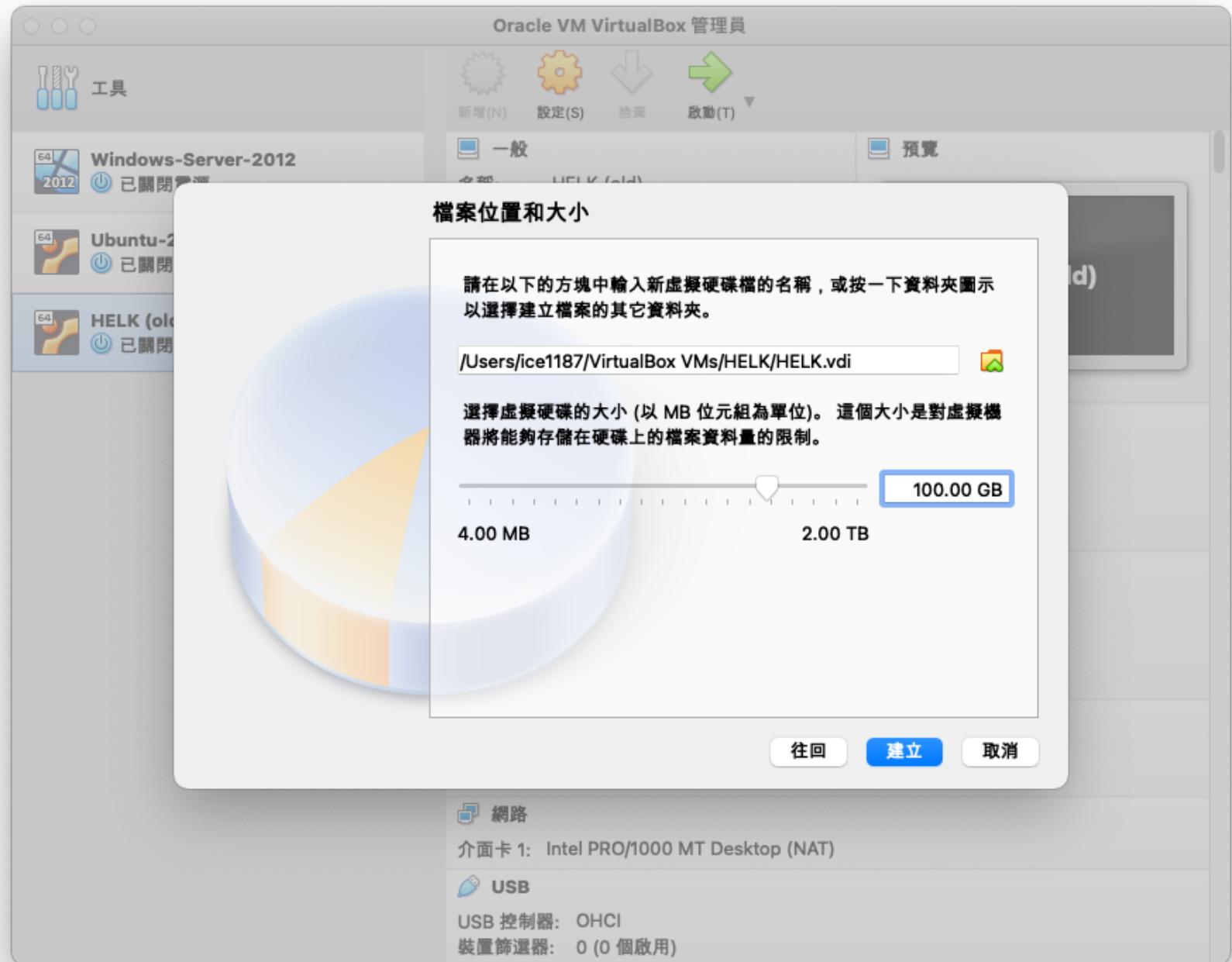
設定 Disk

- 選擇「立即建立虛擬硬碟」
- 選擇「VDI」
- 選擇「動態分配」



設定 Disk

- 選擇「立即建立虛擬硬碟」
- 選擇「VDI」
- 選擇「動態分配」
- 選擇 60 ~ 100 GB



設定

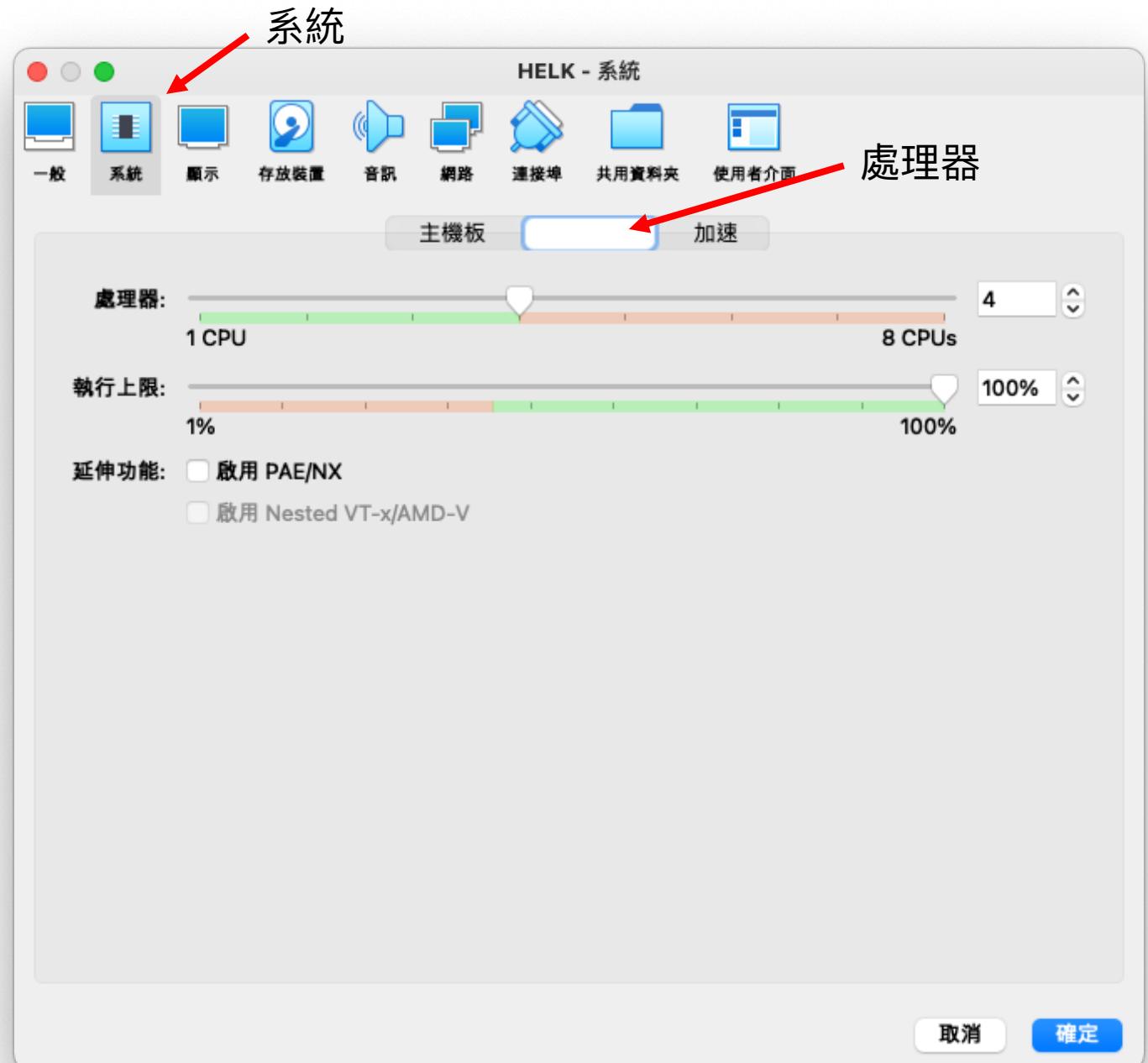
設定硬體

- 選擇「設定」



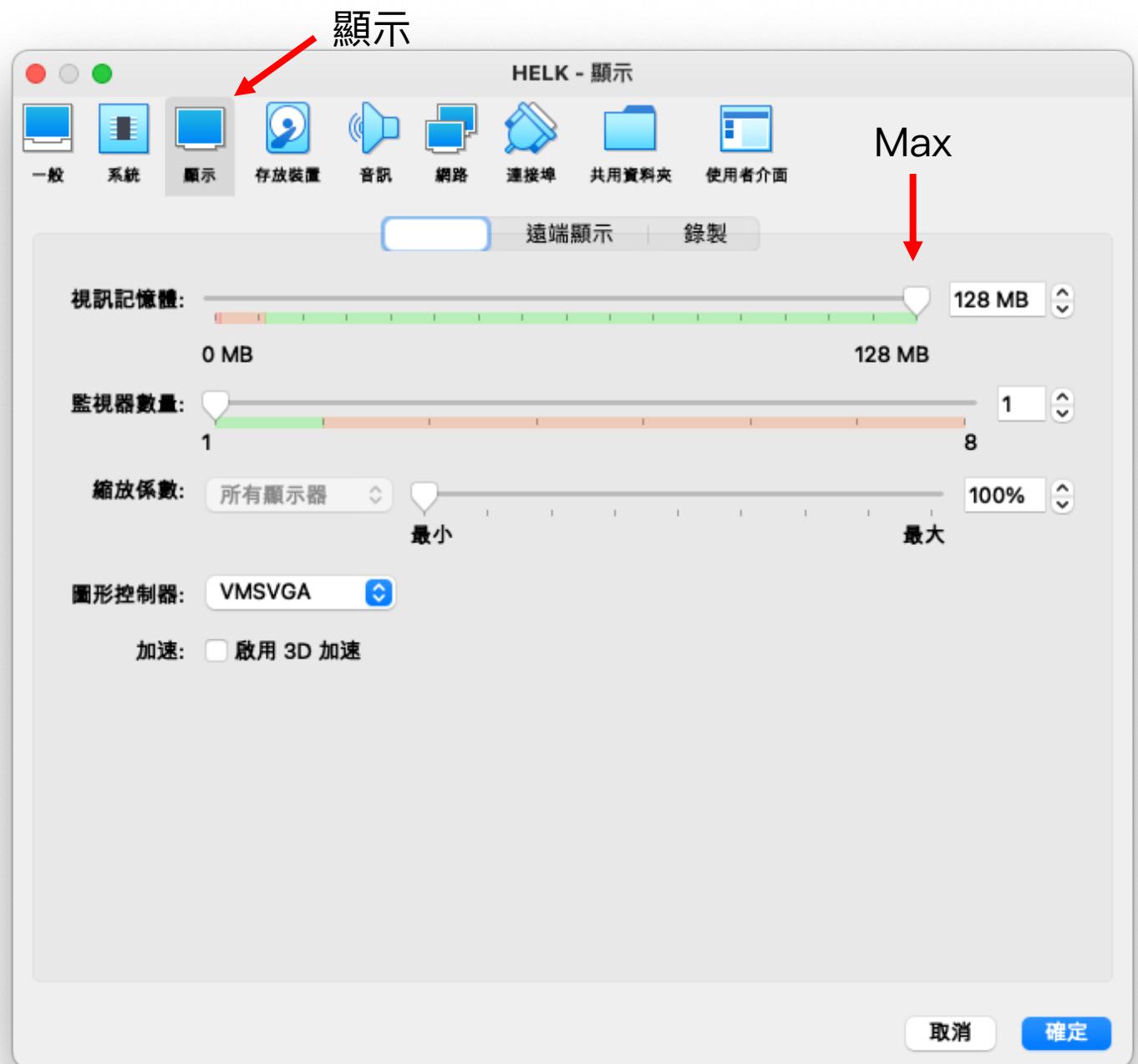
設定硬體

- 選擇「設定」
- 選擇「系統」->「處理器」
->「4 CPU」



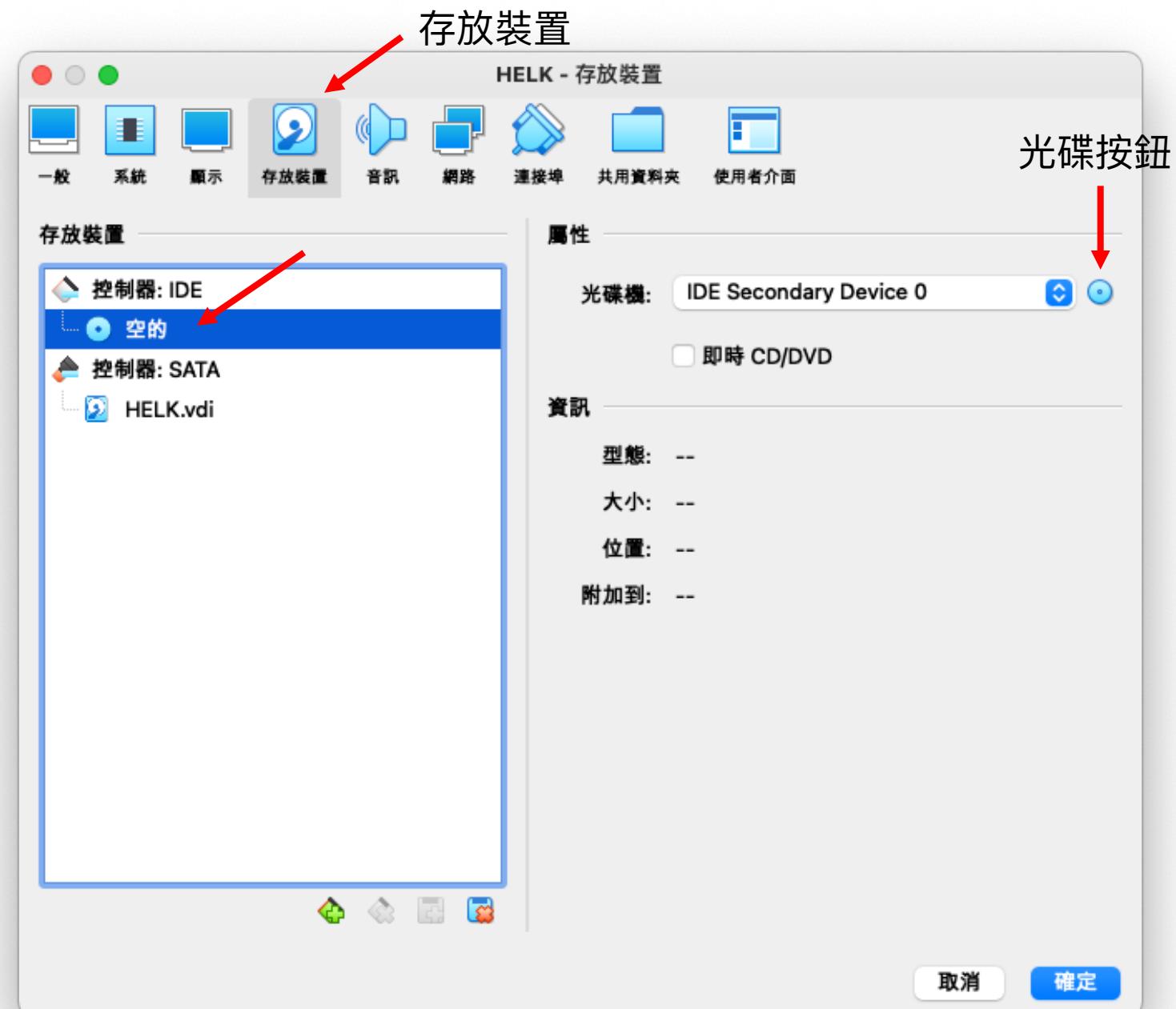
設定硬體

- 選擇「設定」
- 選擇「系統」->「處理器」
->「4 CPU」
- 選擇「顯示」->「視訊記憶體」拉到 Max



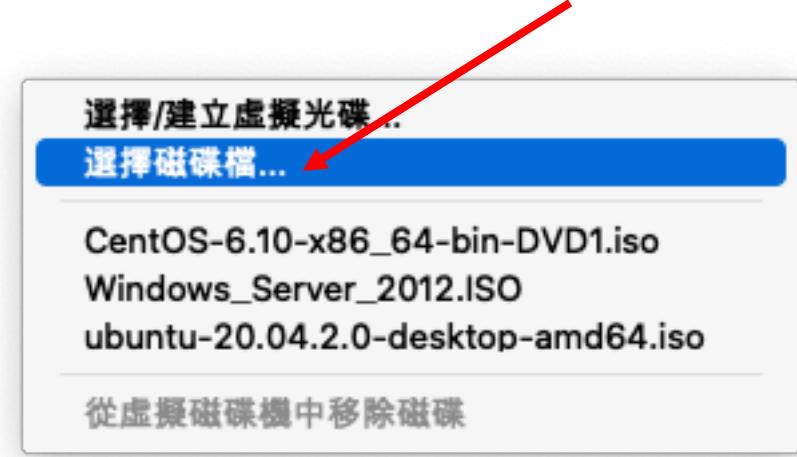
設定硬體：ISO

- 選擇「存放裝置」-> IDE 下的「空的」-> 右方光碟按鈕



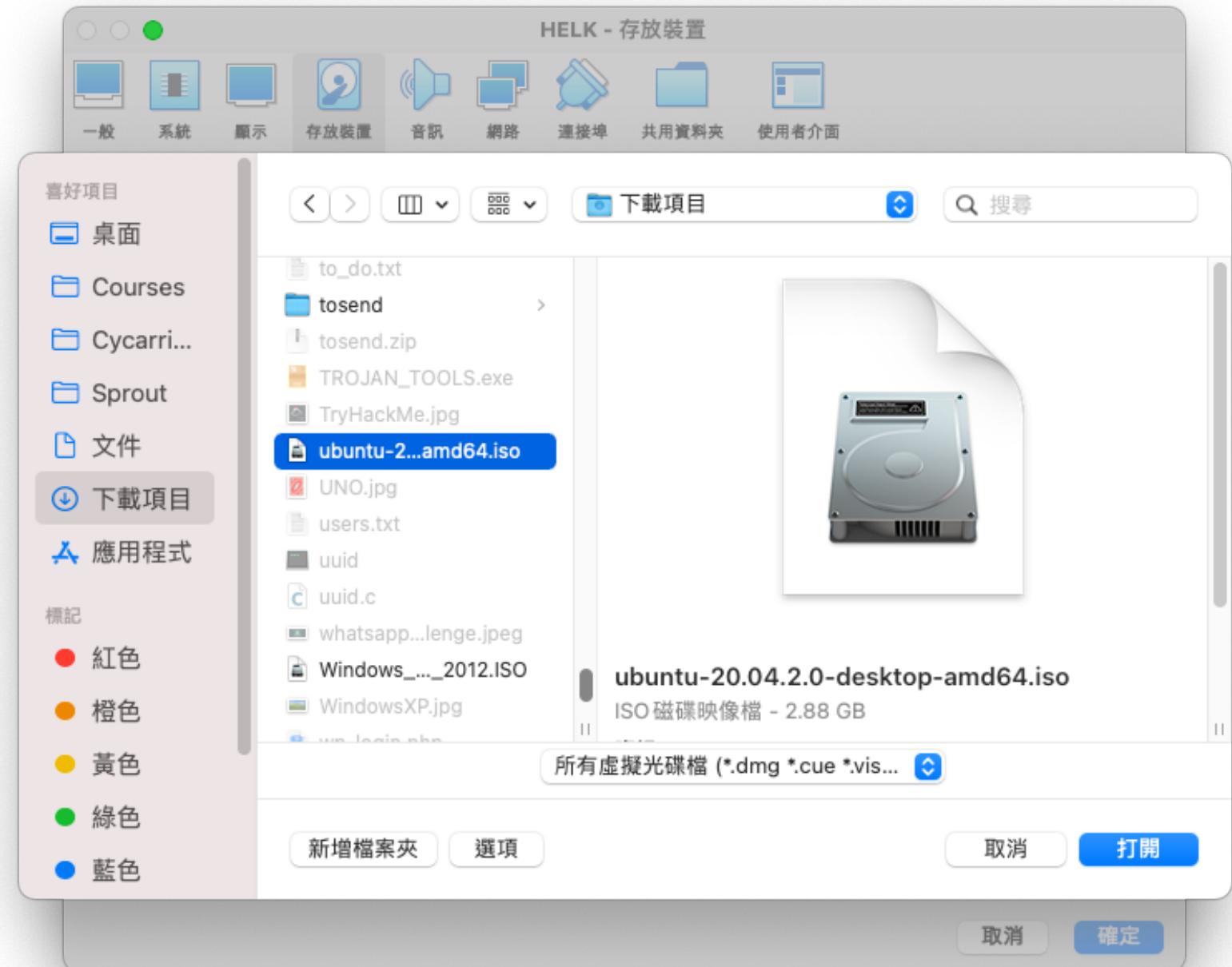
設定硬體：ISO

- 選擇「存放裝置」-> IDE 下的「空的」-> 右方光碟按鈕
- 選擇「選擇磁碟檔...」



設定硬體：ISO

- 選擇「存放裝置」-> IDE 下的「空的」-> 右方光碟按鈕
- 選擇「選擇磁碟檔...」
- 選擇下載好的 Ubuntu .iso 檔



安裝 Ubuntu

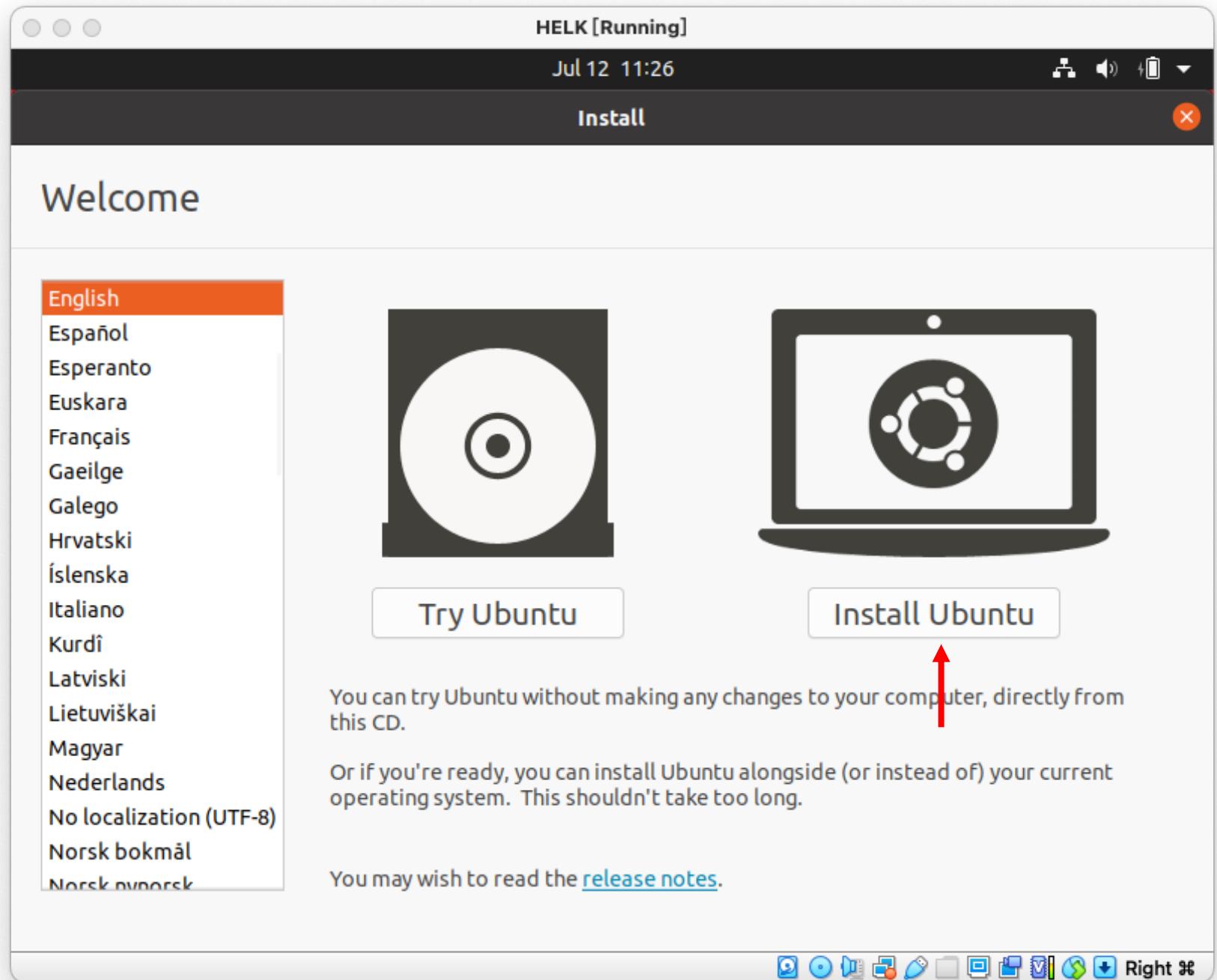
- 點兩下「HELK」開機



安裝 Ubuntu

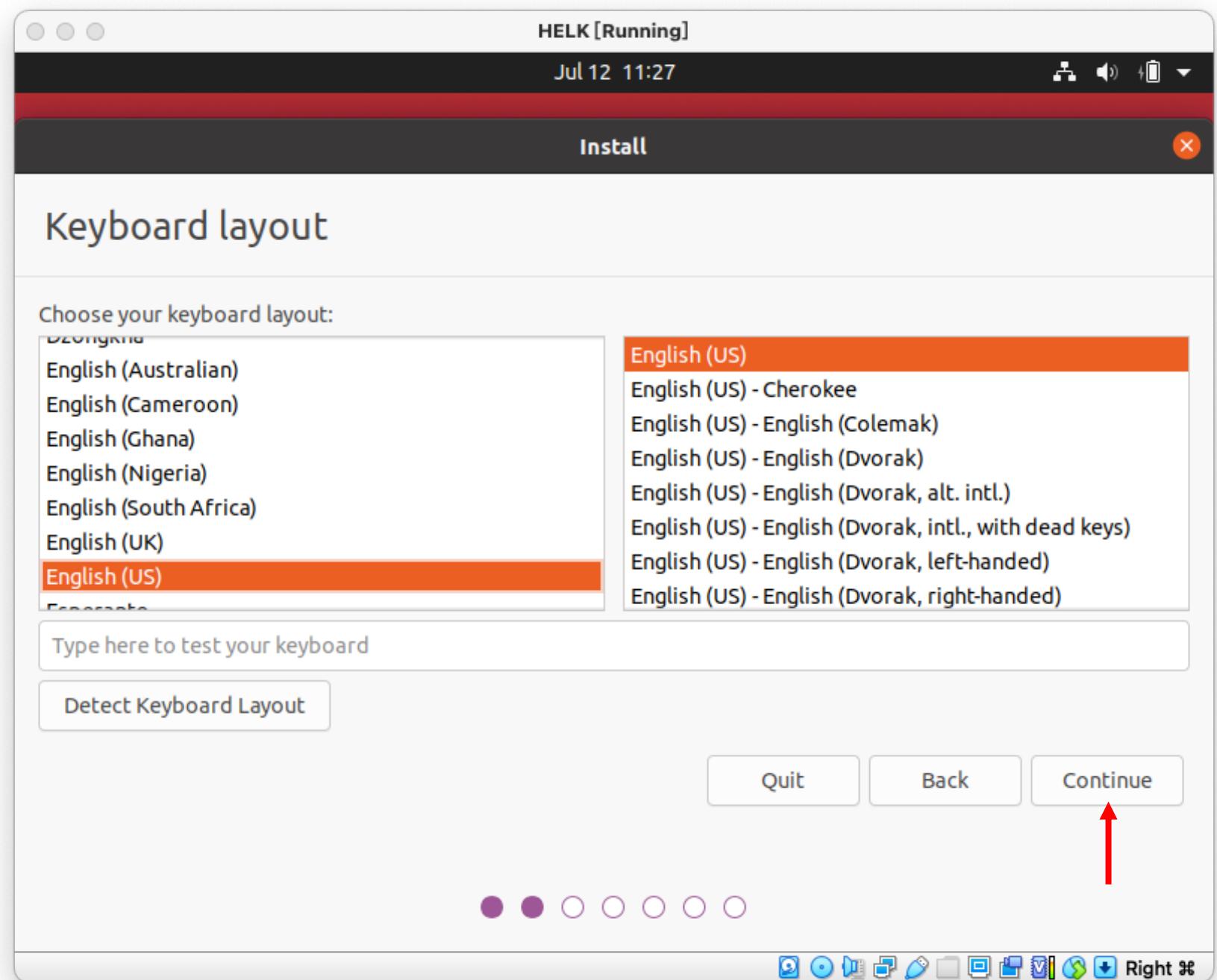
安裝 Ubuntu

- 點兩下「HELK」開機
- 選擇「Install Ubuntu」



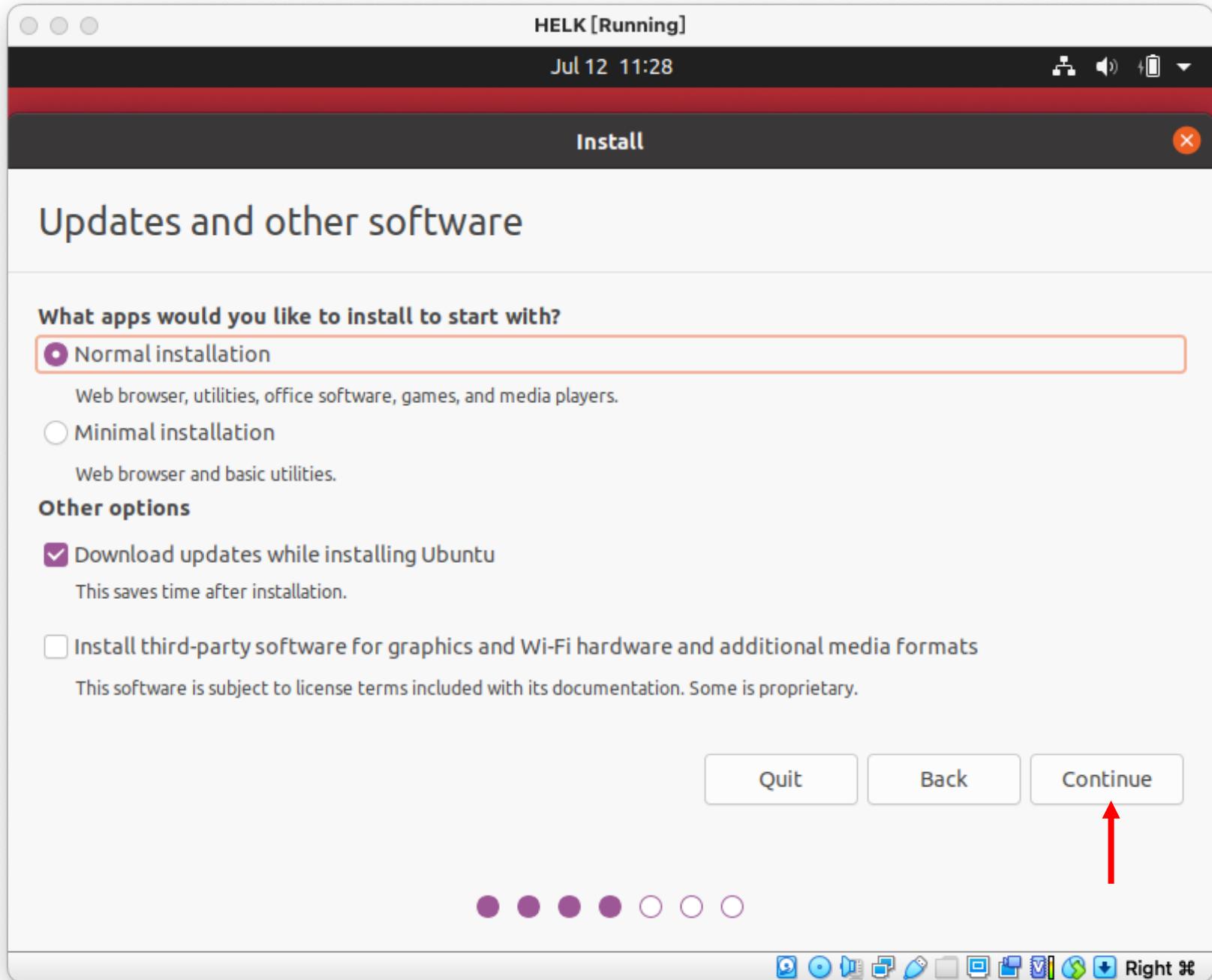
安裝 Ubuntu

- 點兩下「HELK」開機
- 選擇「Install Ubuntu」
- 選擇「Continue」



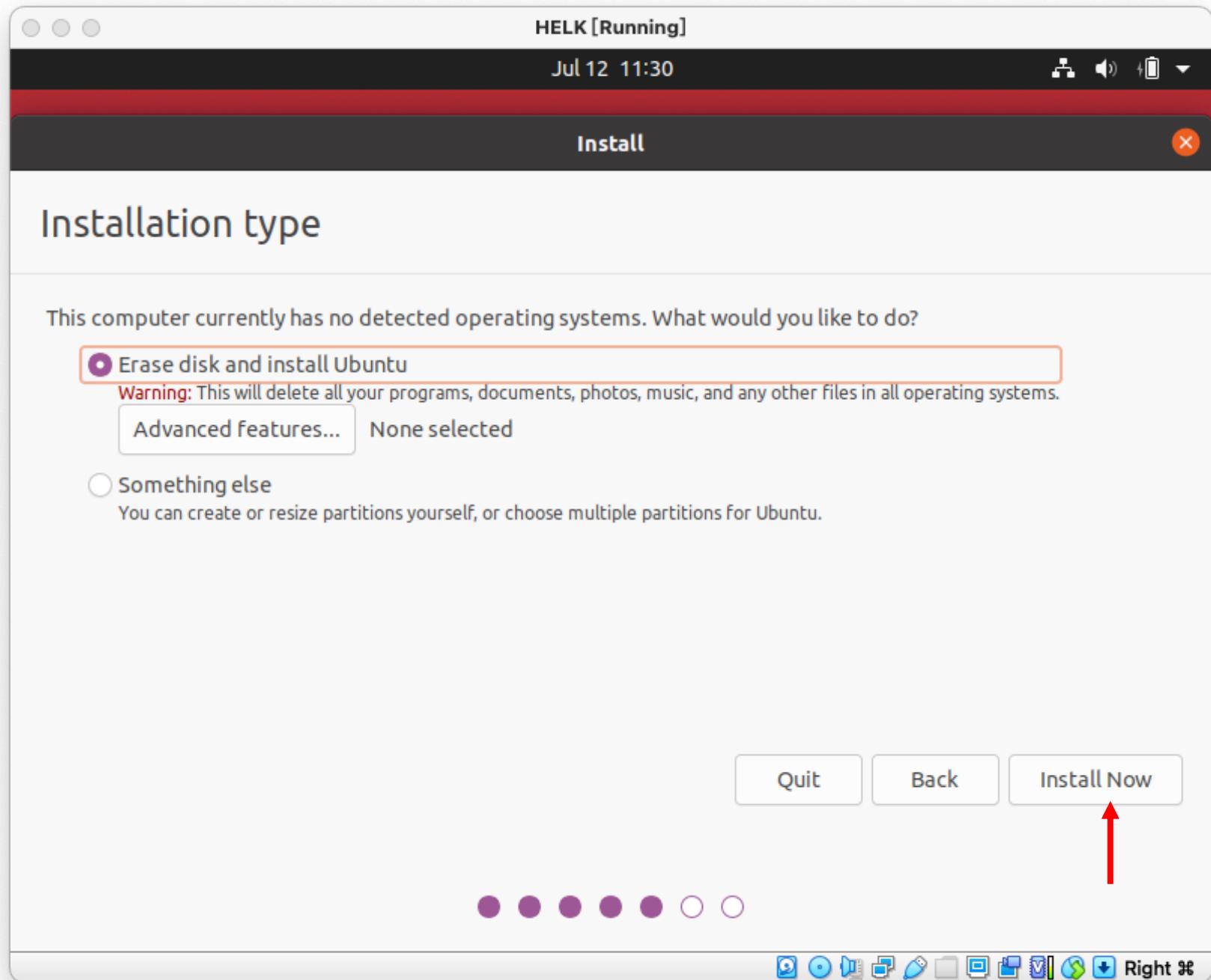
安裝 Ubuntu

- 點兩下「HELK」開機
- 選擇「Install Ubuntu」
- 選擇「Continue」
- 選擇「Continue」



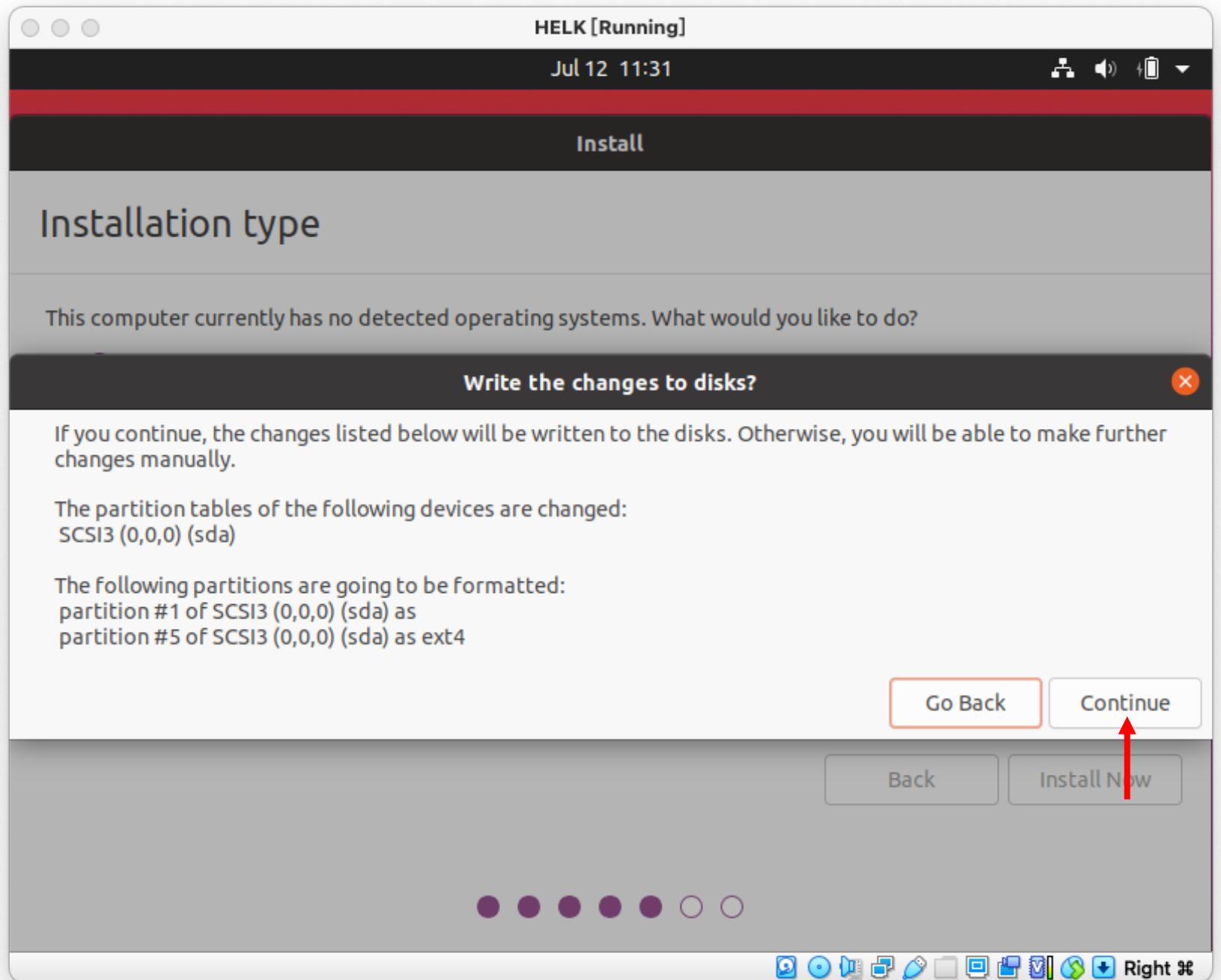
安裝 Ubuntu

- 點兩下「HELK」開機
- 選擇「Install Ubuntu」
- 選擇「Continue」
- 選擇「Continue」
- 選擇「Install Now」



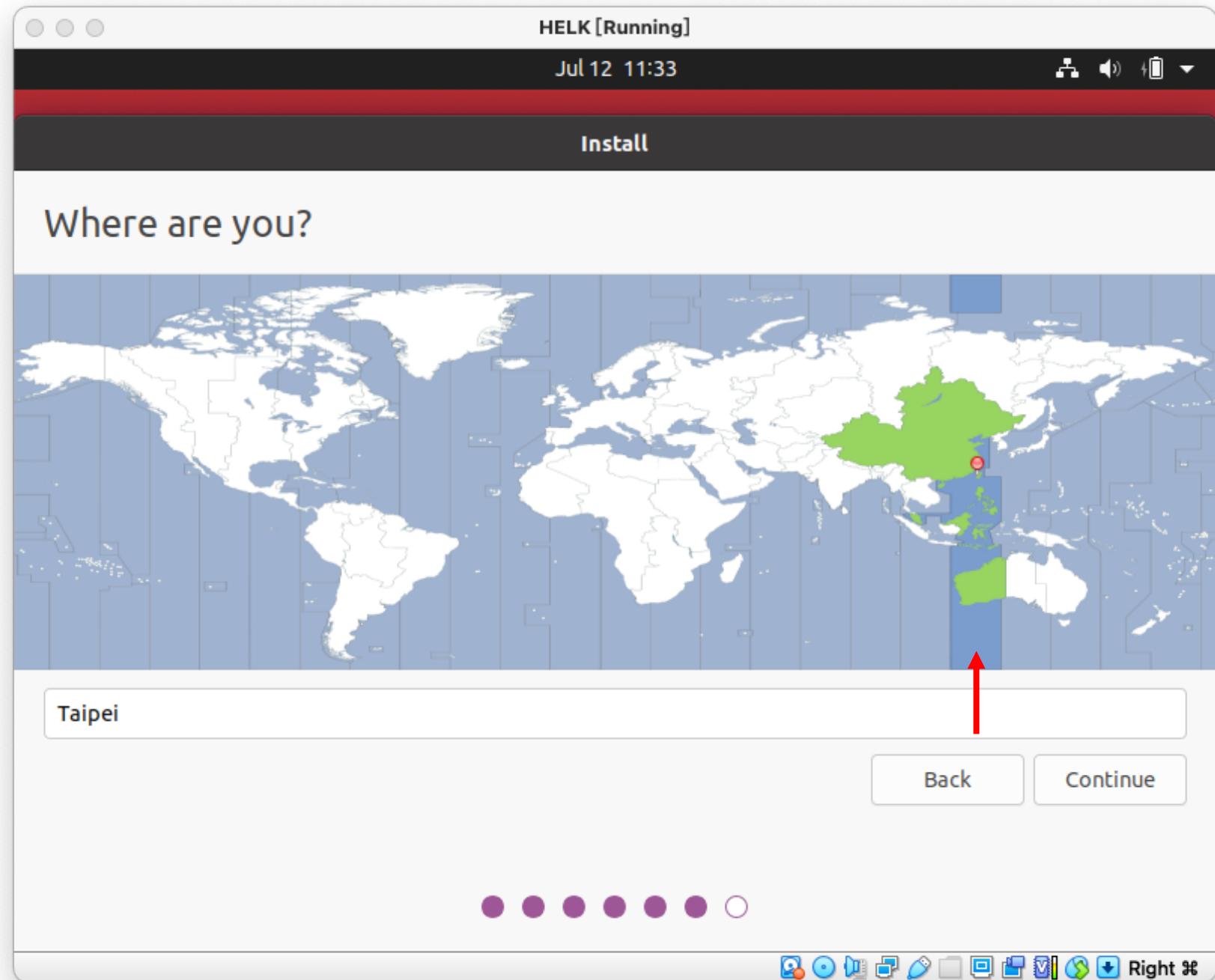
安裝 Ubuntu

- 點兩下「HELK」開機
- 選擇「Install Ubuntu」
- 選擇「Continue」
- 選擇「Continue」
- 選擇「Install Now」
- 選擇「Continue」



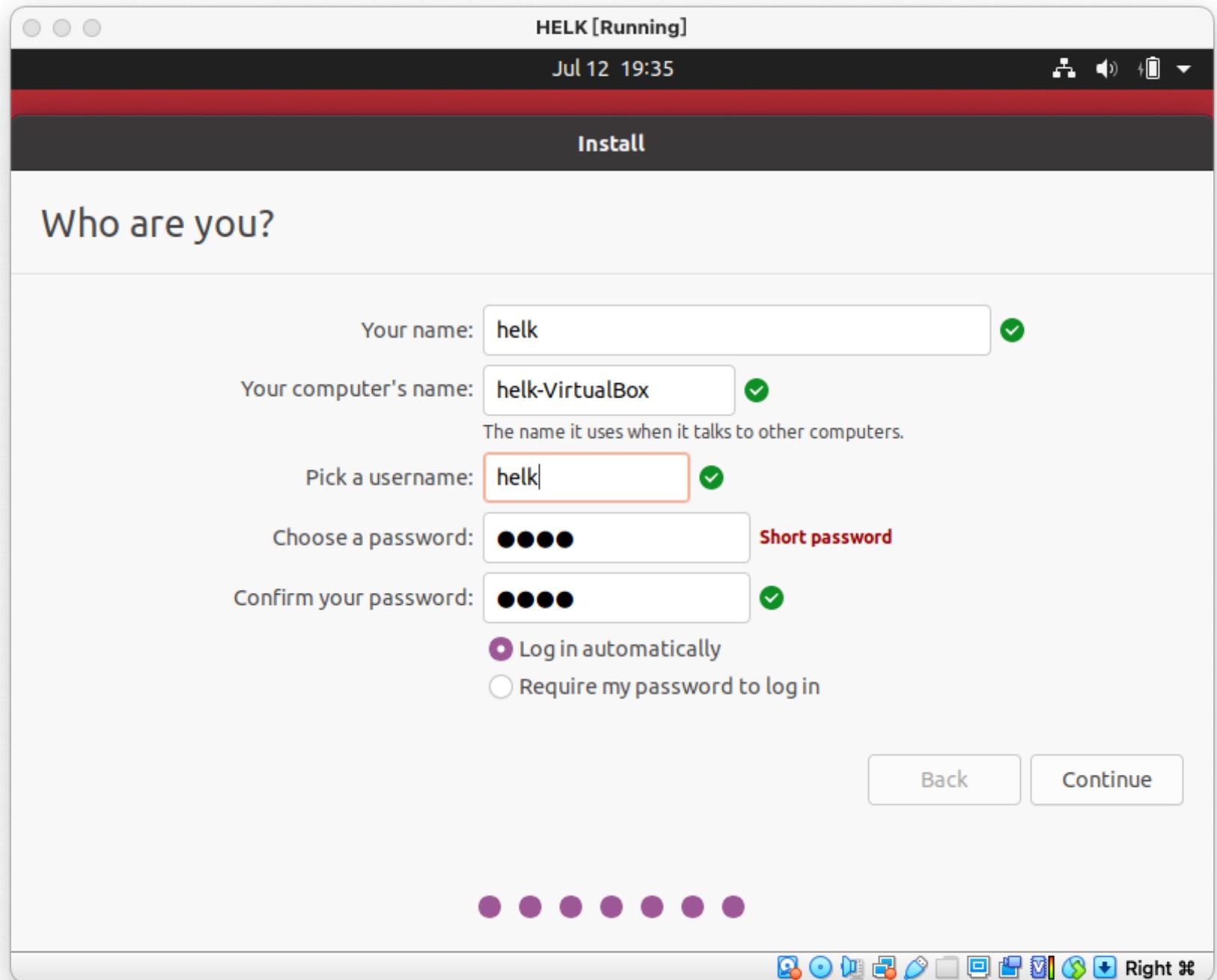
安裝 Ubuntu

- 點兩下「HELK」開機
- 選擇「Install Ubuntu」
- 選擇「Continue」
- 選擇「Continue」
- 選擇「Install Now」
- 選擇「Continue」
- 選擇「Taipei」



安裝 Ubuntu

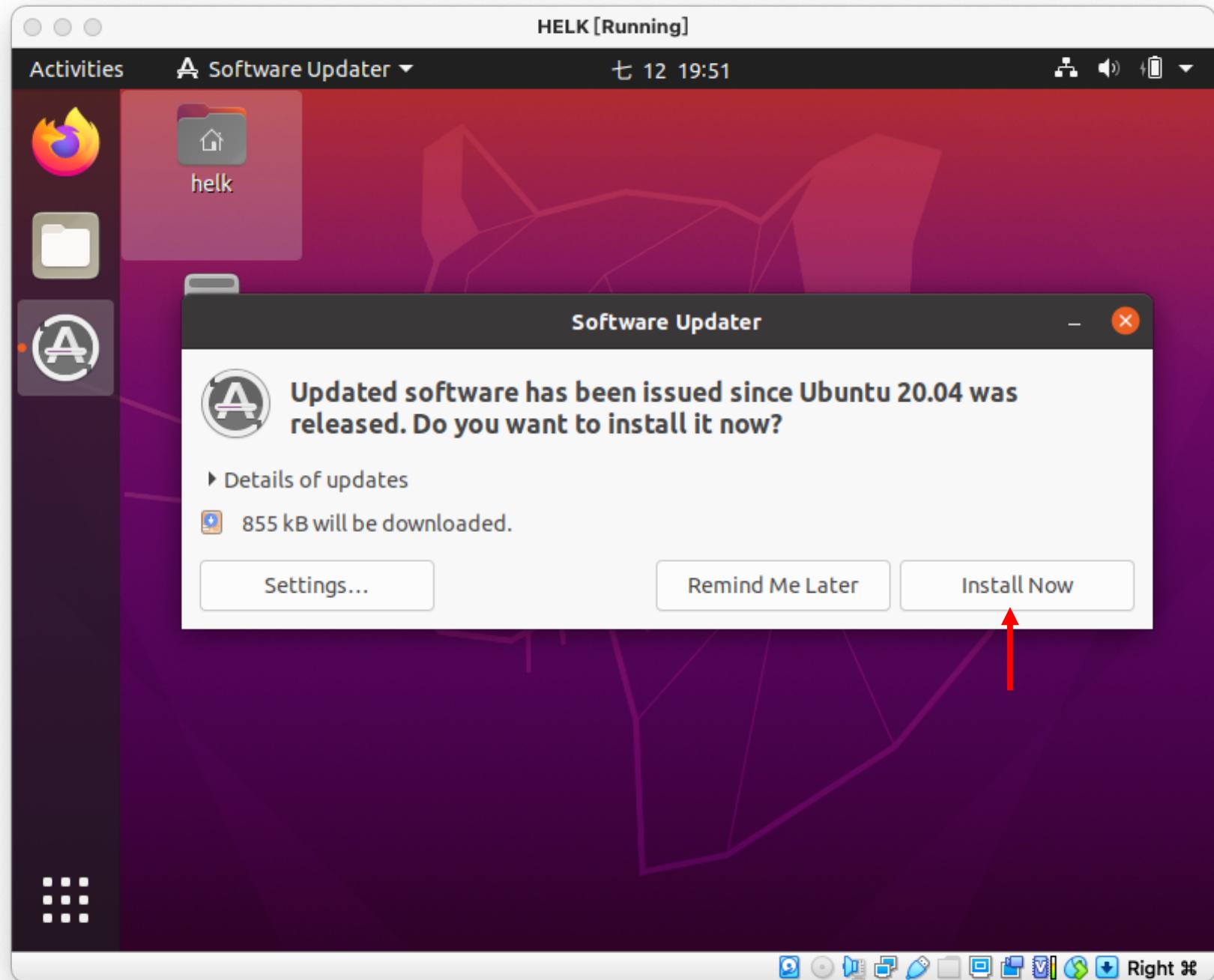
- 點兩下「HELK」開機
- 選擇「Install Ubuntu」
- 選擇「Continue」
- 選擇「Continue」
- 選擇「Install Now」
- 選擇「Continue」
- 選擇「Taipei」
- 設定 user、password



設定 Ubuntu

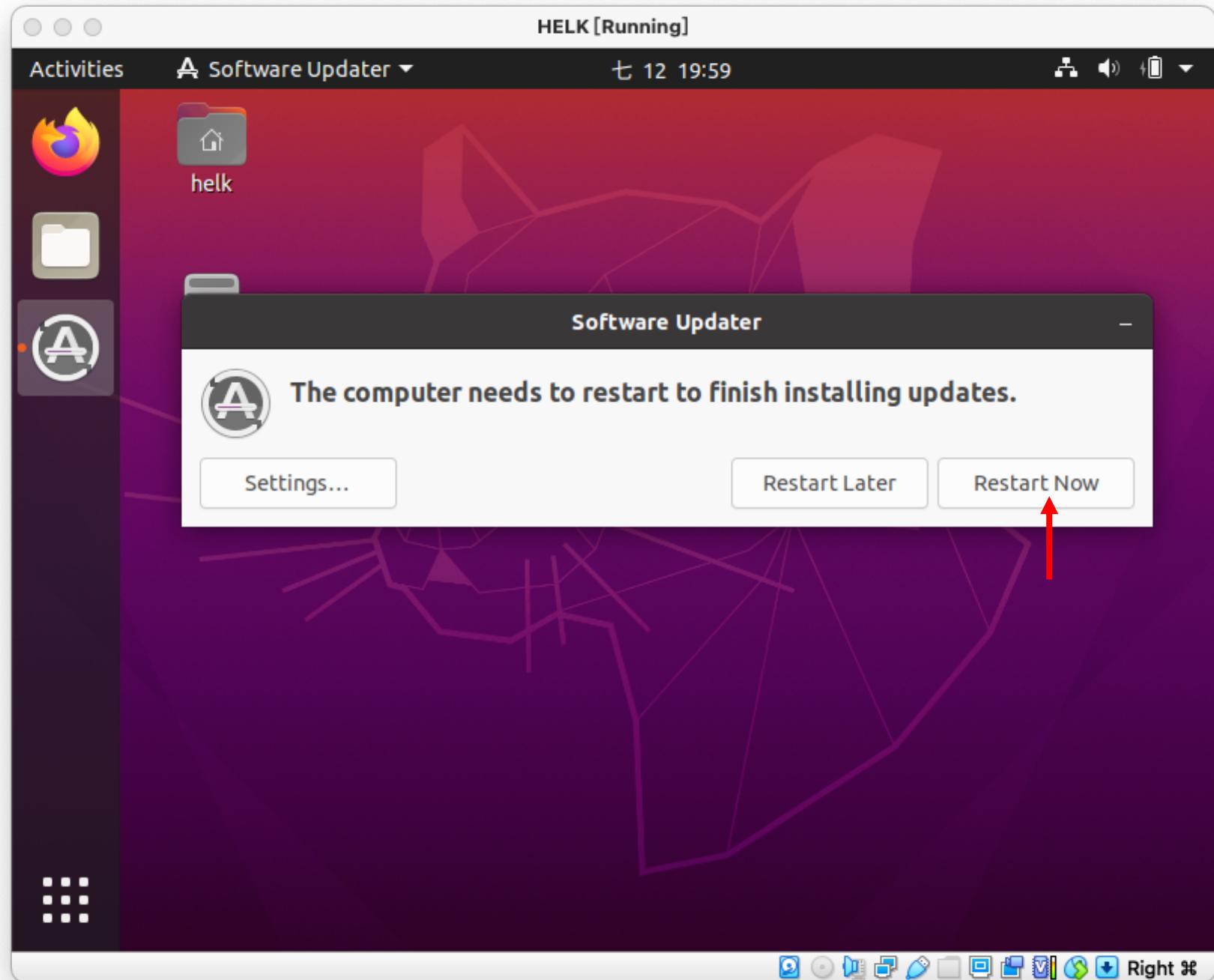
設定 Ubuntu

- 選擇「Install Now」



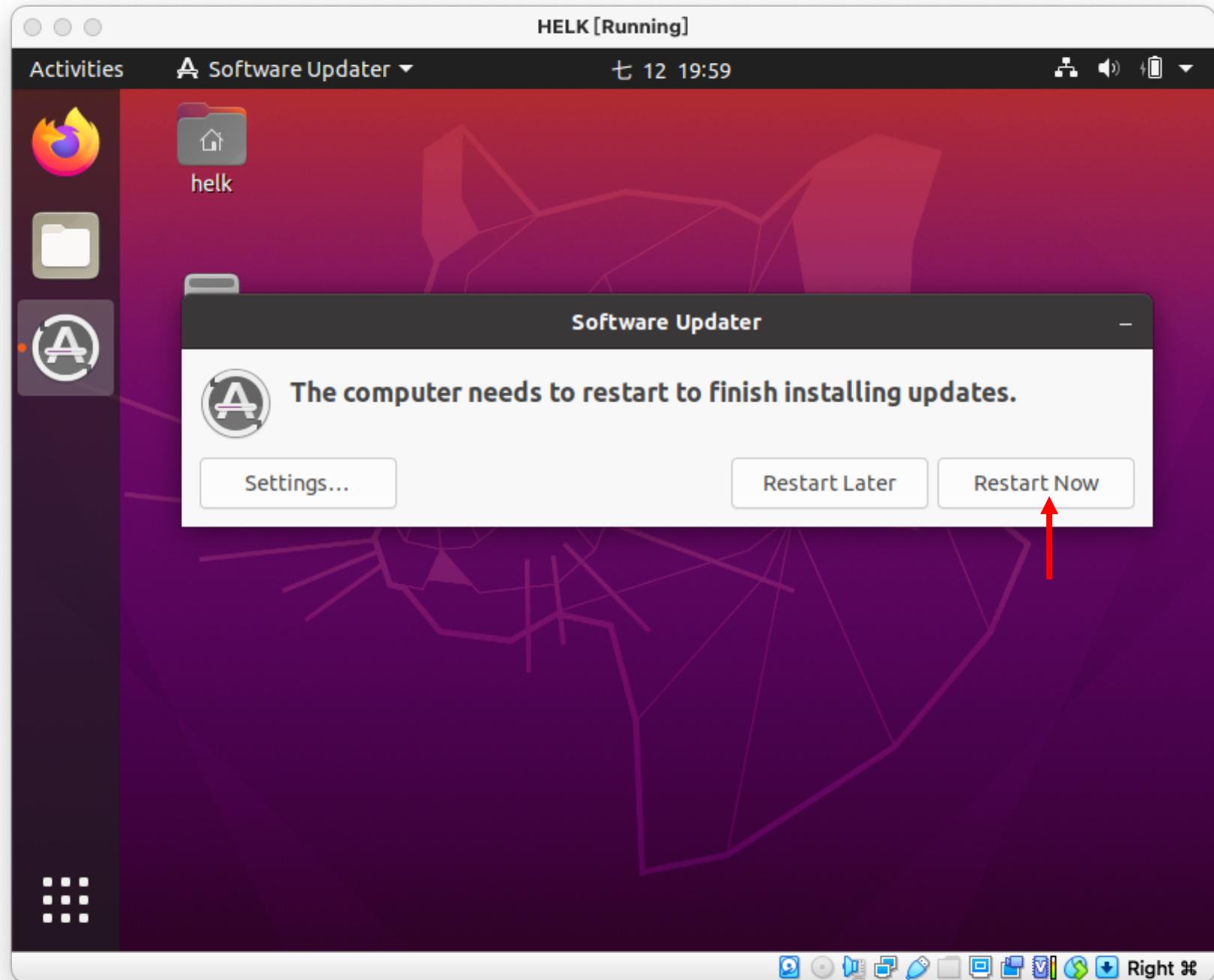
設定 Ubuntu

- 選擇「Install Now」
- 等跑完，選擇「Restart Now」



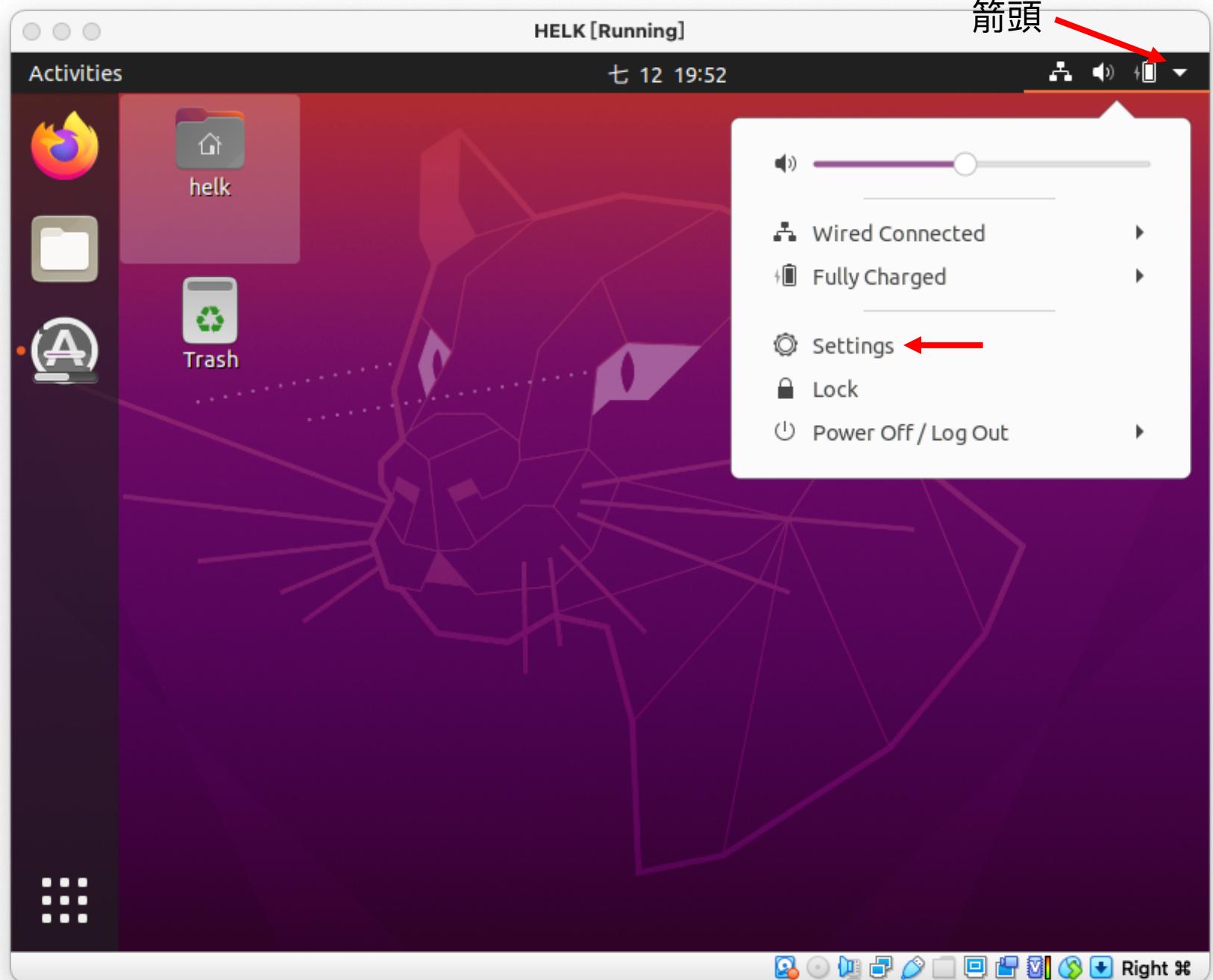
設定 Ubuntu

- 選擇「Install Now」
- 等跑完，選擇「Restart Now」



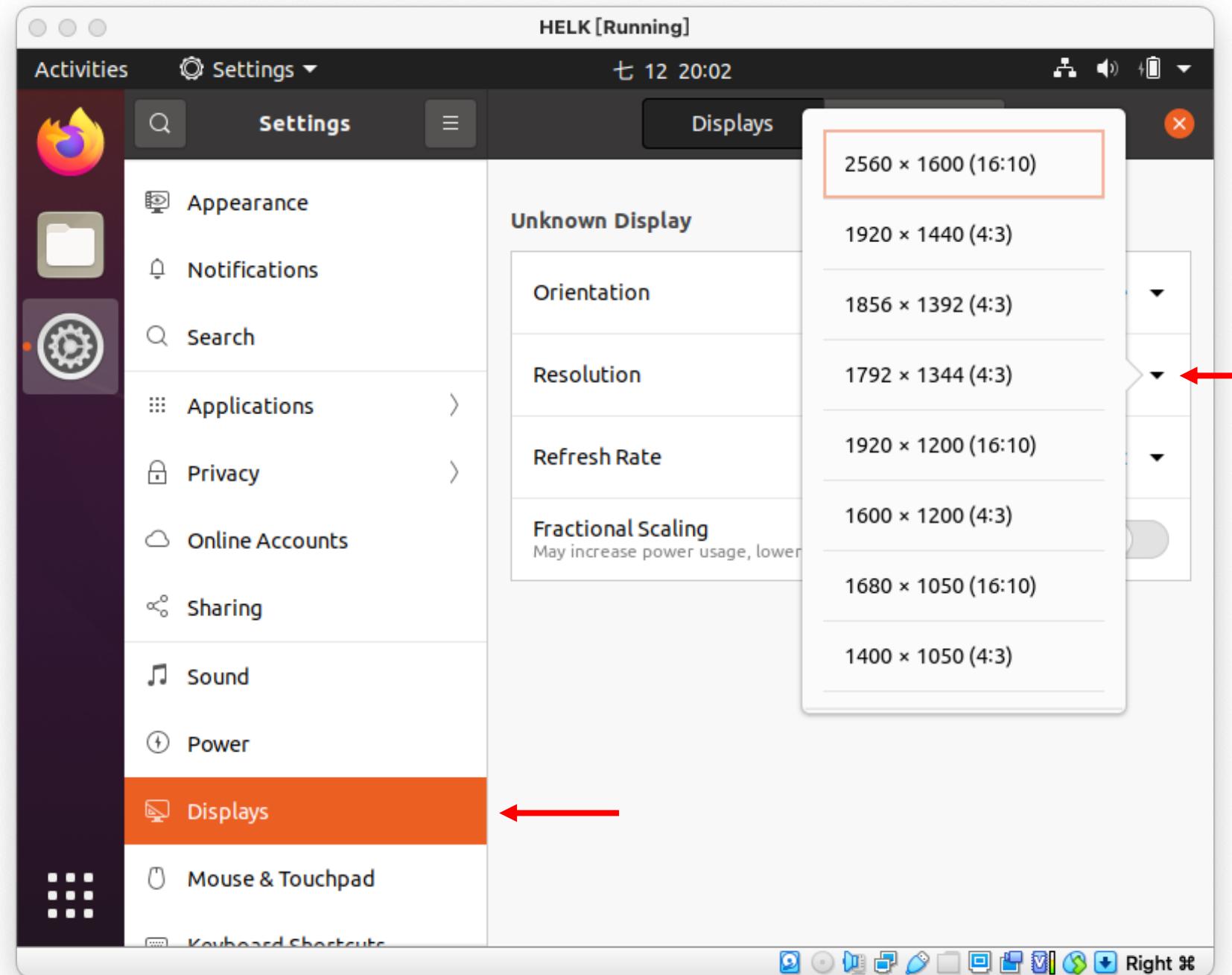
設定 Ubuntu

- 選擇「Install Now」
- 等跑完，選擇「Restart Now」
- 點選「箭頭」 -> 「Settings」



設定 Ubuntu

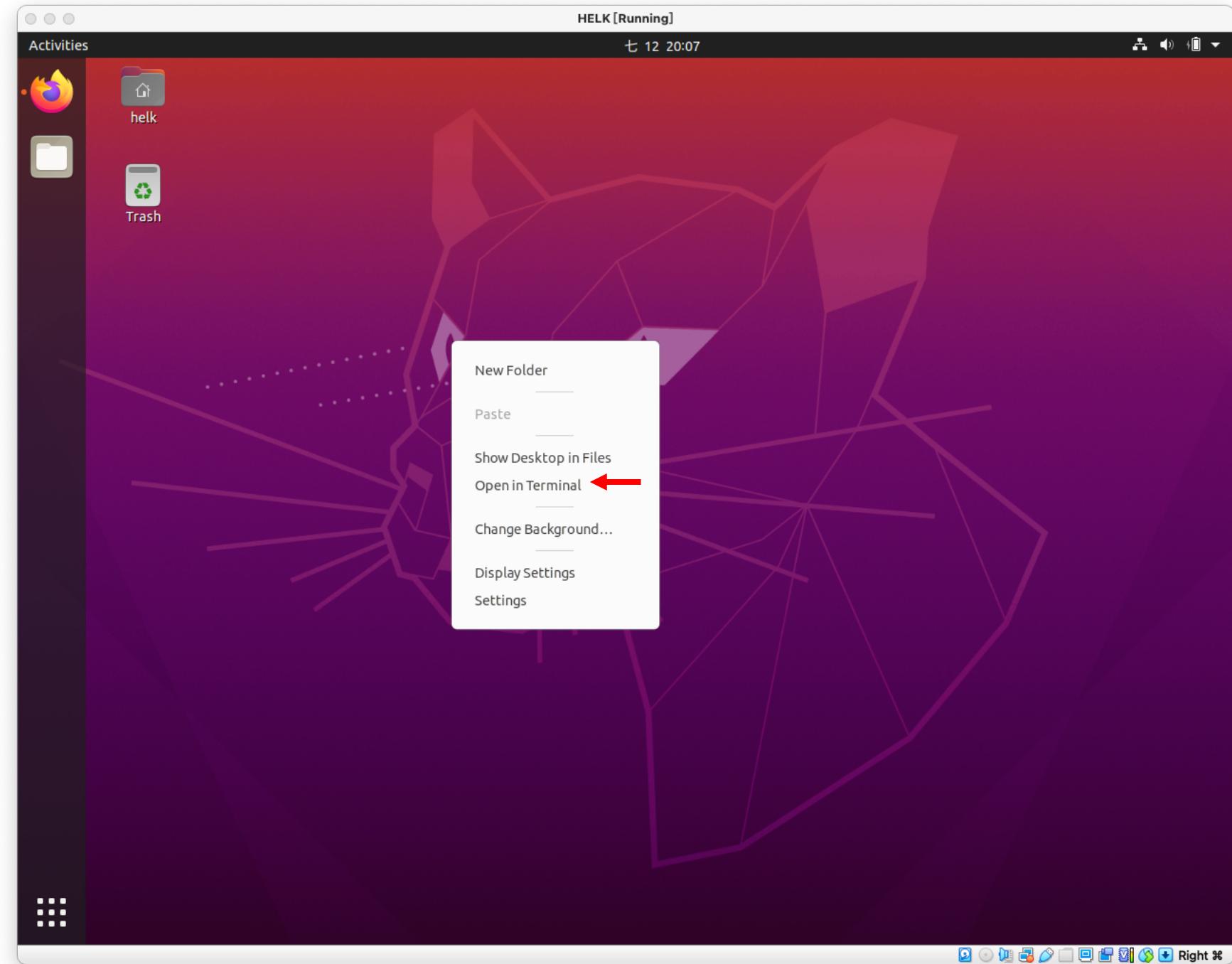
- 選擇「Install Now」
- 等跑完，選擇「Restart Now」
- 點選「箭頭」->「Settings」
- 選擇「Displays」->
「Resolution」，然後調整到
適當的視窗大小



安裝 Lab 環境

安裝 Kafkacat

- 在桌面「右鍵」->「Open in Terminal」



The screenshot shows a Linux desktop environment with a dark theme. A terminal window titled "HELK [Running]" is open, displaying the following text:

```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

helk@helk-VirtualBox:~/Desktop$ sudo apt install kafka-cat git
[sudo] password for helk: █
```

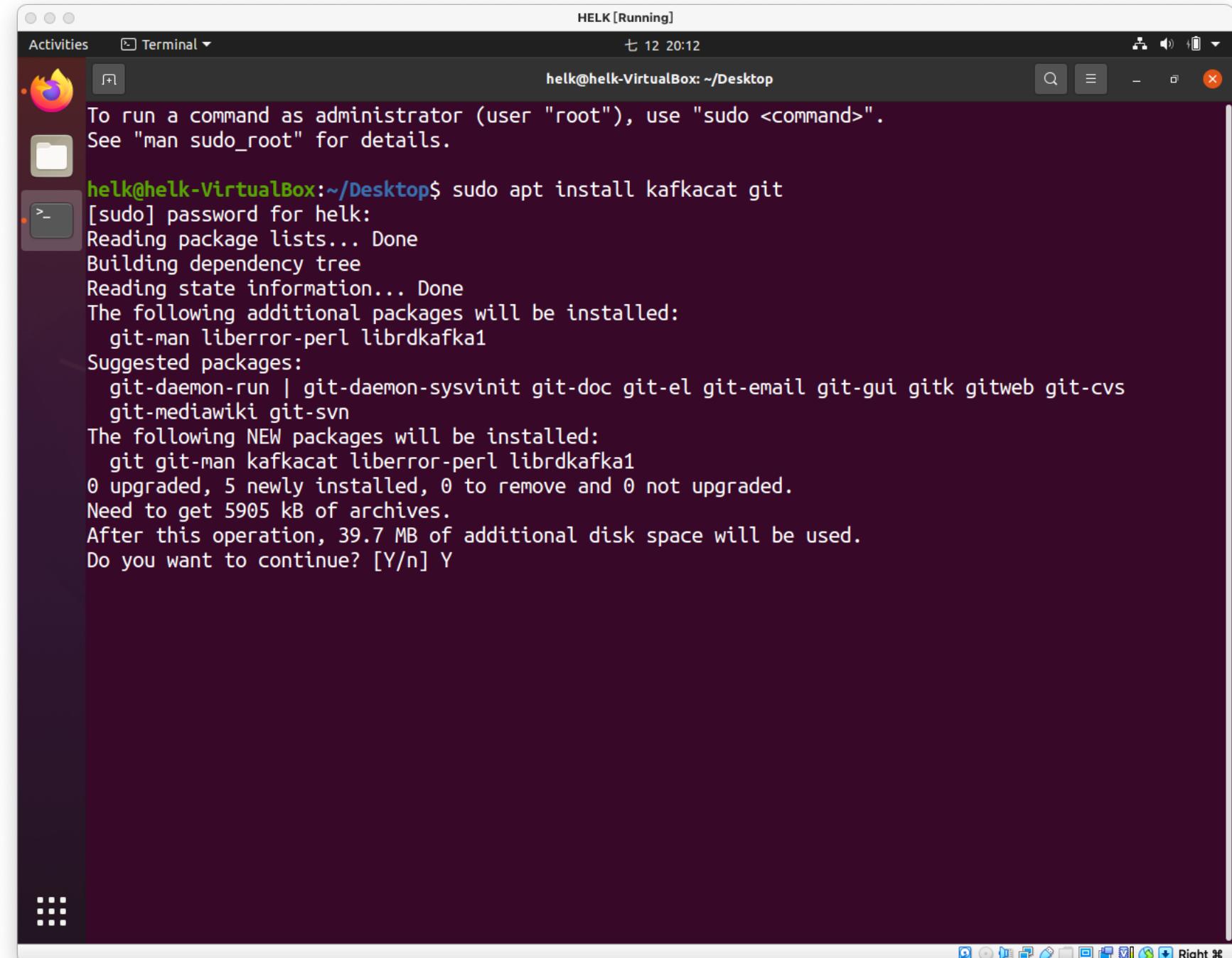
The desktop interface includes a dock at the bottom with various application icons.

安裝 KafkaCat

- 在桌面「右鍵」->「Open in Terminal」
- 輸入「`sudo apt install kafka-cat git`」-> 輸入密碼

安裝 Kafkacat

- 在桌面「右鍵」->「Open in Terminal」
- 輸入「sudo apt install kafkacat git」->輸入密碼
- 輸入「Y」

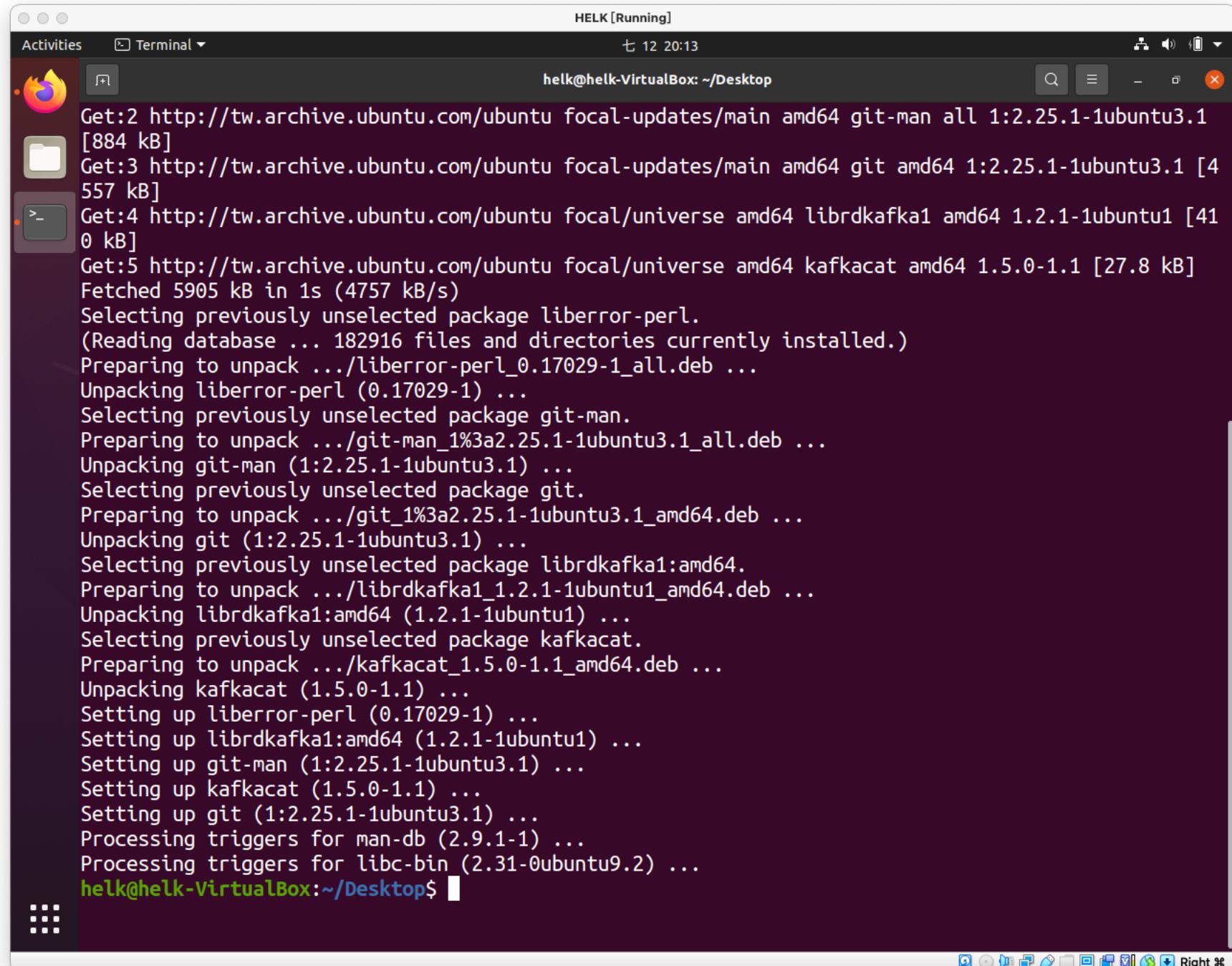


To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

```
helk@helk-VirtualBox:~/Desktop$ sudo apt install kafkacat git
[sudo] password for helk:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl librdkafka1
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs
  git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man kafkacat liberror-perl librdkafka1
0 upgraded, 5 newly installed, 0 to remove and 0 not upgraded.
Need to get 5905 kB of archives.
After this operation, 39.7 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

安裝 Kafkacat

- 在桌面「右鍵」->「Open in Terminal」
- 輸入「sudo apt install kafkacat git」->輸入密碼
- 輸入「Y」
- 若沒有出現錯誤，應該就是沒有問題



The screenshot shows a terminal window titled "HELK [Running]" running on a desktop environment. The terminal output is as follows:

```
Get:2 http://tw.archive.ubuntu.com/ubuntu focal-updates/main amd64 git-man all 1:2.25.1-1ubuntu3.1 [884 kB]
Get:3 http://tw.archive.ubuntu.com/ubuntu focal-updates/main amd64 git amd64 1:2.25.1-1ubuntu3.1 [4557 kB]
Get:4 http://tw.archive.ubuntu.com/ubuntu focal/universe amd64 librdkafka1 amd64 1.2.1-1ubuntu1 [410 kB]
Get:5 http://tw.archive.ubuntu.com/ubuntu focal/universe amd64 kafkacat amd64 1.5.0-1.1 [27.8 kB]
Fetched 5905 kB in 1s (4757 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 182916 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17029-1_all.deb ...
Unpacking liberror-perl (0.17029-1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.25.1-1ubuntu3.1_all.deb ...
Unpacking git-man (1:2.25.1-1ubuntu3.1) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.25.1-1ubuntu3.1_amd64.deb ...
Unpacking git (1:2.25.1-1ubuntu3.1) ...
Selecting previously unselected package librdkafka1:amd64.
Preparing to unpack .../librdkafka1_1.2.1-1ubuntu1_amd64.deb ...
Unpacking librdkafka1:amd64 (1.2.1-1ubuntu1) ...
Selecting previously unselected package kafkacat.
Preparing to unpack .../kafkacat_1.5.0-1.1_amd64.deb ...
Unpacking kafkacat (1.5.0-1.1) ...
Setting up liberror-perl (0.17029-1) ...
Setting up librdkafka1:amd64 (1.2.1-1ubuntu1) ...
Setting up git-man (1:2.25.1-1ubuntu3.1) ...
Setting up kafkacat (1.5.0-1.1) ...
Setting up git (1:2.25.1-1ubuntu3.1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
helk@helk-VirtualBox:~/Desktop$
```

安裝 HELK

- 打開連結 -> 複製 HELK 網址

HELK [Running]
七 12 20:16

detection-hackathon-apt29 GitHub - Cyb3rWard0g/HELK Installation — The HELK

master detection-hackathon-apt29 / SIEMs / HELK / Go to file

Cyb3rWard0g Updated docs cda2d9f on May 4, 2020 History

.. README.md Updated docs 15 months ago

README.md

HELK and APT29 Day 1 & Day2

Pre-requirements

- Install Kafkacat 1.5+
 - Reference: <https://github.com/edenhill/kafkacat#install>
- git clone <https://github.com/Cyb3rWard0g/HELK> ←
- git clone <https://github.com/OTRF/detection-hackathon-apt29> (same VM or local host as HELK)

Send Data to HELK

- cd detection-hackathon-apt29/datasets/day1
- decompress files

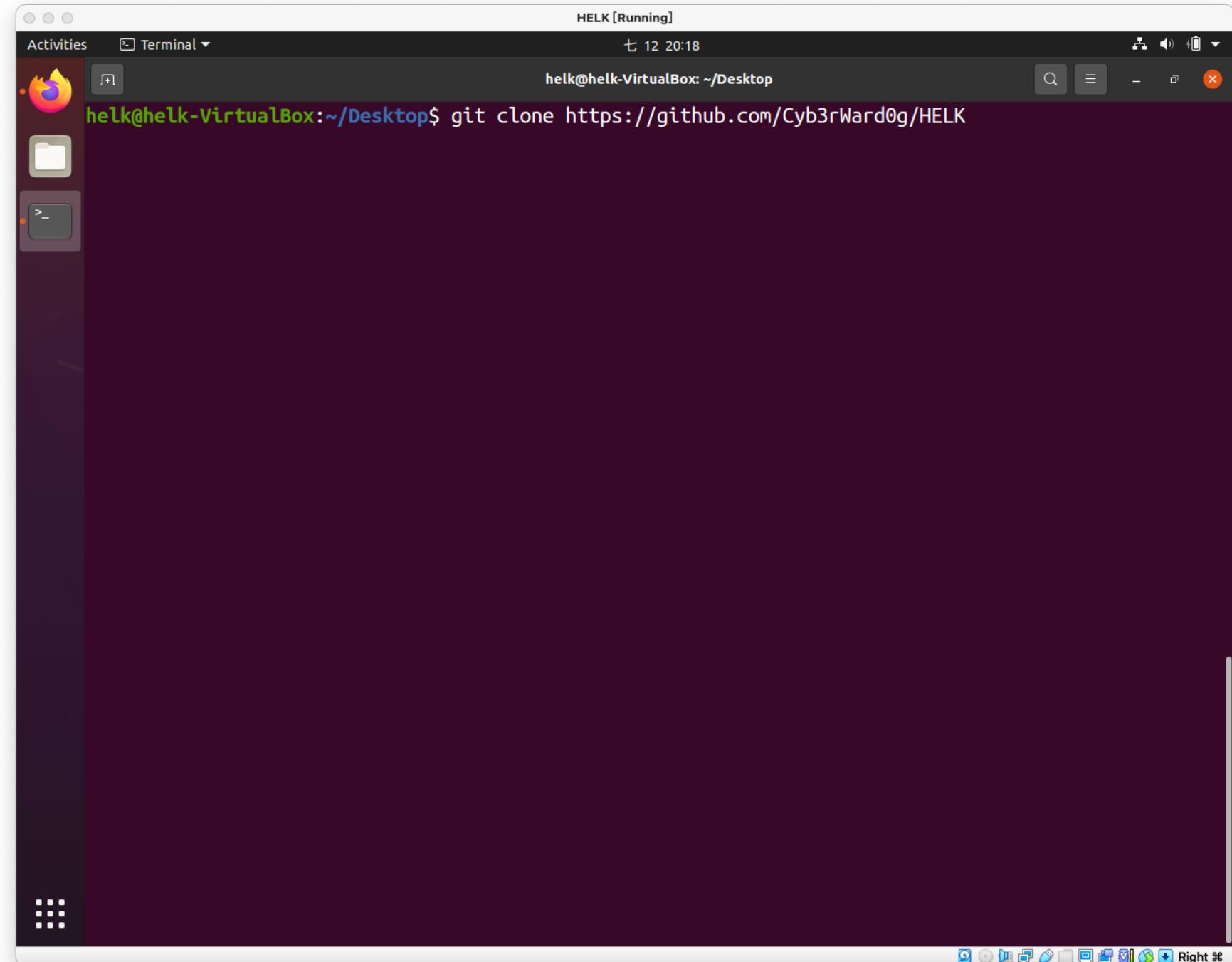
```
unzip apt29_evals_day1_manual.zip
```

- Use Kafkacat to send the data over to the kafka broker deployed by HELK

```
kafkacat -b 127.0.0.1:9092 -t winlogbeat -P -l apt29_evals_day1_manual_2020-05-01225525.json
```

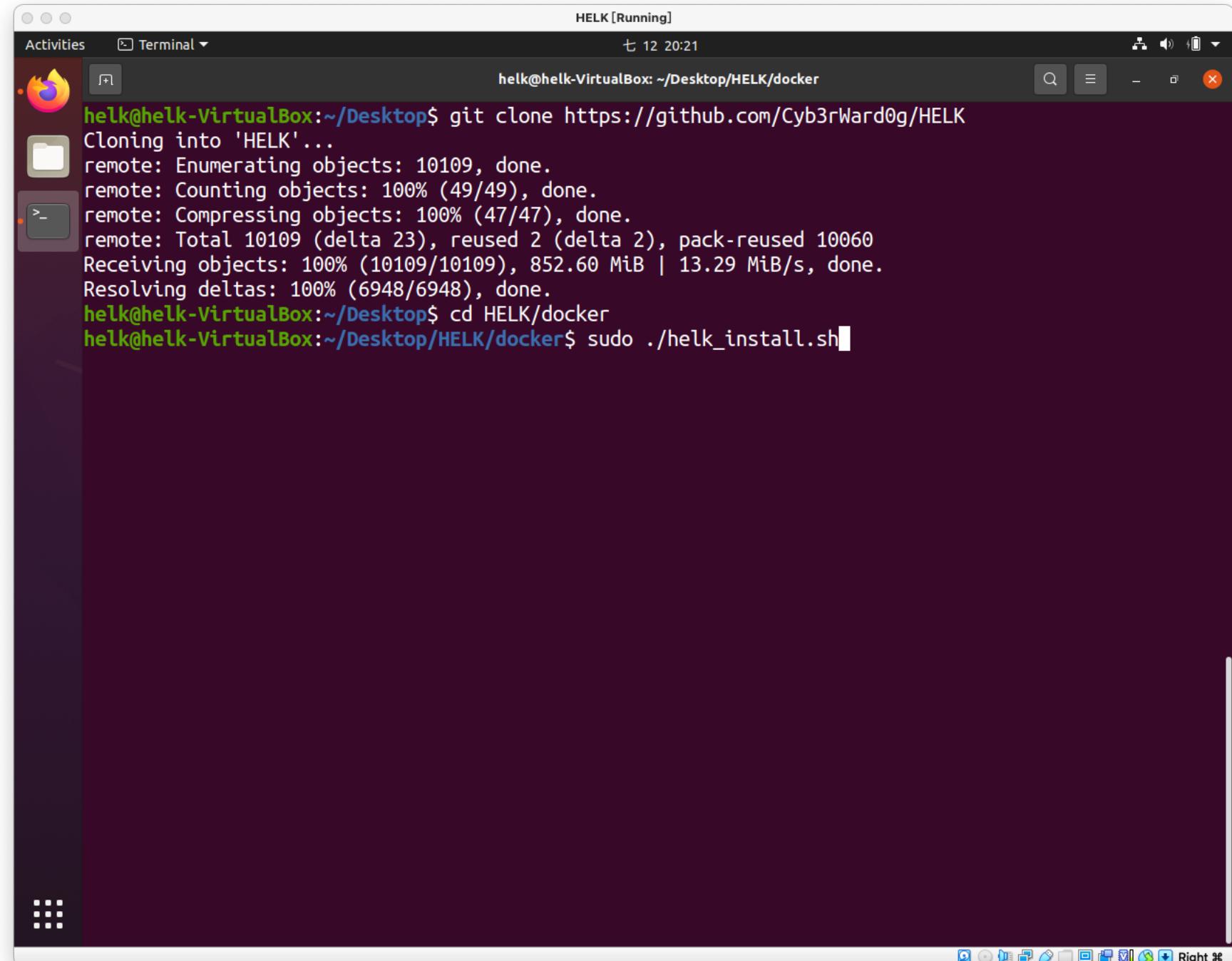
安裝 HELK

- 打開連結 -> 複製 HELK 網址
- 輸入「git clone <網址>」



安裝 HELK

- 打開連結 -> 複製 HELK 網址
- 輸入「git clone <網址>」
- 輸入「cd HELK/docker」 ->
「sudo ./helk_install.sh」



The screenshot shows a terminal window titled "HELK [Running]" running on an Ubuntu desktop. The terminal output is as follows:

```
helk@helk-VirtualBox:~/Desktop$ git clone https://github.com/Cyb3rWard0g/HELK
Cloning into 'HELK'...
remote: Enumerating objects: 10109, done.
remote: Counting objects: 100% (49/49), done.
remote: Compressing objects: 100% (47/47), done.
remote: Total 10109 (delta 23), reused 2 (delta 2), pack-reused 10060
Receiving objects: 100% (10109/10109), 852.60 MiB | 13.29 MiB/s, done.
Resolving deltas: 100% (6948/6948), done.
helk@helk-VirtualBox:~/Desktop$ cd HELK/docker
helk@helk-VirtualBox:~/Desktop/HELK/docker$ sudo ./helk_install.sh
```

安裝 HELK

- 打開連結 -> 複製 HELK 網址
- 輸入「git clone <網址>」
- 輸入「cd HELK/docker」 ->
「sudo ./helk_install.sh」
- Builde choice 輸入「1」

HELK [Running]
七 12 20:22
helk@helk-VirtualBox: ~/Desktop/HELK/docker

```
helk@helk-VirtualBox:~/Desktop$ git clone https://github.com/Cyb3rWard0g/HELK
Cloning into 'HELK'...
remote: Enumerating objects: 10109, done.
remote: Counting objects: 100% (49/49), done.
remote: Compressing objects: 100% (47/47), done.
remote: Total 10109 (delta 23), reused 2 (delta 2), pack-reused 10060
Receiving objects: 100% (10109/10109), 852.60 MiB | 13.29 MiB/s, done.
Resolving deltas: 100% (6948/6948), done.
helk@helk-VirtualBox:~/Desktop$ cd HELK/docker
helk@helk-VirtualBox:~/Desktop/HELK/docker$ sudo ./helk_install.sh

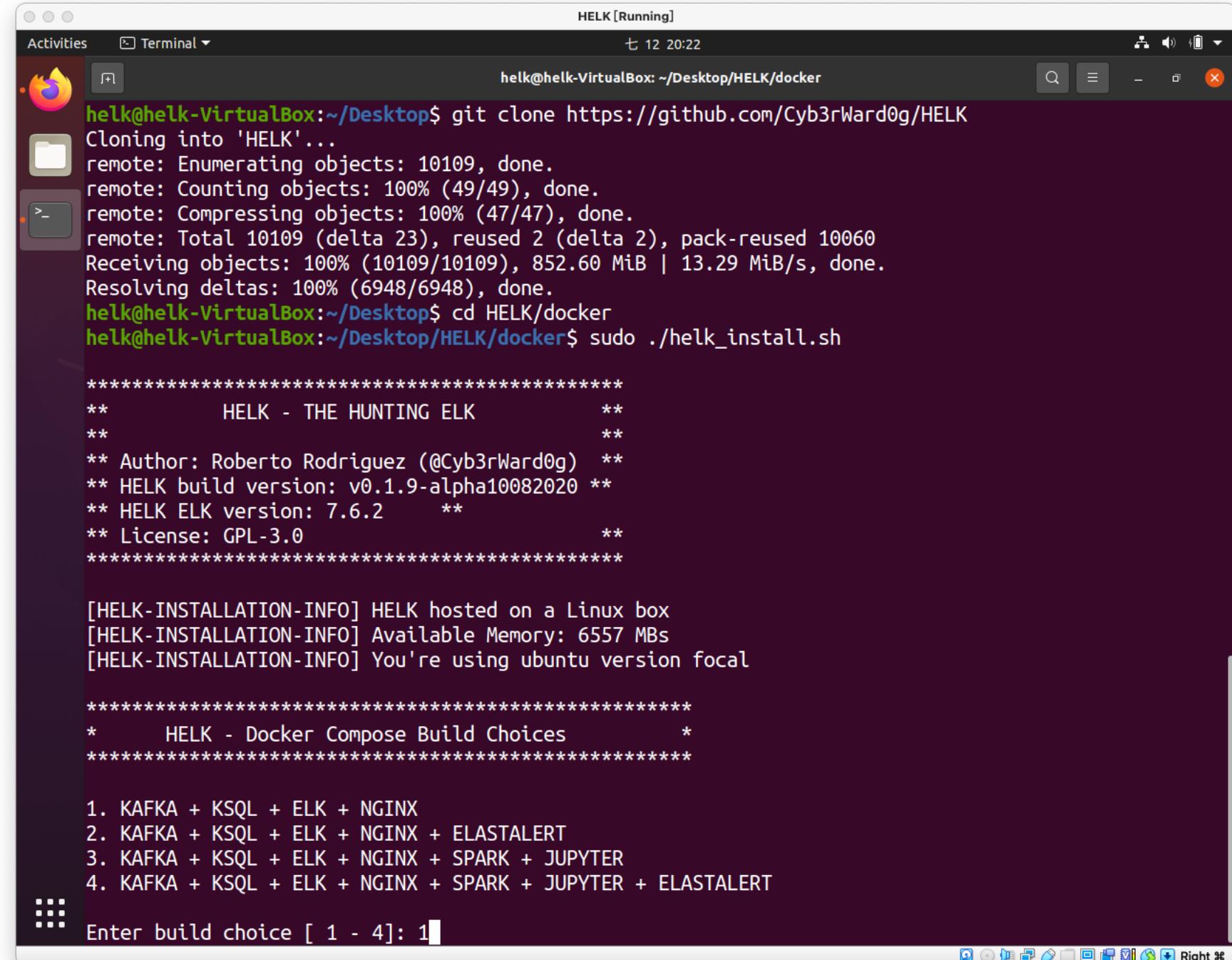
*****
**          HELK - THE HUNTING ELK          **
**          **                                 **
** Author: Roberto Rodriguez (@Cyb3rWard0g)  **
** HELK build version: v0.1.9-alpha10082020  **
** HELK ELK version: 7.6.2                   **
** License: GPL-3.0                           **
*****
```

[HELK-INSTALLATION-INFO] HELK hosted on a Linux box
[HELK-INSTALLATION-INFO] Available Memory: 6557 MBs
[HELK-INSTALLATION-INFO] You're using ubuntu version focal

```
*****
*          HELK - Docker Compose Build Choices      *
*****
```

1. KAFKA + KSQL + ELK + NGINX
2. KAFKA + KSQL + ELK + NGINX + ELASTALERT
3. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER
4. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER + ELASTALERT

Enter build choice [1 - 4]: 1



安裝 HELK

- 打開連結 -> 複製 HELK 網址
- 輸入「git clone <網址>」
- 輸入「cd HELK/docker」 ->
「sudo ./helk_install.sh」
- Build choice 輸入「1」
- HELK IP 直接「Enter」

The screenshot shows a terminal window titled "HELK [Running]" running on a Linux desktop environment. The terminal output is as follows:

```
Enter build choice [ 1 - 4]: 1
[HELK-INSTALLATION-INFO] HELK build set to helk-kibana-analysis
[HELK-INSTALLATION-INFO] Set HELK IP. Default value is your current IP: 10.0.2.15
[HELK-INSTALLATION-INFO] HELK IP set to 10.0.2.15
[HELK-INSTALLATION-INFO] Please make sure to create a custom Kibana password and store it securely for future use.
[HELK-INSTALLATION-INFO] Set HELK Kibana UI Password: hunting^C
helk@helk-VirtualBox:~/Desktop/HELK/docker$ sudo ./helk_install.sh

*****
**          HELK - THE HUNTING ELK          **
**                                      **
** Author: Roberto Rodriguez (@Cyb3rWard0g)  **
** HELK build version: v0.1.9-alpha10082020  **
** HELK ELK version: 7.6.2                  **
** License: GPL-3.0                         **
*****


[HELK-INSTALLATION-INFO] HELK hosted on a Linux box
[HELK-INSTALLATION-INFO] Available Memory: 6556 MBs
[HELK-INSTALLATION-INFO] You're using ubuntu version focal

*****
*          HELK - Docker Compose Build Choices      *
*****


1. KAFKA + KSQL + ELK + NGINX
2. KAFKA + KSQL + ELK + NGINX + ELASTALERT
3. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER
4. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER + ELASTALERT

Enter build choice [ 1 - 4]: 1
[HELK-INSTALLATION-INFO] HELK build set to 1
[HELK-INSTALLATION-INFO] Set HELK IP. Default value is your current IP: 10.0.2.15
```

安裝 HELK

- 打開連結 -> 複製 HELK 網址
- 輸入「git clone <網址>」
- 輸入「cd HELK/docker」 ->
「sudo ./helk_install.sh」
- Build choice 選擇「1」
- HELK IP 直接「Enter」
- 設定 Kibana password

HELK [Running]
Activities Terminal ▾ helk@helk-VirtualBox ~/Desktop/HELK/docker
[HELK-INSTALLATION-INFO] Please make sure to create a custom Kibana password and store it securely for future use.
[HELK-INSTALLATION-INFO] Set HELK Kibana UI Password: hunting^C
helk@helk-VirtualBox:~/Desktop/HELK/docker\$ sudo ./helk_install.sh

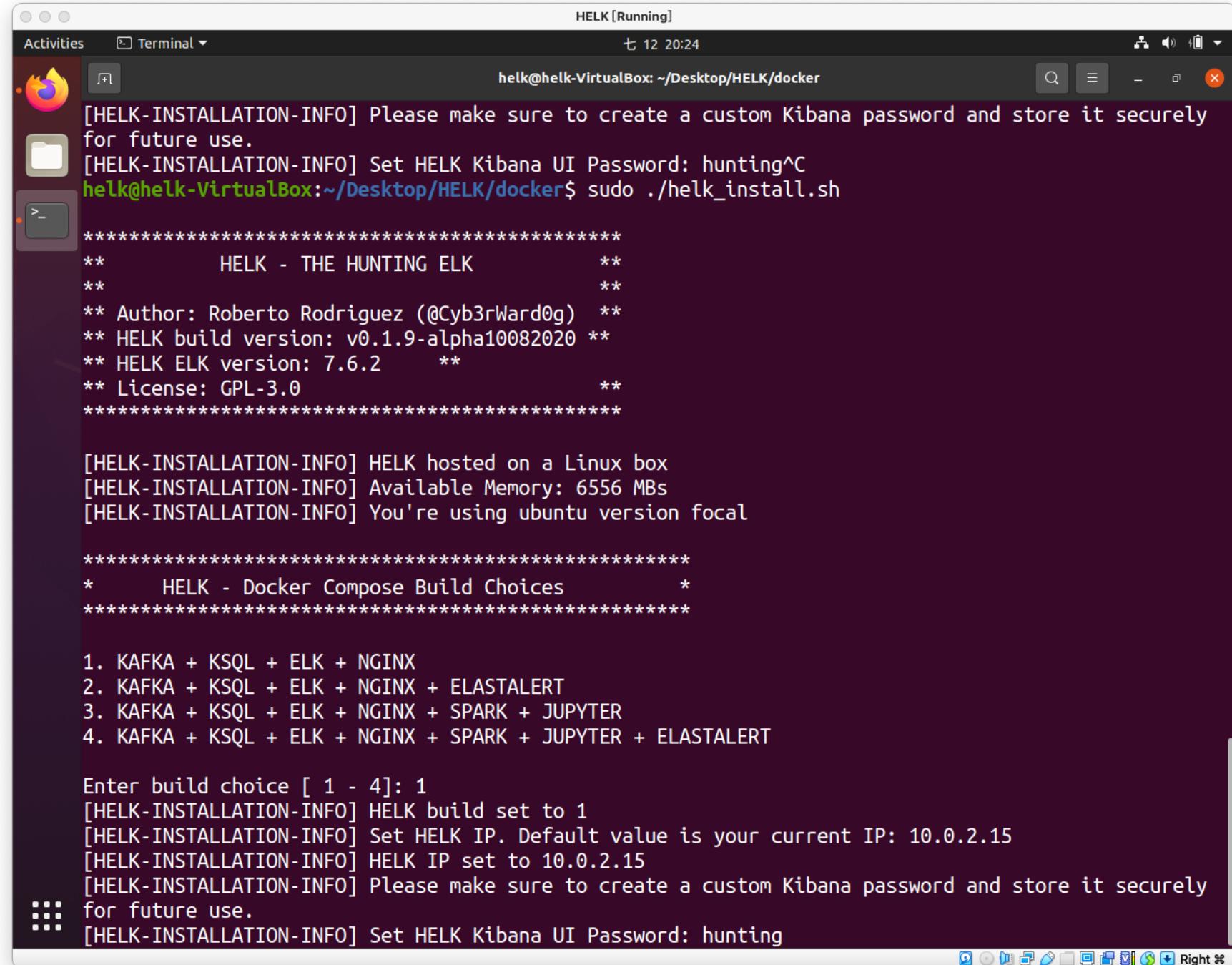
```
*****  
**          HELK - THE HUNTING ELK          **  
**  
** Author: Roberto Rodriguez (@Cyb3rWard0g) **  
** HELK build version: v0.1.9-alpha10082020 **  
** HELK ELK version: 7.6.2      **  
** License: GPL-3.0                  **  
*****
```

[HELK-INSTALLATION-INFO] HELK hosted on a Linux box
[HELK-INSTALLATION-INFO] Available Memory: 6556 MBs
[HELK-INSTALLATION-INFO] You're using ubuntu version focal

```
*****  
*          HELK - Docker Compose Build Choices          *  
*****
```

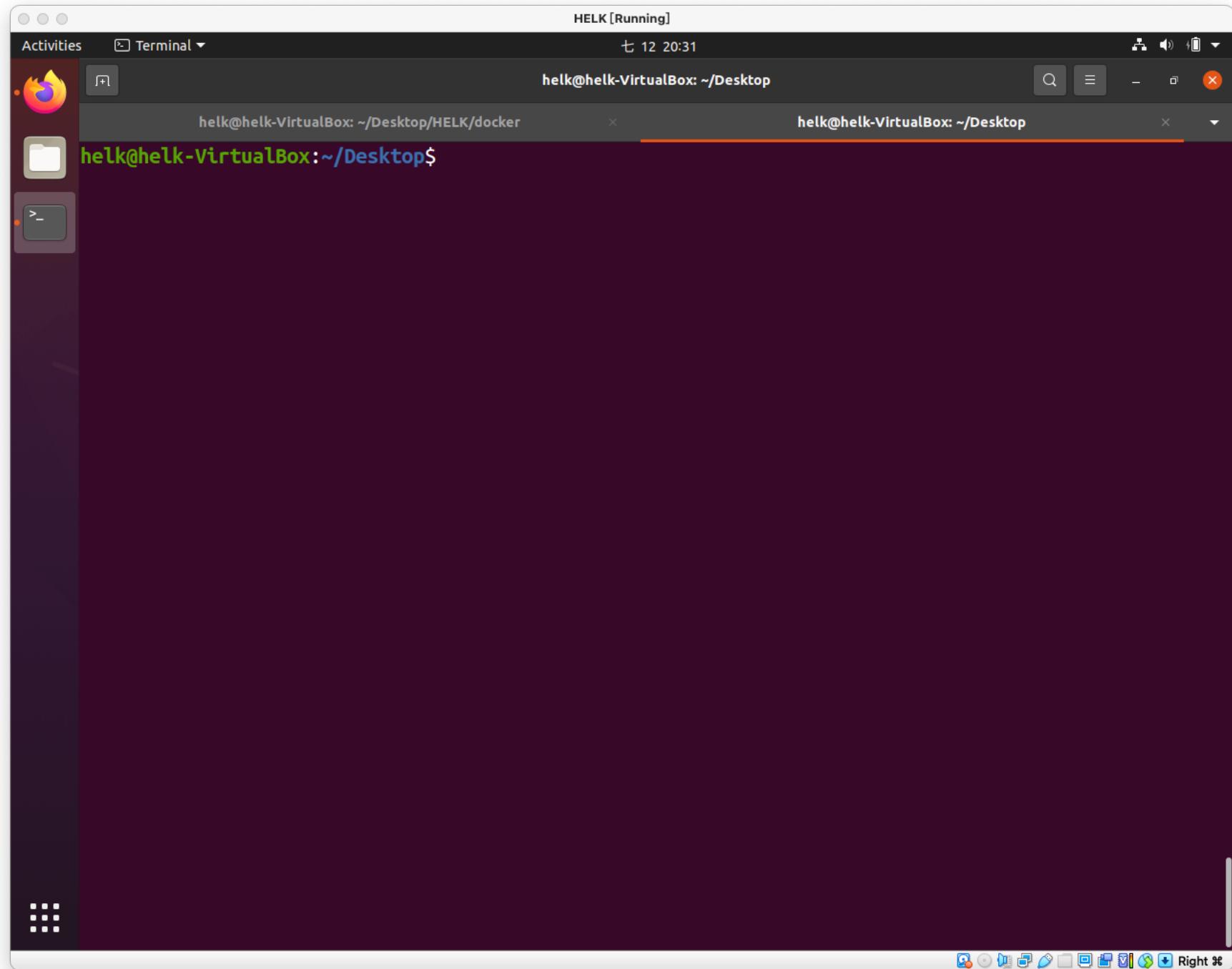
1. KAFKA + KSQL + ELK + NGINX
2. KAFKA + KSQL + ELK + NGINX + ELASTALERT
3. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER
4. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER + ELASTALERT

Enter build choice [1 - 4]: 1
[HELK-INSTALLATION-INFO] HELK build set to 1
[HELK-INSTALLATION-INFO] Set HELK IP. Default value is your current IP: 10.0.2.15
[HELK-INSTALLATION-INFO] HELK IP set to 10.0.2.15
[HELK-INSTALLATION-INFO] Please make sure to create a custom Kibana password and store it securely for future use.
[HELK-INSTALLATION-INFO] Set HELK Kibana UI Password: hunting



下載 datasets

- 輸入「Ctrl+Shift+t」打開 tab



下載 datasets

- 輸入「Ctrl+Shift+t」打開 tab
- 打開連結 -> 複製第三個網址

The screenshot shows a Firefox browser window titled "HELK [Running]" with the URL <https://github.com/OTRF/detection-hackathon-apt29/tree/master/SIEMs/HELK>. The page displays a commit by "Cyb3rWard0g" with the message "Updated docs". The commit was made on May 4, 2020, and is 15 months old. Below the commit, there is a file listing for "README.md" which has been updated. The main content of the page is titled "HELK and APT29 Day 1 & Day2" and includes a "Pre-requirements" section with the following list:

- Install Kafkacat 1.5+
 - Reference: <https://github.com/edenhill/kafkacat#install>
- git clone <https://github.com/Cyb3rWard0g/HELK>
- git clone <https://github.com/OTRF/detection-hackathon-apt29> (same VM or local host as HELK)

A red arrow points to the third item in this list.

The next section is "Send Data to HELK" with the following steps:

- cd detection-hackathon-apt29/datasets/day1
- decompress files

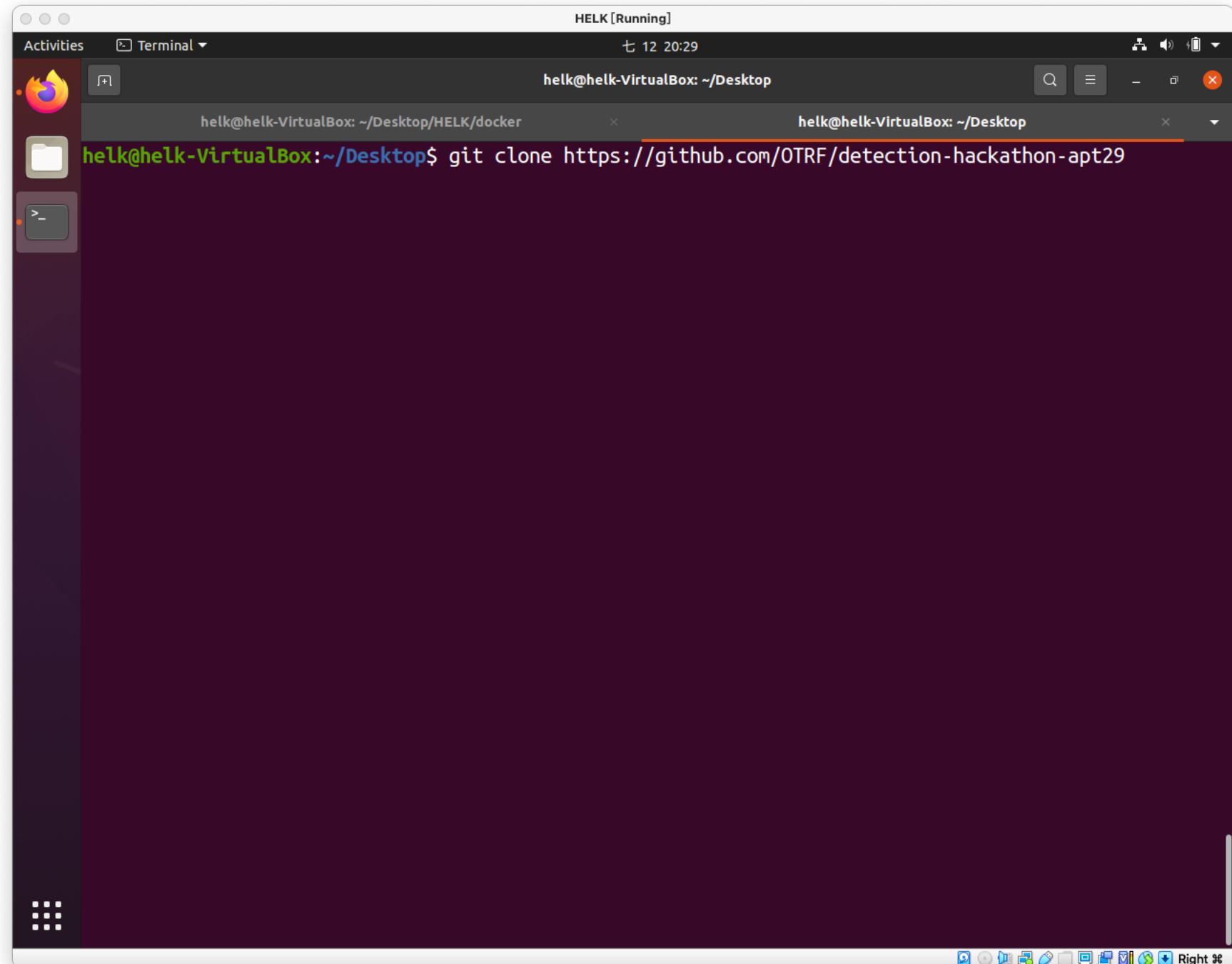
```
unzip apt29_evals_day1_manual.zip
```

- Use Kafkacat to send the data over to the kafka broker deployed by HELK

```
kafkacat -b 127.0.0.1:9092 -t winlogbeat -P -l apt29_evals_day1_manual_2020-05-01225525.json
```

下載 datasets

- 輸入「Ctrl+Shift+t」打開 tab
- 打開連結 -> 複製第三個網址
- 輸入「git clone <網址>」



下載 datasets

- 輸入「Ctrl+Shift+t」打開 tab
- 打開連結 -> 複製第三個網址
- 輸入「git clone <網址>」
- 輸入「unzip ...day1...」

The screenshot shows a terminal window titled "HELK [Running]" running on a Linux desktop. The terminal has a dark background and light-colored text. It displays the following command and its execution:

```
helk@helk-VirtualBox:~/Desktop$ git clone https://github.com/OTRF/detection-hackathon-apt29
Cloning into 'detection-hackathon-apt29'...
remote: Enumerating objects: 340, done.
remote: Counting objects: 100% (240/240), done.
remote: Compressing objects: 100% (127/127), done.
remote: Total 340 (delta 120), reused 211 (delta 105), pack-reused 100
Receiving objects: 100% (340/340), 140.46 MiB | 18.75 MiB/s, done.
Resolving deltas: 100% (138/138), done.
helk@helk-VirtualBox:~/Desktop$ unzip detection-hackathon-apt29/datasets/day1/apt29_evals_day1_manual.zip
```

The last command, "unzip detection-hackathon-apt29/datasets/day1/apt29_evals_day1_manual.zip", is highlighted with a red underline.

下載 datasets

- 輸入「Ctrl+Shift+t」打開 tab
- 打開連結 -> 複製第三個網址
- 輸入「git clone <網址>」
- 輸入「unzip ...day1...」
- 輸入「unzip ...day2...」

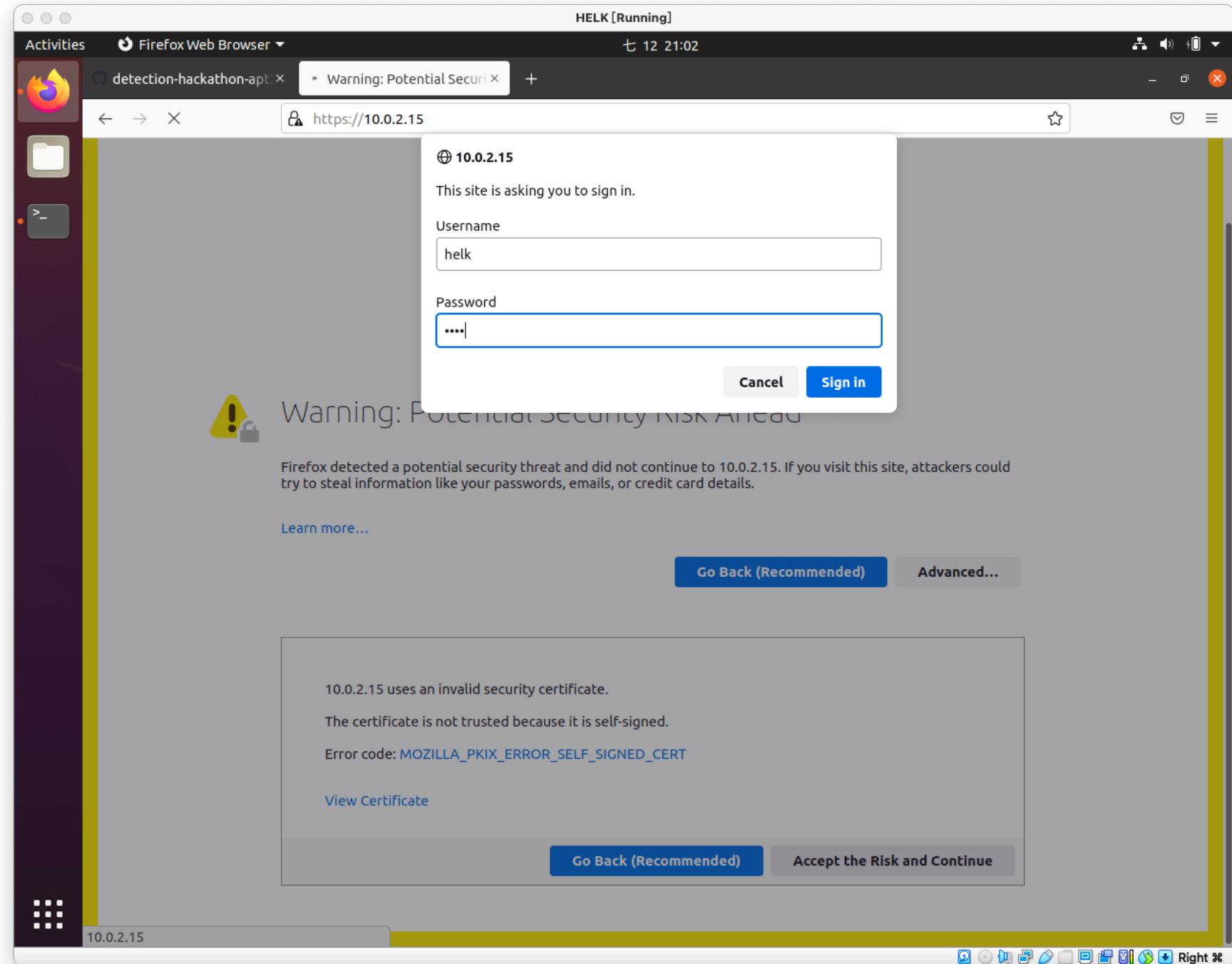
```
helk@helk-VirtualBox: ~/Desktop$ git clone https://github.com/OTRF/detection-hackathon-apt29
Cloning into 'detection-hackathon-apt29'...
remote: Enumerating objects: 340, done.
remote: Counting objects: 100% (240/240), done.
remote: Compressing objects: 100% (127/127), done.
remote: Total 340 (delta 120), reused 211 (delta 105), pack-reused 100
Receiving objects: 100% (340/340), 140.46 MiB | 18.75 MiB/s, done.
Resolving deltas: 100% (138/138), done.
helk@helk-VirtualBox:~/Desktop$ unzip detection-hackathon-apt29/datasets/day1/apt29_evals_day1_manual.zip
Archive: detection-hackathon-apt29/datasets/day1/apt29_evals_day1_manual.zip
  inflating: apt29_evals_day1_manual_2020-05-01225525.json
helk@helk-VirtualBox:~/Desktop$ unzip detection-hackathon-apt29/datasets/day2/apt29_evals_day2_manual.zip
Archive: detection-hackathon-apt29/datasets/day2/apt29_evals_day2_manual.zip
  inflating: apt29_evals_day2_manual_2020-05-02035409.json
helk@helk-VirtualBox:~/Desktop$
```

HELK 資訊

```
HELK KIBANA URL: https://10.0.2.15  
HELK KIBANA USER: helk  
HELK KIBANA PASSWORD: helk  
HELK ZOOKEEPER: 10.0.2.15:2181  
HELK KSQL SERVER: 10.0.2.15:8088
```

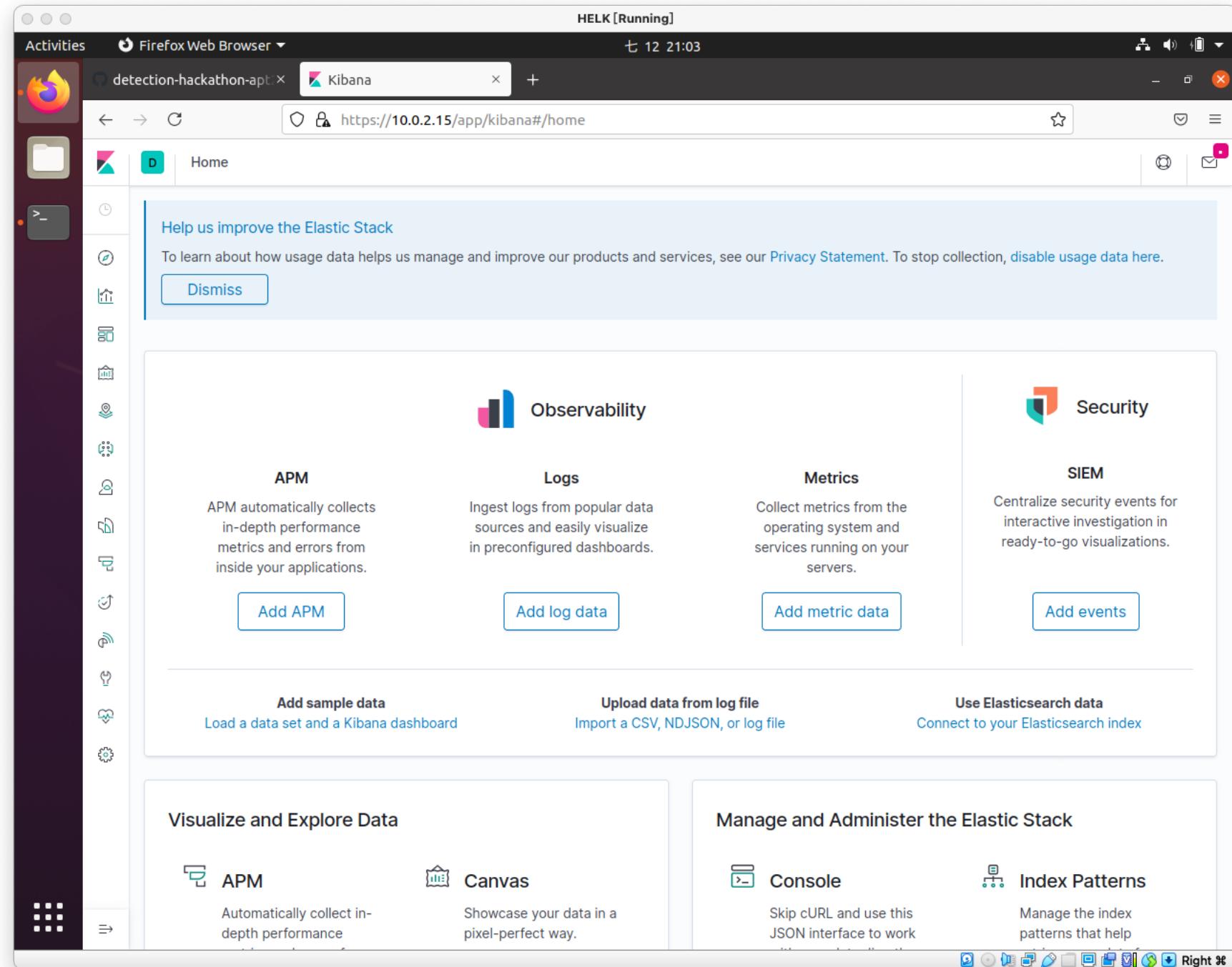
啟動 HELK

- 打開「<https://<ip>>」 ->
「Advanced...」 -> 「Accept
the Risk and Continue」，
然後輸入 user、password



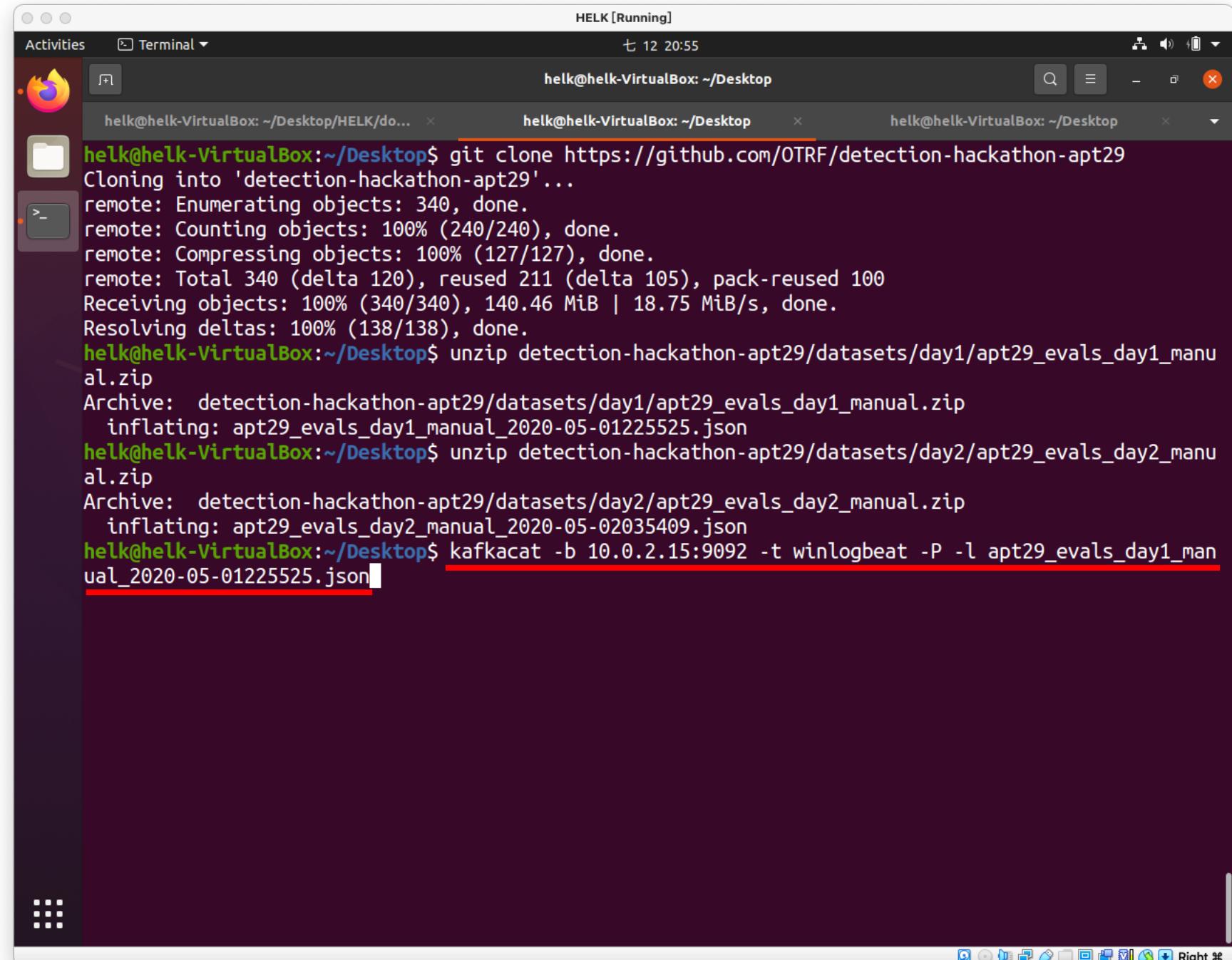
啟動 HELK

- 打開「<https://<ip>>」 ->
「Advanced...」 -> 「Accept
the Risk and Continue」，
然後輸入 user、password
- 若出現 502 Bad Gateway 表
示還在啟動中，直到出現
Kibana 畫面表示啟動完成



匯入 dataset

- 輸入「... <ip>:9092 ...day1...」



```
helk@helk-VirtualBox:~/Desktop$ git clone https://github.com/OTRF/detection-hackathon-apt29
Cloning into 'detection-hackathon-apt29'...
remote: Enumerating objects: 340, done.
remote: Counting objects: 100% (240/240), done.
remote: Compressing objects: 100% (127/127), done.
remote: Total 340 (delta 120), reused 211 (delta 105), pack-reused 100
Receiving objects: 100% (340/340), 140.46 MiB | 18.75 MiB/s, done.
Resolving deltas: 100% (138/138), done.
helk@helk-VirtualBox:~/Desktop$ unzip detection-hackathon-apt29/datasets/day1/apt29_evals_day1_manual.zip
Archive:  detection-hackathon-apt29/datasets/day1/apt29_evals_day1_manual.zip
      inflating: apt29_evals_day1_manual_2020-05-01225525.json
helk@helk-VirtualBox:~/Desktop$ unzip detection-hackathon-apt29/datasets/day2/apt29_evals_day2_manual.zip
Archive:  detection-hackathon-apt29/datasets/day2/apt29_evals_day2_manual.zip
      inflating: apt29_evals_day2_manual_2020-05-02035409.json
helk@helk-VirtualBox:~/Desktop$ kafkacat -b 10.0.2.15:9092 -t winlogbeat -P -l apt29_evals_day1_manual_2020-05-01225525.json
```

Activities Terminal ▾

HELK [Running]
七 12 21:08

helk@helk-VirtualBox: ~/Desktop

helk@helk-VirtualBox: ~/Desktop/HELK/do... x helk@helk-VirtualBox: ~/Desktop x helk@helk-VirtualBox: ~/Desktop x

helk@helk-VirtualBox: ~/Desktop\$ ls
apt29_evals_day1_manual_2020-05-01225525.json detection-hackathon-apt29
apt29_evals_day2_manual_2020-05-02035409.json HELK
helk@helk-VirtualBox:~/Desktop\$ kafkacat -b 10.0.2.15:9092 -t winlogbeat -P -l apt29_evals_day2_ma
ual_2020-05-02035409.json
helk@helk-VirtualBox:~/Desktop\$

Right ⌘

匯入 dataset

- 輸入「... <ip>:9092 ...day1...」
- 輸入「... <ip>:9092 ...day2...」

匯入 dataset

- 輸入「... <ip>:9092 ...day1...」
- 輸入「... <ip>:9092 ...day2...」
- 打開 Kibana，點選「Visualize」
→ 「Global_Count」

The screenshot shows a Firefox browser window running on a Mac OS X desktop. The title bar indicates the browser is titled 'detection-hackathon-apt' and the tab is 'Kibana'. The URL in the address bar is https://10.0.2.15/app/kibana#/visualize?_g=h@9037929. A red arrow points from the text '點選『Visualize』' to the 'Visualize' button in the top navigation bar. Another red arrow points from the text '→ 『Global_Count』' to the 'Global_Count' visualization entry in the list below.

HELK [Running]
七 12 21:09

Firefox Web Browser

detection-hackathon-apt Kibana

Visualize

Help us improve the Elastic Stack
To learn about how usage data helps us manage and improve our products and services, see our [Privacy Statement](#). To stop collection, [disable usage data here](#).

Dismiss

Visualizations

+ Create visualization

Title	Type	Description	Actions
Global_Count	Metric		
Global_EventIDs	Pie		
Global_Hashes_Sha256	Data Table		
Global_Host_Name	Pie		
Global_Logon_Type	Data Table		
Global_Process_Name	Data Table		
Global_Process_Parent_Name	Data Table		
Global_Service_Name	Data Table		
Global_User_Name	Pie		
Global_dst_ip	Data Table		

Search...

Global_Count

Global_EventIDs

Global_Hashes_Sha256

Global_Host_Name

Global_Logon_Type

Global_Process_Name

Global_Process_Parent_Name

Global_Service_Name

Global_User_Name

Global_dst_ip

Right ⌘

匯入 dataset

- 輸入「... <ip>:9092 ...day1...」
- 輸入「... <ip>:9092 ...day2...」
- 打開 Kibana，點選「Visualize」
→ 「Global_Count」
- 設定 Date 至「2020/04/30 ~ 2020/05/03」→ 「Update」

HELK [Running]
七 12 21:55

detection-hackathon-apt.x Global_Count - Kibana +
https://10.0.2.15/app/kibana#/visualize/edit/97478120-1dd7-11e8-8f1b-1b86647d4817?_g=h@4cb8b518

Dismiss

Help us improve the Elastic Stack
To learn about how usage data helps us manage and improve our products and services, see our [Privacy Statement](#). To stop collection, [disable usage data here](#).

Save Share Inspect Refresh

Lucene

Date Apr 1, 2017 @ 21:00:00.00 → May 3, 2021 @ 21:00:00.00

Update Refresh

logs-* Data Options

Metrics

Buckets

Absolute Relative Now

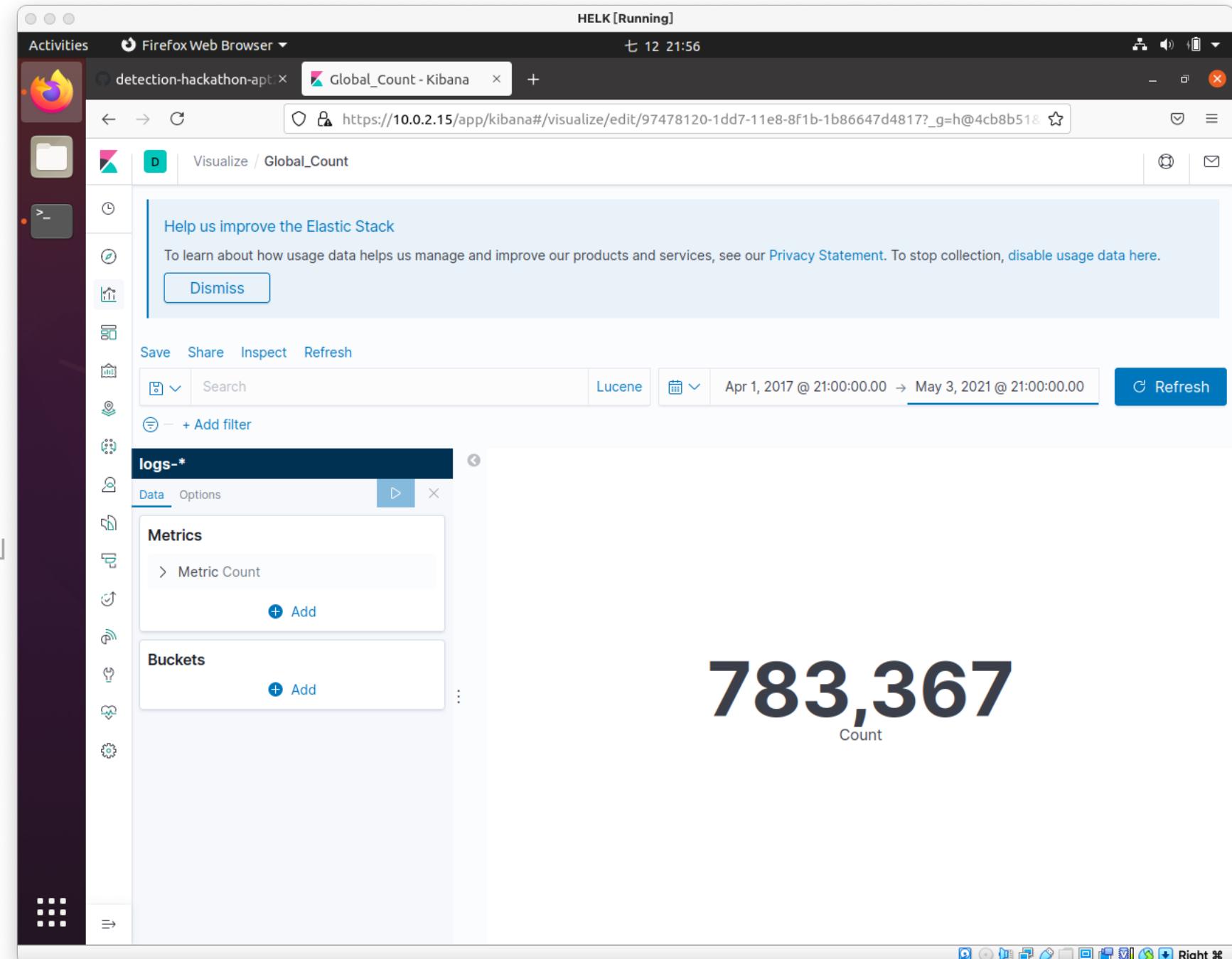
May 2021

SU	MO	TU	WE	TH	FR	SA	
25	26	27	28	29	30	1	18:30
2	3	4	5	6	7	8	19:00
9	10	11	12	13	14	15	19:30
16	17	18	19	20	21	22	20:00
23	24	25	26	27	28	29	20:30
30	31	1	2	3	4	5	21:00

End date May 3, 2021 @ 21:00:00.000

匯入 dataset

- 輸入「... <ip>:9092 ...day1...」
- 輸入「... <ip>:9092 ...day2...」
- 打開 Kibana，點選「Visualize」
→ 「Global_Count」
- 設定 Date 至「2020/04/30 ~ 2020/05/03」→ 「Update」
- 點選「Update」，等到數值為 783,367，應該就是匯入完成



啟動、關閉 HELK

- 下次開機會需要使用右方的 start 指令啟動 HELK

```
sudo docker-compose -f /home/helk/Desktop/HELK/docker/helk-kibana-analysis-basic.yml start  
sudo docker-compose -f /home/helk/Desktop/HELK/docker/helk-kibana-analysis-basic.yml stop
```