LSAP HW3 Report

1 LDAP Directory Service

(1) 安裝與初始設定

安裝 LDAP 伺服器與工具，並把 Base DN/組織與管理者密碼寫入初始資料庫。
Domain: im.ntu.edu.tw → Base DN: dc=im,dc=ntu,dc=edu,dc=tw
Organization: B11705053
設定 admin 密碼（用於日後 ldapadd/modify）

```
sudo slapcat | grep -A5 'dn: dc=im,dc=ntu,dc=edu,dc=tw'
```

Figure 1: 初始化設定

(2) 建立 OU（People / Groups）

建立常見的兩個組織單位，讓人與群組有清楚的容器。

```
cat > ou_base.ldif <<'LDIF'
dn: ou=People,dc=im,dc=ntu,dc=edu,dc=tw
objectClass: organizationalUnit
ou: People

dn: ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
objectClass: organizationalUnit
ou: Groups
LDIF

sudo ldapadd -x -D "cn=admin,dc=im,dc=ntu,dc=edu,dc=tw" -W -f ou_base.ldif
```

用 LDIF 描述兩個條目，再用 ldapadd 以 admin 身分匯入。

```
ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(objectClass=organizationalUnit)" ou
```

```
classuser@vm01:~$ ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(objectClass=organizationalUnit)" ou
# extended LDIF
#
# LDAPv3
# base <dc=im,dc=ntu,dc=edu,dc=tw> with scope subtree
# filter: (objectClass=organizationalUnit)
# requesting: ou
#

# People, im.ntu.edu.tw
dn: ou=People,dc=im,dc=ntu,dc=edu,dc=tw
ou: People

# Groups, im.ntu.edu.tw
dn: ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
ou: Groups

# search result
search: 2
result: 0 Success

# numResponses: 3
# numEntries: 2
```

Figure 2: 建立 OU

(3) 建立群組（eng / intern）

建立 UNIX 風格的 POSIX 群組，後面使用者可用 gidNumber 對應。

```
cat > groups.ldif <<'LDIF'
dn: cn=eng,ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
objectClass: posixGroup
cn: eng
gidNumber: 5100

dn: cn=intern,ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
objectClass: posixGroup
cn: intern
gidNumber: 5101
LDIF

sudo ldapadd -x -D "cn=admin,dc=im,dc=ntu,dc=edu,dc=tw" -W -f groups.ldif
```

建兩個群組條目，分別給工程/實習用。

```
ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(cn=eng)" cn gidNumber
ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(cn=intern)" cn gidNumber
```

```
# eng, Groups, im.ntu.edu.tw
dn: cn=eng,ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
cn: eng
gidNumber: 5100
```

(a) 建立工程群組

```
# intern, Groups, im.ntu.edu.tw
dn: cn=intern,ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
cn: intern
gidNumber: 5101
```

(b) 建立實習群組

(4) 建立三個使用者並加入群組

1. 依題目建立 3 個帳號（first/last/student-id），並綁定到對應群組。
2. 產生密碼雜湊（SSHA）：slappasswd
3. 建立使用者條目（uidNumber/gidNumber/家目錄/登入殼等）：
4. 把使用者加入群組（memberUid）：可看 Figure 5

```
cat > group_members.ldif <<'LDIF'
```

```
dn: cn=eng,ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
changetype: modify
add: memberUid
memberUid: ting-yu
-
add: memberUid
memberUid: chen

dn: cn=intern,ou=Groups,dc=im,dc=ntu,dc=edu,dc=tw
changetype: modify
add: memberUid
memberUid: b11705053
LDIF

sudo ldapmodify -x -D "cn=admin,dc=im,dc=ntu,dc=edu,dc=tw" -W -f group_members.ldif
```

```
ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(uid=ting-yu)" uid uidNumber gidNumber
ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(uid=chen)" uid uidNumber gidNumber
ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(uid=b11705053)" uid uidNumber gidNumber

ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(cn=eng)" cn gidNumber memberUid
ldapsearch -x -b "dc=im,dc=ntu,dc=edu,dc=tw" "(cn=intern)" cn gidNumber memberUid
```

(a) 群組對應 (b) 群組對應 (c) 群組對應

(d) 群組對應 (e) 群組對應

Figure 4: 群組對應（兩行排版）

(5) Generate a Certificate Authority (CA)

我們用自己的 CA 來簽發 LDAP 伺服器憑證，並把這張 CA 的「公鑰憑證」加入系統信任，讓 ldap 工具能信任你後面簽出來的 server.crt。

```
# 建目錄
sudo mkdir -p /etc/ssl/ldap
cd /etc/ssl/ldap
# 5.1 產生 CA 私鑰（4096 bits）
sudo openssl genrsa -out ca.key 4096
sudo chmod 600 ca.key
# 5.2 產生 CA 憑證（有效 10 年，可調整 -days）
sudo openssl req -x509 -new -sha256 -days 3650 \
  -key ca.key -out ca.crt \
  -subj "/C=TW/ST=Taiwan/L=Taipei/O=B11705053/OU=LSAP/CN=LSAP-HW3-CA"
# 5.3 將 CA 加入這台機器的信任庫（給 ldap* 客戶端用）
```

```
sudo cp /etc/ssl/ldap/ca.crt /usr/local/share/ca-certificates/ldap_ca.crt
sudo update-ca-certificates
```

```
classuser@vm01:/etc/ssl/ldap$ openssl x509 -in /usr/local/share/ca-certificates/ldap_ca.crt \
  -noout -subject -issuer -dates -serial -fingerprint -sha256
subject=C = TW, ST = Taiwan, L = Taipei, O = B11705053, OU = LSAP, CN = LSAP-HW3-CA
issuer=C = TW, ST = Taiwan, L = Taipei, O = B11705053, OU = LSAP, CN = LSAP-HW3-CA
notBefore=Oct 26 16:05:49 2025 GMT
notAfter=Oct 24 16:05:49 2035 GMT
serial=409B9B5E451512E1BF029E9B618E722E96D12F6D
sha256 Fingerprint=9E:02:DC:2C:3C:9A:A7:11:18:1E:54:A4:12:2F:D1:03:26:15:A4:3B:CC:93:41:83:FF:5E:7A:26:BC:CA:73:EC
```

Figure 5: ca.crt

(6) Generate Server Key & CSR

建 server.cnf（要含 SAN），生私鑰 server.key 並設權限

```
# 6.0 OpenSSL req 設定（SAN 內含 FQDN，IP 直連再開 IP.1）
sudo tee /etc/ssl/ldap/server.cnf >/dev/null <<'CONF'
[ req ]
default_bits = 2048
prompt = no
default_md = sha256
req_extensions = v3_req
distinguished_name = dn
[ dn ]
C = TW
ST = Taiwan
L = Taipei
O = IM Corp
OU = LSAP
CN = __FQDN__
[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = digitalSignature, keyEncipherment
extendedKeyUsage = serverAuth
subjectAltName = @alt_names
[ alt_names ]
DNS.1 = __FQDN__
DNS.2 = localhost
# IP.1 = __IPADDR__
CONF
# 6.1 套入變數
sudo sed -i "s/__FQDN__/${FQDN}/g" /etc/ssl/ldap/server.cnf
if [ -n "${IPADDR}" ]; then
  sudo sed -i "s/# IP.1/ IP.1/" /etc/ssl/ldap/server.cnf
  sudo sed -i "s/__IPADDR__/${IPADDR}/" /etc/ssl/ldap/server.cnf
fi
# 6.2 伺服器私鑰 + 權限（slapd 需可讀）
sudo openssl genrsa -out /etc/ssl/ldap/server.key 2048
sudo chgrp openldap /etc/ssl/ldap/server.key
sudo chmod 640 /etc/ssl/ldap/server.key
# 6.3 產生 CSR（帶 v3_req/SAN）
sudo openssl req -new -key /etc/ssl/ldap/server.key \
  -out /etc/ssl/ldap/server.csr \
  -config /etc/ssl/ldap/server.cnf -reqexts v3_req
sudo chmod 644 /etc/ssl/ldap/server.csr
```

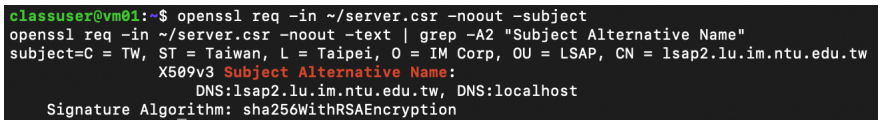

Figure 6: server.csr

(7) Sign the Server Certificate

準備簽章用的 openssl-ext.cnf（寫清楚 SAN/EKU），用 (5) 產生的 ca.key / ca.crt 對 (6) 的 server.csr 簽發 server.crt，設正確權限，並用 openssl verify 與 -text 檢查 Issuer/SAN/EKU。

```
# 7.0 建簽章 extensions：SAN / EKU / SKI / AKI
sudo tee /etc/ssl/ldap/openssl-ext.cnf >/dev/null <<'EXT'
basicConstraints=CA:FALSE
keyUsage=digitalSignature, keyEncipherment
extendedKeyUsage=serverAuth
subjectAltName=@alt_names
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer
[alt_names]
DNS.1=lsap2.lu.im.ntu.edu.tw
DNS.2=localhost
# IP.1=你的IP（若會用 IP 直連才填）
EXT
# 7.1 用 CA 簽 CSR → 產生 server.crt（1 年）
sudo openssl x509 -req -in /etc/ssl/ldap/server.csr \
  -CA /etc/ssl/ldap/ca.crt -CAkey /etc/ssl/ldap/ca.key -CAcreateserial \
  -out /etc/ssl/ldap/server.crt -days 365 -sha256 \
  -extfile /etc/ssl/ldap/openssl-ext.cnf
# 權限
sudo chmod 644 /etc/ssl/ldap/server.crt
```

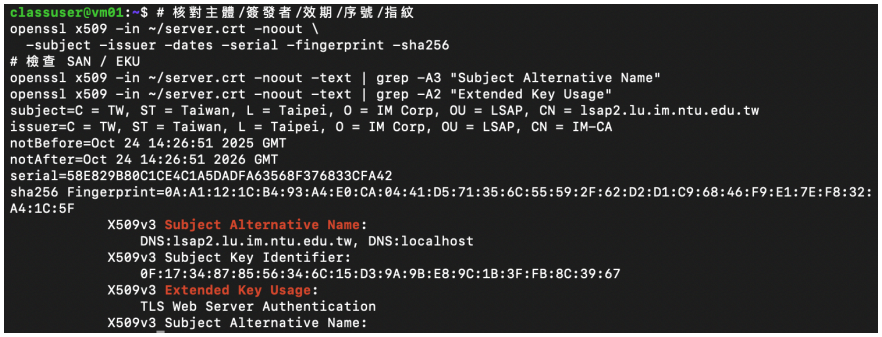

Figure 7: server.crt

(8) Enable LDAPS and Trust Your CA Locally

讓 slapd 使用自簽 CA 與伺服器憑證，開啟 ldaps:///（TCP 636），並在客戶端信任 CA，使 LDAP 工具能以 TLS 成功查詢。

```
classuser@vm01:~$ ldapsearch -x -H "ldaps://lsap2.lu.im.ntu.edu.tw:636" \
  -b "dc=im,dc=ntu,dc=edu,dc=tw" -s base namingContexts
# extended LDIF
#
# LDAPv3
# base <dc=im,dc=ntu,dc=edu,dc=tw> with scope baseObject
# filter: (objectclass=*)
# requesting: namingContexts
#

# im.ntu.edu.tw
dn: dc=im,dc=ntu,dc=edu,dc=tw

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Figure 8: TLS 成功查詢

(9) Apache Directory Studio（ADS）連線與截圖

以圖形化工具驗證 LDAPS 與樹狀內容，提供作業需要的截圖。

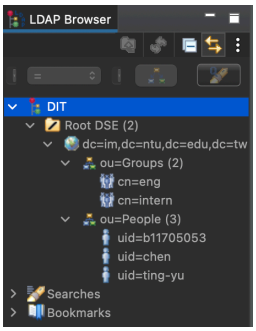

Figure 9: LDAPS 與樹狀內容

2 Custom APT Repository

(1) 打包與本機驗證（.deb）

先確定套件可安裝與可執行。

```
ls -l ~/b11705053-statistics_1.0.0_amd64.deb
sudo dpkg -i ~/b11705053-statistics_1.0.0_amd64.deb || sudo apt -f install -y
dpkg -s b11705053-statistics | grep -E 'Status|Version'
which b11705053-statistics
b11705053-statistics -h
```

```
classuser@vm01:~$ b11705053-statistics -h
Usage:
  b11705053-statistics --fit <train.csv> --out <model.json>
    train.csv rows: y,x1,x2,...,xp  (headerless, numeric)

  b11705053-statistics --apply <model.json> --in <X.csv> --out <pred.txt>
    X.csv rows: x1,x2,...,xp (headerless, numeric)
    pred.txt: one prediction per line

  b11705053-statistics -h | --help
Notes:
  * CSV must be numeric. Whitespace allowed around commas.
  * Clear errors with line/column on parse failure.
```

Figure 10: 幫助訊息

(2) 建立倉庫目錄與放入套件

建出 APT 標準結構，將 .deb 置於 pool/。

```
sudo install -d /var/www/html/apt/{pool/main,dists/stable/main/binary-amd64}
sudo cp ~/b11705053-statistics_1.0.0_amd64.deb /var/www/html/apt/pool/main/
```

```
classuser@vm01:~$ cd ~
echo -e "11,1,2\n13,2,3\n17,3,5\n23,4,8" > train.csv
echo -e "5,8\n6,9" > X.csv
b11705053-statistics --fit train.csv --out model.json
b11705053-statistics --apply model.json --in X.csv --out pred.txt
cat pred.txt
Model saved to model.json with 2 features.
Predictions written to pred.txt (2 lines)
23.0000000000
25.0000000000
```

Figure 11: 確認輸出

(3) 產生索引與 Release

用 dpkg-scanpackages 生成 Packages，再以 apt-ftparchive 生成 Release。

```
cd /var/www/html/apt
sudo bash -c 'dpkg-scanpackages --multiversion pool > dists/stable/main/binary-amd64/
    Packages'
sudo gzip -kf dists/stable/main/binary-amd64/Packages

sudo tee apt-ftparchive.conf >/dev/null <<'CONF'
APT::FTPArchive::Release {
  Origin "b11705053";
  Label "b11705053 APT";
  Suite "stable";
  Codename "stable";
  Architectures "amd64";
  Components "main";
  Description "Custom repo for b11705053-statistics";
};
CONF

sudo bash -c 'apt-ftparchive -c apt-ftparchive.conf release dists/stable > dists/stable/
    Release'
```

```
classuser@vm01:~$ dpkg -c ~/b11705053-statistics_1.0.0_amd64.deb | grep -E '/usr
/bin/b11705053-statistics$|/usr/share/man/man1/.*\.gz$'
-rwxr-xr-x root/root     39168 2025-10-25 04:00 ./usr/bin/b11705053-statistics
-rw-r--r-- root/root       541 2025-10-25 04:00 ./usr/share/man/man1/b11705053-s
tatistics.1.gz
```

Figure 12: deb 內含正確檔案

(4) GPG 金鑰與簽署

以私鑰簽出 InRelease/Release.gpg，並匯出公鑰供客戶端信任。

```
gpg --quick-generate-key "b11705053 APT Repo <you@example.com>" default default never
gpg --list-keys "b11705053 APT Repo <you@example.com>"

gpg --default-key "b11705053 APT Repo <you@example.com>" --clearsign \
  -o /tmp/InRelease /var/www/html/apt/dists/stable/Release
gpg --default-key "b11705053 APT Repo <you@example.com>" -abs \
  -o /tmp/Release.gpg /var/www/html/apt/dists/stable/Release
sudo mv /tmp/InRelease /var/www/html/apt/dists/stable/InRelease
sudo mv /tmp/Release.gpg /var/www/html/apt/dists/stable/Release.gpg

gpg --export -a "b11705053 APT Repo <you@example.com>" \
 | sudo tee /var/www/html/apt/b11705053-archive-keyring.gpg.asc >/dev/null
```

(5) 以 Apache 提供 HTTP

讓倉庫經由 HTTP 對外可取用（你使用 Apache，而非臨時 127.0.0.1:8000）。

```
sudo apt install -y apache2
sudo systemctl enable --now apache2

curl -I http://<IP>/apt/dists/stable/InRelease
curl -I http://<IP>/apt/dists/stable/main/binary-amd64/Packages.gz
```

(6) 客戶端加入來源並安裝

客戶端 dearmor 匯入公鑰、加入 source、更新索引並安裝。

```
sudo mkdir -p /usr/share/keyrings
curl -fsSL http://<IP>/apt/b11705053-archive-keyring.gpg.asc \
 | sudo gpg --dearmor -o /usr/share/keyrings/b11705053-archive-keyring.gpg
sudo chmod 644 /usr/share/keyrings/b11705053-archive-keyring.gpg

echo "deb [arch=amd64 signed-by=/usr/share/keyrings/b11705053-archive-keyring.gpg] \
http://<IP>/apt stable main" | sudo tee /etc/apt/sources.list.d/b11705053.list

sudo apt update
apt-cache policy b11705053-statistics | sed -n '1,12p'
sudo apt install -y b11705053-statistics
```