Exercise 29.2: Centralized Authentication using LDAP and TLS - testing the LDAP server and adding users.

In this exercise we will verify the basic operation of the **LDAP** server. Specifically the ability of the server to respond to queries, and add a **POSIX** group and user. Once the user is added we can monitor the connection with **wireshark** and observe the plain text transfer of information to and from the **LDAP** server. Since it is a poor security policy to have clear text user information, we will encrypt the data stream.

Very Important

> This lab is going to use clear text when communication with the **ldap** server. The use of encryption keys is out of scope for this course. Clear text is **NOT RECOMMENDED** in any sort of production environment. To accommodate our lab environment we will disable certificate verification. The ability to not verify the keys is handy in our lab but **DO NOT** use this technique on any production systems.

This exercise will be using the machine used in previous exercises to communicate with the **LDAP** server.

It is assumed that the `ready-for.sh` script has been run, the solutions and resource files loaded and extracted to a directory.

1. Verify or install openldap-clients packages, use the appropriate command.

   Ubuntu

   ```
   # apt install sssd sssd-ldap sssd-tools oddjob-mkhomedir ldap-utils
   ```

   CentOS

   ```
   # yum install sssd sssd-ldap sssd-tools oddjob-mkhomedir openldap-clients
   ```

2. Using the **ip address** of the **LDAP** server displayed in the **Turnkey OpenLDAP Appliance Summary** screen in the previous exercise, perform a simple **LDAP** search of the **example.com** domain. We should see information about the base records for the domain example.com

   ```
   # ldapsearch -x -H ldap://192.168.0.23 -b "dc=example,dc=com" -s sub"objectclass=*"
   ```

   ```
   # extended LDIF
   #
   # LDAPv3
   # base <dc=example,dc=com> with scope subtree
   # filter: (objectclass=*)
   # requesting: ALL
   #

   # example.com
   dn: dc=example,dc=com
   objectClass: top
   objectClass: dcObject
   objectClass: organization
   o: example.com
   dc: example

   # admin, example.com
   dn: cn=admin,dc=example,dc=com
   objectClass: simpleSecurityObject
   objectClass: organizationalRole
   cn: admin
   description: LDAP administrator
   ```

```
# Groups, example.com
dn: ou=Groups,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Groups

# users, Groups, example.com
dn: cn=users,ou=Groups,dc=example,dc=com
cn: users
gidNumber: 100
objectClass: posixGroup
objectClass: top

# Users, example.com
dn: ou=Users,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Users

# Hosts, example.com
dn: ou=Hosts,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Hosts

# Idmaps, example.com
dn: ou=Idmaps,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Idmaps

# samba, example.com
dn: cn=samba,dc=example,dc=com
cn: samba
objectClass: simpleSecurityObject
objectClass: organizationalRole
description: SAMBA Access Account

# Aliases, example.com
dn: ou=Aliases,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Aliases

# nsspam, example.com
dn: cn=nsspam,dc=example,dc=com
cn: nsspam
objectClass: simpleSecurityObject
objectClass: organizationalRole
description: NSS/PAM Access Account

# search result
search: 2
result: 0 Success

# numResponses: 11
# numEntries: 10
```

3. Add a new group record using the Please see SOLUTIONS/s_29/`groups.ldif` file included in the **solutions** directory.

```
# ldapadd -x -D "cn=admin,dc=example,dc=com" -W -H ldap://192.168.0.23 -f groups.ldif
```

```
 Enter LDAP Password:
adding new entry "cn=luser1,ou=Groups,dc=example,dc=com "
```

4. The **ldapsearch** command should now show a new group.

```
# ldapsearch -x -H ldap://192.168.0.23 -b "dc=example,dc=com" -s sub"objectclass=*"
```

```
 extended LDIF
#
# LDAPv3
# base <dc=example,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# example.com
dn: dc=example,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: example.com
dc: example

# admin, example.com
dn: cn=admin,dc=example,dc=com
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator

# Groups, example.com
dn: ou=Groups,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Groups

# users, Groups, example.com
dn: cn=users,ou=Groups,dc=example,dc=com
cn: users
gidNumber: 100
objectClass: posixGroup
objectClass: top

# Users, example.com
dn: ou=Users,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Users

# Hosts, example.com
dn: ou=Hosts,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Hosts

# Idmaps, example.com
dn: ou=Idmaps,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Idmaps
```

```
# samba, example.com
dn: cn=samba,dc=example,dc=com
cn: samba
objectClass: simpleSecurityObject
objectClass: organizationalRole
description: SAMBA Access Account

# Aliases, example.com
dn: ou=Aliases,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Aliases

# nsspam, example.com
dn: cn=nsspam,dc=example,dc=com
cn: nsspam
objectClass: simpleSecurityObject
objectClass: organizationalRole
description: NSS/PAM Access Account

# luser1, Groups, example.com
dn: cn=luser1,ou=Groups,dc=example,dc=com
cn: luser1
objectClass: posixGroup
objectClass: top
gidNumber: 999001
memberUid: luser1

# search result
search: 2
result: 0 Success

# numResponses: 12
# numEntries: 11
```

Notice in the output above the **numResponses** and **numEntries** have increased when we added the new group record.

5. Add a new user record using the Please see SOLUTIONS/s_29/users.ldif file included in the **solutions** directory.

```
# ldapadd -x -D "cn=admin,dc=example,dc=com" -W -H ldap://192.168.0.23 -f users.ldif
```

```
Enter LDAP Password:
adding new entry "cn=luser1,dc=example,dc=com"
```

6. The **ldapsearch** command should now show a new user.

```
# ldapsearch -x -H ldap://192.168.0.23 -b "dc=example,dc=com" -s sub"objectclass=*"
```

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# example.com
dn: dc=example,dc=com
objectClass: top
objectClass: dcObject
```

```
objectClass: organization
o: example.com
dc: example

# admin, example.com
dn: cn=admin,dc=example,dc=com
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator

# Groups, example.com
dn: ou=Groups,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Groups

# users, Groups, example.com
dn: cn=users,ou=Groups,dc=example,dc=com
cn: users
gidNumber: 100
objectClass: posixGroup
objectClass: top

# Users, example.com
dn: ou=Users,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Users

# Hosts, example.com
dn: ou=Hosts,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Hosts

# Idmaps, example.com
dn: ou=Idmaps,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Idmaps

# samba, example.com
dn: cn=samba,dc=example,dc=com
cn: samba
objectClass: simpleSecurityObject
objectClass: organizationalRole
description: SAMBA Access Account

# Aliases, example.com
dn: ou=Aliases,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Aliases

# nsspam, example.com
dn: cn=nsspam,dc=example,dc=com
cn: nsspam
objectClass: simpleSecurityObject
objectClass: organizationalRole
description: NSS/PAM Access Account

# luser1, Groups, example.com
```

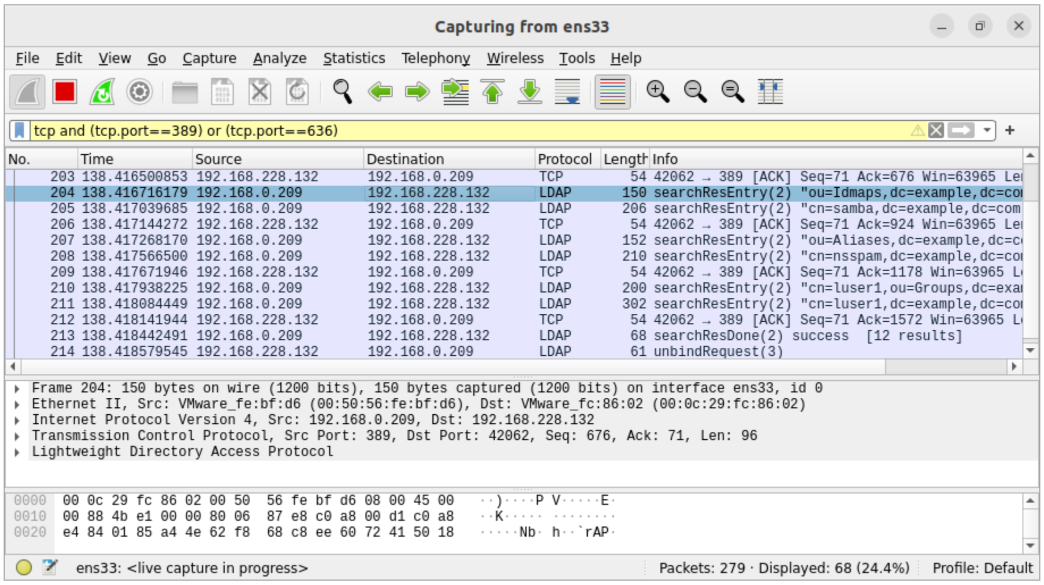
```
dn: cn=luser1,ou=Groups,dc=example,dc=com
cn: luser1
objectClass: posixGroup
objectClass: top
gidNumber: 999001
memberUid: luser1

# luser1, example.com
dn: cn=luser1,dc=example,dc=com
uid: luser1
cn: luser1
givenName: luser1
sn: linux
homeDirectory: /home/users/luser1
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
uidNumber: 999001
gidNumber: 999001

# search result
search: 2
result: 0 Success

# numResponses: 13
# numEntries: 12
```

7. Install or verify wireshark is installed.

```
# yum install wireshark-gnome
# wireshark
```

8. Capture an ldapsearch command with wireshark, notice the data is clear text.

   Wireshark has excellent filters, use the filter "`tcp and (tcp.port==389) or (tcp.port==636)`" to have **wireshark** only display **LDAP** communication.



```
tcp and (tcp.port==389) or (tcp.port==636)
```

```
No.    Time          Source           Destination      Protocol Length Info
   203 138.416500853 192.168.228.132  192.168.0.209    TCP        54 42062 → 389 [ACK] Seq=71 Ack=676 Win=63965 Le
   204 138.416716179 192.168.0.209    192.168.228.132  LDAP      150 searchResEntry(2) "ou=Idmaps,dc=example,dc=co
   205 138.417039685 192.168.0.209    192.168.228.132  LDAP      206 searchResEntry(2) "cn=samba,dc=example,dc=com
   206 138.417144272 192.168.228.132  192.168.0.209    TCP        54 42062 → 389 [ACK] Seq=71 Ack=924 Win=63965 Le
   207 138.417268170 192.168.0.209    192.168.228.132  LDAP      152 searchResEntry(2) "ou=Aliases,dc=example,dc=c
   208 138.417566500 192.168.0.209    192.168.228.132  LDAP      210 searchResEntry(2) "cn=nsspam,dc=example,dc=co
   209 138.417671946 192.168.228.132  192.168.0.209    TCP        54 42062 → 389 [ACK] Seq=71 Ack=1178 Win=63965 L
   210 138.417938225 192.168.0.209    192.168.228.132  LDAP      200 searchResEntry(2) "cn=luser1,ou=Groups,dc=exa
   211 138.418084449 192.168.0.209    192.168.228.132  LDAP      302 searchResEntry(2) "cn=luser1,dc=example,dc=co
   212 138.418141944 192.168.228.132  192.168.0.209    TCP        54 42062 → 389 [ACK] Seq=71 Ack=1572 Win=63965 L
   213 138.418442491 192.168.0.209    192.168.228.132  LDAP       68 searchResDone(2) success  [12 results]
   214 138.418579545 192.168.228.132  192.168.0.209    LDAP       61 unbindRequest(3)
```

```
Frame 204: 150 bytes on wire (1200 bits), 150 bytes captured (1200 bits) on interface ens33, id 0
Ethernet II, Src: VMware_fe:bf:d6 (00:50:56:fe:bf:d6), Dst: VMware_fc:86:02 (00:0c:29:fc:86:02)
Internet Protocol Version 4, Src: 192.168.0.209, Dst: 192.168.228.132
Transmission Control Protocol, Src Port: 389, Dst Port: 42062, Seq: 676, Ack: 71, Len: 96
Lightweight Directory Access Protocol
```

```
0000  00 0c 29 fc 86 02 00 50  56 fe bf d6 08 00 45 00   ··)····P V·····E·
0010  00 88 4b e1 00 00 80 06  87 e8 c0 a8 00 d1 c0 a8   ··K····· ········
0020  e4 84 01 85 a4 4e 62 f8  68 c8 ee 60 72 41 50 18   ·····Nb· h··`rAP·
```

```
ens33: <live capture in progress>    Packets: 279 · Displayed: 68 (24.4%)    Profile: Default
```

Figure 29.3: **Wireshark LDAP packet trace**