

Workshop 2 – Openldap Utilities

Part1 : Idapadd

Features:

1. Facilitates the addition of objects to the DIT
2. 'Idapadd' is a symbolic link to 'Idapmodify -a'

Note: 'Idapmodify' performs the same functions as 'Idapadd'

Idapadd options:

- W: Prompt for simple authentication. This is used instead of specifying the password on the command line
- x: Use simple authentication instead of SASL.
- f: Interpret the content of files.
- D: Use the Distinguished Name *binddn* to bind to the LDAP directory.
- h: Specify an alternate host on which the ldap server is running.

Step 1: classic search with ldapclient

```
# ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=*)"
```

Step 2: Add initial entries to directory

Create a filename build_root_ou.ldif with below content:

```
dn: dc=lpi,dc=org
dc: lpi
objectclass= dcobject
objectclass= organizationalunit
ou=lpiorg
```

step 3: create root OU

```
#Idapadd -D "cn=Manager,dc=lpi,dc=org" -W -x -f build_root_ou.ldif
```

Step 4: create organizational unit

Create a filename Add_ous.ldif with below content:

```
dn: ou=engineering,dc=lpi,dc=org
ou: engineering
objectclass: organizationalunit
```

```
dn: ou=support,dc=lpi,dc=org
ou: support
objectclass: organizationalunit
```

```
dn: ou=sales,dc=lpi,dc=org
ou: sales
objectclass: organizationalunit
```

step 5: create other OU

```
# Idapadd -D "cn=Manager,dc=lpi,dc=org" -W -x -f Add_ous.ldif
```

Step 6: search for the created OU

```
# ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=*)"
```

Step 7: create a user

#ldapadd -D "cn=manager,dc=lpi,dc=org" -x -W

(enter pass)

And put as the stdin on the prompt the below amounts:

dn: cn=mehdi mehranfarid,ou=engineering,dc=lpi,dc=org
cn: mehdi mehranfarid
sn: mehdi
mail: mehdi@lpi.org
departmentNumber: Development
telephoneNumber: 203-543-8958
objectclass: inetOrgPerson

CLTR+D

Step 8: search for the created OU

ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=*)"

Step 9: create other User

Create a filename Add_second_user.ldif with below content:

dn: cn=lpi1,ou=engineering,dc=lpi,dc=org
cn: lpi1
sn: lpi1
mail: lpi1@lpi.org
departmentNumber: Support
telephoneNumber: 203-543-8956
objectclass: inetOrgPerson

Step 9: create a user

#ldapadd -D "cn=manager,dc=lpi,dc=org" -x -W -f Add_second_user.ldif

Step 10: search for the created OU

ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=*)"

Step 11: repeat this task for lpi2,lpi3 & lpi4 users for all ous

Part 2 : ldapdelete

Features:

1. A mean to DIT objects
2. Usage : ldapdelete -D "cn=manager,dc=domain,dc=org" -W -x -h hostname -v "Dn_to_delete"
3. It is an standalone application.

ldapdelete options:

- W: Prompt for simple authentication. This is used instead of specifying the password on the command line
- x: Use simple authentication instead of SASL.
- h: Specify an alternate host on which the ldap server is running.
- v: Use verbose mode, with many diagnostics written to standard output.
- c : Continues even errors are encountered.

Step 1: Delete an object (user lpi3) from the command-line

ldapdelete -D "cn=manager,dc=lpi,dc=org" -W -x -h 127.0.0.1 -v "cn=lpi3,ou=engineering,dc=lpi,dc=org"

Step 2: Delete objects from : ou=sales,dc=lpi,dc=org – via STDI

Create a file named "del_ou_objects.ldif" with below content:

```
cn=lpi1,ou=sales,dc=lpi,dc=org
cn=lpi2,ou=sales,dc=lpi,dc=org
cn=lpi3,ou=sales,dc=lpi,dc=org
cn=lpi4,ou=sales,dc=lpi,dc=org
```

Step 3: Copy the del_ou_objects.ldif content

Step 4: # ldapdelete -D "cn=manager,dc=lpi,dc=org" -W -x -h 127.0.0.1 -v
(enter pass / paste / ctrl+d)

Note: -c option with ldapdelete continues even errors are encountered.

Part 3 : ldapmodify

Features:

- 1- functionality found in:
 - a. ldapadd
 - b. ldapdelete
 - c. ldapmodify
- 2- Manipulate objects in DIT

Step 1: Add some OUs to DIT

Create a filename Add_ous.ldif with below content:

```
dn: ou=finance,dc=lpi,dc=org
ou: finance
objectclass: organizationalunit
```

```
dn: ou=marketing,dc=lpi,dc=org
ou: marketing
objectclass: organizationalunit
```

```
dn: ou=billing,dc=lpi,dc=org
ou: billing
objectclass: organizationalunit
```

Step 2: Adding the OUs:

ldapmodify -D "cn=manager,dc=lpi,dc=org" -x -W -a -v -f Add_ous.ldif

Step 3: modify

Create a filename user-mod.ldif with below content:

```
dn: cn=mehdi mehranfarid,ou=engineering,dc=lpi,dc=org
changetype: modify
replace: telephonenumber
telephoneNumber: 4777623
-
replace: departmentNumber
departmentNumber: billing
```

Step 4:

ldapmodify -D "cn=manager,dc=lpi,dc=org" -x -W -v -f user-mod.ldif

For test:

ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=*)"

Step 5: delete

Create a filename user-del.ldif with below content:

```
dn: cn=lpi1,ou=engineering,dc=lpi,dc=org
changetype: delete
```

step 6: # ldapmodify -D "cn=manager,dc=lpi,dc=org" -x -W -v -f user-del.ldif

Part 4: Ldapsearch

Features:

1. Ability to search the DIT
2. Support Boolean operators: &(And) , |(Or), !(Not)
3. Supports wildcards: *, +

Usage: ldapsearch [options] "(attribute filter operator value)"

Filtering operators:

1. =(String Comparisons)
2. <, >, <=, >=
3. ~= (regex match for approximate results)

Value(s)

1. Absolute string
2. Wildcards

Step1: Search the DIT for everything (Returning all objects)

ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=*)"

Step2: Search for specific OUs (engineering OU)

ldapsearch -x -b "ou=engineering,dc=lpi,dc=org" "(objectclass=*)"

Step3: Search the DIT for organizationalunit (OUs)

ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=organizationalUnit)"

Step4: Search the DIT for all Users

ldapsearch -x -b "dc=lpi,dc=org" "(objectclass=inetorgperson)"

Step5: Search the Users of an specific OU

ldapsearch -x -b "ou=engineering,dc=lpi,dc=org" "(objectclass=inetorgperson)"

Step6: Search the DIT for objects with an specific cn AND sn

ldapsearch -x -b "dc=lpi,dc=org" "(&(cn=mehdi mehranfarid)(sn=mehdi))"

Note: This search is case-insensitive.

Step7: Search the DIT for objects with an specific cn AND sn (Using wildcards)

ldapsearch -x -b "dc=lpi,dc=org" "(&(cn=lpi*)(sn=lpi*))"