

NASA2022 Homework3

Network Administration

Short Answer

1. **Pass:** 如果傳來的封包match rule的設定，封包便可以通過Firewall。如果該rule允許state tracking，會create一個state table entry，將允許相關的流量回傳時能夠通過。

Block: 如果傳來的封包match rule的設定，封包便會被丟棄。

Reject: 如果傳來的封包match rule的設定，封包便會被丟棄，並且所支援的protocol會在傳送一個封包告訴sender該connection失敗了。

[Ref]

2. 選擇net是當ip address指在rule指定的subnet底下，就會match；選擇address是代表ip address要和rule所指定的一樣才會match

[Ref]

3. Stateful firewall: 此種防火牆可以用來持續追蹤通過它的各種網路連線，是一種用來區分不同連線種類下的合法封包。只有符合主動連線的封包才能通過該防火牆，其他的都會被拒絕。用來進行動態封包過濾。打個比方：假如我發起一個port5000去連google的port80，且這個流量是被允許的，防火牆就會預期google回應一個以port80回應我的port5000的連線，所以會有一個state table去記錄所有出去的流量，只要在table內的都會被放行，當偵測到流量結束，這個暫時的放行的rule就會關閉。

Stateless firewall: 透過destination address、source或其他key value來停估一個流量是否存在威脅。用預設好的rules來強制執行是否要允許或deny某個流量，但如果有偽裝成trusted communication的unauthorized communication，系統常無法辨識其與真正需要的communication之間的差異。

pfSense是**stateful firewall**，因為他可以記錄連線的資訊，因此可以自動允許reply的流量。

[Ref1][Ref2][Ref3]

pfSense

1. **Step1:** 到 `interfaces/VLANS` 底下去新增3個vlan，分別為5/8/99。

Step2: 再到 `interfaces/Interface Assignments` 將他們的IPv4 Configuration Type設成Static IPv4，並把IPv4 address分別設成10.5.0.1/24、10.8.0.1/24、10.99.0.1/24。

Step3: 到 `Services/DHCP Server` 把VLAN 5、8、99的"Enable DHCP server on VLAN5 interface"選項開啟，並把range設成和available range一樣，並把DNS server都加上8.8.8.8以及8.8.4.4便完成。

(討論對象：b09505014王聖文)

2. **Step1:** 到 `Firewall/Aliases/IP` 底下，分別新增一個Name是GOOGLE_DNS和CSIE_WORKSTATIONS的alias，Type都選Host(s)，接著分別把題目所要求的value都加進去即可。

Step2: 到 `Firewall/Aliases/Ports` 底下，新增一個Name為ADMIN_PORTS的alias，Type選Port(s)，接著分別把題目所要求的value都加進去即可。

[Ref]

3. **Step1:** 到 `System/Advanced/Admin Access`，在Secure Shell選項下把"Enable Secure Shell"選項開啟。

Step2: 到 `Firewall/Rules/VLAN99` 底下完成以下rule設定，即可完成。

<input type="checkbox"/>	<input checked="" type="checkbox"/> 0 / 0 B	IPv4 TCP	VLAN99 net	*	This Firewall	ADMIN_PORTS	*	none					
--------------------------	---	----------	------------	---	---------------	-------------	---	------	--	--	--	--	--

[Ref]

4. **Step1:** 到 `Firewall/Rules/VLAN99` 底下完成以下rule設定，即可完成。

<input type="checkbox"/>	<input checked="" type="checkbox"/> 0 / 0 B	IPv4 TCP	*	*	*	853 (DNS over TLS)	*	none					
<input type="checkbox"/>	<input checked="" type="checkbox"/> 0 / 2 KIB	IPv4 TCP/UDP	*	*	*	53 (DNS)	*	none					
<input type="checkbox"/>	<input checked="" type="checkbox"/> 0 / 0 B	IPv4+6 *	*	*	CSIE_WORKSTATIONS	*	*	none					
<input type="checkbox"/>	<input checked="" type="checkbox"/> 0 / 0 B	IPv4+6 *	*	*	This Firewall	*	*	none					

traceroute linux1.csie.org 的結果：

```
localhost:~# traceroute linux1.csie.org
traceroute to linux1.csie.org (140.112.30.32), 30 hops max, 46 byte packets
1 10.99.0.1 (10.99.0.1) 0.199 ms 0.395 ms 0.248 ms
2 10.0.2.2 (10.0.2.2) 0.424 ms 0.360 ms 0.383 ms
3 * * *
4 192.168.203.230 (192.168.203.230) 13.025 ms 6.056 ms 5.796 ms
5 wl127.cc.ntu.edu.tw (140.112.4.254) 5.343 ms 4.910 ms 5.057 ms
6 140.112.0.170 (140.112.0.170) 5.511 ms 140.112.0.210 (140.112.0.210) 5.102 ms 4.636 ms
7 140.112.0.237 (140.112.0.237) 4.466 ms 140.112.0.217 (140.112.0.217) 4.143 ms 140.112.0.237
(140.112.0.237) 4.776 ms
8 140.112.149.122 (140.112.149.122) 16.105 ms 15.016 ms 10.010 ms
9 linux1.csie.ntu.edu.tw (140.112.30.32) 5.749 ms 4.580 ms 4.636 ms
localhost:~#
```

traceroute linux2.csie.org的結果：

```
● ● ● 99 [Running]
localhost:~# traceroute linux2.csie.org
traceroute to linux2.csie.org (140.112.30.33), 30 hops max, 46 byte packets
1 10.99.0.1 (10.99.0.1) 0.464 ms 0.368 ms 0.342 ms
2 10.0.2.2 (10.0.2.2) 0.561 ms 0.566 ms 0.944 ms
3 * * *
4 192.168.203.230 (192.168.203.230) 13.553 ms 6.299 ms 6.553 ms
5 wl127.cc.ntu.edu.tw (140.112.4.254) 5.564 ms 5.724 ms 4.675 ms
6 140.112.0.210 (140.112.0.210) 4.455 ms 140.112.0.170 (140.112.0.170) 4.934 ms 140.112.0.210
(140.112.0.210) 4.356 ms
7 140.112.0.237 (140.112.0.237) 4.122 ms 140.112.0.217 (140.112.0.217) 5.541 ms 140.112.0.237
(140.112.0.237) 5.251 ms
8 140.112.149.122 (140.112.149.122) 6.235 ms 6.641 ms 6.277 ms
9 linux2.csie.ntu.edu.tw (140.112.30.33) 5.354 ms 4.549 ms 4.636 ms
localhost:~#
```

traceroute linux3.csie.org的結果：

```
● ● ● 99 [Running]
localhost:~# traceroute linux3.csie.org
traceroute to linux3.csie.org (140.112.30.34), 30 hops max, 46 byte packets
1 10.99.0.1 (10.99.0.1) 25.826 ms 0.471 ms 0.720 ms
2 10.0.2.2 (10.0.2.2) 0.627 ms 0.546 ms 0.533 ms
3 * * *
4 192.168.203.230 (192.168.203.230) 11.951 ms 6.607 ms 8.079 ms
5 wl127.cc.ntu.edu.tw (140.112.4.254) 4.225 ms 4.014 ms 4.082 ms
6 140.112.0.210 (140.112.0.210) 10.796 ms 140.112.0.170 (140.112.0.170) 4.125 ms 140.112.0.210
(140.112.0.210) 3.896 ms
7 140.112.0.217 (140.112.0.217) 4.322 ms 4.086 ms 140.112.0.237 (140.112.0.237) 5.661 ms
8 140.112.149.122 (140.112.149.122) 6.843 ms 6.031 ms 7.032 ms
9 * * *
10 * * *
11 * * *
12 * * *
13 * * *
14 * * *
15 * *c *
16^C
localhost:~#
```

(linux3在當日應該是未開機，因為我嘗試用terminal要連線linux3也連不上)

在VLAN99的機器ssh連到pfSense的結果：

```
● ● ● 99 [Running]
localhost:~# ssh admin@10.99.0.1
(admin@10.99.0.1) Password for admin@pfSense.home.arp:
VirtualBox Virtual Machine - Netgate Device ID: dc683b1661405c954734

*** Welcome to pfSense 2.6.0-RELEASE (amd64) on pfSense ***

WAN (wan)      -> pcn0      -> v4/DHCP4: 10.0.2.15/24
LAN (lan)      -> pcn1      -> v4: 192.168.1.1/24
VLAN5 (opt1)   -> pcn1.5   -> v4: 10.5.0.1/24
VLAN99 (opt2)  -> pcn1.99  -> v4: 10.99.0.1/24
VLAN8 (opt3)   -> pcn1.8   -> v4: 10.8.0.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces           10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults 13) Update from console
5) Reboot system               14) Disable Secure Shell (sshd)
6) Halt system                 15) Restore recent configuration
7) Ping host                   16) Restart PHP-FPM

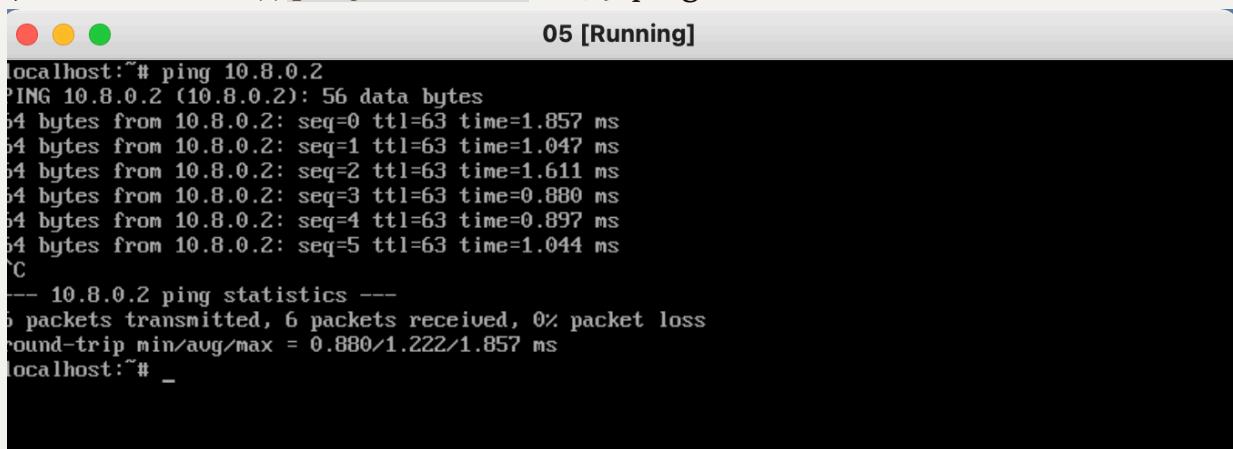
Enter an option: _
```

(討論對象：b09505014王聖文)

5. Step1: 到 Firewall/Rules/VLAN5 完成以下設定即可

<input type="checkbox"/>	<input checked="" type="checkbox"/>	0 / 0 B	IPv4 ICMP	*	*	VLAN8 net	*	*	none	   
--------------------------	-------------------------------------	---------	-----------	---	---	-----------	---	---	------	---

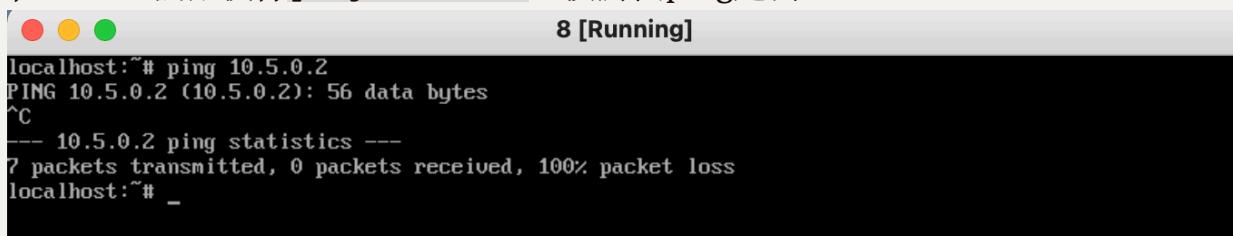
在VLAN5機器執行 ping 10.8.0.2，可以ping過去：



05 [Running]

```
localhost:~# ping 10.8.0.2
PING 10.8.0.2 (10.8.0.2): 56 data bytes
64 bytes from 10.8.0.2: seq=0 ttl=63 time=1.857 ms
64 bytes from 10.8.0.2: seq=1 ttl=63 time=1.047 ms
64 bytes from 10.8.0.2: seq=2 ttl=63 time=1.611 ms
64 bytes from 10.8.0.2: seq=3 ttl=63 time=0.880 ms
64 bytes from 10.8.0.2: seq=4 ttl=63 time=0.897 ms
64 bytes from 10.8.0.2: seq=5 ttl=63 time=1.044 ms
^C
--- 10.8.0.2 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 0.880/1.222/1.857 ms
localhost:~# _
```

在VLAN8機器執行 ping 10.5.0.2，沒辦法ping過去：



8 [Running]

```
localhost:~# ping 10.5.0.2
PING 10.5.0.2 (10.5.0.2): 56 data bytes
^C
--- 10.5.0.2 ping statistics ---
7 packets transmitted, 0 packets received, 100% packet loss
localhost:~# _
```

(討論對象：b09505014王聖文)

6. Step1: 到 Firewall/Schedules 先新增一個schedule，將時間range設在May 10的0:00-23:59。

Step2: 到 Firewall/Rules/VLAN5 底下，新增一個rule如下，schedule的設定可以透過新增rule之頁面底下有一個Extra Options，將下面的Advanced Option點開後，可以選擇想要的schedule。

<input type="checkbox"/>	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input checked="" type="checkbox"/>	X	0 / 0 B	IPv4+6 *	*	*	*	*	none	 nasa_hw3	   	

[Ref]

7. 到 Diagnostics/Backup & Restore 底下點擊Download configuration as XML，即可完成。

(討論對象：b09505014王聖文)

System Administration

KVM & Virsh

1. Step1: 先用

```
qemu-img create -f qcow2 /tmp2/b08902149/ubuntu.qcow2 8G 建立一個要給虛擬機器用的虛擬磁碟檔案。
```

Step2: 再透過以下指令安裝VM：

```
virt-install --virt-type kvm --name b08902149 --vcpus 2 --  
memory 2048 --disk /tmp2/b08902149/ubuntu.qcow2  
--graphics vnc,listen=0.0.0.0,password=08902149  
--noautoconsole --os-variant=ubuntu20.04  
--cdrom=/tmp2/nasa-hw3/ubuntu.iso
```

Step3: 之後以 `virsh vncdisplay b08902149` 看是使用哪個port得到output為 :1 。

Step4: 用本機下載vnc viewer之後，連線到 `linux11.csie.ntu.edu.tw:1` 進行安裝。

Step5: 根據題目要求設定server name、username，完成ubuntu的安裝。

Step6: 如果VM莫名的掛掉，可以回到工作站上打

```
virsh start b08902149
```

`virsh list` 的畫面：

```
[b08902149@linux11 [~] virsh list  
  Id   名稱      狀態  
-----  
  2    b08902149  執行中
```

開機完成的畫面：(開機過程會有一些[ok]的畫面，但不曉得為什麼會被洗掉所以沒截圖到)



登入後畫面：

```
Ubuntu 20.04.4 LTS nasa-hw3 tty1
nasa-hw3 login: b08902149
Password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-107-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Sat 09 Apr 2022 11:24:46 AM UTC

System load:          0.0
Usage of /:           49.5% of 7.07GB
Memory usage:         9%
Swap usage:           0%
Processes:            124
Users logged in:     0
IPv4 address for emp1s0: 10.0.2.15
IPv6 address for emp1s0: fec0::5054:ff:fe4d:c712

23 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sat Apr  9 10:52:17 UTC 2022 on tty1
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

b08902149@nasa-hw3:~$
```

[Ref1] [Ref2]

2. Step1: `sudo vi /etc/default/grub`，將檔案改成以下：

```
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
#   info -f grub -n 'Simple configuration'

GRUB_DEFAULT=0
GRUB_TIMEOUT_STYLE=hidden
GRUB_TIMEOUT=2
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed=115200"
GRUB_DISTROITOR= lsb_release -i -s 2>/dev/null || echo Debian
GRUB_CMDLINE_LINUX_DEFAULT="maybe-ubiquity"
GRUB_CMDLINE_LINUX="console=tty1 console=ttyS0,115200"

# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that obtains
# the memory map information from GRUB (GNU Mach, Kernel of FreeBSD ...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"

# Uncomment to disable graphical terminal (grub-pc only)
#GRUB_TERMINAL=console

# The resolution used on graphical terminal
# note that you can use only modes which your graphic card supports via VBE
# you can see them in real GRUB with the command 'vbeinfo'
#GRUB_GFXMODE=640x480

# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter to Linux
#GRUB_DISABLE_LINUX_UUID=true

# Uncomment to disable generation of recovery mode menu entries
#GRUB_DISABLE_RECOVERY="true"

# Uncomment to get a beep at grub start
#GRUB_INIT_TUNE="480 440 1"
~

"/etc/default/grub" 35L, 1319C written
b08902149@nasa-hw3:~$
```

Step2: `sudo update-grub`

Step3: 到工作站打 `virsh reboot b08902149`

在工作站上打 virsh console b089021409 的結果：

```
[b08902149@linux11 [~] virsh console b08902149
Connected to domain 'b08902149'
Escape character is ^] (Ctrl + ])

[nasa-hw3 login: b08902149
[Password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-107-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:      https://landscape.canonical.com
 * Support:         https://ubuntu.com/advantage

 System information as of Sat 09 Apr 2022 12:00:43 PM UTC

 System load:          0.19
 Usage of /:           49.5% of 7.07GB
 Memory usage:         9%
 Swap usage:           0%
 Processes:            133
 Users logged in:     0
 IPv4 address for enp1s0: 10.0.2.15
 IPv6 address for enp1s0: fec0::5054:ff:fe4d:c712

23 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sat Apr  9 11:24:47 UTC 2022 on tty1
b08902149@nasa-hw3:~$
```

[\[Ref1\]](#)[\[Ref2\]](#)

Docker

1. 以下指令用來在vm安裝docker和docker-compose：

```
sudo apt update
sudo apt install docker
sudo apt install docker-compose
```

`sudo docker version` 與 `sudo docker-compose version` 的結果：

```
[b08902149@nasa-hw3:~$ sudo docker version
[[sudo] password for b08902149:
[Client:
  Version:          20.10.7
  API version:     1.41
  Go version:      go1.13.8
  Git commit:       20.10.7-0ubuntu5~20.04.2
  Built:           Mon Nov  1 00:34:17 2021
  OS/Arch:          linux/amd64
  Context:          default
  Experimental:    true

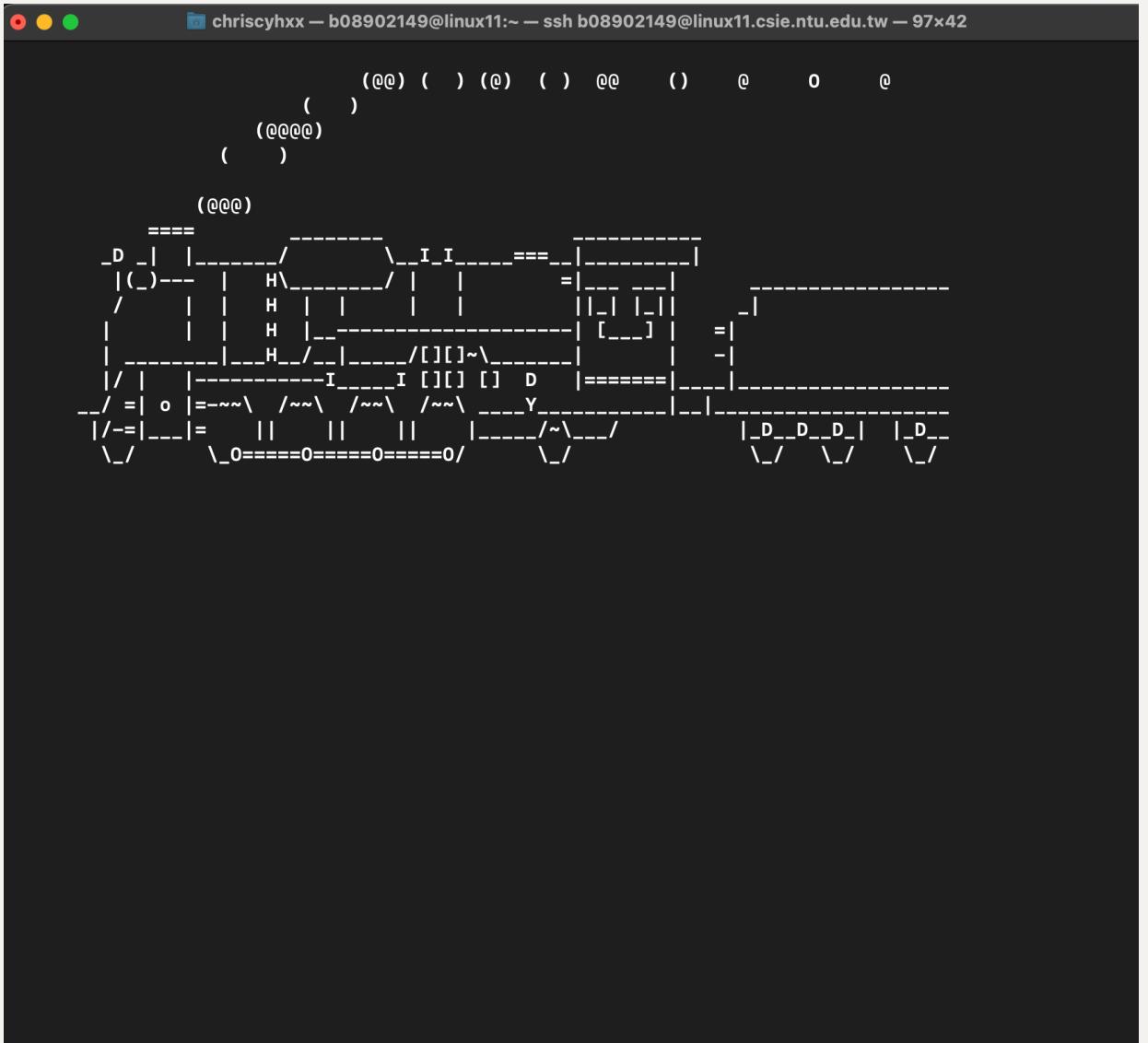
Server:
  Engine:
    Version:          20.10.7
    API version:     1.41 (minimum version 1.12)
    Go version:      go1.13.8
    Git commit:       20.10.7-0ubuntu5~20.04.2
    Built:           Fri Oct 22 00:45:53 2021
    OS/Arch:          linux/amd64
    Experimental:    false
  containerd:
    Version:          1.5.5-0ubuntu3~20.04.2
    GitCommit:
  runc:
    Version:          1.0.1-0ubuntu2~20.04.1
    GitCommit:
  docker-init:
    Version:          0.19.0
    GitCommit:
[b08902149@nasa-hw3:~$ sudo docker-compose version
docker-compose version 1.25.0, build unknown
docker-py version: 4.1.0
CPython version: 3.8.10
OpenSSL version: OpenSSL 1.1.1f  31 Mar 2020
```

(討論對象：b05504066 李旻翰)

2. Dockerfile如下：

```
FROM alpine:latest
RUN apk add --update-cache \
    gcc \
    make \
    ncurses-dev \
    libc-dev
WORKDIR /sl
COPY . /sl
RUN cd /sl
RUN make
CMD [ "./sl" ]
```

執行 `docker run --rm -it sl` 的畫面：



[Ref1][Ref2]

(討論對象：`b05504066 李旻翰`)

3. 我所用到的指令：

`sudo docker container ls -a` 可以看到還有哪些container。

`sudo docker container stop $(sudo docker container ls -aq)` 中止所有 container。

`sudo docker container rm $(sudo docker container ls -aq)` 移除所有 container。

`sudo docker image ls` 可以看到還有哪些image。

`sudo docker image prune -a` 將所有沒在使用的images都刪掉。

`sudo docker-compose up -d` 將service在背景跑起來。

以下為 `sudo docker-compose ps` 結果：

Name	Command	State	Ports
golang-shorturl_cleaner_1	./cleaner	Up	
golang-shorturl_memcached_1	memcached	Up	11211/tcp
golang-shorturl_mongo_1	mongod	Up	27017/tcp
golang-shorturl_server_1	/bin/sh -c ./server	Up	0.0.0.0:8000->8000/tcp ,:::8000->8000/tcp

透過以下指令得到短網址：

```
sudo curl -X POST http://localhost:8000/api/v1/urls
-H 'Content-Type: application/json'
-d '{"url": "https://youtu.be/j8PxqqliIno", "expireAt": "2023-02-08T09:20:41Z"}'
```

可以得到短網址<http://localhost:8000/AKUqC1>，

`curl -v http://localhost:8000/AKUqC1` 結果如下：

```
b08902149@nasa-hw3:~/golang-shorturl$ sudo curl -X POST http://localhost:8000/api/v1/urls -H 'Content-Type: application/json'
-d '{"url": "https://youtu.be/j8PxqqliIno", "expireAt": "2023-02-08T09:20:41Z"}'
{"id": "AKUqC1", "shortUrl": "http://localhost:8000/AKUqC1"}
b08902149@nasa-hw3:~/golang-shorturl$ curl -v http://localhost:8000/AKUqC1
* Trying 127.0.0.1:8000...
* TCP_NODELAY set
* Connected to localhost (127.0.0.1) port 8000 (#0)
> GET /AKUqC1 HTTP/1.1
> Host: localhost:8000
> User-Agent: curl/7.68.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 307 Temporary Redirect
< Location: https://youtu.be/j8PxqqliIno
< Date: Sun, 10 Apr 2022 04:11:45 GMT
< Content-Length: 0
<
* Connection #0 to host localhost left intact
b08902149@nasa-hw3:~/golang-shorturl$
```

[Ref1][Ref2]

(討論對象：`b05504066 李晏翰`)

Clean Up

透過以下指令將vm清掉：

```
virsh destroy b08902149 2> /dev/null
virsh undefine b08902149
rm /tmp2/b08902149/ubuntu.qcow2
```

以下得到 `virsh list --all` 的結果：

```
[b08902149@linux11 ~] virsh list --all
```

Id	名稱	狀態
----	----	----

```
b08902149@linux11 ~]
```

[Ref]

(討論對象：b09505014王聖文)