HW11

Author: B12705014 陳泊華

Web Terminology

1.

```
Performance: Nginx outperforms Apache in terms of being lightweight and ha
Configuration: Nginx's configuration files are concise and easy to read, w
Tuning: Nginx demonstrates more stable performance under high concurrency
```

2. Static Web Server provides static file service, such as HTML, Javascript, CSS, while dynamic web server can execute server-side programs and dynamically generates personalized web content and functions.

3.

```
CSR (Client-Side Rendering): smooth and interactive user experience (thoug
SSR (Server-Side Rendering): beneficial for applications with dynamic data
SSG (Static Site Generation): provides excellent performance and security
ISR (Incremental Static Regeneration): some parts of the page can be stati
```

4. proxy acts as the middle agent that sits in front of a group of client machines and help forward requests. There are several advantages:
   (1) Access control: Proxies can control client access to specific content or services through access rules and filters, thereby protecting sensitive resources from unauthorized access.
   (2) Protocol Conversion: Proxies can facilitate the conversion and mapping between different protocols, enabling cross-protocol communication and interoperability.
   (3) Network Mitigation: During network congestion or attacks, proxies can regulate and balance network traffic, alleviate server loads, and protect network stability and availability.

5. Reverse proxy, unlike forward proxy, serves as a connection between a client and multiple servers. Based on this feature, there are several advantages of reverse proxy:
   (1) Load balancing: decides which server is the most suitable and manage the load.

(2) Caching: records temporary and static data that can be responded without contacting the behind servers.
(3) Security: A reverse proxy can hide the actual ip address of its servers to enhance security.

reference:
(i) Nginx vs. apache
https://www.nexcess.net/blog/nginx-vs-apache/ (https://www.nexcess.net/blog/nginx-vs-apache/)
(ii) Static vs. dynamic
https://www.mlytics.com/blog/static-and-dynamic-websites-whats-the-difference/ (https://www.mlytics.com/blog/static-and-dynamic-websites-whats-the-difference/)
(iii) SEO
https://ranking.works/knowledge/ssr-csr-渲染/ (https://ranking.works/knowledge/ssr-csr-%E6%B8%B2%E6%9F%93/)
(iv) forward vs. reverse proxy
https://www.jyt0532.com/2019/11/18/proxy-reverse-proxy/ (https://www.jyt0532.com/2019/11/18/proxy-reverse-proxy/)

Web Server Configurations

6. Installing VM

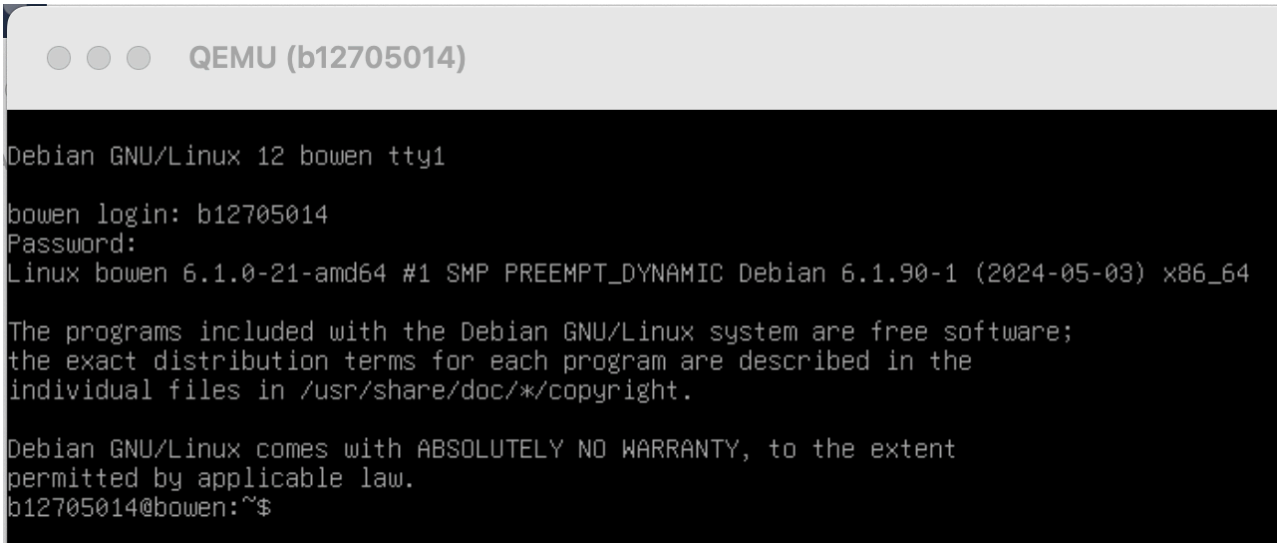
On ws4:

```
virt-install --virt-type kvm \
    --name b12705014 \
    --vcpus=2 \
    --ram 8192 \
    --disk path=/tmp2/b12705014/hw11_debian.qcow2,size=10 \
    --network default \
    --graphics vnc,password=nasa2024,listen=0.0.0.0 \
    --os-variant debian12 \
    --noautoconsole
    --cdrom=/tmp2/nasa-hw11/debian.iso
```

Eject the iso then re-import

```
$ virsh change-media b12705014 --path sda --eject --live
```

VNC Screenshot



- Add to sudoer's file
  change password to enable the user
  ```
  $ sudo passwd b12705014
  $ usermod -aG sudo b12705014
  ```
  reference: https://www.digitalocean.com/community/tutorials/how-to-create-a-new-sudo-enabled-user-on-ubuntu (https://www.digitalocean.com/community/tutorials/how-to-create-a-new-sudo-enabled-user-on-ubuntu)

- Port forwarding
  on the work station:
  ssh
  ```
  $ virsh qemu-monitor-command --hmp b12705014 'hostfwd_add ::8099-:22'
  ```
  http
  ```
  $ virsh qemu-monitor-command --hmp b12705014 'hostfwd_add ::8059-:80'
  ```
  https
  ```
  $ virsh qemu-monitor-command --hmp b12705014 'hostfwd_add ::8069-:443'
  ```

- Install Nginx packages
  ```
  $ sudo apt install nginx mariadb-server php-mysql
  ```

- start nginx service
  ```
  $ sudo systemctl start nginx
  $ sudo systemctl enable nginx
  ```

```
$ systemctl status nginx
```

```
b12705014@bowen:/etc/nginx$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
     Active: active (running) since Thu 2024-05-09 12:10:01 EDT; 53s ago
       Docs: man:nginx(8)
    Process: 6222 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 6223 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 6224 (nginx)
      Tasks: 3 (limit: 9474)
     Memory: 2.3M
        CPU: 24ms
     CGroup: /system.slice/nginx.service
             ├─6224 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ├─6225 "nginx: worker process"
             └─6226 "nginx: worker process"
b12705014@bowen:/etc/nginx$
```

reference:

(i) vm

https://linux.vbird.org/linux_server/rocky9/0130vmtuning.php

(https://linux.vbird.org/linux_server/rocky9/0130vmtuning.php)

(ii) hw5

(iii) TA slides

7. Firewall Settings

sudo iptables -A INPUT -p tcp –dport 22 -j ACCEPT
sudo iptables -A INPUT -p tcp –dport 80 -j ACCEPT
sudo iptables -A INPUT -p tcp –dport 443 -j ACCEPT
sudo iptables -A INPUT -j DROP

Check:

```
$ telnet 10.0.2.15 <port>
```

```
b12705014@bowen:~$ telnet 10.0.2.15 8888
Trying 10.0.2.15...
Connected to 10.0.2.15.
Escape character is '^]'.
^]
HTTP/1.1 400 Bad Request
Server: nginx/1.22.1
Date: Mon, 13 May 2024 05:43:25 GMT
Content-Type: text/html
Content-Length: 157
Connection: close

<html>
<head><title>400 Bad Request</title></head>
<body>
<center><h1>400 Bad Request</h1></center>
<hr><center>nginx/1.22.1</center>
</body>
</html>
Connection closed by foreign host.
b12705014@bowen:~$ sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
[sudo] password for b12705014:
b12705014@bowen:~$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
b12705014@bowen:~$ sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
b12705014@bowen:~$ sudo iptables -A INPUT -j DROP
b12705014@bowen:~$ telnet 10.0.2.15 8888
Trying 10.0.2.15...
```

reference:

(i) iptables

https://www.weithenn.org/2009/04/iptables-linux.html (https://www.weithenn.org/2009/04/iptables-linux.html)

(ii)

https://www.digitalocean.com/community/tutorials/how-to-test-your-firewall-configuration-with-nmap-and-tcpdump (https://www.digitalocean.com/community/tutorials/how-to-test-your-firewall-configuration-with-nmap-and-tcpdump)

8. The Main Page

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Hello! My name is B12705014

```
b12705014@bowen:/var/www/html$ cat index.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
        <h1> Hello! My name is B12705014 </h1>
</body>
</html>
```

reference:
(i) https://www.baeldung.com/linux/nginx-default-document-root
(https://www.baeldung.com/linux/nginx-default-document-root)

9. User Directory

- Edit location in /etc/nginx/sites-available/default

```
location ~ ^/~(.+?)(/.*)?$ {
        alias /home/$1/public_html$2;
        index  index.html index.htm;
        autoindex on;
}
```

- reload nginx

```
$ sudo systemctl reload nginx
```

Check:

reference:

(i) https://www.server-world.info/en/note?os=Ubuntu_20.04&p=nginx&f=4

(https://www.server-world.info/en/note?os=Ubuntu_20.04&p=nginx&f=4)

(ii) https://www.linkedin.com/pulse/user-directories-nginx-micael-vinhas/

(https://www.linkedin.com/pulse/user-directories-nginx-micael-vinhas/)

(iii) TA slides

10. Secret

Edit site config to add filter

```
$ vim /etc/nginx/sites-available/default
```

```
        location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a 404.
                try_files $uri $uri/ =404;
        }

        location = /secret.html {
                allow 192.168.28.0/24;
                deny all;
        }
```

```
$ cd /var/www/html
$ sudo vim secret.html
```

Test:

- add the designated ip

  ```
  $ sudo ip addr add 192.168.28.1/24 dev enp1s0
  $ ip a
  ```

```
b12705014@bowen:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 52:54:00:6a:7f:45 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp1s0
       valid_lft 63510sec preferred_lft 63510sec
    inet 192.168.28.1/24 scope global enp1s0
       valid_lft forever preferred_lft forever
    inet6 fec0::5054:ff:fe6a:7f45/64 scope site dynamic mngtmpaddr
       valid_lft 86392sec preferred_lft 14392sec
    inet6 fe80::5054:ff:fe6a:7f45/64 scope link
       valid_lft forever preferred_lft forever
```

- wget

```
b12705014@bowen:~$ wget https://192.168.28.1/secret.html --no-check-certificate
--2024-05-12 01:47:17--  https://192.168.28.1/secret.html
Connecting to 192.168.28.1:443... connected.
WARNING: The certificate of '192.168.28.1' is not trusted.
WARNING: The certificate of '192.168.28.1' doesn't have a known issuer.
The certificate's owner does not match hostname '192.168.28.1'
HTTP request sent, awaiting response... 200 OK
Length: 27 [text/html]
Saving to: 'secret.html'

secret.html                      100%[=============================================================>]      27  --.-KB/s    in 0s

2024-05-12 01:47:17 (15.7 MB/s) - 'secret.html' saved [27/27]

b12705014@bowen:~$ ls
index.html  public_html  secret.html
b12705014@bowen:~$ wget https://10.0.2.15/secret.html --no-check-certificate
--2024-05-12 01:47:55--  https://10.0.2.15/secret.html
Connecting to 10.0.2.15:443... connected.
WARNING: The certificate of '10.0.2.15' is not trusted.
WARNING: The certificate of '10.0.2.15' doesn't have a known issuer.
The certificate's owner does not match hostname '10.0.2.15'
HTTP request sent, awaiting response... 403 Forbidden
2024-05-12 01:47:55 ERROR 403: Forbidden.

b12705014@bowen:~$
```

```
b12705014@bowen:/var/www/html$ wget 10.0.2.15/secret.html
--2024-05-11 22:51:24--  http://10.0.2.15/secret.html
Connecting to 10.0.2.15:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 27 [text/html]
secret.html.1: Permission denied

Cannot write to 'secret.html.1' (Permission denied).
b12705014@bowen:/var/www/html$ wget 192.168.28.1/secret.html
--2024-05-11 22:51:46--  http://192.168.28.1/secret.html
Connecting to 192.168.28.1:80...
^C
b12705014@bowen:/var/www/html$ wget localhost/secret.html
--2024-05-11 22:52:04--  http://localhost/secret.html
Resolving localhost (localhost)... ::1, 127.0.0.1
Connecting to localhost (localhost)|::1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 27 [text/html]
secret.html.1: Permission denied

Cannot write to 'secret.html.1' (Permission denied).
b12705014@bowen:/var/www/html$
```

- browser

*or use `$ curl --interface 192.168.28.1/24 /secret.html`

reference:
(i) https://serverfault.com/questions/137907/how-to-restrict-access-to-directory-and-subdirs (https://serverfault.com/questions/137907/how-to-restrict-access-to-directory-and-subdirs)

11. Log

```
$ sudo tail/cat /var/log/nginx
```

```
b12705014@bowen:/etc/nginx$ cd /var/log/nginx
b12705014@bowen:/var/log/nginx$ ls
access.log  error.log
b12705014@bowen:/var/log/nginx$ cat access.log
cat: access.log: Permission denied
b12705014@bowen:/var/log/nginx$ sudo cat access.log
10.0.2.2 - - [09/May/2024:12:24:32 -0400] "GET / HTTP/1.1" 200 10701 "-" "curl/8.7.1"
36.227.69.186 - - [09/May/2024:12:32:29 -0400] "GET / HTTP/1.1" 200 3437 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.3
6 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
36.227.69.186 - - [09/May/2024:12:32:29 -0400] "GET /icons/openlogo-75.png HTTP/1.1" 404 187 "http://ws4.csie.ntu.edu.tw:8079/" "Mozilla/5.0 (M
acintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
36.227.69.186 - - [09/May/2024:12:32:29 -0400] "GET /favicon.ico HTTP/1.1" 404 187 "http://ws4.csie.ntu.edu.tw:8079/" "Mozilla/5.0 (Macintosh;
Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
36.227.69.186 - - [09/May/2024:12:32:44 -0400] "GET / HTTP/1.1" 200 3437 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.3
6 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
36.227.69.186 - - [09/May/2024:12:32:44 -0400] "GET /icons/openlogo-75.png HTTP/1.1" 404 187 "http://140.112.30.189:8079/" "Mozilla/5.0 (Macint
osh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
36.227.69.186 - - [09/May/2024:12:32:44 -0400] "GET /favicon.ico HTTP/1.1" 404 187 "http://140.112.30.189:8079/" "Mozilla/5.0 (Macintosh; Intel
 Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
36.227.69.186 - - [09/May/2024:12:34:03 -0400] "GET /index.html HTTP/1.1" 200 3437 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWe
bKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
36.227.69.186 - - [09/May/2024:12:34:03 -0400] "GET /icons/openlogo-75.png HTTP/1.1" 404 187 "http://140.112.30.189:8079/index.html" "Mozilla/5
.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36"
```

12. OpneSSL

(a) TLS, abbreviation of Transport Layer Security, is a protocol that establishes securely authenticated and encrypted traffic between a client and a server. PKI, abbreviation for Public Key Infrastructure, on the other hand, is a hierarchy of cryptographically-backed trust relationships between the following categories of interested party:

```
(1) Subscribers wish to use the PKI to prove their identity to the world a
(2) Relying parties wish to rely on the PKI to authenticate and connect wi
(3) Certificate Authorities (CA) are responsible for verifying the identit
```

In a nutshell, TLS is a powerful tool for encryption, while PKI provides a good way for authentication.

(b) CA, which stands for certificate authority, is a trusted third party that issues and manages certificates and enables authorized transmission and bears the responsibility of key integrity. Certificate, on the other hand, is issued by certificate authorities (CAs) and contain information about the certificate holder, such as their domain name, organization name, and public key. They also include a digital signature from the issuing CA to verify the authenticity of the certificate.

(c)

- generate rootCA private key

  ```
  $ openssl genrsa -des3 -out ca.bowen.key 1024
  ```

- generate rootCA crt
  ```
  $ openssl req -new -key ca.bowen.key -x509 -days 3650 -out ca.bowen.crt
  ```

- generate nginx server private key
  ```
  $ openssl genrsa -out /etc/ssl/private/nasa-hw11.csie.ntu.edu.key 2048
  ```

- generate nginx server csr (request)
  ```
  $ openssl req -new -key /etc/ssl/private/nasa-hw11.csie.ntu.edu.key -out /etc/ssl/certs/asa-hw11.csie.ntu.edu.csr
  ```

```
b12705014@bowen:/etc/ssl$ sudo openssl req -new -key /etc/ssl/private/nasa-hw11.csie.ntu.edu.key -out /etc/ssl/certs/asa-hw11.csie.ntu.edu.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:TW
State or Province Name (full name) [Some-State]:Taipei
Locality Name (eg, city) []:Taipei
Organization Name (eg, company) [Internet Widgits Pty Ltd]:nasa
Organizational Unit Name (eg, section) []:b12705014
Common Name (e.g. server FQDN or YOUR name) []:nasa-hw11.csie.ntu.edu
Email Address []:bowenchen0227@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

- let rootCA sign certificate to server
  ```
  $ openssl x509 -req -days 365 -in /etc/ssl/certs/asa-hw11.csie.ntu.edu.csr -CA /usr/local/share/ca-certificates/ca.bowen.crt -CAkey /usr/local/share/ca-certificates/ca.bowen.key -CAcreateserial -out /etc/ssl/certs/nasa-hw11.csie.ntu.edu.crt
  ```

```
b12705014@bowen:/usr/local/share/ca-certificates$ sudo openssl x509 -req -days 365 -in /etc/ssl/certs/asa-hw11.csie.ntu.edu.csr -CA /usr/local/share/ca-certificates/ca.bowen.crt -CAkey /usr/local/share/ca-certificates/ca.bowen.key -CAcreateserial -out /etc/ssl/certs/nasa-hw11.csie.ntu.edu.crt
Certificate request self-signature ok
subject=C = TW, ST = Taipei, L = Taipei, O = nasa, OU = b12705014, CN = nasa-hw11.csie.ntu.edu, emailAddress = bowenchen0227@gmail.com
Enter pass phrase for /usr/local/share/ca-certificates/ca.bowen.key:
```

Check:

```
$ openssl x509 -in /etc/ssl/certs/nasa-hw11.csie.ntu.edu.crt -text -noou
```

```
b12705014@bowen:/etc/nginx/sites-available$ openssl x509 -in /etc/ssl/certs/nasa-hw11.csie.ntu.edu.crt -text -noou
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number:
            47:cf:a1:05:e0:68:2b:10:fe:30:b9:98:62:9e:93:cc:c9:26:d6:cb
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = TW, ST = Taipei, L = Taipei, O = nasa, OU = b12705014, CN = bowen rootCA, emailAddress = bowenchen0227@gmail.com
        Validity
            Not Before: May 12 01:18:20 2024 GMT
            Not After : May 12 01:18:20 2025 GMT
        Subject: C = TW, ST = Taipei, L = Taipei, O = nasa, OU = b12705014, CN = nasa-hw11.csie.ntu.edu, emailAddress = bowenchen0227@gmail.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
```

reference:

(i) CA certificate https://linux.vbird.org/somepaper/20070222-root-ca.pdf (https://linux.vbird.org/somepaper/20070222-root-ca.pdf)

(ii) TLS & PKI

https://www.cockroachlabs.com/docs/stable/security-reference/transport-layer-

security (https://www.cockroachlabs.com/docs/stable/security-reference/transport-layer-security)

(iii) OpenSSL for HTTPs

https://hackmd.io/@F6INImbsTzOe4izuqqLoWQ/rk__7uMIc (https://hackmd.io/@F6INImbsTzOe4izuqqLoWQ/rk__7uMIc)

https://www.l-penguin.idv.tw/article/iis-ssl.htm (https://www.l-penguin.idv.tw/article/iis-ssl.htm)

13. Reverse Proxy

- Create and edit the content of the website

```
$ cd /var/www
$ sudo mkdir htmlA
$ sudo vim index.html
```

```
$ cd /etc/nginx/sites-available
$ sudo cp default hostA
```

- Edit (1) listen port (2) root location

```
# Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

# Default server configuration
#
server {
        listen 8888;
        listen [::]:8888;

        #listen       443 ssl http2;
        #listen       [::]:443 ssl http2;

        # SSL configuration
        #
        #listen 443 ssl;
        #listen [::]:443 ssl;
        #server_name         nasa-hw11.csie.ntu.edu;
        #ssl_protocols       TLSv1 TLSv1.1 TLSv1.2 TLSv1.3;
        #ssl_ciphers         HIGH:!aNULL:!MD5;

        # 憑證與金鑰的路徑
        #ssl_certificate /etc/ssl/certs/nasa-hw11.csie.ntu.edu.crt;
        #ssl_certificate_key /etc/ssl/private/nasa-hw11.csie.ntu.edu.key;
        #
        # Note: You should disable gzip for SSL traffic.
        # See: https://bugs.debian.org/773332
        #
        # Read up on ssl_ciphers to ensure a secure configuration.
        # See: https://bugs.debian.org/765782
        #
        # Self signed certs generated by the ssl-cert package
        # Don't use them in a production server!
        #
        # include snippets/snakeoil.conf;

        root /var/www/htmlA;

        # Add index.php to the list if you are using PHP
        index index.html index.htm index.nginx-debian.html;
```

- Add link to enable hostA

```
$ cd /etc/nginx
$ sudo ln -s ../sites-available/hostA sites-enabled/hostA
```

- Reload

```
$ sudo systemctl reload nginx
```

Then do the same thing for hostB.

Check:

```
$ netstat -an |grep LISTEN
```

```
b12705014@bowen:/etc/nginx$ netstat -an |grep LISTEN
tcp        0      0 0.0.0.0:8888            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:9999            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp6       0      0 :::8888                 :::*                    LISTEN
tcp6       0      0 :::443                  :::*                    LISTEN
tcp6       0      0 :::80                   :::*                    LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::9999                 :::*                    LISTEN
```

- Proxy config

```
        location / {
    proxy_pass  http://192.168.28.1;
    proxy_redirect off;
    proxy_set_header Host $proxy_host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;

}
```

Check:

```
$ wget http://10.0.2.15/hostA
```

```
b12705014@bowen:~$ wget http://10.0.2.15/hostA
--2024-05-13 01:09:19--  http://10.0.2.15/hostA
Connecting to 10.0.2.15:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 20 [text/html]
Saving to: ‘hostA’

hostA               100%[===================>]      20  --.-KB/s    in 0s

2024-05-13 01:09:19 (1.41 MB/s) - ‘hostA’ saved [20/20]
```

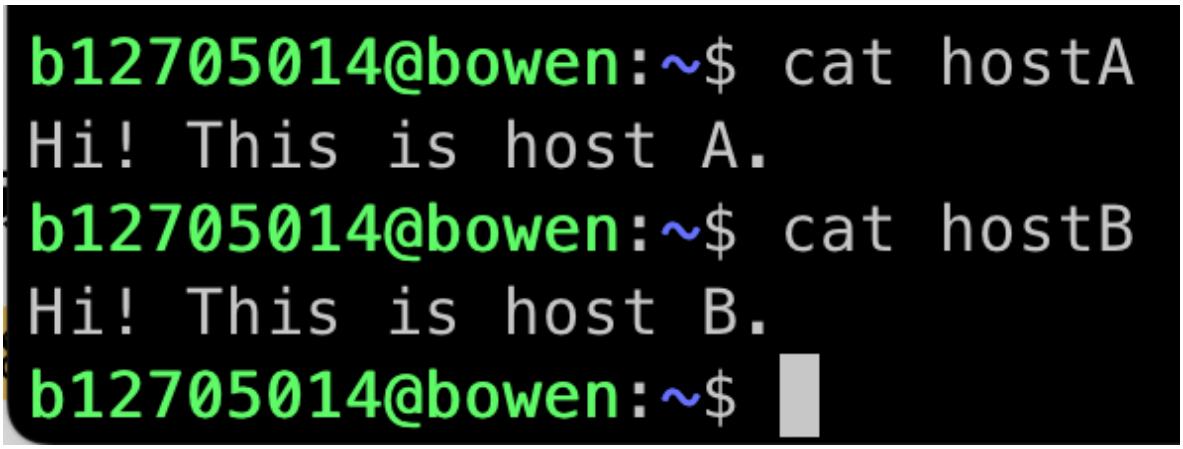
```
$ wget http://10.0.2.15/hostB
```

```
b12705014@bowen:~$ wget http://10.0.2.15/hostB
--2024-05-13 01:32:43--  http://10.0.2.15/hostB
Connecting to 10.0.2.15:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 20 [text/html]
Saving to: ‘hostB’

hostB               100%[===================>]      20  --.-KB/s    in 0s

2024-05-13 01:32:43 (317 KB/s) - ‘hostB’ saved [20/20]

b12705014@bowen:~$
```



references:
(i) TA slides
(ii) Other questions above
(iii) https://docs.nginx.com/nginx/admin-guide/web-server/reverse-proxy/ (https://docs.nginx.com/nginx/admin-guide/web-server/reverse-proxy/)