NASA HW8

B11901164 陳秉緯

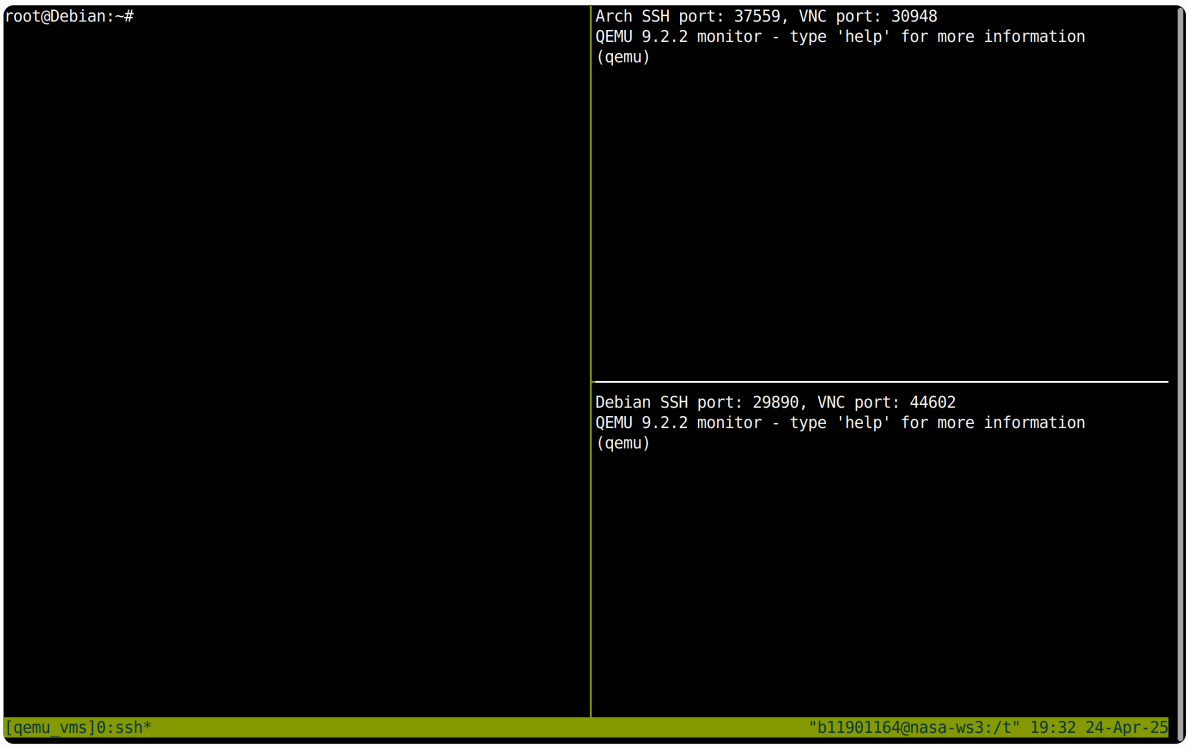
討論：R13941146 李毓庭, B12901194 賴睿廷

I. 前置作業

1. Server

ref: https://docs.google.com/presentation/d/1QOBSuBnh2F55daXRpcfpHbN-fNiUS3Hz2edsyFqzFQQ/edit?usp=sharing

1. `/tmp2/hw8/run-vm.sh b11901194` 生成此題所需要的虛擬機環境



2. `ssh -p 29890 root@local` 輸入 password: nasa2025 進入 Debian
3. `apt install -y slapd ldap-utils` install OpenLDAP and Utilities , 輸入 password
4. `nano suffix.ldif`：

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

5. `ldapmodify -Y EXTERNAL -H ldapi:/// -f suffix.ldif` apply modification

6. `nano rootdn.ldif`:

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

7. `ldapmodify -Y EXTERNAL -H ldapi:/// -f rootdn.ldif` apply modification
8. `nano 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
```

9. `ldapadd -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -H ldapi:/// -f base.ldif` apply modification
10. `slappasswd` hash the password and copy the hash result

    `{SSHA}2s1hybgIa5Rss7SG2MbQ+Gb/SJqtX/hL`
11. `nano rootpw.ldif`:

```
dn: olcDatabase={1}mdb,cn=config
changetype: modify
replace: olcRootPW
olcRootPW: {SSHA}2s1hybgIa5Rss7SG2MbQ+Gb/SJqtX/hL
```

12. `ldapmodify -Y EXTERNAL -H ldapi:/// -f rootpw.ldif` apply modification

13. 在 Server 執行 `ldapsearch -x -b dc=nasa,dc=csie,dc=ntu`

```
root@Debian:~# ldapsearch -x -b dc=nasa,dc=csie,dc=ntu
# extended LDIF
#
# LDAPv3
# base <dc=nasa,dc=csie,dc=ntu> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# nasa.csie.ntu
dn: dc=nasa,dc=csie,dc=ntu
dc: nasa
objectClass: top
objectClass: domain

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

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

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

# search result
search: 2
result: 0 Success

# numResponses: 5
# numEntries: 4
root@Debian:~#
```

2. Client

ref: https://wiki.archlinux.org/title/OpenLDAP

1. `ssh -p 37559 root@local` 輸入 password: nasa2025 進入 Arch
2. install openldap:

```
pacman -Syu
pacman -S openldap
```

3. 在 Debian `ip a` 得知 ip: `192.168.167.70`
4. 在 Arch 上 `ldapsearch -x -H ldap://192.168.167.70 -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -b dc=nasa,dc=csie,dc=ntu`

```
[root@Arch ~]# ldapsearch -x -H ldap://192.168.167.70 -D "cn=admin,dc=nasa,dc=csie,dc=ntu"
 -W -b "dc=nasa,dc=csie,dc=ntu"
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=nasa,dc=csie,dc=ntu> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# nasa.csie.ntu
dn: dc=nasa,dc=csie,dc=ntu
dc: nasa
objectClass: top
objectClass: domain

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

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

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

# search result
search: 2
result: 0 Success

# numResponses: 5
# numEntries: 4
```

II. Task

1. LDAPS (LDAP over SSL)

ref: https://medium.com/munchy-bytes/configuring-openldap-with-ssl-for-secure-directory-access-on-ubuntu-22-04-90e877dcdbb9,

https://askubuntu.com/questions/936382/openldap-error-configuring-starttls-ldap-modify-other-e-g-implementation-sp

1.
```
mkdir /etc/ldap/certs
cd /etc/ldap/certs
```

2. `openssl req -x509 -new -nodes -days 365 -keyout ldap-key.pem -out ldap-cert.pem -subj "/CN=Debian"` 產生金鑰
3. 更改權限：
```
chown openldap:openldap /etc/ldap/certs/*.pem
chmod 600 key.pem
```

4. 把 `ldap-cert.pem` 移到 `ca-certificates` 內然後 update：
```
cp /etc/ldap/certs/ldap-cert.pem /usr/local/share/ca-certificates/ldap-cert.crt
update-ca-certificates
```

5. `nano /etc/ldap/certs/tls.ldif`：
```
dn: cn=config
changetype: modify
replace: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/ldap/certs/ldap-key.pem
-
replace: olcTLSCertificateFile
olcTLSCertificateFile: /etc/ldap/certs/ldap-cert.pem
```

6. `ldapmodify -Y EXTERNAL -H ldapi:/// -f tls.ldif` apply modification
7. `nano /etc/default/slapd`：
```
SLAPD_SERVICES="ldap:/// ldapi:/// ldaps:///"
```

8. `systemctl restart slapd` 重啟 slapd

9. 截圖：

```
root@Debian:/etc/ldap/certs# ldapsearch -x -ZZ -b dc=nasa,dc=csie,dc=ntu
# extended LDIF
#
# LDAPv3
# base <dc=nasa,dc=csie,dc=ntu> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# nasa.csie.ntu
dn: dc=nasa,dc=csie,dc=ntu
dc: nasa
objectClass: top
objectClass: domain

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

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

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

# search result
search: 3
result: 0 Success

# numResponses: 5
# numEntries: 4
```

```
root@Debian:/etc/ldap/certs# ldapsearch -x -H ldaps:/// -b dc=nasa,dc=csie,dc=ntu
# extended LDIF
#
# LDAPv3
# base <dc=nasa,dc=csie,dc=ntu> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# nasa.csie.ntu
dn: dc=nasa,dc=csie,dc=ntu
dc: nasa
objectClass: top
objectClass: domain

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

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

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

# search result
search: 2
result: 0 Success

# numResponses: 5
# numEntries: 4
```

10. `nano another-tls.ldif`：

```
dn: cn=config
changetype: modify
replace: olcSecurity
olcSecurity: tls=1
```

11. `ldapmodify -Y EXTERNAL -H ldapi:/// -f another-tls.ldif` apply modification
12. 在 Arch 內：`scp root@192.168.167.70:/etc/ldap/certs/ldap-cert.pem /etc/ssl/certs/server.pem` 複製 `ldap-cert.pem` 過來
13. `vim /etc/hosts` 加入：

```
192.168.167.70 Debian
```

14. 讓 ca 可以被 Arch 信任：

```
cp /etc/ssl/certs/server.pem /etc/ca-certificates/trust-
source/anchors/server.pem
trust extract-compat
```

15. 用有加密的方法：

```
[root@Arch certs]# ldapsearch -x -ZZ -H ldap://Debian -b dc=nasa,dc=csie,dc=ntu
# extended LDIF
#
# LDAPv3
# base <dc=nasa,dc=csie,dc=ntu> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# nasa.csie.ntu
dn: dc=nasa,dc=csie,dc=ntu
dc: nasa
objectClass: top
objectClass: domain

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

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

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

# search result
search: 3
result: 0 Success

# numResponses: 5
# numEntries: 4
```

用未加密的方法：

```
[root@Arch certs]# ldapsearch -x -H ldap://Debian -b dc=nasa,dc=csie,dc=ntu
ldap_bind: Confidentiality required (13)
        additional info: TLS confidentiality required
```

2. SSSD

ref: https://docs.google.com/presentation/d/1QOBSuBnh2F55daXRpcfpHbN-fNiUS3Hz2edsyFqzFQQ/edit?usp=sharing, https://wiki.archlinux.org/title/LDAP_authentication, https://wiki.archlinux.org/title/Sudo#Example_entries, https://wiki.archlinux.org/title/SSSD

1. 在 Debian 內 `nano groups.ldif`：

```
dn: cn=ta,ou=group,dc=nasa,dc=csie,dc=ntu
objectClass: posixGroup
cn: ta
gidNumber: 10000
```

```
dn: cn=student,ou=group,dc=nasa,dc=csie,dc=ntu
objectClass: posixGroup
cn: student
gidNumber: 10001
```

新增兩個群組 ta 及 student

2. `ldapadd -x -ZZ -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -f groups.ldif` apply modification
3. `slappasswd` 輸入 password: nasa2025，得到 `{SSHA}1fYaRDHmpvBsGWziAxUAkvJOa5ZDhfll`
4. `nano user.ldif`：

```
dn: uid=ta,ou=people,dc=nasa,dc=csie,dc=ntu
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: ta
sn: user
cn: ta
uidNumber: 20000
gidNumber: 10000
homeDirectory: /home/ta
loginShell: /bin/bash
userPassword: {SSHA}1fYaRDHmpvBsGWziAxUAkvJOa5ZDhfll

dn: uid=student,ou=people,dc=nasa,dc=csie,dc=ntu
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: student
sn: user
cn: student
uidNumber: 20001
gidNumber: 10001
homeDirectory: /home/student
loginShell: /bin/bash
userPassword: {SSHA}1fYaRDHmpvBsGWziAxUAkvJOa5ZDhfll
```

添加兩個新的使用者，一個在 ta 群組，一個在 student 群組

5. `ldapadd -x -ZZ -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -f user.ldif` apply modification
6. 在 Arch 內：`pacman -S sssd sudo`
7. `vim /etc/sssd/sssd.conf`：

```
[sssd]
services = nss, pam, sudo, ssh
config_file_version = 2
domains = ldap

[domain/ldap]
```

```
id_provider = ldap
auth_provider = ldap
ldap_uri = ldaps://Debian
ldap_search_base = dc=nasa,dc=csie,dc=ntu
ldap_tls_cacert = /etc/ssl/certs/ca-certificates.crt
enumerate = true
```

設定 sssd

8. `chmod 600 /etc/sssd/sssd.conf` 更改權限
9. `systemctl enable --now sssd` 啟動 sssd
10. `vim /etc/pam.d/system-login`：

```
session    required   pam_mkhomedir.so skel=/etc/skel umask=0077
```

SSH 初次登入自動新增家目錄

11. `vim /etc/sudoers.d/ta`：

```
%ta ALL=(ALL:ALL) ALL
```

讓 ta group 的使用者有 sudo 的權限

12. `vim /etc/nsswitch.conf`：

```
passwd: files sss systemd
group: files sss [SUCCESS=merge] systemd
shadow: files sss systemd
```

13. `vim /etc/pam.d/system-auth` 加入：

```
auth sufficient pam_sss.so forward_pass
account [default=bad success=ok user_unknown=ignore authinfo_unavail=ignore]
pam_sss.so
password sufficient pam_sss.so
session     required      pam_mkhomedir.so skel=/etc/skel/ umask=0077
session optional pam_sss.so
```

```
#%PAM-1.0

auth       required                    pam_faillock.so      preauth
# Optionally use requisite above if you do not want to prompt for the password
# on locked accounts.
-auth      [success=2 default=ignore]  pam_systemd_home.so
auth       sufficient                  pam_sss.so           forward_pass
auth       [success=1 default=bad]     pam_unix.so          try_first_pass nullok
auth       [default=die]               pam_faillock.so      authfail
auth       optional                    pam_permit.so
auth       required                    pam_env.so
auth       required                    pam_faillock.so      authsucc
# If you drop the above call to pam_faillock.so the lock will be done also
# on non-consecutive authentication failures.

-account   [success=1 default=ignore]  pam_systemd_home.so
account    [default=bad success=ok user_unknown=ignore authinfo_unavail=ignore] pam_sss.so
account    required                    pam_unix.so
account    optional                    pam_permit.so
account    required                    pam_time.so

-password  [success=1 default=ignore]  pam_systemd_home.so
password   sufficient                  pam_sss.so
password   required                    pam_unix.so          try_first_pass nullok shadow
password   optional                    pam_permit.so

-session   optional                    pam_systemd_home.so
session    required                    pam_mkhomedir.so skel=/etc/skel/ umask=0077
session    required                    pam_limits.so
session    required                    pam_unix.so
session    optional                    pam_sss.so
session    optional                    pam_permit.so
```

14. `systemctl restart sssd` 重啟 sssd
15. 在 Arch `ip a` 得知 ip: `192.168.167.69`
16. `ssh ta@192.168.167.69`

```
[root@Arch sudoers.d]# ssh ta@192.168.167.69
The authenticity of host '192.168.167.69 (192.168.167.69)' can't be established.
ED25519 key fingerprint is SHA256:ze6F2uaaKhEQYXuSL+7GChICTBOUv0epU5bXRdyywWc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.167.69' (ED25519) to the list of known hosts.
ta@192.168.167.69's password:
Creating directory '/home/ta'.
[ta@Arch ~]$ sudo echo Hello World

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

For security reasons, the password you type will not be visible.

[sudo] password for ta:
Hello World
```

17 `ssh student@192.168.167.69`

```
[root@Arch sudoers.d]# ssh student@192.168.167.69
student@192.168.167.69's password:
Creating directory '/home/student'.
[student@Arch ~]$ sudo echo Hello World

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

For security reasons, the password you type will not be visible.

[sudo] password for student:
student is not in the sudoers file.
```

3. ACL (Access Control Lists)

ref: https://documentation.ubuntu.com/server/how-to/openldap/access-control/index.html#getting-the-acls, https://serverfault.com/questions/962747/why-does-anonymous-user-could-access-userpassword-attribute-of-openldap

1. 在 Debian：`nano /etc/ldap/slapd.d/cn=config.ldif` 把 `tls=1` 改成 `tls=0`
2. `systemctl restart slapd` 重啟 slapd
3. `nano conf_rootpw.ldif`：

```
dn: olcDatabase={0}config,cn=config
changetype: modify
replace: olcRootPW
olcRootPW: {SSHA}2s1hybgIa5Rss7SG2MbQ+Gb/SJqtX/hL
```

4. `ldapmodify -Y EXTERNAL -H ldapi:/// -f conf_rootpw.ldif` apply modification
5. `nano /etc/ldap/slapd.d/cn=config.ldif` 把 `tls=0` 改成 `tls=1`
6. `systemctl restart slapd` 重啟 slapd
7. `ldapsearch -x -LLL -D cn=admin,cn=config -W -H ldaps://localhost -b cn=config '(olcDatabase={1}mdb)' olcAccess`

```
root@Debian:~# ldapsearch -x -LLL -D cn=admin,cn=config -W -H ldaps://localhost -b cn=config '(olcDatabase={1}mdb)' olcAccess
Enter LDAP Password:
dn: olcDatabase={1}mdb,cn=config
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by * none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read
```

8. `nano acl.ldif`：

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

```
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by *
none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to attrs=cn,uid,uidNumber,gidNumber,homeDirectory by self read
by users read by anonymous none
olcAccess: {3}to * by * read
```

- {0}: 允許使用者修改自己的 userPassword;允許匿名使用者進行身份驗證，但無法讀取密碼;禁止所有其他使用者存取 userPassword
- {1}: 允許使用者修改自己的 shadowLastChange;允許所有使用者讀取該屬性
- {2}: 允許使用者讀取自己的cn, uid, uidNumber, gidNumber, homeDirectory;允許已驗證的使用者讀取這些屬性;禁止匿名使用者存取這些屬性
- {3}: 允許所有使用者讀取其他未明確指定的屬性

9. `ldapmodify -x -D cn=admin,cn=config -W -H ldaps://Debian -f acl.ldif` apply modification

4. LDAP Schema Extension

ref: https://www.openldap.org/doc/admin22/schema.html, https://www.openldap.org/doc/admin24/schema.html, https://guillaumemaka.com/2013/07/17/openldap-create-a-custom-ldap-schema/, https://stackoverflow.com/questions/45511696/creating-a-new-objectclass-and-attribute-in-openldap

1. `nano schema.ldif`:

```
dn: cn=student,cn=schema,cn=config
objectClass: olcSchemaConfig
cn: student
```

2. `ldapadd -x -D cn=admin,cn=config -W -H ldaps://localhost -f schema.ldif` apply modification

3. `ldapsearch -x -D cn=admin,cn=config -W -H ldaps://localhost -b cn=schema,cn=config '(objectClass=olcSchemaConfig)' cn` 查詢 student 編號

```
root@Debian:~# ldapsearch -x -D cn=admin,cn=config -W -H ldaps://localhost -b cn=schema,cn=config '(objectClass=olcSchemaConfig)' cn
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <cn=schema,cn=config> with scope subtree
# filter: (objectClass=olcSchemaConfig)
# requesting: cn
#

# schema, config
dn: cn=schema,cn=config
cn: schema

# {0}core, schema, config
dn: cn={0}core,cn=schema,cn=config
cn: {0}core

# {1}cosine, schema, config
dn: cn={1}cosine,cn=schema,cn=config
cn: {1}cosine

# {2}nis, schema, config
dn: cn={2}nis,cn=schema,cn=config
cn: {2}nis

# {3}inetorgperson, schema, config
dn: cn={3}inetorgperson,cn=schema,cn=config
cn: {3}inetorgperson

# {4}student, schema, config
dn: cn={4}student,cn=schema,cn=config
cn: {4}student

# search result
search: 2
result: 0 Success

# numResponses: 7
# numEntries: 6
```

4. `nano student.ldif`：

```
dn: cn={4}student,cn=schema,cn=config
changetype: modify
add: olcAttributeTypes
olcAttributeTypes: {0}( 1.3.6.1.4.1.4203.666.1.90 NAME 'studentName' DESC
'Name of the student' EQUALITY caseIgnoreMatch SYNTAX
1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
-
add: olcAttributeTypes
olcAttributeTypes: {1}( 1.3.6.1.4.1.4203.666.1.91 NAME 'examGroupID' DESC
'Exam group id' EQUALITY integerMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )
-
add: olcObjectClasses
olcObjectClasses: {0}( 1.3.6.1.4.1.4203.666.2.90 NAME 'StudentInformation'
DESC 'Student information with name and exam group' SUP inetOrgPerson
STRUCTURAL MUST ( studentName $ examGroupID ) )
```

定義兩個新的屬性: studentName、examGroupID，一個新的 objectClass: StudentInformation

5. `nano /etc/ldap/slapd.d/cn=config.ldif` 把 `tls=1` 改成 `tls=0`

6. `systemctl restart slapd` 重啟 slapd

7. `ldapmodify -Y EXTERNAL -H ldapi:/// -f student.ldif` apply modification

8. `ldapsearch -Y EXTERNAL -H ldapi:/// -b cn={4}student,cn=schema,cn=config` 確認 schema 是否匯入成功

```
root@Debian:~# ldapsearch -Y EXTERNAL -H ldapi:/// -b cn={4}student,cn=schema,cn=config
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
# extended LDIF
#
# LDAPv3
# base <cn={4}student,cn=schema,cn=config> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# {4}student, schema, config
dn: cn={4}student,cn=schema,cn=config
objectClass: olcSchemaConfig
cn: {4}student
olcAttributeTypes: {0}( 1.3.6.1.4.1.4203.666.1.90 NAME 'studentName' DESC 'Nam
 e of the student' EQUALITY caseIgnoreMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.
 15 SINGLE-VALUE )
olcAttributeTypes: {1}( 1.3.6.1.4.1.4203.666.1.91 NAME 'examGroupID' DESC 'Exa
 m group id' EQUALITY integerMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE
 -VALUE )
olcObjectClasses: {0}( 1.3.6.1.4.1.4203.666.2.90 NAME 'StudentInformation' DES
 C 'Student information with name and exam group' SUP inetOrgPerson STRUCTURAL
  MUST ( studentName $ examGroupID ) )

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

9. `nano student_one.ldif`：

```
dn: uid=student_one,ou=people,dc=nasa,dc=csie,dc=ntu
objectClass: posixAccount
objectClass: StudentInformation
uid: student_one
sn: student
cn: student_one
uidNumber: 30000
gidNumber: 10001
homeDirectory: /home/student_one
loginShell: /bin/bash
studentName: student_one
examGroupID: 0
```

新增一個 student 使用者叫 student_one

10. `nano group.ldif`：

```
dn: cn=students,ou=group,dc=nasa,dc=csie,dc=ntu
objectClass: groupOfNames
```

```
cn: students
member: uid=student_one,ou=people,dc=nasa,dc=csie,dc=ntu
```

新增一個 group，member 有 student_one

11. `ldapadd -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -H ldaps://localhost -f group.ldif` apply modification

12. `ldapsearch -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -H ldaps://localhost -b cn=students,ou=group,dc=nasa,dc=csie,dc=ntu` 查詢 group

```
root@Debian:~# ldapsearch -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -H ldaps://localhost -b cn=students,ou=group,dc=nasa,dc=csie,dc=ntu
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <cn=students,ou=group,dc=nasa,dc=csie,dc=ntu> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# students, group, nasa.csie.ntu
dn: cn=students,ou=group,dc=nasa,dc=csie,dc=ntu
objectClass: groupOfNames
cn: students
member: uid=student_one,ou=people,dc=nasa,dc=csie,dc=ntu

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

13. `ldapsearch -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -H ldaps://localhost -b uid=student_one,ou=people,dc=nasa,dc=csie,dc=ntu` 查詢 student 使用者

```
root@Debian:~# ldapsearch -x -D cn=admin,dc=nasa,dc=csie,dc=ntu -W -H ldaps://localhost -b uid=student_one,ou=people,dc=nasa,dc=csie,dc=ntu
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <uid=student_one,ou=people,dc=nasa,dc=csie,dc=ntu> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# student_one, people, nasa.csie.ntu
dn: uid=student_one,ou=people,dc=nasa,dc=csie,dc=ntu
objectClass: posixAccount
objectClass: StudentInformation
uid: student_one
sn: student
cn: student_one
uidNumber: 30000
gidNumber: 10001
homeDirectory: /home/student_one
loginShell: /bin/bash
studentName: student_one
examGroupID: 0

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```