Administración de unidades organizativas, usuarios e grupos en LDAP con ldap-utils

Sumario

Introdución

- Neste apartado imos administrar o directorio ldap facendo uso das utilidades de **ldap-utils** para poder administrar o servidor LDAP: introducir, modificar, borrar, buscar e extraer información, mantelo en óptimo funcionamento, etc.

- Na imaxe amósase cales son esas utilidades, das cales usaremos: ldapadd, ldapdelete, ldapmodify, ldappasswd e ldapsearch. Este último xa o vimos no apartado anterior.
- Lembrar que non existe un comando ldap... senón que se escribiu ldap e logo premeuse a tecla TAB 2 veces para ver que comandos comezan por ldap...

```
root@dserver00:~# ldap
ldapadd      ldapdelete   ldapmodify   ldappasswd   ldapurl
ldapcompare  ldapexop     ldapmodrdn   ldapsearch   ldapwhoami
root@dserver00:~# ldap
```

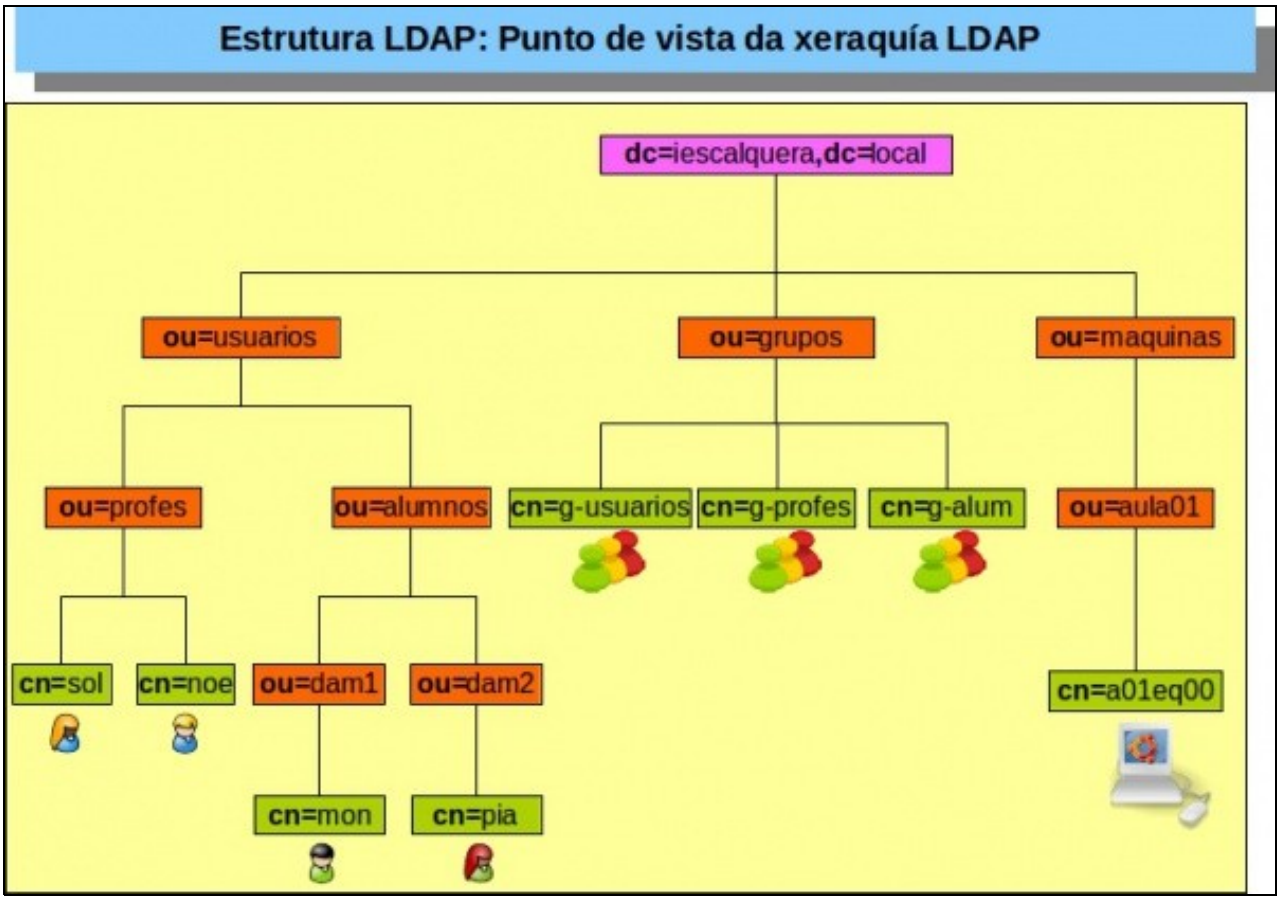
- Na seguinte imaxe amosamos os usuarios e grupos cos que imos traballar neste apartado e nos seguintes.
- Neste apartado traballaremos con:
    - usuarios: **sol** e **noe**
    - grupos: **g-usuarios** e os relacionados cos **profes**.

USUARIOS E GRUPOS

Grupos Usuarios	Nome Completo	g-usarios (10000)	g-profes (10001)	g-dam1-profes (10002)	g-dam2-profes (10003)	g-alum (10004)	g-dam1-alum (10005)	g-dam2-alum (10006)
Descric.		Todolos usuarios de LDAP	Todo o profesorado	Profesorado de 1º da DAM	Profesorado de 2º DAM	Todo o alumnado	Alumnado de 1º da DAM	Alumnado de 2º da DAM
sol (10000)	Profe – Sol Lúa	✔(1º)	✔	✔	✔			
noe (10001)	Profe – Noé Ras	✔(1º)	✔		✔			
mon (10002)	Dam1 – Mon Mon	✔(1º)				✔	✔	
tom (10003)	Dam1 – Tom Tom	✔(1º)				✔	✔	
pia (10004)	Dam2 – Pía Glez	✔(1º)				✔		✔
paz (10005)	Dam2 – Paz Fdez	✔(1º)				✔		✔

- Observar:
  - Onde pon (1º) indica ese usuario ten ese grupo por **grupo principal ou primario**.
  - Todos os usuarios teñen como grupo primario *g-usuarios*, independentemente de se é profesor/a ou alumnado.
  - Para todo **curso** hai dous grupos,
    - *g-<curso>-profes*: terá como membros ao profesorado dese curso
    - *g-<curso>-alum*: terá como membros ao alumnado dese curso
  - Todo usuario pertence como mínimo a tres grupos:
    - *g-usuarios* obrigatoriamente
    - *g-<curso>-<tipo usuario>*
    - *g-<tipo de usuario>* (*profes ou alum*)
  - Esta organización é pensada para cando xestionemos os permisos nas carpetas na Parte III do curso.
  - A profesora **sol** é profesora nos dous cursos.
  - O grupos comezan coa letra **g-**, para á hora de buscalos entre todos os grupos locais e de LDAP poder ter todos os grupos que creamos por nós máis fáciles de localizar. Todos son **g-algo**.

- Finalmente, a seguinte imaxe amosa as Unidades Organizativas (OU) coas que imos traballar e representa algúns dos usuarios e grupos da táboa anterior.



LEMBRA...

Antes de comezar con esta sección é aconsellable que o lector/a domine a xestión de usuarios en GNU/Linux. Pódese atopar axuda na sección de Usuarios e grupos en Ubuntu do curso Curso Platega: Ubuntu Desktop. Un sistema dual (MS Windows / GNU/Linux) .

Ferramentas de ldap-utils

- Indicar que estas ferramentas tamén se poden instalar en calquera equipo cliente e permite administrar o servidor LDAP, só que non podemos usar a *uri* **ldapi:**/// e temos que usar para conectarnos a el **ldap://servidor** ou **ldaps://servidor:636** se o servizo seguro está configurado no servidor e no cliente.

- Como indicamos, o paquete *ldap-utils* contén varios comandos útiles para manexar a información almacenada no directorio LDAP. Algúns dos máis destacables son:

- **ldapsearch**: Como xa vimos nun punto do apartado anterior: (Consultar a BD: ldapsearch) permite buscar información no directorio, devolvendo o resultado da busca en formato LDIF. Para facer a busca, poderemos indicarlle o filtro polo que queremos buscar; por exemplo '(uid=xyz)' buscará as entradas que teñan un atributo *uid* co valor *xyz*. Tamén lle podemos indicar despois os atributos que queremos ver de cada entrada atopada no resultado da busca. Os parámetros máis importantes xa vistos antes son:
    - **-x**: usar autenticación simple.
    - **-Y *EXTERNAL***: usa autenticación **SASL** (Simple Authentication and Security Layer - capa de seguridade e autenticación simple)
        - O mecanismo de SASL que usamos neste caso é **EXTERNAL**, onde a autenticación está implícita no contexto, isto é, o cliente e servidor teñen instalados uns certificados para autenticarse un contra o outro. Neste caso o cliente (ldapsearch) e o servizo (slapd) están no mesmo equipo en dserver00, co cal eses certificados xa están no equipo.
        - Nos seguintes enlaces hai máis información sobre este tipo de autenticación:
            - http://es.wikipedia.org/wiki/SASL
            - http://www.openldap.org/doc/admin21/sasl.html
            - http://www.openldap.org/doc/admin21/tls.html
    - **-D *dn***: dn para conectarse ao LDAP indicando o nome de usuario co que nos imos conectar.

- **-w *contrasinal***: Indicar o contrasinal para conectarse ao LDAP.
- **-W**: Obrigamos a que o comando pida o contrasinal para conectarse ao LDAP en lugar de recibilo como parámetro.
- **-H** *ldapuri***: Especificar a URI coa que nos imos conectar ao servidor ldap. Por exemplo ldap://localhost ou ldapi://**
- **filtro** indica que nos devolva as entradas que coincidan cos valores dos atributos que indicamos no filtro.
- **-b *base de busca***: para indicar en que obxecto da árbore comezar a buscar
- **-s** *base*|*one*|*sub***: indica se queremos que nos devolva os atributos e valores:**
    - ◊ **base**: só do obxecto que estamos consultado
    - ◊ **one**: só dos obxectos que están un nivel por debaixo do obxecto consultado.
    - ◊ **sub**: do obxecto consultado e de toda a súa subárbore. É o valor por defecto.

- **ldapadd**, **ldapmodify**: Os dous comandos enlazan con mesmo executable, polo que realmente fan a mesma función de engadir ou modificar entradas no LDAP (*ldapadd* equivale a *ldapmodify -a*). Os parámetros máis importantes son (moitos deles vistos en ldapsearch):
    - **-x**: usar autenticación simple.
    - **-Y *EXTERNAL***: usa autenticación **SASL**
    - **-D *dn***: dn para conectarse ao LDAP indicando o nome de usuario co que nos imos conectar.
    - **-w *contrasinal***: Indicar o contrasinal para conectarse ao LDAP.
    - **-W**: Obrigamos a que o comando pida o contrasinal para conectarse ao LDAP en lugar de recibilo como parámetro.
    - **-H** *ldapuri***: Especificar a URI coa que nos imos conectar ao servidor ldap. Por exemplo ldap://localhost ou ldapi://**
    - **filtro** indican que nos devolva as entradas que coincidan cos valores dos atributos que indicamos no filtro.
    - **-f *ficheiro***: Toma a información que se debe engadir, modificar ou borrar do *ficheiro* en lugar da entrada estándar
        - ◊ Na páxina do manual do comando (http://linux.die.net/man/1/ldapadd) explica o formato deste ficheiro para engadir, modificar e borrar entradas.
        - ◊ Basicamente o formato do ficheiro baséase no formato LDIF e ten a seguinte estrutura:

```
dn: cn=objectXX,dc=example,dc=com
changetype: modify
replace: mail
mail: modme@example.com
-
add: title
title: Grand Poobah
-
add: jpegPhoto
jpegPhoto:< file:///tmp/modme.jpeg
-
delete: description
-
```

- **ldapdelete**: De sintaxe similar ao comando anterior, recibe como entrada (ou ben pola entrada estándar ou ben dun ficheiro) unha lista de dn's para eliminalas do LDAP.
    - **-r** para poder borrar unha rama que ten fillos de forma recursiva
    - No seguinte enlace está o manual do comando: http://linux.die.net/man/1/ldapdelete

- **ldappasswd**: É para cambiar o contrasinal dun usuario en LDAP. Sóese usar no servidor para o que root cambie o contrasinal dun usuario. De sintaxe similar ao comando anterior.
    - O comando **passwd** segue existindo nos clientes e é o que debe seguir sendo usado polos usuarios normais, pois xa hai utilidades que se encargan de enlazar co LDAP para que sexa aí onde se reflictan os cambios.
    - Cando cambiemos o contrasinal sempre imos deixar o mesmo que tiñamos **abc123.**, para non esquecerse!!!.

Administración de Unidades Organizativas (OU)

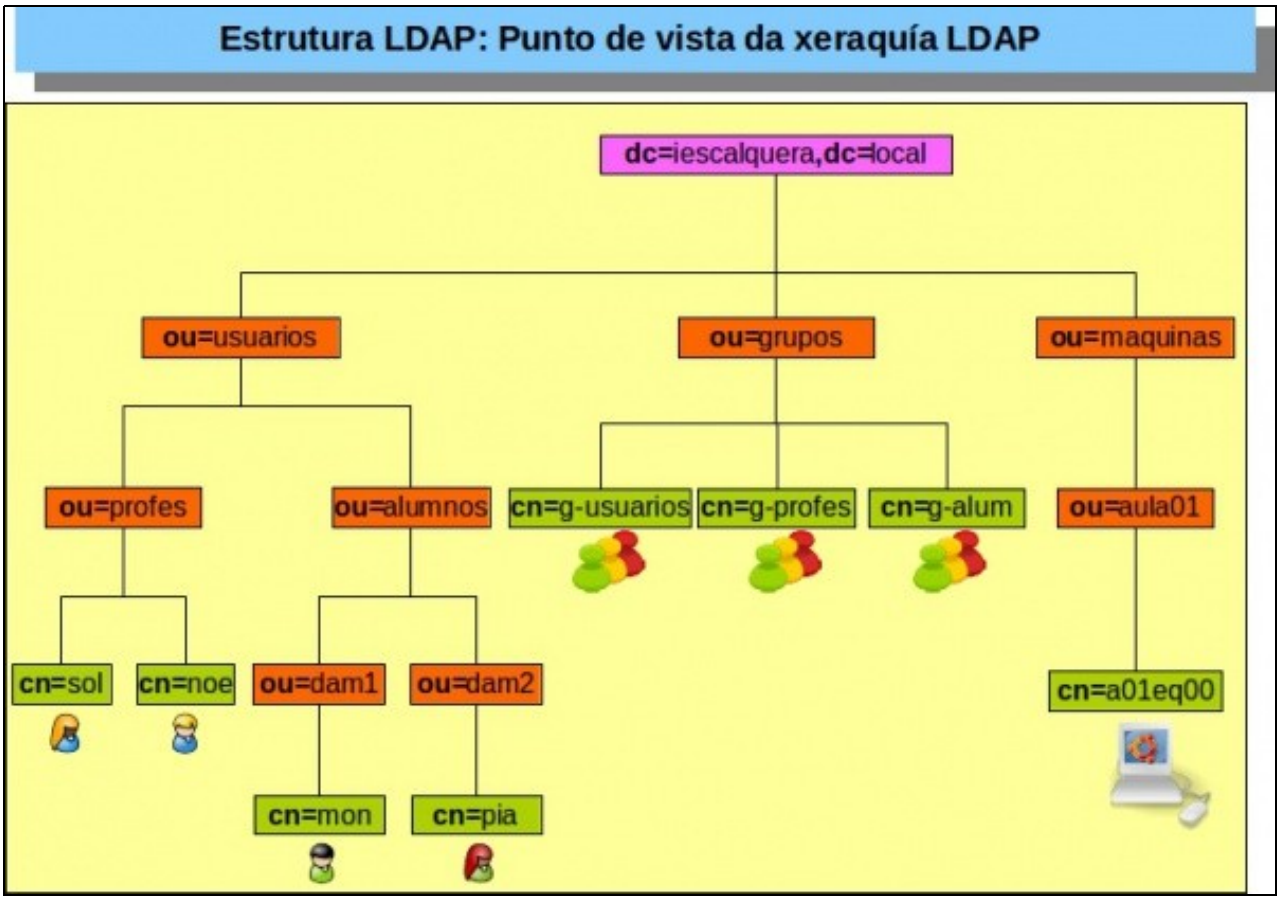
- O seguinte exemplo amosa o mínimo que debe conter un ficheiro base para manexar as OUs.

```
#ModeloOU.ldif
dn: ou=exemplo-unidade-organizativa,dc=exemplo,dc=local
objectClass: organizationalUnit
ou: exemplo-unidade-organizativa
```

- Como sabemos da teoría o **objectClass organizationalUnit** defínese no **esquema core**.
    - ♦ Observar a liña 2, só se precisa indicar o atributo **ou** que é o que ten o exemplo anterior.
    - ♦ Os demais atributos son opcionais.

```
objectclass ( 2.5.6.5 NAME 'organizationalUnit' SUP top STRUCTURAL
        MUST ou
    ( userPassword $ searchGuide $ seeAlso $ businessCategory $
                x121Address $ registeredAddress $ destinationIndicator $
                preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $
                telephoneNumber $ internationaliSDNNumber $
                facsimileTelephoneNumber $ street $ postOfficeBox $ postalCode $
                postalAddress $ physicalDeliveryOfficeName $ st $ l $ description
```

- Neste apartado da árbore DIT imos crear as ramas todas as OUs excepto **dam2** e **maquinas**.



Crear Unidades Organizativas

- Creamos un ficheiro **ou.ldif** co seguinte contido.

```
#Ficheiro ou.ldif

#Puxemos un atributo opcional para a descrición.

#Creamos a rama onde crearemos outras OUs para albergar usuarios.
dn: ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: usuarios
description: OU para almacenar usuarios.


#Creamos a rama profes DENTRO da rama usuarios.
#O nome da "ou" está mal para ser modificado despois.
#A descrición non está para ser engadida despois.
#Hai un atributo street que logo borraremos.
dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: profe
street: Rua Borrar n 3 (non usar tiles)
```

e executamos o comando:

```
ldapadd -D cn=admin,dc=iescalquera,dc=local -W -f ou.ldif
Enter LDAP Password:
adding new entry "ou=usuarios,dc=iescalquera,dc=local"

adding new entry "ou=profes,ou=usuarios,dc=iescalquera,dc=local"
```

- Se no ficheiro **ou.ldif** se atopa un erro nunha entrada non se segue coas seguintes, salvo que se use o parámetro **-c**.

- Comprobamos o efecto da execución da instrución anterior.

```
ldapsearch -x -b ou=usuarios,dc=iescalquera,dc=local
# extended LDIF
#
# LDAPv3
# base <ou=usuarios,dc=iescalquera,dc=local> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# usuarios, iescalquera.local
dn: ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: usuarios
description: OU para almacenar usuarios.

# profes, usuarios, iescalquera.local
dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: profe
ou: profes
street: Rua Borrar n 3 (non usar tiles)

# search result
search: 2
result: 0 Success

# numResponses: 3
# numEntries: 2
```

- Observar nas liñas 19-20 como ao equivocarnos na ou que non coincidía coa do dn, creou tamén a ou do dn. Pronto o arranxaremos.

Modificar Unidades Organizativas

- Creamos un ficheiro **ou_modif.ldif** co seguinte contido.
- Imos realizar operacións na OU profes.
    - Cambiar o valor do atributo *ou*
    - Engadirlle unha descrición
    - Eliminar a unha rúa.

```
#Ficheiro ou_modif.ldif

#Indicamos o obxecto que desexamos modificar
dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
changetype: modify
replace: ou
ou: profes
-
add: description
description: OU para almacenar usuarios profes.
-
delete: street
```

- Executamos o seguinte comando:

```
ldapadd -D cn=admin,dc=iescalquera,dc=local -W -f ou_modif.ldif
Enter LDAP Password:
modifying entry "ou=profes,ou=usuarios,dc=iescalquera,dc=local"
```

- Observamos o efecto da instrución anterior.
- Observar que só temos un atributo *ou*, que non está o atributo *street* e si *description*.

```
ldapsearch -x -b ou=profes,ou=usuarios,dc=iescalquera,dc=local
# extended LDIF
#
# LDAPv3
# base <ou=profes,ou=usuarios,dc=iescalquera,dc=local> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# profes, usuarios, iescalquera.local
dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: profes
description: OU para almacenar usuarios profes.

# search result
search: 2
result: 0 Success
```

Eliminar Unidades Organizativas

- Para eliminar unidades organizativas podemos usar:
    - **ldapadd**, onde lle pasamos o ficheiro cos DNs dos obxectos e a función a realizar con eles (eliminar).
    - **ldapdelete**, onde lle podemos pasar o ficheiro directamente cos DNs dos obxectos a eliminar ou mesmo podemos indicarllos por consola.

- **Exemplo 1: ldapadd**

```
#Ficheiro ou_borrar1.ldif
```

```
#Debemos borrar antes os obxectos fillos que o pai.
#Non o imos facer para propiciar un erro

#dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
#changetype: delete


dn: ou=usuarios,dc=iescalquera,dc=local
changetype: delete
```

- Executamos o seguinte comando e vemos que non nos borra a *ou=usuarios* porque contén fillos.
- Se descomentamos as liñas anteriores referentes á *ou=profes* o resultado sería positivo.

```
ldapadd -D cn=admin,dc=iescalquera,dc=local -W -f ou_borrar1.ldif
Enter LDAP Password:
deleting entry "ou=usuarios,dc=iescalquera,dc=local"
ldap_delete: Operation not allowed on non-leaf (66)
        additional info: subordinate objects must be deleted first
```

- **Exemplo 2: ldapdelete**
    - Podemos crear un ficheiro semellante ao seguinte onde só indicamos os DNs dos obxectos a borrar.
    - Igual que antes hai que indicar os obxectos fillos ante ca o pai.
    - Se se usa o parámetro **-r** podemos indicar só o pai e vai borrar os obxectos fillos en modo recursivo.

```
#Ficheiro ou_borrar2.ldif

#Debemos borrar antes os obxectos fillos que o pai.
#Non o imos facer para propiciar un erro

dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
dn: ou=usuarios,dc=iescalquera,dc=local
```

- Nesta ocasión non imos usar o ficheiro anterior, que se cargaría con -f igual que en ldapadd.
- Imos borrar toda á arbore de usuarios en modo recursivo (-r) coa instrución:

```
ldapdelete -D cn=admin,dc=iescalquera,dc=local -W "ou=usuarios,dc=iescalquera,dc=local" -r
Enter LDAP Password:
```

Unidades Organizativas: exercicio para o/a lector/a

- Todo o visto até agora é aplicable a calquera tipo de obxecto do ldap (usuarios, grupos, máquinas, etc)

- Agora o lector ou lectora debe ser quen de crear un ficheiro **ou_final.ldif** coas unidades organizativas seguintes:
    - dn: ou=usuarios,dc=iescalquera,dc=local
    - dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
    - dn: ou=grupos,dc=iescalquera,dc=local
    - dn: ou=alum,ou=usuarios,dc=iescalquera,dc=local
    - dn: ou=dam1,ou=alum,ou=usuarios,dc=iescalquera,dc=local
- Todas elas con descrición.

- Observar que non se crea a rama **dam2** dentro de **alum** nin a rama **maquinas**. Iso é para deixar cousas sen facer e realizalas a posteriori con outras utilidades.

- No directorio deberán estar os seguintes obxectos:

```
ldapsearch -x -b dc=iescalquera,dc=local objectClass=organizationalUnit
# extended LDIF
#
# LDAPv3
# base <dc=iescalquera,dc=local> with scope subtree
```

```
# filter: objectClass=organizationalUnit
# requesting: ALL
#

# usuarios, iescalquera.local
dn: ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: usuarios
description: OU para almacenar usuarios.

# profes, usuarios, iescalquera.local
dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: profes
description: OU para almacenar usuarios/as profes

# grupos, iescalquera.local
dn: ou=grupos,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: grupos
description: OU para almacenar os grupos

# alum, usuarios, iescalquera.local
dn: ou=alum,ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: alum
description: OU para almacenar os alumnos

# dam1, alum, usuarios, iescalquera.local
dn: ou=dam1,ou=alum,ou=usuarios,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: dam1
description: OU para almacenar os alumnos de DAM1
```

Administración de grupos

- Seguimos agora coa xestión dos grupos.
- O seguinte exemplo amosa o mínimo que debe conter un ficheiro base para manexar Grupos.

```
#ModeloGrupo.ldif
dn:cn=exemplo-nome-grupo,ou=exemplo-unidade-organizativa,dc=exemplo, dc=local
objectClass: posixGroup
cn: exemplo-nome-grupo
gidNumber: N-gid-Mellor-maior-de-9999
```

- Como vemos baséase no **objectClass: posixGroup** que se define no esquema **NIS**.
    - Obriga a ter 2 atributos: cn e gidNumber

```
objectclass ( 1.3.6.1.1.1.2.2 NAME 'posixGroup'
        DESC 'Abstraction of a group of accounts'
        SUP top STRUCTURAL
        MUST ( cn $ gidNumber )
        MAY ( userPassword $ memberUid $ description ) )
```

Crear grupos

- Neste apartado, dentro de "dn: ou=grupos,dc=iescalquera,dc=local", imos crear os grupos:
    - **g-usuarios**
    - e os grupos relacionados cos profes.
    - Observar que cada grupo ten un ID, debemos poñelo de forma manual.
    - Se poñemos o mesmo ID a distintos grupos, ldap non vai controlar iso, pero logo teremos problemas cos usuarios e os grupos aos que pertencen.

USUARIOS E GRUPOS

Grupos Usuarios	Nome Completo	g-usarios (10000)	g-profes (10001)	g-dam1-profes (10002)	g-dam2-profes (10003)	g-alum (10004)	g-dam1-alum (10005)	g-dam2-alum (10006)
Descric.		Todolos usuarios de LDAP	Todo o profesorado	Profesorado de 1º da DAM	Profesorado de 2º DAM	Todo o alumnado	Alumnado de 1º da DAM	Alumnado de 2º da DAM
sol (10000)	Profe - Sol Lúa	✔(1º)	✔	✔	✔			
noe (10001)	Profe - Noé Ras	✔(1º)	✔		✔			
mon (10002)	Dam1 - Mon Mon	✔(1º)				✔	✔	
tom (10003)	Dam1 - Tom Tom	✔(1º)				✔	✔	
pia (10004)	Dam2 - Pía Glez	✔(1º)				✔		✔
paz (10005)	Dam2 - Paz Fdez	✔(1º)				✔		✔

- Creamos un ficheiro **grupos.ldif** co seguinte contido

```
#Ficheiro grupos.ldif

#Xa sabemos que se non se engade o atributo cn:g-usuarios este vaise crear igual tomado da entrada DN.
#Quen o desexe pode engadir unha descrición.
dn: cn=g-usuarios,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
cn: g-usuarios
gidNumber: 10000
```

- Executamos o comando:

```
ldapadd -D cn=admin,dc=iescalquera,dc=local -W -f grupos.ldif
Enter LDAP Password:
adding new entry "cn=g-usuarios,ou=grupos,dc=iescalquera,dc=local"
```

- Observar as liñas 17-20 que indican que no directorio ldap xa está o grupo **g-usuarios**.

```
ldapsearch -x -b ou=grupos,dc=iescalquera,dc=local
# extended LDIF
#
# LDAPv3
# base <ou=grupos,dc=iescalquera,dc=local> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# grupos, iescalquera.local
dn: ou=grupos,dc=iescalquera,dc=local
objectClass: organizationalUnit
ou: grupos
description: OU para almacenar os grupos

# g-usuarios, grupos, iescalquera.local
dn: cn=g-usuarios,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
cn: g-usuarios
gidNumber: 10000
```

```
# search result
search: 2
result: 0 Success

# numResponses: 3
# numEntries: 2
```

Modificar/Eliminar grupos

- Procédese igual que se viu nas Unidades Organizativas.

Grupos: exercicio para o/a lector/a

- O lector ou lectora debe ser quen de crear un ficheiro **grupos_final.ldif** coas entradas seguintes e apoiándose na táboa superior:
    - dn: cn=g-usuarios,ou=grupos,dc=iescalquera,dc=local
    - dn: cn=g-profes,ou=grupos,dc=iescalquera,dc=local
    - dn: cn=g-dam1-profes,ou=grupos,dc=iescalquera,dc=local
    - dn: cn=g-dam2-profes,ou=grupos,dc=iescalquera,dc=local

- O atributo descrición é opcional.
- Lembrar que cando se execute **ldapadd** pódese engadir o parámetro **-c** por se atopa algunha entrada dentro do directorio que non pare e siga procesando o ficheiro.

- Unha vez introducidas as entradas no ldap débese obter o seguinte (lembrar que facía o parámetro -s):

```
ldapsearch -x -b ou=grupos,dc=iescalquera,dc=local -s one
# extended LDIF
#
# LDAPv3
# base <ou=grupos,dc=iescalquera,dc=local> with scope oneLevel
# filter: (objectclass=*)
# requesting: ALL
#

# g-usuarios, grupos, iescalquera.local
dn: cn=g-usuarios,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
cn: g-usuarios
gidNumber: 10000

# g-profes, grupos, iescalquera.local
dn: cn=g-profes,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
gidNumber: 10001
cn: g-profes

# g-dam1-profes, grupos, iescalquera.local
dn: cn=g-dam1-profes,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
gidNumber: 10002
cn: g-dam1-profes

# g-dam2-profes, grupos, iescalquera.local
dn: cn=g-dam2-profes,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
gidNumber: 10003
cn: g-dam2-profes
```

Administración de usuarios

- O seguinte exemplo amosa o mínimo que debe conter un ficheiro base para manexar Usuarios.

```
#ModeloUsuario.ldif
dn: uid=usuario,ou=ou_exemplo,dc=exemplo,dc=local
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
```

```
uid: usuario
sn: apelido
givenName: Nome de pila
cn: Nome Completo
displayName: Nome para amosar
uidNumber: 10000
gidNumber: 10000
userPassword: contrasinal
gecos: Informacion sobre o usuario (Opcional. Sen tiles)
loginShell: /bin/bash
homeDirectory: /home/usuario
mail: usuario@exemplo.local
initials: UA

#As seguintes entradas son opcionais, e serven para controlar a caducidade do contrasinal.
shadowExpire: -1
shadowFlag: 0
shadowWarning: 7
shadowMin: 8
shadowMax: 999999
shadowLastChange: 10877
```

- Como vimos no esquema **NIS** están os **objectClass: posixAccount** e **shadowAccount**
- Nas liñas 4 e 10 indican cales son os atributos obrigatorios. Neste caso *uid* é obrigatorio para as dúas clases de obxectos.

```
objectclass ( 1.3.6.1.1.1.2.0 NAME 'posixAccount'
        DESC 'Abstraction of an account with POSIX attributes'
        SUP top AUXILIARY
        MUST ( cn $ uid $ uidNumber $ gidNumber $ homeDirectory )
        MAY ( userPassword $ loginShell $ gecos $ description ) )

objectclass ( 1.3.6.1.1.1.2.1 NAME 'shadowAccount'
        DESC 'Additional attributes for shadow passwords'
        SUP top AUXILIARY
        MUST uid
        MAY ( userPassword $ shadowLastChange $ shadowMin $
              shadowMax $ shadowWarning $ shadowInactive $
              shadowExpire $ shadowFlag $ description ) )
```

- E no esquema **inetOrgPerson** está a *objectClass* do mesmo nome.
- Non hai atributos obrigatorios.
- Nas liñas 8 e 9 están 3 dos atributos do exemplo: *displayName*, *givenName* e *initials*.

```
objectclass     ( 2.16.840.1.113730.3.2.2
    NAME 'inetOrgPerson'
        DESC 'RFC2798: Internet Organizational Person'
    SUP organizationalPerson
    STRUCTURAL
        MAY (
                audio $ businessCategory $ carLicense $ departmentNumber $
                displayName $ employeeNumber $ employeeType $ givenName $
                homePhone $ homePostalAddress $ initials $ jpegPhoto $
                labeledURI $ mail $ manager $ mobile $ o $ pager $
                photo $ roomNumber $ secretary $ uid $ userCertificate $
                x500uniqueIdentifier $ preferredLanguage $
                userSMIMECertificate $ userPKCS12 )
        )
```

Crear usuarios

- Neste apartado, dentro de **dn: ou=profes,ou=usuarios,dc=iescalquera,dc=local**, imos crear os usuarios:
    - **sol**
    - **noe**
- Observar que cada usuario ten un ID, debemos poñelo de forma manual. Se poñemos o mesmo ID a distintos usuarios, ldap non controla iso, pero logo teremos problemas cos usuarios.

USUARIOS E GRUPOS

Grupos Usuarios	Nome Completo	g-usarios (10000)	g-profes (10001)	g-dam1-profes (10002)	g-dam2-profes (10003)	g-alum (10004)	g-dam1-alum (10005)	g-dam2-alum (10006)
Descric.		Todolos usuarios de LDAP	Todo o profesorado	Profesorado de 1º da DAM	Profesorado de 2º DAM	Todo o alumnado	Alumnado de 1º da DAM	Alumnado de 2º da DAM
sol (10000)	Profe - Sol Lúa	✔(1º)	✔	✔	✔			
noe (10001)	Profe - Noé Ras	✔(1º)	✔		✔			
mon (10002)	Dam1 - Mon Mon	✔(1º)				✔	✔	
tom (10003)	Dam1 - Tom Tom	✔(1º)				✔	✔	
pia (10004)	Dam2 - Pía Glez	✔(1º)				✔		✔
paz (10005)	Dam2 - Paz Fdez	✔(1º)				✔		✔

- **NOTA**: As tiles poden causar problemas ao copiar e pegar co cal, se non se desexa non é preciso usalas.

- Creamos un ficheiro **usuarios.ldif** co seguinte contido

```
#Ficheiro usuarios.ldif

#Observar onde se poñen tiles e onde non. Non é preciso poñelas
#Observar tamén que se indica o gid do grupo primario (10000 -> g-usuarios)
#Finalmente indicase que o contrasinal nunca expira.

dn: uid=sol,ou=profes,ou=usuarios,dc=iescalquera,dc=local
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: sol
sn: Lúa
cn: Profe - Sol Lúa
givenName: Sol
displayName: Profe - Sol Lúa
uidNumber: 10000
gidNumber: 10000
userPassword: abc123.
gecos: Profe - Sol Lua
loginShell: /bin/bash
homeDirectory: /home/sol
mail: sol@iescalquera.local
initials: SL
shadowExpire: -1
```

- Executamos o comando

```
ldapadd -D cn=admin,dc=iescalquera,dc=local -W -f usuarios.ldif
Enter LDAP Password:
adding new entry "uid=sol,ou=profes,ou=usuarios,dc=iescalquera,dc=local"
```

- Observar que aínda que indicamos que home do usuario é /**home**/**sol** non se vai crear. Iso arranxarémolo nas seguintes seccións e na parte III do curso.

```
ls /home
   administrador  lost+found
```

Engadir usuarios aos grupos secundarios

- Para engadir usuarios aos grupos secundarios debemos modificar cada grupo e engadir o atributo: **memberUid** co seu valor.
- Engadiremos tantos atributos como usuarios desexemos que pertenzan a ese grupo como secundario.

- Creamos o ficheiro **grupos_secundarios.ldif**

```
#Ficheiro grupos_secundarios.ldif

#Observar que os grupos sepáranse por unha liña en branco
#Se tivéramos máis usuarios para un grupo seguiríamos poñendo: memberUid: usuario
dn: cn=g-profes,ou=grupos,dc=iescalquera,dc=local
changetype: modify
add: memberUid
memberUid:sol

dn: cn=g-dam1-profes,ou=grupos,dc=iescalquera,dc=local
changetype: modify
add: memberUid
memberUid:sol

dn: cn=g-dam2-profes,ou=grupos,dc=iescalquera,dc=local
changetype: modify
add: memberUid
memberUid:sol
```

- Executamos o comando

```
ldapadd -D cn=admin,dc=iescalquera,dc=local -W -f grupos_secundarios.ldif -c
Enter LDAP Password:
modifying entry "cn=g-profes,ou=grupos,dc=iescalquera,dc=local"

modifying entry "cn=g-dam1-profes,ou=grupos,dc=iescalquera,dc=local"

modifying entry "cn=g-dam2-profes,ou=grupos,dc=iescalquera,dc=local"
```

Eliminar/modificar usuarios

- Faríase igual que coas Unidades Organizativas.
- Pero aquí hai un detalle: se eliminamos un usuario non significa que se elimine a entrada correspondente nos grupos secundarios aos que pertenza ese usuarios.
- Por tanto tamén habería que borrar a entrada *memberUid* nos grupos aos que pertence ese usuario, porque senón o sistema quedaría incoherente, pois o sistema deixa eliminar o usuario sen máis.

Usuarios: exercicio para o/a lector/a

- O lector ou lectora debe ser quen de crear un ficheiro **usuarios_final.ldif** para dar de alta aos seguintes usuarios (Observar a táboa superior):
- Non é preciso poñer as tiles.
    - uid=sol,ou=profes,ou=usuarios,dc=iescalquera,dc=local
    - uid=noe,ou=profes,ou=usuarios,dc=iescalquera,dc=local
- E crear un ficheiro **grupos_secundarios_final.ldif** para configurar os grupos secundarios aos que pertencen eses usuarios como se indica na táboa dos usuarios.

- Unha vez realizados os procesos debemos obter:

- **Usuarios:**
    - Notar que ldapsearch non amosa os caracteres UTF-8 (os que levan til, neste caso). Pero eses valores están ben almacenados.

```
ldapsearch -x -b ou=profes,ou=usuarios,dc=iescalquera,dc=local
# extended LDIF
#
# LDAPv3
# base <ou=profes,ou=usuarios,dc=iescalquera,dc=local> with scope oneLevel
```

```
# filter: (objectclass=*)
# requesting: ALL
#

# sol, profes, usuarios, iescalquera.local
dn: uid=sol,ou=profes,ou=usuarios,dc=iescalquera,dc=local
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: sol
sn:: TMO6YQ==
cn:: UHJvZmUgLSBTb2wgTMO6YQ==
givenName: Sol
displayName:: UHJvZmUgLSBTb2wgTMO6YQ==
uidNumber: 10000
gidNumber: 10000
gecos: Profe - Sol Lua
loginShell: /bin/bash
homeDirectory: /home/sol
mail: sol@iescalquera.local
initials: SL
shadowExpire: -1

# noe, profes, usuarios, iescalquera.local
dn: uid=noe,ou=profes,ou=usuarios,dc=iescalquera,dc=local
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: noe
sn: Ras
cn:: UHJvZmUgLSBOb8OpIFJhcw==
givenName: Noe
displayName:: UHJvZmUgLSBOb8OpIFJhcw==
uidNumber: 10001
gidNumber: 10000
gecos: Profe - Noe Ras
loginShell: /bin/bash
homeDirectory: /home/noe
mail: noe@iescalquera.local
initials: NR
shadowExpire: -1
```

- **Grupos:**
- Fixarse nas liñas marcadas.

```
ldapsearch -x -b ou=grupos,dc=iescalquera,dc=local
# extended LDIF
#
# LDAPv3
# base <ou=grupos,dc=iescalquera,dc=local> with scope oneLevel
# filter: (objectclass=*)
# requesting: ALL
#

# g-usuarios, grupos, iescalquera.local
dn: cn=g-usuarios,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
cn: g-usuarios
gidNumber: 10000

# g-profes, grupos, iescalquera.local
dn: cn=g-profes,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
gidNumber: 10001
cn: g-profes
memberUid: sol
memberUid: noe

# g-dam1-profes, grupos, iescalquera.local
dn: cn=g-dam1-profes,ou=grupos,dc=iescalquera,dc=local
```

```
objectClass: posixGroup
gidNumber: 10002
cn: g-dam1-profes
memberUid: sol

# g-dam2-profes, grupos, iescalquera.local
dn: cn=g-dam2-profes,ou=grupos,dc=iescalquera,dc=local
objectClass: posixGroup
gidNumber: 10003
cn: g-dam2-profes
memberUid: sol
memberUid: noe
```

Cambiar o contrasinal dun usuario

- Se non se especifica o contrasinal o sistema xera un aleatoriamente

```
ldappasswd -D cn=admin,dc=iescalquera,dc=local -W uid=sol,ou=profes,ou=usuarios,dc=iescalquera,dc=local
Enter LDAP Password:
New password: EL1U.58e
```

- Co parámetro **-S** pódese especificar o contrasinal desexado para o usuario.

```
ldappasswd -D cn=admin,dc=iescalquera,dc=local -W uid=sol,ou=profes,ou=usuarios,dc=iescalquera,dc=local -S
New password:
Re-enter new password:
Enter LDAP Password:
```

- Nos dous casos tamén pide o contrasinal de admin para poder facer as modificacións no directorio.

- Outra forma de cambiar o contrasinal sería creando un ficheiro ldif de modificacións

```
dn: uid=sol,ou=profes,ou=usuarios,dc=iescalquera,dc=local
changetype: modify
replace: userPassword
userPassword:novo contrasinal
```

- E cargar o ficheiro con **ldapadd**.

Ferramentas incluídas co servidor LDAP

- **slapindex**: Este comando pode ser moi útil en caso de apagados accidentais do servidor LDAP, xa que é posible que os índices usados para acceder á información do directorio se corrompan, o que pode producir erros na busca ou inserción de información no directorio ou incluso que o servidor LDAP non poida arrancar. A función do comando é rexenerar os índices a partir da información almacenada no directorio, creando así de novo as estruturas necesarias para que o servizo LDAP funcione correctamente.

- O comando debe executarse co mesmo usuario que executa o servidor LDAP (no caso de Debian/Ubuntu o usuario *openldap*), xa que os ficheiros de índices que este comando crea só poderán ser modificados polo usuario que executa o comando e executalo con outro usuario podería dar problemas de permisos ao arrancar o servidor LDAP.

```
root@dserver00:~# service slapd stop
  [ ok ] Stopping OpenLDAP: slapd.
root@dserver00:~# su - openldap -s /bin/bash
openldap@dserver00:~$ /usr/sbin/slapindex -v
  indexing id=00000001
  indexing id=00000002
  indexing id=00000014
  indexing id=00000015
  indexing id=00000016
  indexing id=00000017
  indexing id=00000018
  indexing id=00000019
```

```
  indexing id=0000001a
  indexing id=00000024
  indexing id=00000025
openldap@dserver00:~$ exit
  logout
root@dserver00:~# service slapd start
  [ ok ] Starting OpenLDAP: slapd.
```

-- Antonio de Andrés Lema e Carlos Carrión Álvarez --