NASA HW9

B11901164 陳秉緯

討論：R13941146 李毓庭, B12901194 賴睿廷

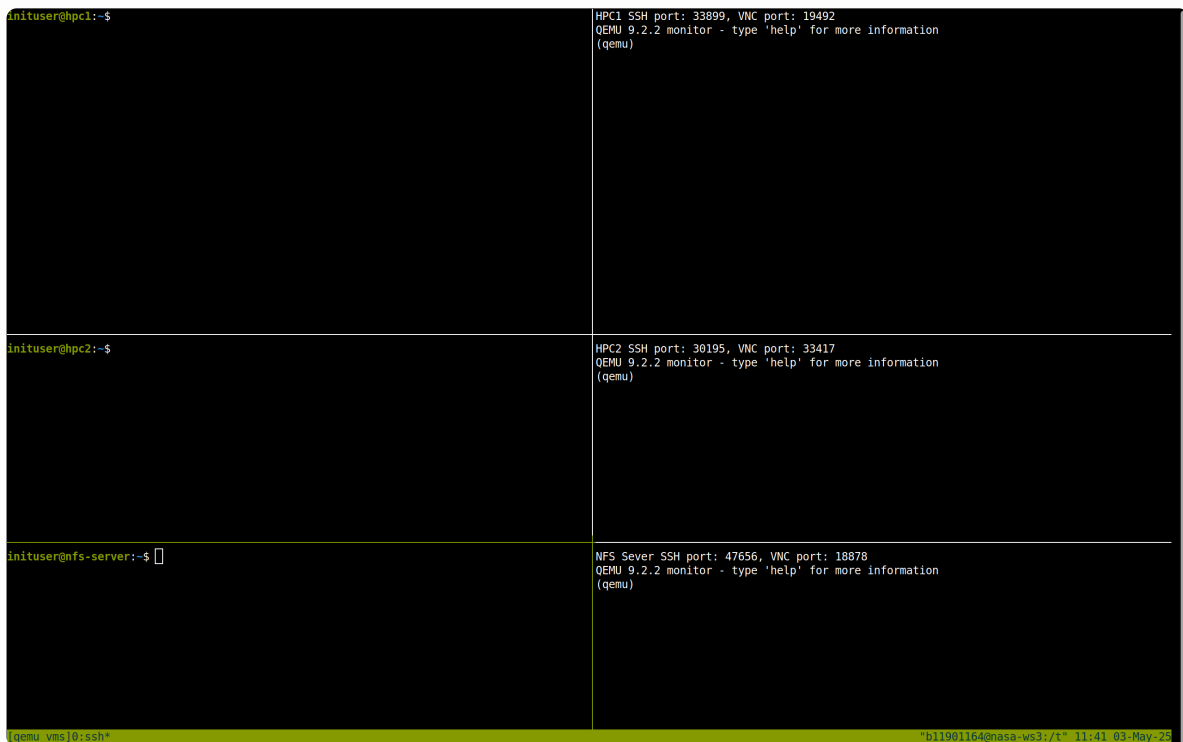## 第一部分：系統環境與 NFS 基礎安裝

1. 環境說明

ref:
https://docs.google.com/presentation/d/12CehpBWNYFjKwv4hO1BSLZGJz_u0fkogbm1qKort
Kys/edit?usp=sharing

1. 啟動三台 VM：

```
cd /tmp2/b11901164
/tmp2/hw9-release/run-vm.sh b11901164 5
```


2. 根據助教分配的 ip 分別設定 Server , hpc1 , 跟 hpc2 的 ip:
   - 停用 cloud-init 的網路設定：
     ```
     sudo mkdir -p /etc/cloud/cloud.cfg.d
     echo 'network: {config: disabled}' | sudo tee /etc/cloud/cloud.cfg.d/99-
     disable-network-config.cfg
     ```

- 建立 netplan 設定檔：

```
sudo vim /etc/netplan/01-static-ip.yaml
```

```yaml
network:
  version: 2
  ethernets:
    ens4:
      dhcp4: no
      addresses:
        - 192.150.9.19/16 # Server IP
      nameservers:
        addresses: [8.8.8.8, 1.1.1.1]
      routes:
        - to: default
          via: 192.150.0.1
```

- `sudo netplan apply` 套用設定
- `ip a show ens4` 確認設定是否正確

2. 題目要求

1.

On Server

1. `sudo mkdir -p /srv/nfs-share`
2. `sudo vim /etc/exports` 加入：

   ```
   /srv/nfs-share 192.150.9.20(rw,sync,no_subtree_check)
   192.150.9.21(rw,sync,no_subtree_check)
   ```

3. ```
   sudo chown 1000:1000 /srv/nfs-share
   sudo chmod 755 /srv/nfs-share
   sudo exportfs -arv
   ```

4. `sudo systemctl enable --now nfs-server` 啟動 nfs-server

2.

On hpc1

1. `sudo mkdir -p /mnt/nfs-share`
2. `sudo mount -t nfs 192.150.9.19:/srv/nfs-share /mnt/nfs-share`
3. `vim /mnt/nfs-share/from_hpc1.txt`：

   ```
   Hello from hpc1
   ```

On hpc2

重複步驟 2-1 ~ 2-2

5. `cd /mnt/nfs-share && ls -al && cat from_hpc1.txt`

```
inituser@hpc2:~$ cd /mnt/nfs-share && ls -al && cat from_hpc1.
txt
total 12
drwxr-xr-x 2 inituser inituser 4096 Apr 29 02:55 .
drwxr-xr-x 3 root     root     4096 Apr 29 02:57 ..
-rw-rw-r-- 1 inituser inituser   16 Apr 29 02:55 from_hpc1.txt
Hello from hpc1
```

6. `vim from_hpc2.txt` :

```
    Hello from hpc2
```

On hpc1

7. `cd /mnt/nfs-share && ls -al && cat from_hpc2.txt`

```
inituser@hpc1:~$ cd /mnt/nfs-share && ls -al && cat from_hpc2.txt
total 16
drwxr-xr-x 2 inituser inituser 4096 Apr 29 03:11 .
drwxr-xr-x 3 root     root     4096 Apr 29 02:51 ..
-rw-rw-r-- 1 inituser inituser   16 Apr 29 02:55 from_hpc1.txt
-rw-rw-r-- 1 inituser inituser   16 Apr 29 03:11 from_hpc2.txt
Hello from hpc2
```

3. 繳交內容

- 見 2. 題目要求

- 截圖 :

  1. Screenshot 1:

```
inituser@nfs-server:~$ systemctl status nfs-kernel-server
● nfs-server.service - NFS server and services
     Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; preset: enabled)
    Drop-In: /run/systemd/generator/nfs-server.service.d
             └─order-with-mounts.conf
     Active: active (exited) since Tue 2025-04-29 02:21:53 UTC; 54min ago
   Main PID: 876 (code=exited, status=0/SUCCESS)
        CPU: 11ms

Apr 29 02:21:52 nfs-server systemd[1]: Starting nfs-server.service - NFS server and services...
Apr 29 02:21:53 nfs-server systemd[1]: Finished nfs-server.service - NFS server and services.
```

2. Screenshot 2:

3. Screenshot 3:

第二部分：多使用者帳號與權限控管

1. 題目要求

ref: https://docs.google.com/presentation/d/1QOBSuBnh2F55daXRpcfpHbN-fNiUS3Hz2edsyFqzFQQ/edit?usp=sharing, http://old.linux.vbird.org/somepaper/20050817-ldap-1.pdf

1.

On Server

1. `sudo apt install -y slapd ldap-utils`

2. `vim suffix.ldif` :

```
dn: olcDatabase={1}mdb,cn=config
changetype: modify
```

```
        replace: olcSuffix
        olcSuffix: dc=nasa,dc=csie,dc=ntu
```

3. `sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f suffix.ldif` apply change

4. `vim rootdn.ldif` :

```
    dn: olcDatabase={1}mdb,cn=config
    changetype: modify
    replace: olcRootDN
    olcRootDN: cn=admin,dc=nasa,dc=csie,dc=ntu
```

5. `sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f rootdn.ldif` apply change

6. `slappasswd` and copy the result

7. `vim rootpw.ldif` :

```
    dn: olcDatabase={1}mdb,cn=config
    changetype: modify
    replace: olcRootPW
    olcRootPW: {SSHA}xiOeZFSc5hUpkNaQn6n0MEiYhYBfLjXe
```

8. `sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f rootpw.ldif` apply change

9. `vim base.ldif` :

```
    dn: dc=nasa,dc=csie,dc=ntu
    dc: nasa
    objectClass: top
    objectClass: domain

    dn: cn=admin,dc=nasa,dc=csie,dc=ntu
    cn: admin
    objectClass: organizationalRole
    description: admin account

    dn: ou=people,dc=nasa,dc=csie,dc=ntu
    ou: people
    objectClass: organizationalUnit

    dn: ou=group,dc=nasa,dc=csie,dc=ntu
    ou: group
    objectClass: organizationalUnit
```

10. `sudo ldapadd -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -H ldapi:/// -f base.ldif` 輸
    入 password
    但是出現錯誤
    ldap_bind: Invalid credentials (49)

11. `sudo dpkg-reconfigure slapd` 重新設定password: nasa2025 然後再做一次步驟 10，就可
    以了

12. `vim group.ldif` :

```
dn: cn=astro,ou=group,dc=nasa,dc=csie,dc=ntu
objectClass: posixGroup
cn: astro
gidNumber: 20000
```

13. `sudo ldapadd -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -f group.ldif` apply change

14. add astro1, astro2, and astro3:

- `vim astro1.ldif` :

  ```
  dn: uid=astro1,ou=people,dc=nasa,dc=csie,dc=ntu
  objectClass: inetOrgPerson
  objectClass: posixAccount
  objectClass: shadowAccount
  cn: astro1
  sn: astro1
  uid: astro1
  uidNumber: 20001
  gidNumber: 20000
  homeDirectory: /home/astro1
  loginShell: /bin/bash
  userPassword: {SSHA}xiOeZFSc5hUpkNaQn6n0MEiYhYBfLjXe
  ```

- `sudo ldapadd -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -f astro1.ldif` apply change

- `vim astro2.ldif` :

  ```
  dn: uid=astro1,ou=people,dc=nasa,dc=csie,dc=ntu
  objectClass: inetOrgPerson
  objectClass: posixAccount
  objectClass: shadowAccount
  cn: astro2
  sn: astro2
  uid: astro2
  uidNumber: 20002
  gidNumber: 20000
  homeDirectory: /home/astro2
  loginShell: /bin/bash
  userPassword: {SSHA}xiOeZFSc5hUpkNaQn6n0MEiYhYBfLjXe
  ```

- `sudo ldapadd -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -f astro2.ldif` apply change

- `vim astro3.ldif` :

  ```
  dn: uid=astro1,ou=people,dc=nasa,dc=csie,dc=ntu
  objectClass: inetOrgPerson
  objectClass: posixAccount
  objectClass: shadowAccount
  cn: astro3
  sn: astro3
  uid: astro3
  ```

```
        uidNumber: 20003
        gidNumber: 20000
        homeDirectory: /home/astro3
        loginShell: /bin/bash
        userPassword: {SSHA}xiOeZFSc5hUpkNaQn6n0MEiYhYBfLjXe
```

- `sudo ldapadd -x -D "cn=admin,dc=nasa,dc=csie,dc=ntu" -W -f astro3.ldif`
  apply change

On hpc1 and hpc2

15. `sudo apt install libnss-ldapd libpam-ldapd nslcd`

16. `sudo vim /etc/nslcd.conf` 改成以下內容：

   `uri ldap://192.150.9.19/ base dc=nasa,dc=csie,dc=ntu binddn cn=admin,dc=nasa,dc=csie,dc=ntu bindpw nasa2025`

17. `sudo vim /etc/nsswitch.conf` 改成以下內容：

   `passwd: files systemd ldap group: files systemd ldap shadow: files systemd ldap`

18. `sudo pam-auth-update` 選建立家目錄後按 Enter，但是後來發現登入 astro1 之後 create /home/astro1 失敗，所以 `sudo vim /etc/pam.d/common-session` 加入：

   `session required pam_mkhomedir.so skel=/etc/skel umask=0022`

19. `sudo systemctl restart nslcd` restart

```
inituser@hpc1:~$ su - astro1
Password:
Creating directory '/home/astro1'.
astro1@hpc1:~$ su - astro2
Password:
Creating directory '/home/astro2'.
astro2@hpc1:~$ su - astro3
Password:
Creating directory '/home/astro3'.
```

```
inituser@hpc2:~$ su - astro1
Password:
Creating directory '/home/astro1'.
astro1@hpc2:~$ su - astro2
Password:
Creating directory '/home/astro2'.
astro2@hpc2:~$ su - astro3
Password:
Creating directory '/home/astro3'.
```

On Server,

21. `cd /srv/nfs-share`

22. `sudo mkdir astro1_dir astro2_dir astro3_dir` create them

23. 設定權限：

`sudo chown 20001:20000 astro1_dir sudo chown 20002:20000 astro2_dir sudo chown 20003:20000 astro3_dir sudo chmod 700 astro1_dir sudo chmod 700 astro2_dir sudo chmod 700 astro3_dir`

24. `sudo vim /etc/exports` 改成以下：

`! /srv/nfs-share 192.150.9.20(rw,sync,no_subtree_check,root_squash) 192.150.9.21(rw,sync,no_subtree_check,root_squash)`

25. 更新：

`sudo exportfs -arv sudo systemctl restart nfs-server`

2. 繳交內容

- 見 1. 題目要求
- 截圖：

  1. Screenshot 1:

```
inituser@hpc1:~$ id astro1
uid=20001(astro1) gid=20000(astro) groups=20000(astro)
inituser@hpc1:~$ id astro2
uid=20002(astro2) gid=20000(astro) groups=20000(astro)
inituser@hpc1:~$ id astro3
uid=20003(astro3) gid=20000(astro) groups=20000(astro)
```

  2. Screenshot 2:

```
inituser@nfs-server:/srv/nfs-share$ ls -al
total 28
drwxr-xr-x 5 inituser inituser 4096 May  3 08:32 .
drwxr-xr-x 3 root     root     4096 May  3 05:57 ..
drwx------ 2    20001    20000 4096 May  3 08:32 astro1_dir
drwx------ 2    20002    20000 4096 May  3 08:32 astro2_dir
drwx------ 2    20003    20000 4096 May  3 08:32 astro3_dir
-rw-rw-r-- 1 inituser inituser   16 May  3 06:01 from_hpc1.txt
-rw-rw-r-- 1 inituser inituser   16 May  3 06:28 from_hpc2.txt
```

  3. Screenshot 3:

```
astro1@hpc1:/mnt/nfs-share/astro1_dir$ vim from_hpc1.txt
astro1@hpc1:/mnt/nfs-share/astro1_dir$ ls
from_hpc1.txt
astro1@hpc1:/mnt/nfs-share/astro1_dir$ cat from_hpc1.txt
Hello from hpc1
```

```
inituser@hpc1:/mnt/nfs-share$ ls
astro1_dir  astro2_dir  astro3_dir  from_hpc1.txt  from_hpc2.txt
inituser@hpc1:/mnt/nfs-share$ cd astro1_dir
-bash: cd: astro1_dir: Permission denied
inituser@hpc1:/mnt/nfs-share$ su - astro2
Password:
astro2@hpc1:~$ cd /mnt/nfs-share/
astro2@hpc1:/mnt/nfs-share$ ls
astro1_dir  astro2_dir  astro3_dir  from_hpc1.txt  from_hpc2.txt
astro2@hpc1:/mnt/nfs-share$ cd astro1_dir
-bash: cd: astro1_dir: Permission denied
astro2@hpc1:/mnt/nfs-share$ su - astro3
Password:
astro3@hpc1:~$ cd /mnt/nfs-share/
astro3@hpc1:/mnt/nfs-share$ ls
astro1_dir  astro2_dir  astro3_dir  from_hpc1.txt  from_hpc2.txt
astro3@hpc1:/mnt/nfs-share$ cd astro1_dir
-bash: cd: astro1_dir: Permission denied
astro3@hpc1:/mnt/nfs-share$
```

第三部分：效能與大規模檔案測試

ref: ,

1. NFS 參數 :

On Server

1. `sudo vim /etc/exports` :
   - for A and B:
     ```
     /srv/nfs-share 192.150.9.20(rw,sync,no_subtree_check,root_squash) 192.150.9
     ```
   - for C and D:
     ```
     /srv/nfs-share 192.150.9.20(rw,async,no_subtree_check,root_squash) 192.150
     ```

2. restart:
   ```
   sudo exportfs -arv
   sudo systemctl restart nfs-server
   ```

On hpc1 and hpc2

3. `sudo umount /mnt/nfs-share`
4. for A and C: `sudo mount -t nfs -o rsize=8192,wsize=8192 192.150.9.19:/srv/nfs-share /mnt/nfs-share`

   for B and D: `sudo mount -t nfs -o rsize=32768,wsize=32768 192.150.9.19:/srv/nfs-share /mnt/nfs-share`

2. Shell Script :

5. `sudo vim nfs_test_b11901164.sh` :
   ```bash
   #!/bin/bash

   # env
   MOUNT_DIR="/mnt/nfs-share"
   FILENAME="${USER}_dir/testfile_${USER}_$(hostname)"
   FILEPATH="$MOUNT_DIR/$FILENAME"

   # remove the existing file
   if [ -f "$FILEPATH" ]; then
     rm -f "$FILEPATH"
   fi

   # get the load before testing
   ```

```
LOAD_BEFORE=$(uptime | awk -F'load average: ' '{print $2}' | cut -d',' -f1)

echo "建立測試檔案：$FILEPATH"

# write 1GB test file
START_WRITE=$(date +%s.%N)
dd if=/dev/zero of=$FILEPATH bs=1M count=1024 oflag=direct status=none
END_WRITE=$(date +%s.%N)
WRITE_TIME=$(echo "$END_WRITE - $START_WRITE" | bc)
WRITE_SPEED=$(echo "scale=2; 1024 / $WRITE_TIME" | bc)

# Read
START_READ=$(date +%s.%N)
dd if=$FILEPATH of=/dev/null bs=1M iflag=direct status=none
END_READ=$(date +%s.%N)
READ_TIME=$(echo "$END_READ - $START_READ" | bc)
READ_SPEED=$(echo "scale=2; 1024 / $READ_TIME" | bc)

# avg speed
AVG_SPEED=$(echo "scale=2; ($WRITE_SPEED + $READ_SPEED) / 2" | bc)

# get the load after testing
LOAD_AFTER=$(uptime | awk -F'load average: ' '{print $2}' | cut -d',' -f1)

# output
echo "寫入時間：$WRITE_TIME s"
echo "寫入速率：$WRITE_SPEED MB/s"
echo "讀取時間：$READ_TIME s"
echo "讀取速率：$READ_SPEED MB/s"
echo "平均傳輸速率：$AVG_SPEED MB/s"
echo "CPU 使用率/負載：$LOAD_BEFORE → $LOAD_AFTER"
```

6. 如果是單一使用者則只要在hpc1執行 `nfs_test_b11901164.sh`，如果是多使用者則分別在 hpc1 與 hpc2 內的 astro1 ~ astro3 執行 `nfs_test_b11901164.sh`

```
sudo chmod +x nfs_test_b11901164.sh
./nfs_test_b11901164.sh
```

7. shell script 的運作流程以及所使用的工具：一開始先刪除舊的測試檔，再利用 dd 工具寫入 1GB 檔案至 /mnt/nfs-share，然再讀取該檔案，分別計算寫入與讀取的時間與速率。透過 uptime 擷取測試前後的 CPU 負載。

8. 表格：

測試情境	寫入時間 (s)	讀取時間 (s)	平均傳輸速率 (MB/s)	CPU 使用率/負載
組合 A + 單一使用者	189.82	13.13	41.67	1.88
組合 A + 多使用者	850.58	54.61	10.44	4.24
組合 B +	205.44	10.38	51.80	1.36

單一使用者				
組合 B + 多使用者	614.23	55.19	10.11	3.54
組合 C + 單一使用者	27.21	22.54	41.52	1.72
組合 C + 多使用者	139.40	66.22	11.18	4.62
組合 D + 單一使用者	15.58	17.85	61.53	0.62
組合 D + 多使用者	108.65	58.71	13.40	3.71

9. 問答題：

- 有，在多使用者同時讀寫的情況下效能明顯下降。以組合 A 為例，單一使用者平均傳輸速率為 41.67 MB/s，而多使用者情境下降至 10.44 MB/s，其他組合也呈現類似趨勢。

- 原因：

  - 多使用者同時存取導致網路擁塞與 NFS 鎖定（lock contention）
  - sync 模式會強制等待資料寫入磁碟，造成延遲更加明顯
  - rsize/wsize 值越小，封包數越多，傳輸效率更低
  - CPU 負載上升也影響 I/O 效能

- 解決辦法：使用 async 並適當增加 rsize/wsize（如組合 D），能顯著改善效能；若資料一致性不是關鍵，可優先考慮 async 模式，並使用更快的網路介面，如 10GbE 以減少瓶頸。