Week 2

Session 2

Identity and Access Management

Olga Chernukhina

0. Preparation

I decided to use GUI, so I installed OpenLDAP and ApacheDirectoryStudio on my server (10.0.2.4), and connected to it via `ssh -X` on client machine (10.0.2.5).

1. Install OpenLDAP on your server

Run **on server:**

```
sudo apt-get update
```

```
sudo apt-get install slapd ldap-utils
```

```
sudo dpkg-reconfigure slapd
```

- Omit OpenLDAP server configuration - *No*
- DNS domain name - *innopolis.local*
- Organization name - *SB*
- Administrator password - *entered a password twice*
- Database backend - *HDB*
- Remove the database when slapd is purged - *No*
- Move old database - *Yes*

```
sudo ufw allow ldap
```

```
sudo service sldap restart
```

Installing ApacheDirectoryStudio

On client:

Download the Apache Directory Studio tar file from the official Apache site

Copy the file to the server via scp:
`scp ApacheDirectoryStudio-2.0.0.v20200411-M15-linux.gtk.x86_64tar.gz serv@10.0.2.4`

On server:

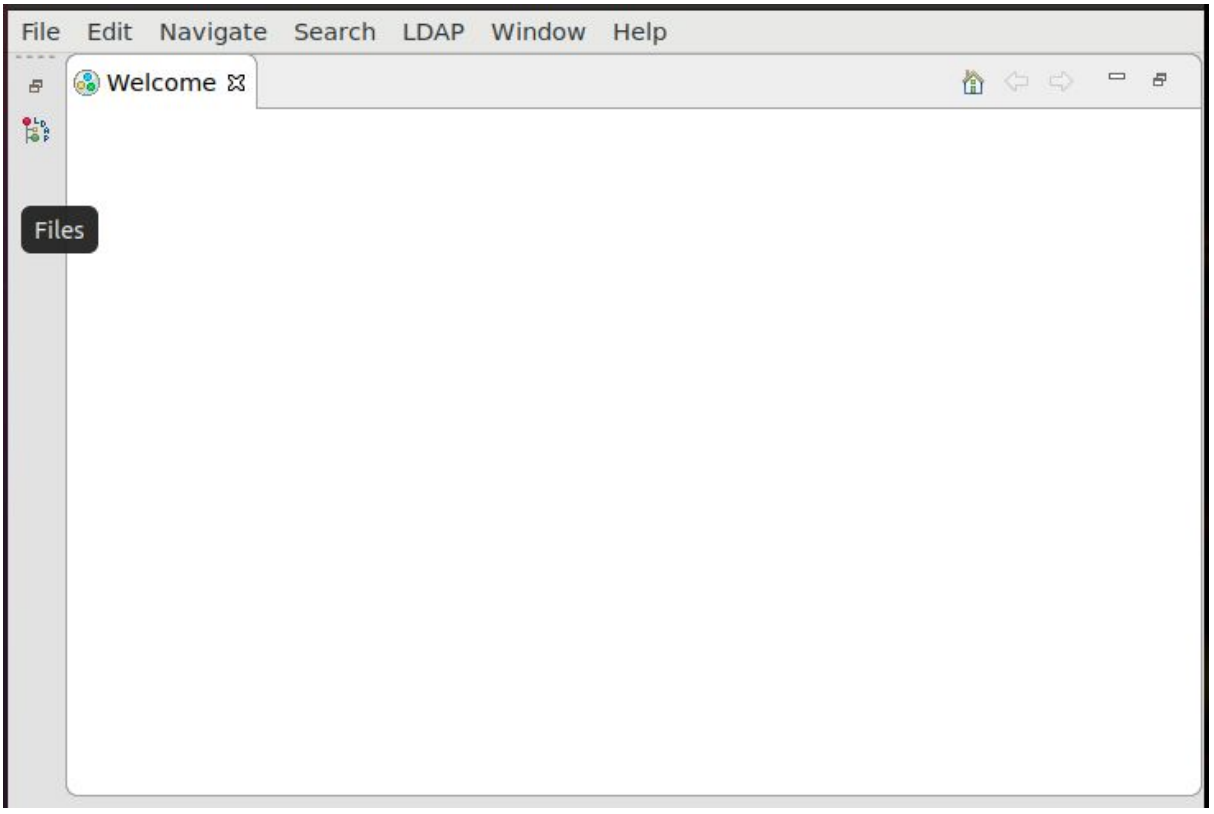
Navigate to the directory with the archive

Unpack it via
`tar xvzf ApacheDirectoryStudio-2.0.0.v20200411-M15-linux.gtk.x86_64tar.gz`

On client:

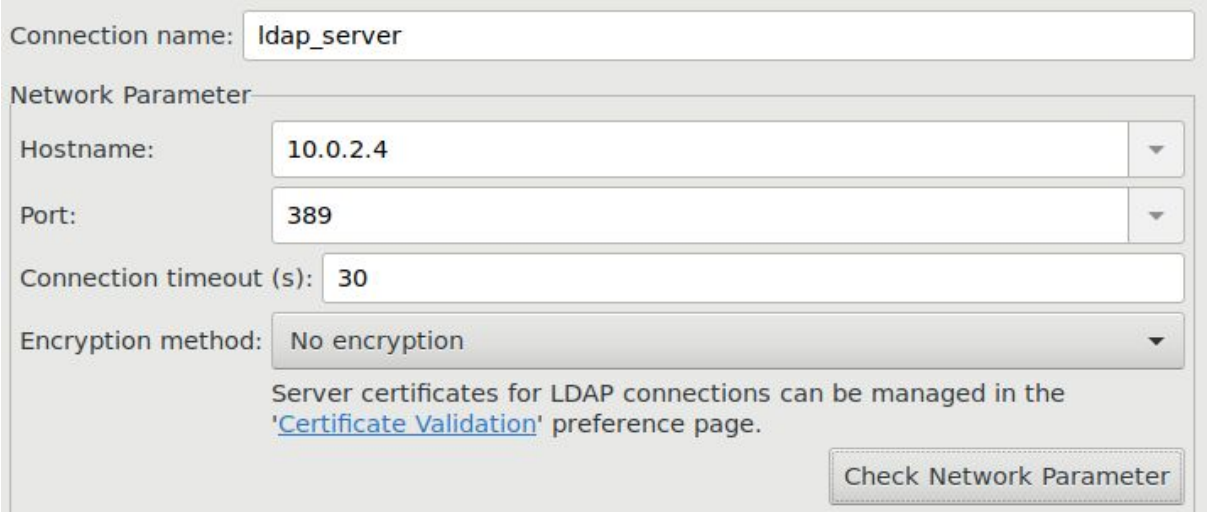
Connect via ssh to the server and run the ADS: `ssh -X serv@10.0.2.4`

```
serv@server:~$ cd ApacheDirectoryStudio
serv@server:/ApacheDirectoryStudio$ ./ApacheDirectoryStudio
```

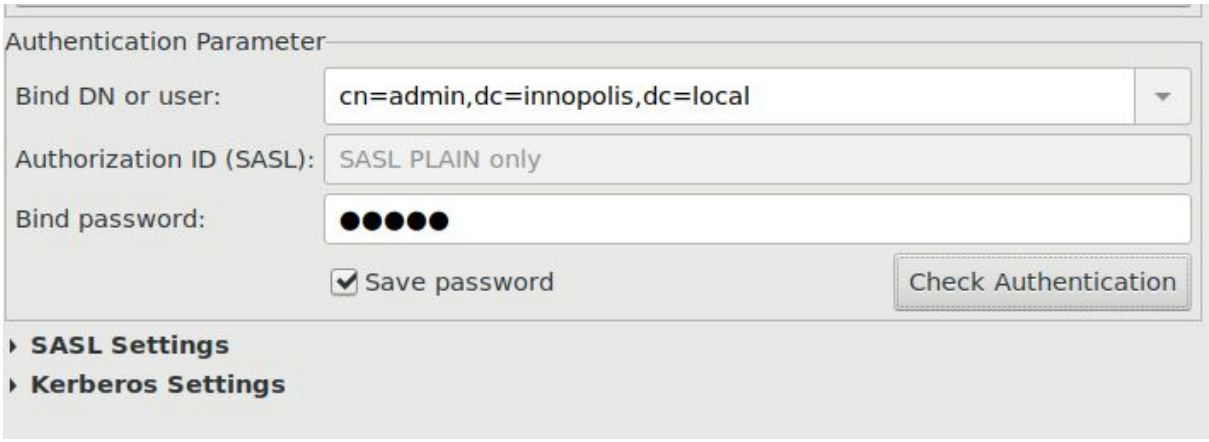


Create a connection to the ldap:

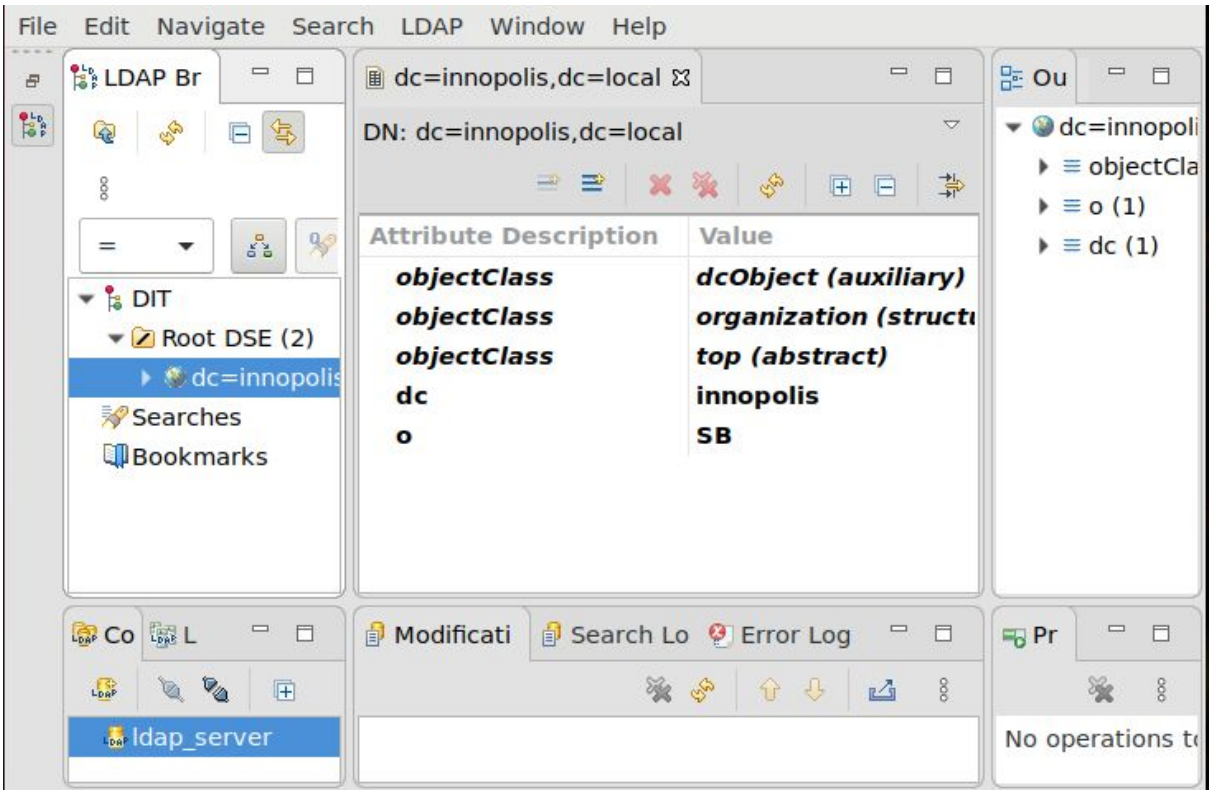
Enter the IP of the server and give any name to the connection:



Enter the LDAP server credentials:

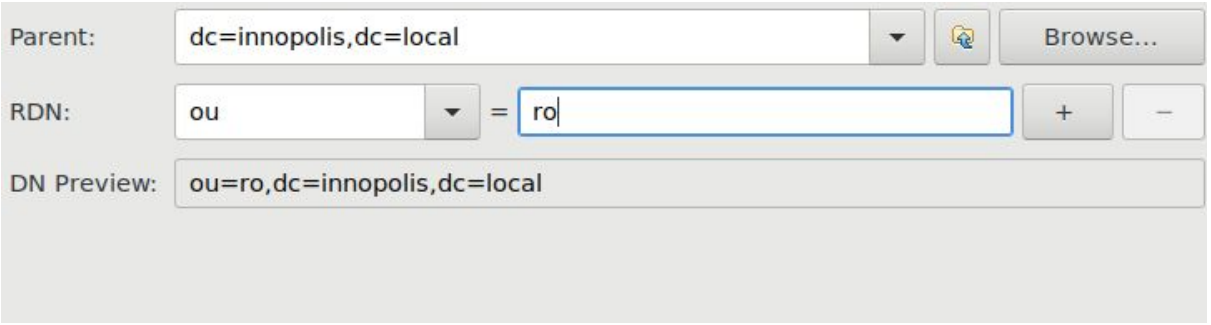


As a result you would see this:



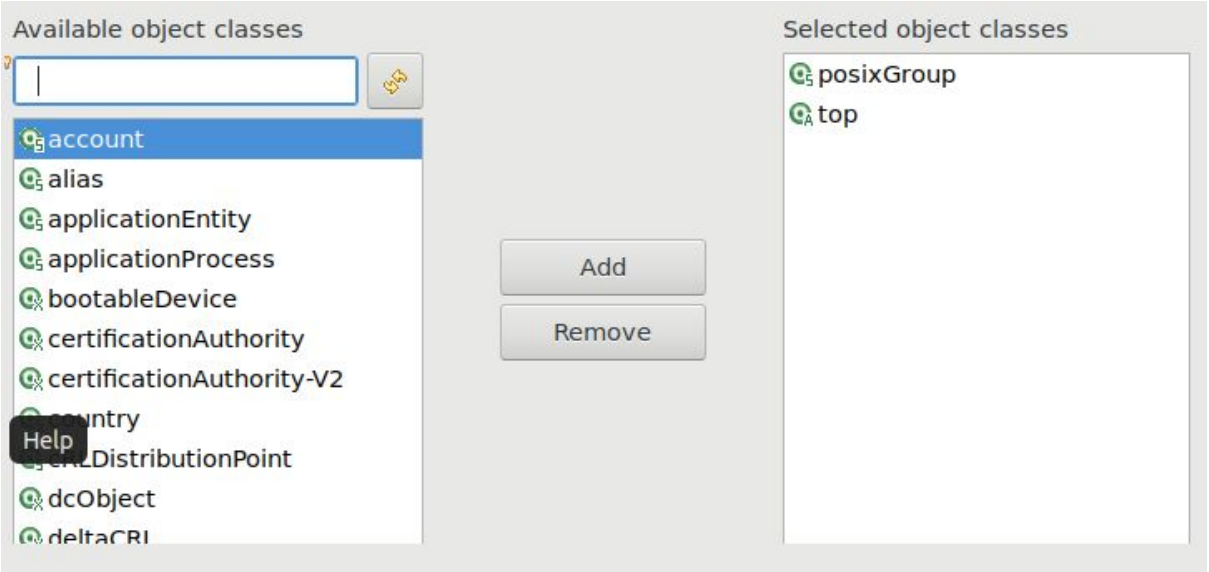
Create two OUs for in your directory base.

I created **OU1 = RO** and **OU2 = SE** in the following way:

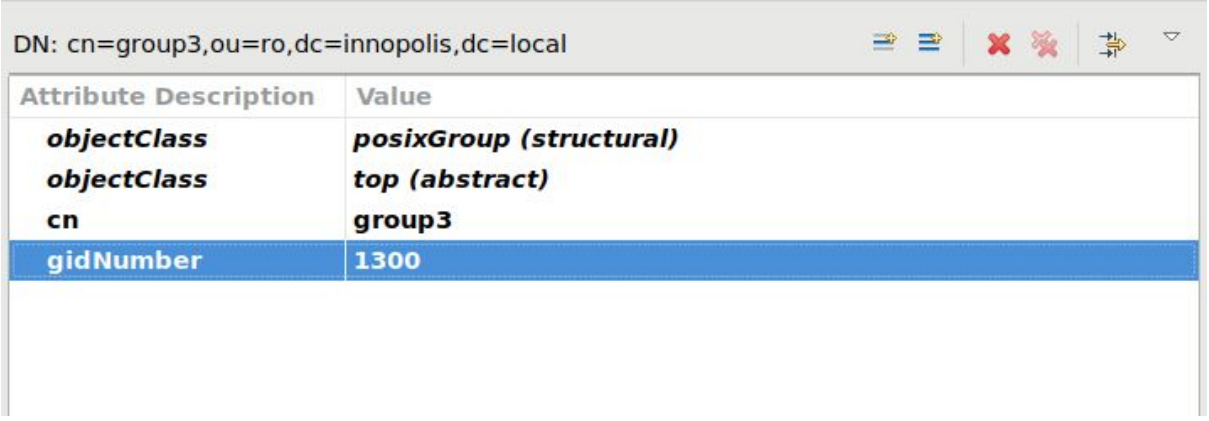


Create two groups in each OU

Choose PosixGroup since we are using Linux:

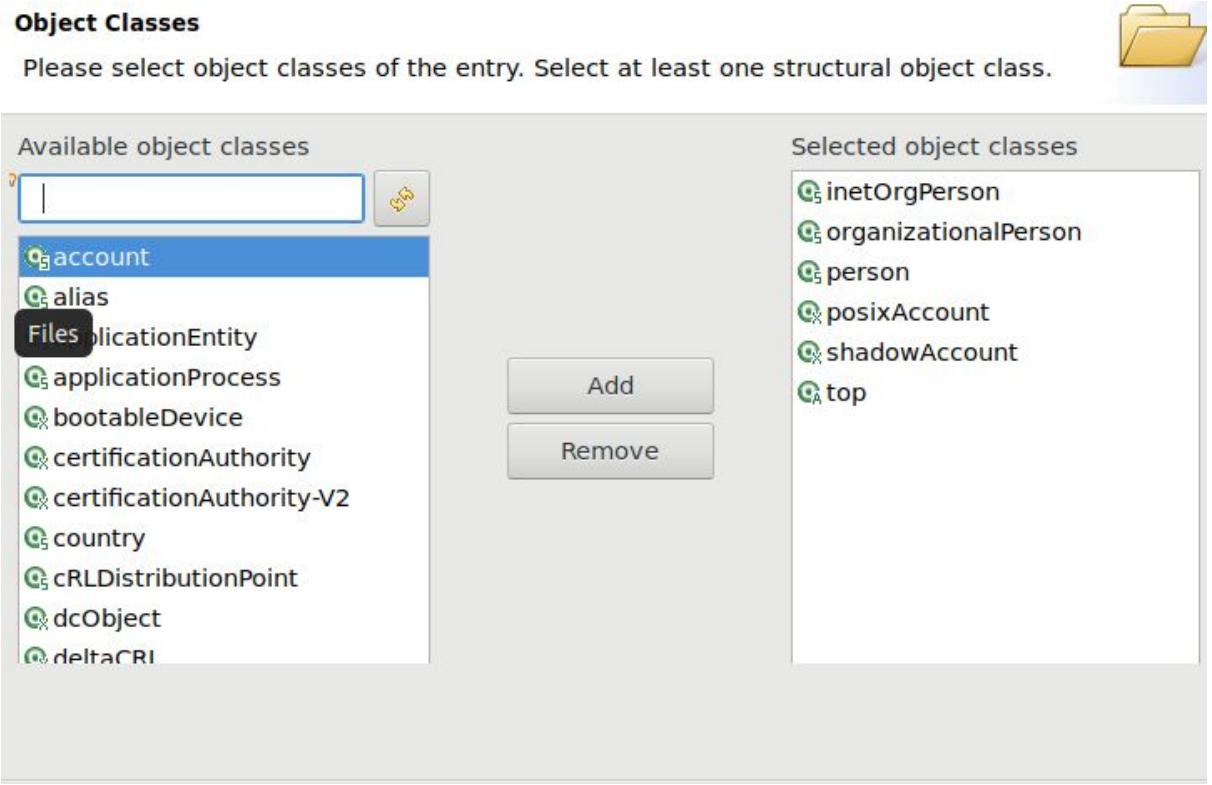


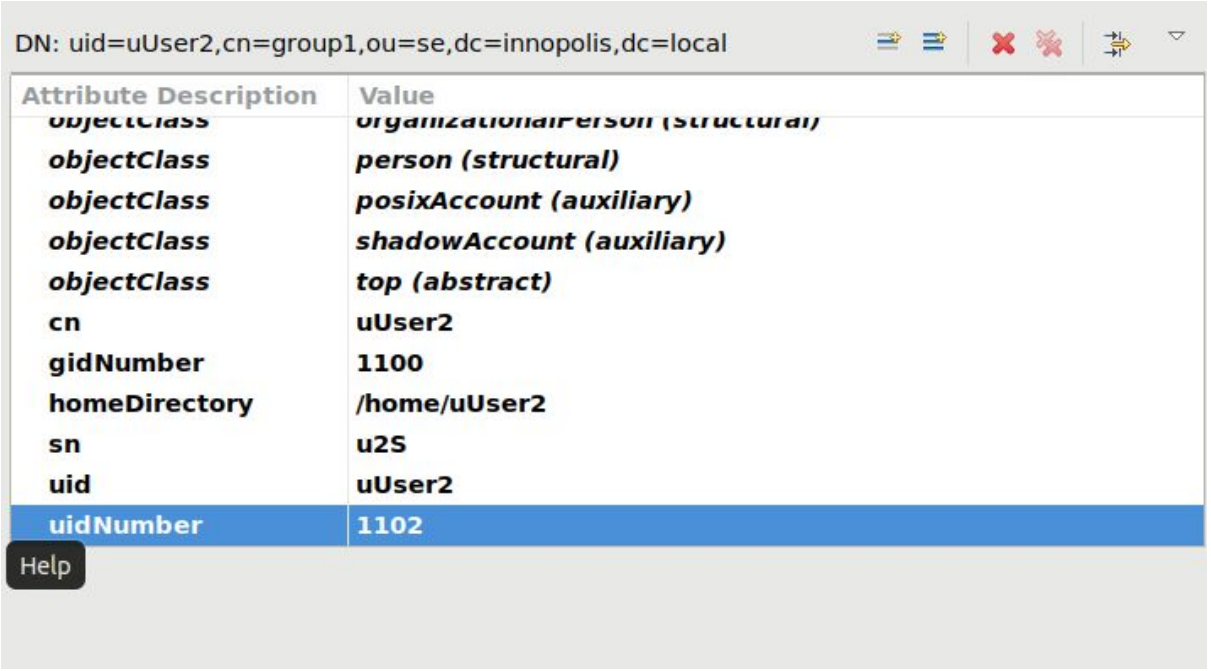
Select group number strictly larger than 1000, since 1-1000 are used for services and root user:



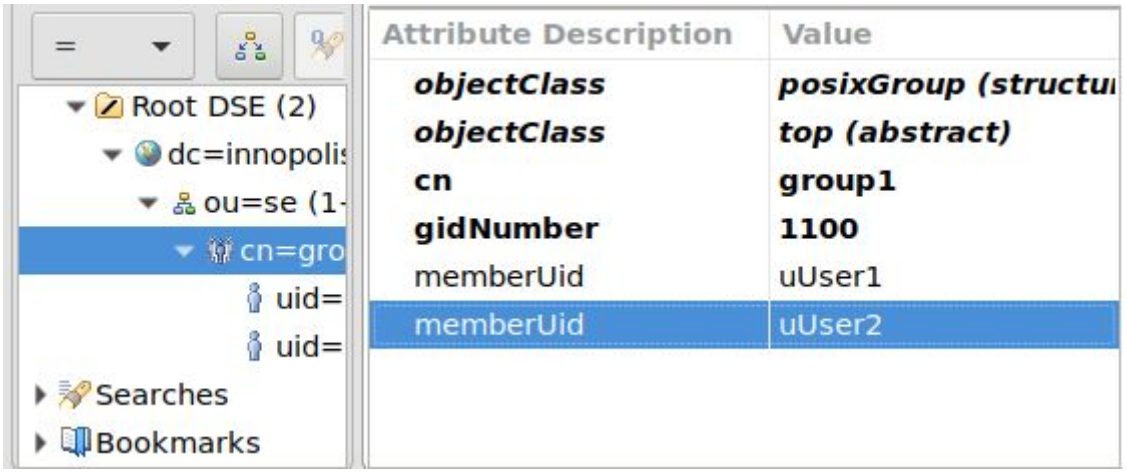
Create one account in each of 4 groups

Apply the following object classes so that the user would be compatible with any system:





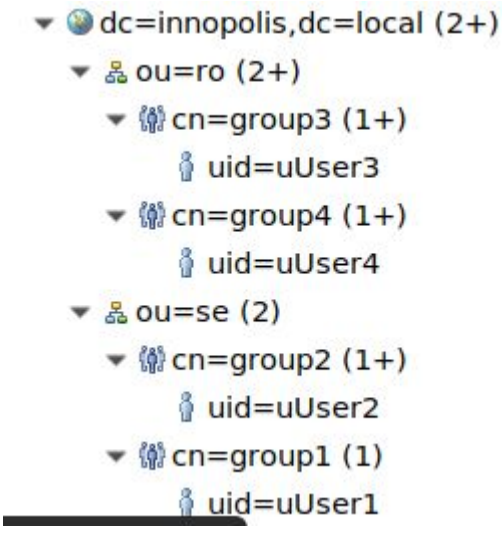
Add a user to the group (later I moved that user to the other group):



Add `loginShell` and `userPassword` fields:

Attribute Description	Value
~~objectClass~~	~~top (abstract)~~
cn	**uUser2**
gidNumber	**1100**
homeDirectory	**/home/uUser2**
sn	**u2S**
uid	**uUser2**
uidNumber	**1102**
loginShell	/bin/bash
userPassword	Plain text password

The final structure is the following:



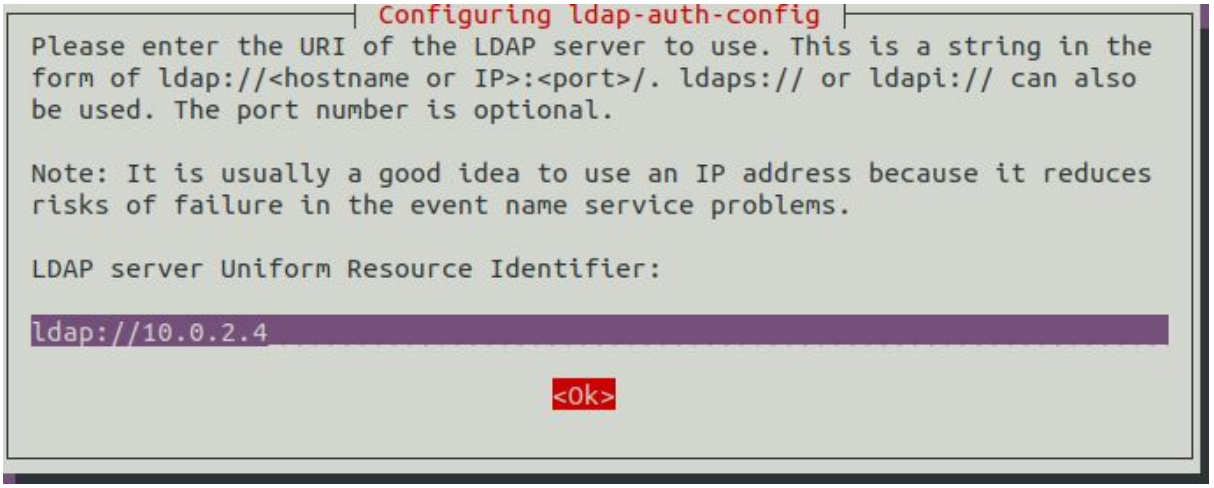
2. Install LDAP Client on your client

On the second client machine (first one is used to ssh to the server):

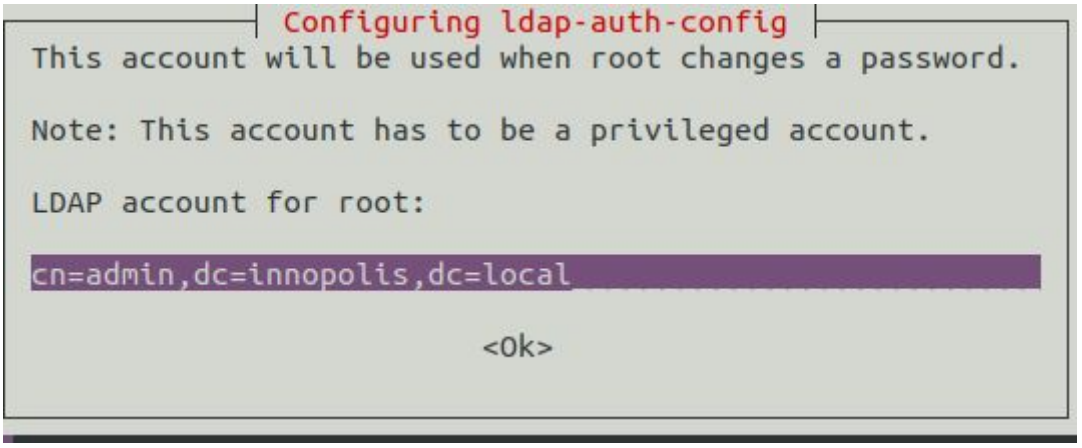
```
apt-get install ldap-auth-client libpam-ldap nscd
```

```
sudo dpkg-reconfigure ldap-auth-config
```

Enter the IP address of the OpenLDAP server:



Enter credentials of the server connection:




```
sudo nano /etc/nsswitch.conf
```

```
  GNU nano 4.8                    /etc/nsswitch.conf
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the `glibc-doc-reference' and `info' packages instal
# `info libc "Name Service Switch"' for information about this fil

p Thunderbird Mail  ldap compat
group:              ldap compat
shadow:             ldap compat
gshadow:            files

hosts:              files mdns4_minimal [NOTFOUND=return] dns
networks:           files

protocols:          db files
services:           db files
ethers:             db files
rpc:                db files

netgroup:           nis
```

`sudo nano /etc/pam.d/common-session`

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

`sudo /etc/init.d/nscd restart`

Run `getent passwd` to check that newly created users have appeared

```
ubuntu@ubuntu:~$ getent passwd
uUser1:x:1101:1100:uUser1:/home/User1:/bin/bash
uUser3:x:1301:1300:uUser3:/home/uUser3:/bin/bash
uUser4:x:1401:1400:uUser4:/home/uUser4:/bin/bash
uUser2:x:1201:1200:uUser2:/home/uUser2:/bin/bash
```

Install ssh: `sudo apt-get install ssh`

Make sure these parameters are set in `/etc/ssh/sshd_config:`

```
PermitRootLogin yes
UsePAM yes
```

SSH from your client using one of the users from each OU (accounts from different OUs)

```
ubuntu@ubuntu:~$ ssh uUser1@10.0.2.6
uUser1@10.0.2.6's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-42-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Tue Nov  3 18:08:35 2020 from 10.0.2.6
uUser1@ubuntu:~$
```

```
ubuntu@ubuntu:~$ ssh uUser3@10.0.2.6
uUser3@10.0.2.6's password:
Creating directory '/home/uUser3'.
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-42-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

uUser3@ubuntu:~$
```

Login to your client using the remaining two accounts (one account from each OU)

```
ubuntu@ubuntu:~$ su - uUser2
Password:
uUser2@ubuntu:~$ id
uid=1201(uUser2) gid=1200(group2) groups=1200(group2)
```

```
ubuntu@ubuntu:~$ su - uUser4
Password:
Creating directory '/home/uUser4'.
uUser4@ubuntu:~$ id
uid=1401(uUser4) gid=1400(group4) groups=1400(group4)
```

3. Bonus

Describe what is MimiKatz in details

Mimikatz is a tool that implements the functionality of the **Windows Credentials Editor** and allows you to extract the **authentication data** of a user in **plain text**. However, it is applicable only for those who have **logged in** to the system.

Such a miracle is associated with the use of the security provider **wdigest**, which stores the password in memory in cleartext. Why to store passwords in plain text when you can log in using a **hash**? In fact, the latter is **not possible everywhere**. Therefore, Windows has a special security provider, wdigest, to support such types of authorization where you need to know the password (and the hash is not enough).

The attack is performed in mimikatz's own terminal using only three commands:

```
mimikatz # privilege::debug
mimikatz # inject::process lsass.exe sekurlsa.dll
mimikatz # @getLogonPasswords
```

To **protect** against an attack, as a **temporary** solution, you can **disable the digest security provider** through the appropriate registry branch:

```
((HKEY_LOCAL_MACHINE SYSTEM CurrentControlSet Control Lsa)
```

However, the user should understand that the attacker can do the same in **reverse order**.

Describe Golden Ticket Attack

Imagine that we intruded into the system, but suddenly lost control of the domain because the **administrator changed the password** for some reason. **Golden Ticket** is used to **prevent losing** gained administrative **access** due to such situations.

In **Kerberos** authentication scheme the **authenticity of Kerberos is not verified** (AS-REQ and AS-REP do not pass through the domain controller). Since the Golden Ticket is a fake TGT, it is sent to the domain controller as part of the TGS-REQ to receive the TGS ticket.

The Kerberos **Golden ticket is a valid Kerberos TGT** ticket because it is e**ncrypted and signed** by a Kerberos domain account (krbtgt). And since the TGT is encrypted with the krbtgt password hash and can be decrypted by any KDC service in the domain, the ticket is perceived as real. To **make** a Golden Ticket, we need to know the following:

SPN of the domain - revealed by `Get-ADDomain`
Sid of the domain - revealed by `Get-ADDomain`
NTLM hash of the krbtgt domain account - discover using `mimikatz`
Name of the user - any name is allowed

Knowing these system parameters, the attack could be implemented via Ticketer, Mimikatz, Meterpreter.

Sources:
https://www.digitalocean.com/community/tutorials/how-to-configure-bind-as-a-caching-or-forwarding-dns-server-on-ubuntu-16-04

https://www.linux.com/topic/networking/how-install-apache-directory-studio-and-connect-openldap-server/

https://linoxide.com/linux-how-to/setup-openldap-server-authenticate-client-workstation/

https://xakep.ru/2020/04/15/windows-ad-persistence/