**INTRODUCCION**

El proyecto consiste en la instalación de un servidor de autenticación de usuarios centralizado para las sucursales utilizando software libre, lo cual facilite la administración de los mismos al tener un único punto de autenticación. Esto nos va a facilitar principalmente la administración de las políticas de claves, actualmente la autenticación en las sucursales es local (/etc/passwd ), lo que vamos a utilizar para este proyecto es OpenLDAP como contenedor de los datos, DRBD para la réplica de la base de datos a un servidor secundario y heartbeat para realizar el failover de IP entre ambos, para el sistema operativo de base utilizamos CentOS 6.3 de 64 bits.

**CONFIGURACION DEL SERVIDOR PRIMARIO**

**Paquetes:**

Para comenzar inicialmente debemos instalar los siguientes paquetes que se enumeran a continuación en el servidor primario slnxldap1:

**OpenLDAP**

openldap-2.4.23-26.el6\_3.2.x86\_64

openldap-servers-2.4.23-26.el6\_3.2.x86\_64

openldap-clients-2.4.23-26.el6\_3.2.x86\_64

**DRBD**

kmod-drbd84-8.4.2-1.el6\_3.elrepo.x86\_64

drbd84-utils-8.4.2-1.el6.elrepo.x86\_64

**Configuración PAM LDAP**

pam\_ldap-185-11.el6.x86\_64

**NSS**

nss-pam-ldapd.x86\_64

**SSL**

openssl-perl-1.0.0-25.el6\_3.1.x86\_64

openssl-1.0.0-25.el6\_3.1.x86\_64

**Configuración de OpenLDAP**

OpenLDAP a partir de su versión 2.3 realizó grandes cambios estructurales que modificaron su forma de funcionamiento y configuración, en versiones posteriores disponíamos de el archivo slapd.conf en el cual realizábamos toda la configuración , ahora OpenLDAP es mas modular y nos permite realizar cambios en su configuración en caliente sin tener que reiniciar el servicio, imaginen una base de datos con 10000 usuarios y que para aplicar una política de acceso tengamos que reiniciar el servicio, esto generaría una disrupción del servicio. Con estos nuevos cambios OpenLDAP nos da más ventajas y nos facilita la administración. Si bien los cambios introdujeron un cierto grado de complejidad a la hora de su configuración, vale la pena el poder aprovechar estas nuevas prestaciones.

Si bien el archivo slapd.conf se mantiene por compatibilidad y podríamos configurar nuestro sistema usando ese archivo, no lo vamos a hacer y optaremos por la nueva modalidad, ahora OpenLDAP se divide en varios archivos los cuales algunos de ellos se encuentran dentro de la siguiente carpeta con extensión ldif:

**/etc/openldap/slapd./cn=config**

cn=module{0}.ldif

cn=schema

cn=schema.ldif

olcDatabase={0}config.ldif

olcDatabase={-1}frontend.ldif

olcDatabase={1}monitor.ldif

olcDatabase={2}bdb

olcDatabase={2}bdb.ldif

Estos archivos son de uso exclusivo de OpenLDAP y no se recomienda que se modifiquen para la configuración en general ya que OpenLDAP en alguna actualización puede sobre escribir los mismos y perderíamos nuestra configuración, pero para la creación inicial de nuestro sistema debemos modificar un archivo en particular donde definimos ciertas cosas, como ser, el nombre de nuestra organización la cual le va a dar la estructura incial, el archivo en cuestión es:

**olcDatabase={2}bdb.ldif**

En este archivo definimos como se va a llamar nuestra organización , es un nombre que va a ser referente para cuando necesitemos conectarnos, realizar consultas, etc y que lleva la siguiente nomenclatura:

**dc=coto,dc=com,dc=ar**

Es un nombre descriptivo, no tiene relación con un nombre de dominio tal cual conocemos o LTD, bien podríamos poner dc=coto,dc=com,dc=it,dc=tl, pero usamos una nomenclatura parecida para sea legible y de que sepamos que se trata de nuestro dominio, nuestro Active Directory o como lo define el propio LDAP un DIT por sus siglas en ingles Directory Information Tree, a continuación vemos el contenido del mismo:

**/etc/openldap/slapd.d/cn=config/olcDatabase={2\}bdb.ldif**

dn: olcDatabase={2}bdb

objectClass: olcDatabaseConfig

objectClass: olcBdbConfig

olcDatabase: {2}bdb

**olcSuffix: dc=coto,dc=com,dc=ar**

olcAddContentAcl: FALSE

olcLastMod: TRUE

olcMaxDerefDepth: 15

olcReadOnly: FALSE

olcRootDN: cn=Manager,dc=coto,dc=com,dc=ar

**olcRootPW:: e1NTSEF9RGUwMCtnVEJhdnNWY1ZBSllnR1ZROUpkRm1YVTlvbnEg**

olcSyncUseSubentry: FALSE

olcMonitoring: TRUE

olcDbDirectory: /var/lib/ldap

olcDbCacheSize: 1000

olcDbCheckpoint: 1024 15

olcDbNoSync: FALSE

olcDbDirtyRead: FALSE

olcDbIDLcacheSize: 0

olcDbIndex: objectClass pres,eq

olcDbIndex: cn pres,eq,sub

olcDbIndex: uid pres,eq,sub

olcDbIndex: uidNumber pres,eq

olcDbIndex: gidNumber pres,eq

olcDbIndex: mail pres,eq,sub

olcDbIndex: ou pres,eq,sub

olcDbIndex: loginShell pres,eq

olcDbIndex: sn pres,eq,sub

olcDbIndex: givenName pres,eq,sub

olcDbIndex: memberUid pres,eq,sub

olcDbIndex: nisMapName pres,eq,sub

olcDbIndex: nisMapEntry pres,eq,sub

olcDbLinearIndex: FALSE

olcDbMode: 0600

olcDbSearchStack: 16

olcDbShmKey: 0

olcDbCacheFree: 1

olcDbDNcacheSize: 0

structuralObjectClass: olcBdbConfig

entryUUID: c240c242-0b9c-1032-88e5-41250fb342bd

creatorsName: cn=config

createTimestamp: 20130215092101Z

olcAccess: {0}to attrs=userPassword by dn="cn=Manager,dc=coto,dc=com,dc=ar" wr

ite by anonymous auth by self write by \* none

olcAccess: {1}to dn.base="" by \* read

olcAccess: {2}to \* by dn="cn=Manager,dc=coto,dc=com,dc=ar" write by \* read

entryCSN: 20130221110734.561765Z#000000#000#000000

modifiersName: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth

modifyTimestamp: 20130221110734Z

Entre otras cosas que encontramos definidas en este archivo, es la clave del usuario Manager que es el administrador del dominio, vemos también los índices, ACL’s (Access Lists ) de acceso etc. Lo que nosotros vamos a configurar y lo único que vamos a tocar en este archivo el olcSuffix y el olcrootPW como se muestra resaltado en negritas, con eso definimos nuestro DIT, luego como veremos más adelante definiremos el resto de la estructura. Como se puede apreciar olcrootPW contiene un hash que contiene la clave del usuario Manager, para generar dicha clave utilizamos el comando **slappasswd** de la siguiente manera:

[root@slnxldap1 Configuracion]# slappasswd

New password:

Re-enter new password:

**{SSHA}/1rReSjxIuwel5lrSrHQCybuILc3HjPF**

Copiamos y pegamos el hash generado y eso es todo. Una vez que modificamos ese parámetro podemos iniciar por primera vez el servicio con **service slapd start** esto openldap va a crear una estructura vacía con una configuración inicial en /var/lib/ldap.

Continuando con la configuración de nuestro servidor, otro de los archivos que debemos modificar es el siguiente:

**/etc/openldap/ldap.conf**

URI ldap://slnxldap1

HOST slnxldap1

BASE dc=coto,dc=com,dc=ar

Este archivo contiene la información utilizada por los clientes en el mismo servidor ( PAM por ejemplo es un cliente LDAP ) , por ejemplo las utilidades de openldap como ser el comando ldapsearch utiliza la información contenida en ese archivo para realizar las búsquedas, si no configuramos este archivo tenemos que pasar por parámetro a las utilidades todo el nombre de nuestro servidor, por ejemplo, con solo ejecutar ldapsearch –x nos va a traer el contenido de nuestra base de datos, de lo contrario tendríamos que poner algo como lo siguiente:

**ldapsearch -x -b "dc=coto,dc=com,dc=ar"**

**Configuración de PAM**

PAM es el encargado de definir como nos vamos a autenticar al sistema, tenemos varios archivos para configurar, uno de ellos es **pam\_ldap** y define entre otras cosas a donde se van a ir a buscar los usuarios dentro de la base de datos, dentro de nuestro DIT, a continuación vemos los campos relevantes que conforman dicha configuración y que son fundamentales para su funcionamiento:

**/etc/pam\_ldap.conf**

host slnxldap1

base dc=coto,dc=com,dc=ar

uri ldap://slnxldap1/

binddn cn=Manager,dc=coto,dc=com,dc=ar

bindpw LdaP//2013

rootbinddn cn=Manager,dc=coto,dc=com,dc=ar

port 389

bind\_policy soft

pam\_lookup\_policy yes

pam\_password exop

nss\_base\_passwd ou=Usuarios,dc=coto,dc=com,dc=ar?one

nss\_base\_shadow ou=Usuarios,dc=coto,dc=com,dc=ar?one

nss\_base\_group ou=Grupos,dc=coto,dc=com,dc=ar?one

**/etc/nslcd.conf**

Otro archivo involucrado es el **nslcd.conf** que es el LDAP Name Service Daemon, en este archivo indicamos también donde se van a realizar las búsquedas de los usuarios, a continuación mostramos la parte relevante y necesaria para su funcionamiento:

uri ldap://slnxldap1/

base dc=coto,dc=com,dc=ar

binddn cn=Manager,dc=coto,dc=com,dc=ar

bindpw LdaP//2013

rootpwmoddn cn=Manager,dc=coto,dc=com,dc=ar

base group ou=Grupos,dc=coto,dc=com,dc=ar

base passwd ou=Usuarios,dc=coto,dc=com,dc=ar

base shadow ou=Usuarios,dc=coto,dc=com,dc=ar

uid nslcd

gid ldap

El siguiente archivo le indica al sistema operativo en qué orden vamos a realizar las búsquedas de los usuarios cuando nos autentiquemos al sistema, en nuestro caso le decimos que primero busque los usuarios localmente /etc/passwd ) y luego en LDAP:

**/etc/nsswitch.conf**

passwd: files ldap

shadow: files ldap

group: files ldap

El siguiente archivo forma parte de los varios que conforman la configuración de PAM, en nuestro caso éste es el único que modificaremos, el orden en que definimos los módulos de autenticación no es trivial, resaltamos en negritas como debe estar definido el módulo de LDAP y con esto terminamos lo que respecta a la autenticación en el servidor:

**/etc/pam.d/system-auth**

#%PAM-1.0

# This file is auto-generated.

# User changes will be destroyed the next time authconfig is run.

auth required pam\_env.so

auth sufficient pam\_unix.so nullok try\_first\_pass

auth requisite pam\_succeed\_if.so uid >= 500 quiet

**auth sufficient pam\_ldap.so use\_first\_pass**

auth required pam\_deny.so

account required pam\_unix.so broken\_shadow

account sufficient pam\_localuser.so

account sufficient pam\_succeed\_if.so uid < 500 quiet

**account [default=bad success=ok user\_unknown=ignore] pam\_ldap.so**

account required pam\_permit.so

password requisite pam\_cracklib.so try\_first\_pass retry=3

password sufficient pam\_unix.so sha512 shadow nullok try\_first\_pass use\_authtok

**password sufficient pam\_ldap.so use\_authtok**

password required pam\_deny.so

session optional pam\_keyinit.so revoke

session required pam\_limits.so

session optional pam\_mkhomedir.so umask=0022 skel=/etc/skel/

session [success=1 default=ignore] pam\_succeed\_if.so service in crond quiet use\_uid

session required pam\_unix.so

**session optional pam\_ldap.so**

**Configuración De La Estructura**

Ya con la autenticación configurada del lado del servidor procederemos a configurar el resto de la estructura de nuestra base de datos y estructura del DIT, esto incluye la creación del usuario Manager tanto para la administración del DIT en si como para la base de datos de configuración, los grupos, los usuarios y la política de claves. Para tal fin debemos utilizar la línea de comandos y generar algunos archivos de texto con determinada estructura. En el servidor **slnxldap1 128.2.1.42** bajo el directorio /root/temp/ldap tenemos algunos ejemplos de los archivos que utilizaremos para tal fin.

Antes que nada, debemos definir el usuario Manager tanto para la DIT como para la **cn=config**, como mencioné al principio de este documento, varias modificaciones estructurales se realizaron a partir de la versión 2.3 de OpenLDAP, entre ellas la administración de la base de datos, así que ahora tenemos un usuario Manager para la administración general de la DIT, donde están los usuario y grupos, y otro usuario Manager para la configuración de la base de datos donde se guarda la configuración referente al funcionamiento del motor de la base en sí, como ser, cargar módulos de monitoreo, módulos extras como en nuestro caso cuando creemos la política de claves, necesitaremos cargar un módulos u overlay que maneje e intérprete nuestras políticas de claves.

Primero que anda creamos al usuario Manager que va a administrar la configuración de OpenLDAP, vemos a continuación el formato que debe tener el archivo:

[root@slnxldap1 Configuracion]# cat rootDN.ldif

# El Admin de la Configuracion

dn: olcDatabase={0}config,cn=config

changetype: modify

replace: olcRootDN

olcRootDN: cn=Manager,cn=config

Utilizando el siguiente comando le damos de alta al usuario:

**ldapmodify -Y EXTERNAL -H ldapi:/// -f rootDN.ldif**

Ahora le asignamos una clave, vemos el formato del archivo en cuestión:

[root@slnxldap1 Configuracion]# cat rootDNPW.ldif

dn: olcDatabase={0}config,cn=config

changetype: modify

replace: olcRootPW

olcRootPW: {SSHA}hfO63bpJrIS/BkoyKxJoSC+ZiTsl+zda

olcRootPW contiene un hash con la clave del usuario, para generar dicho hash utilizamos el siguiente comando:

**slappasswd**

New password:

Re-enter new password:

**{SSHA}rs49arX+JRSPig/4AxmFzEl0n1Y1imla**

Copiamos y pegamos el hash conteniendo la clave y con el siguiente comando le modificamos la clave al usuario Manager:

**ldapmodify -Y EXTERNAL -H ldapi:/// -f rootDNPW.ldif**

Con esto ya tenemos nuestro usuario Manager que va a administrar la configuración de la base de datos, ahora ya estamos en condiciones de agregar el resto de la configuración, ahora lo que vamos a agregar es el nivel de registro, le vamos a decir que registre todo, ya que estamos instalando por primera vez, es bueno registrar todo evento para una mejor depuración por si se presenta algún inconveniente, con -1 le indicamos a la base que registre todo, el archivo en el cual se va a ir guardando los registros es **/var/log/slapd.log**

[root@slnxldap1 Configuracion]# cat loglevel.ldif

dn: cn=config

changetype: modify

replace: olcLogLevel

olcLogLevel: -1

Con el siguiente comando aplicamos el cambio, recordemos que ahora ya tenemos usuario y clave para el usuario Manager, así que ahora nos vamos a tener que autenticar:

**ldapadd -Wx -D "cn=Manager,cn=config" -f loglevel.ldif**

Ahora agreguemos el overlay que va a ser el encargado de manejar nuestras políticas de claves:

[root@slnxldap1 Configuracion]# cat module.ldif

dn: cn=module{0},cn=config

objectClass: olcConfig

objectClass: olcModuleList

objectClass: top

cn: module{0}

olcModuleLoad: {0}/usr/lib64/openldap/ppolicy.la

**ldapadd -Wx -D "cn=Manager,cn=config" -f module.ldiff**

Ahora agregamos algunas ACL’s básicas que nos van a proteger de que usuarios no autorizados hagan algún tipo de consulta, en resumen las siguientes ACL’s indican que no se permiten conexiones como usuarios anónimos y que el usuario Manager tiene acceso a todo, y que los usuario puedan cambiar su clave:

[root@slnxldap1 Configuracion]# cat acl.ldif

dn: olcDatabase={2}bdb,cn=config

changetype: modify

add: olcAccess

olcAccess: to attrs=userPassword by dn="cn=Manager,dc=coto,dc=com,dc=ar" write by anonymous auth by self write by \* none

olcAccess: to dn.base="" by \* read

olcAccess: to \* by dn="cn=Manager,dc=coto,dc=com,dc=ar" write by \* read

**ldapadd -Wx -D "cn=Manager,cn=config" -f acl.ldif**

Ahora creamos la estructura general de nuestro DIT, mostramos a continuación el formato del archivo que contiene la configuración de la misma:

[root@slnxldap1 Estructura]# cat estructura.ldif

dn: dc=coto,dc=com,dc=ar

dc: coto

objectClass: top

objectClass: domain

dn: ou=Policies,dc=coto,dc=com,dc=ar

objectClass: organizationalUnit

objectClass: top

ou: Policies

# Usuarios, coto

dn: ou=Usuarios,dc=coto,dc=com,dc=ar

ou: Users

objectClass: OrganizationalUnit

# Group, coto

dn: ou=Grupos,dc=coto,dc=com,dc=ar

ou: Groups

objectClass: OrganizationalUnit

**ldapadd -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f estructura.ldif**

Ahora que ya creamos nuestra estructura podemos agregamos las políticas de claves, mostramos el formato que contiene el archivo que crea las mismas:

[root@slnxldap1 Policies]# cat ppolicy.ldif

dn: cn=default,ou=Policies,dc=coto,dc=com,dc=ar

cn: default

objectClass: pwdPolicyChecker

objectClass: pwdPolicy

objectClass: person

objectClass: top

pwdMinAge: 604800

pwdMaxAge: 5184000

pwdMaxFailure: 5

pwdMustChange: TRUE

pwdAllowUserChange: TRUE

pwdAttribute: 2.5.4.35

pwdCheckQuality: 2

pwdExpireWarning: 432000

pwdFailureCountInterval: 30

pwdGraceAuthNLimit: 5

pwdInHistory: 5

pwdLockout: TRUE

pwdLockoutDuration: 1800

pwdSafeModify: FALSE

sn: Password Policies

Como las políticas son parte del DIT, o sea, se van a aplicar a los objetos que están incluidos dentro del DIT como ser grupos y usuarios ya no nos autenticamos a cn=config sino al dominio como se ve a continuación, con lo cual nos va a pedir clave de autenticación, creamos dos políticas una para usuarios normales y otra para usuarios y servicios sin expiración:

**ldapadd -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f ppolicy.ldif**

**ldapadd -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f policy-noexpire.ldif**

Ya hemos completado en su totalidad la configuración de nuestra estructura de DIT, ahora solo resta crear los usuarios los cuales podemos utilizar la herramienta web phpldapadmin o también desde la línea de comandos, esto lo veremos más adelante.

**CONFIGURACION DEL CLIENTE**

La configuración del cliente es similar al servidor, vemos a continuación los archivos involucrados, tomando como ejemplo la sucursal 250:

**/etc/openldap/ ldap.conf**

URI ldap://slnxldap1

HOST slnxldap1

BASE dc=coto,dc=com,dc=ar

**/etc/ldap.conf**

host slnxldap1

base dc=coto,dc=com,dc=ar

uri ldap://slnxldap1/

ldap\_version 3

binddn cn=Manager,dc=coto,dc=com,dc=ar

bindpw LdaP//2013

rootbinddn cn=Manager,dc=coto,dc=com,dc=ar

port 389

timelimit 120

bind\_timelimit 120

idle\_timelimit 3600

pam\_lookup\_policy yes

bind\_policy soft

pam\_password exop

nss\_base\_passwd ou=Users,dc=coto,dc=com,dc=ar?one

nss\_base\_shadow ou=Users,dc=coto,dc=com,dc=ar?one

nss\_base\_group ou=Group,dc=coto,dc=com,dc=ar?one

La diferencia del la configuración con el servidor donde le decimos que la búsqueda sea primero local y luego en LDAP, es que en el cliente hacemos a la inversa, primero en LDAP después local, esto no es porque sí, en las sucursales no vamos a borrar los usuarios locales, van a seguir definidos localmente en /etc/passwd, esto no afecta el funcionamiento del servidor y nos da otra forma de contingencia ante una catástrofe de que nuestro servidor de autenticación master y replica no pueden ser accedidos ya sea por un tema de red o alguna otra causa, , por eso indicamos que la búsqueda sea primero en LDAP, recordemos que el usuario root no está definido en LDAP, es siempre local, ante cualquier eventualidad siempre vamos a poder conectarnos como administradores al equipo ante cualquier problema.

**/etc/nsswitch.conf**

passwd: ldap files

shadow: ldap files

group: ldap files

**/etc/pam.d/system-auth**

# PAM modificado para estar en cumplimiento con PCI-DSS - no alterar

auth required /lib/security/$ISA/pam\_env.so

auth required /lib/security/$ISA/pam\_tally2.so deny=5 unlock\_time=1800

auth sufficient /lib/security/$ISA/pam\_unix.so likeauth nullok

**auth sufficient /lib/security/$ISA/pam\_ldap.so use\_first\_pass**

auth required /lib/security/$ISA/pam\_deny.so

account required /lib/security/$ISA/pam\_unix.so

account sufficient /lib/security/$ISA/pam\_succeed\_if.so uid < 100 quiet

**account sufficient /lib/security/$ISA/pam\_ldap.so use\_first\_pass**

account required /lib/security/$ISA/pam\_permit.so

password requisite /lib/security/$ISA/pam\_cracklib.so retry=3 minlen=7 dcredit=-2 ocredit=-2 difok=2 maxrepeat=2

password sufficient /lib/security/$ISA/pam\_unix.so nullok use\_authtok md5 shadow remember=12

**password sufficient /lib/security/$ISA/pam\_ldap.so**

password required /lib/security/$ISA/pam\_deny.so

session required /lib/security/$ISA/pam\_limits.so

**session sufficient /lib/security/$ISA/pam\_ldap.so**

session required /lib/security/$ISA/pam\_unix.so

**/etc/pam.d/sshd**

Agregar esta linea al final, con este modulo de pam el directorio se crea automaticamente si el mismo no existe cuando el Usuario se conecta por primera vez:

session optional pam\_mkhomedir.so skel=/etc/skel umask=0022

**ADMINISTRACION CON PHPLDAPADMIN**

PHPLDAPAdmin es una herramienta web que nos va a permitir administrar muchas cosas en una forma grafica y práctica, nos va a facilitar el alta y baja de usuarios, blanqueo de contraseñas, creación de grupos entre otras cosas, y nos va a mostrar en una forma grafica como está armado nuestra organización o DIT. Si bien nos va a facilitar hacer muchas cosas, hay operaciones que vamos a tener que ejecutar desde la línea de comandos como veremos más adelante.

**Configuración:**

Antes que nada tenemos que descargar el producto desde la siguiente url:

**http://phpldapadmin.sourceforge.net/wiki/index.php/Download**

Luego descomprimimos el paquete en /var/www/html y le asignamos los permisos al usuario y grupo apache:

[root@slnxldap1 html]#

drwxr-xr-x. 3 root root 1024 Feb 15 06:41 .

drwxr-xr-x. 6 root root 1024 Feb 15 06:40 ..

drwxrwxr-x. 11 apache apache 1024 Feb 15 06:46 phpldapadmin

Mover el archivo **config.php.example** a **config.php** dentro de la carpeta **/var/www/html/phpldapadmin/config**, con la configuración por defecto el software anda igual pero si queremos personalizar algunos campos por ejemplo el nombre de nuestro servidor modificamos los siguientes campos:

**$servers->setValue('server','name','Coto C.I.C.S.A LDAP Server');**

**Conectarse al Portal**

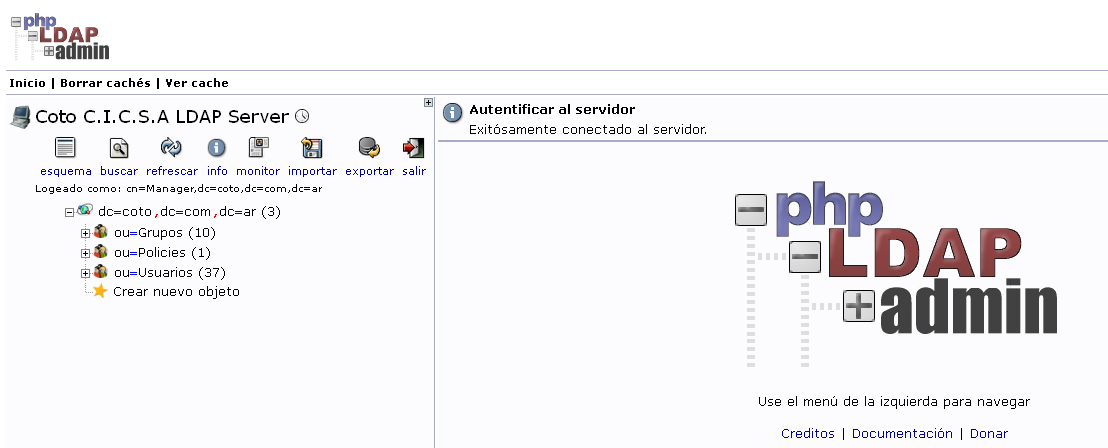
Utilizando nuestro navegador favorito accedemos al portal de phpldapadmin en la siguiente URL, la clave del usuario Manager está registrada en la utilidad Keepass, la forma de conectarse sigue la siguiente nomenclatura:

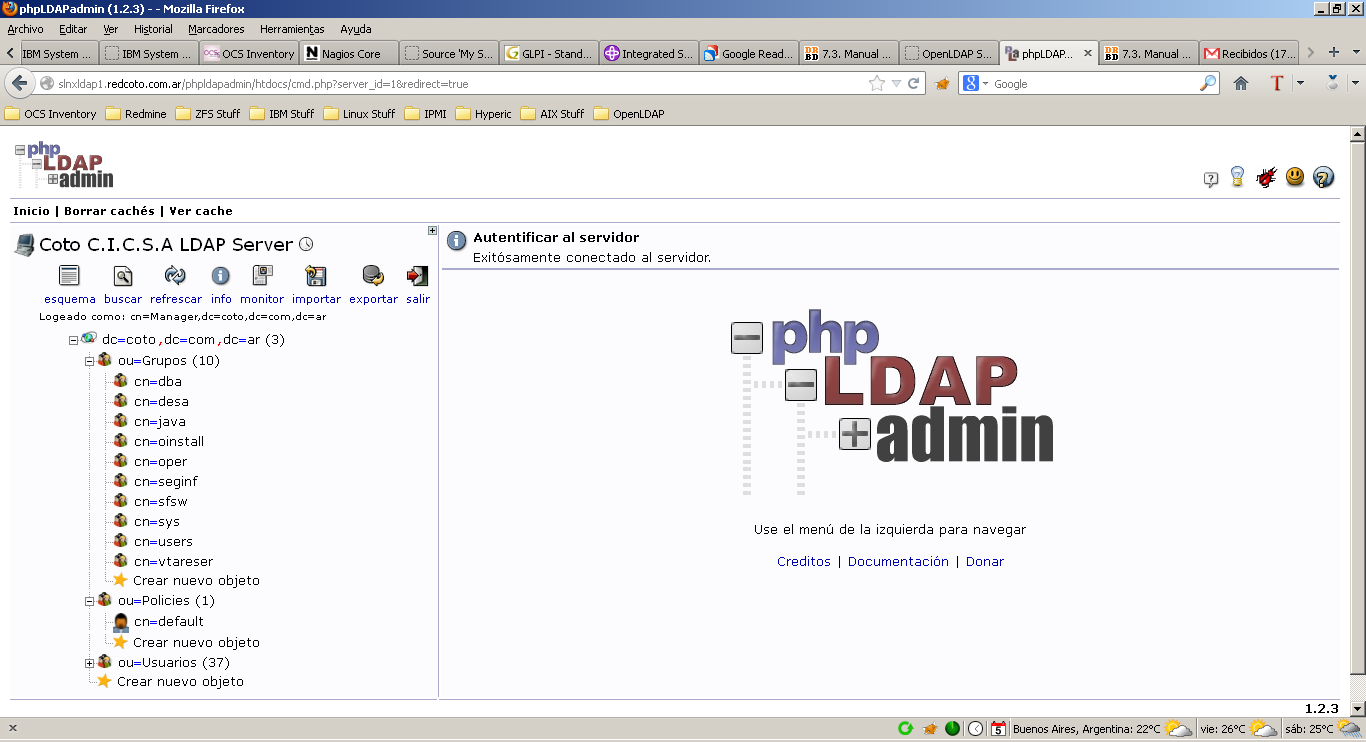
**cn=Manager, dc=coto,dc=com,dc=ar**

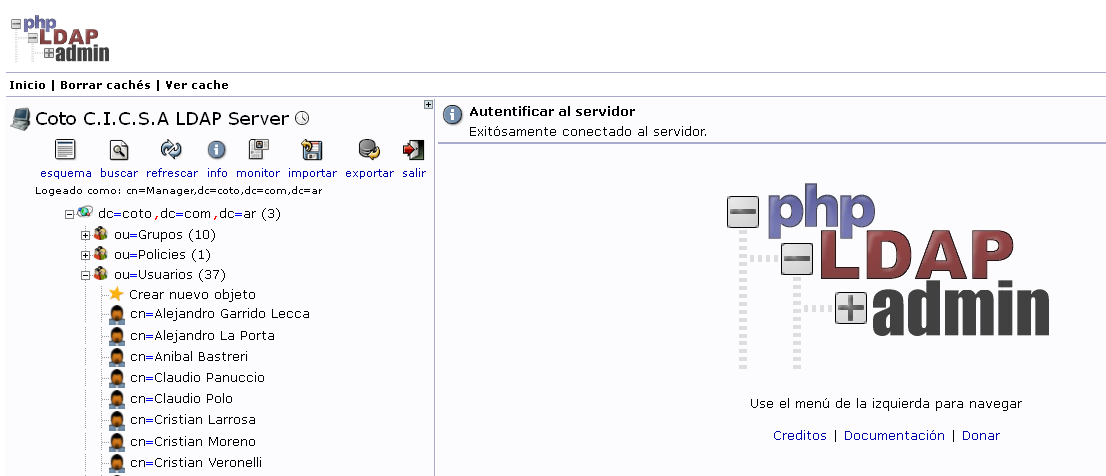
[**http://slnxldap1.redcoto.com.ar/phpldapadmin/htdocs/index.php**](http://slnxldap1.redcoto.com.ar/phpldapadmin/htdocs/index.php)

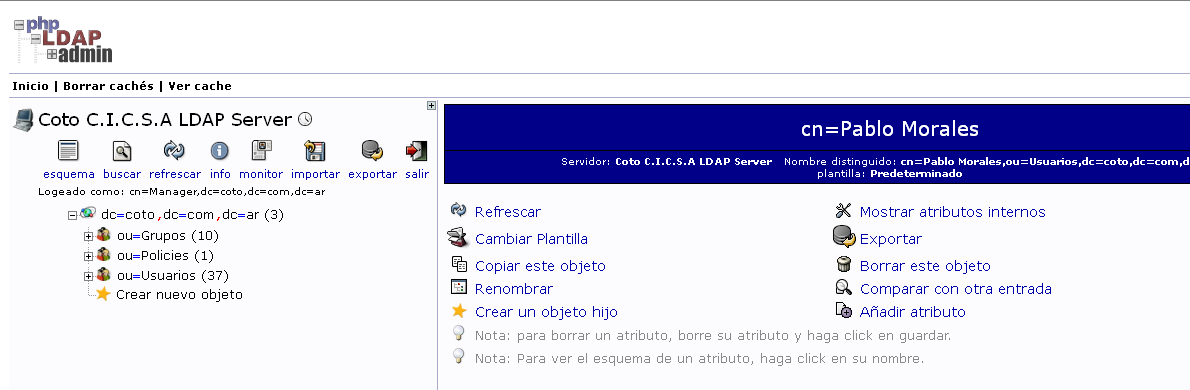


Una vez conectado vemos la siguiente estructura, a la derecha vemos los grupos, los usuarios y nuestra política de claves donde definimos el largo de la misma, cuando expira, etc.



**Acá vemos listados los grupos y la Password Policy:**

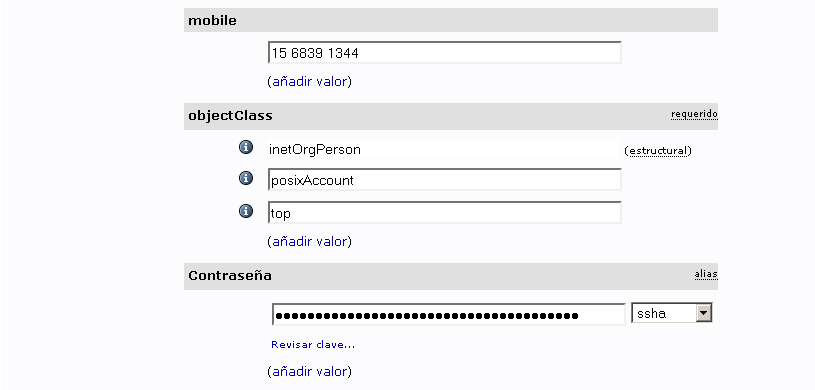
**Acá podemos ver una breve lista de los usuarios:**

**Podemos ver las propiedades de un usuario en particular:**



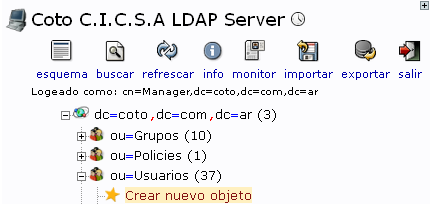


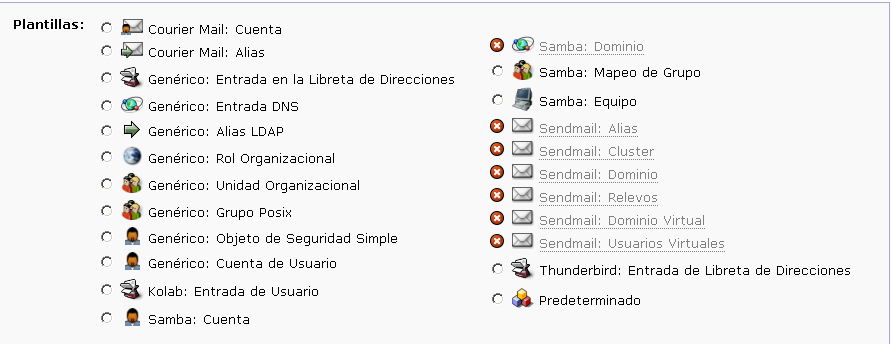




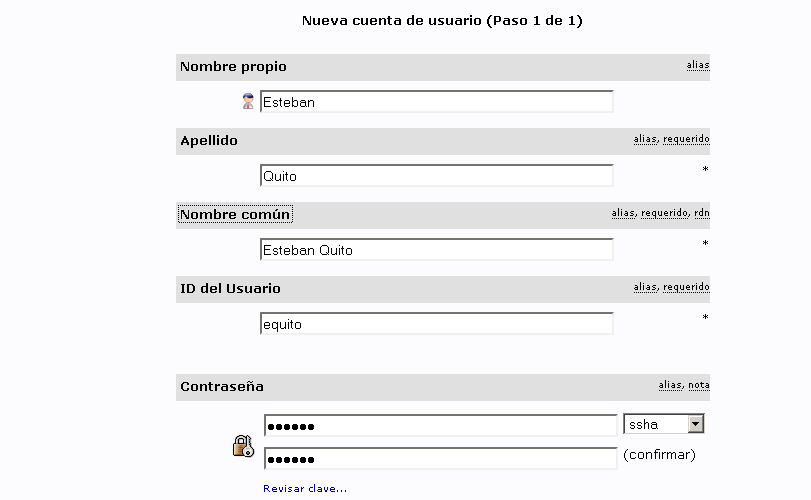


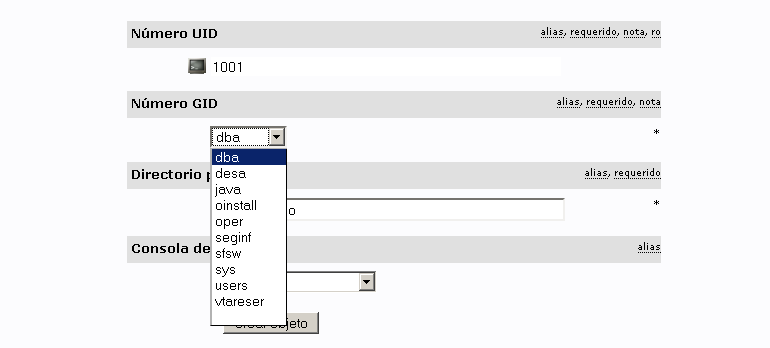
**Para agregar un usuario nuevo hacemos click en Crear nuevo objeto, con lo cual se nos despliega la siguiente pantalla donde debemos seleccionar Genérico Cuenta de Usuario:**

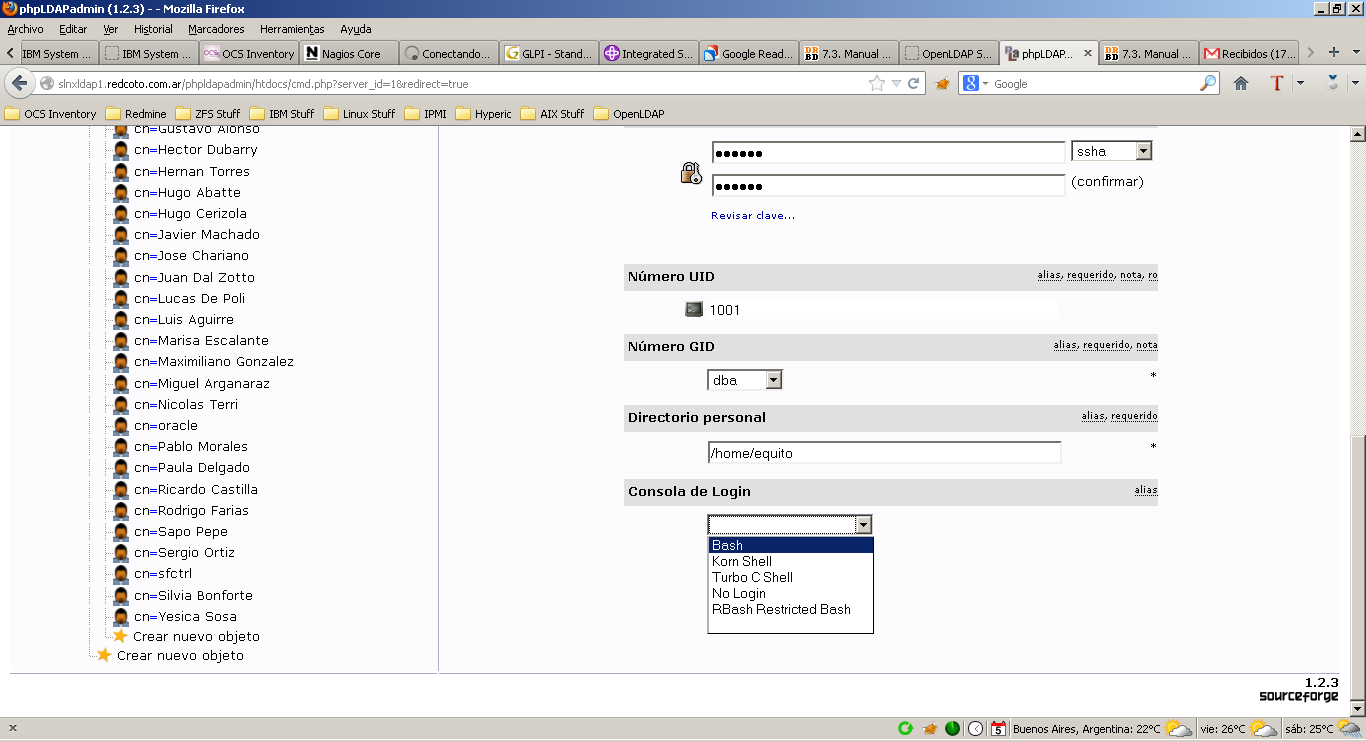


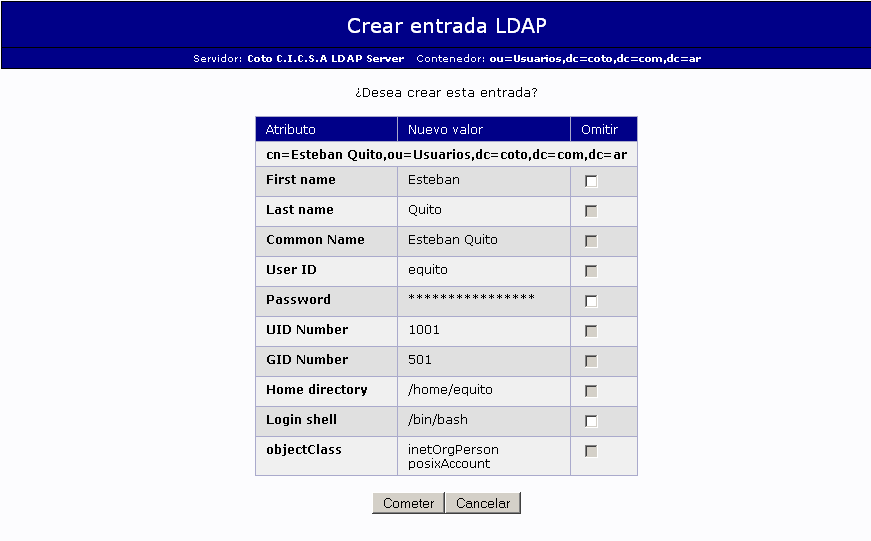


**Se nos despliega la siguiente pantalla donde pasamos a completar los datos del usuario en cuestión:**

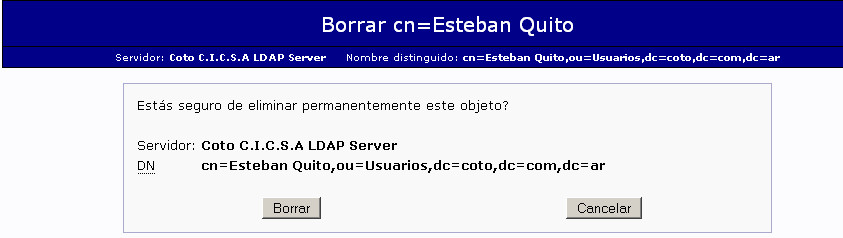


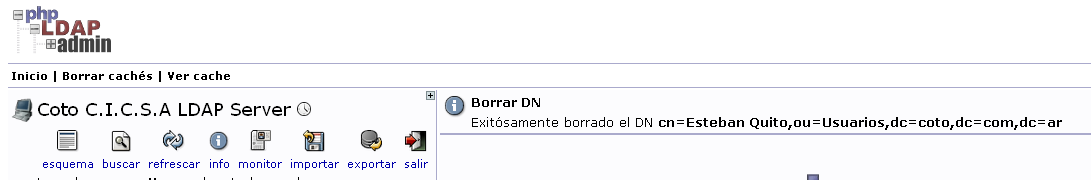
**Seleccionamos al grupo inicial al cual va a pertenecer, en este ejemplo el grupo dba:**

**Seleccionamos el tipo de intérprete de comandos que va a utilizar, entre ellos tenemos, bash, tcsh, nologin, ksh y Restricted Shell:**

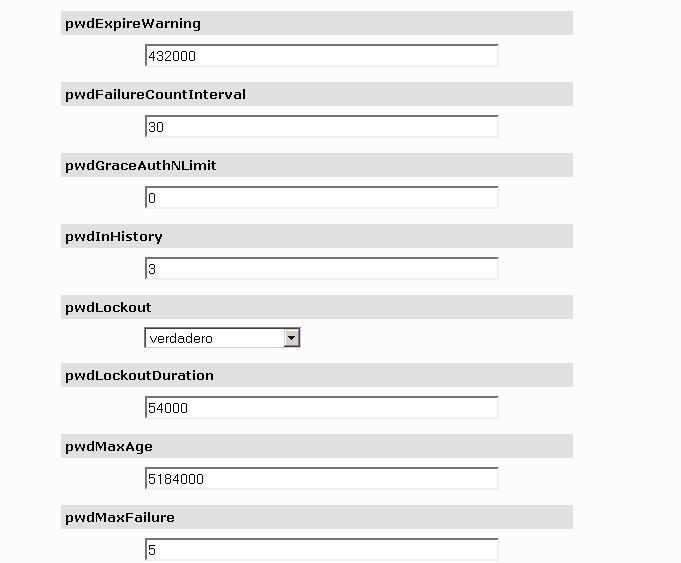
**Grabamos los cambios:**

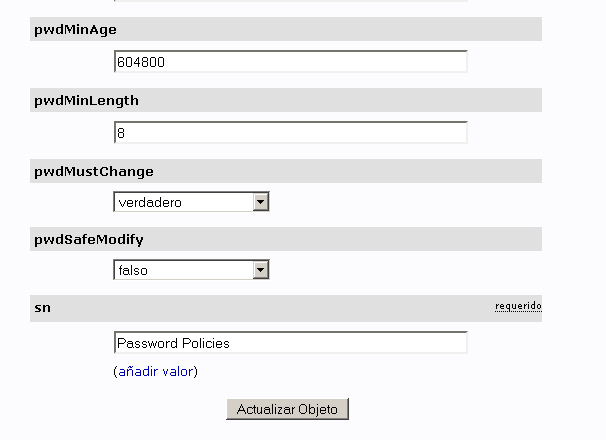
**Vemos el usuario recién creado:**

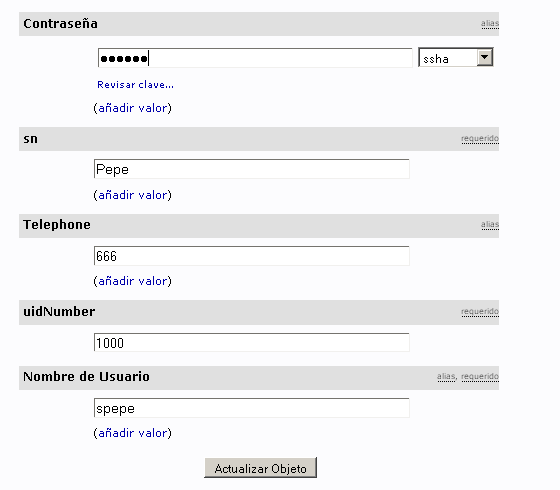
**Borramos el usuario a modo de ejemplo:**



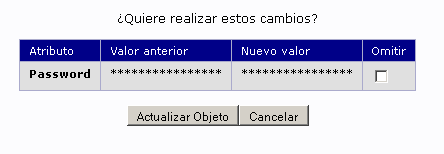
**Acá vemos las propiedades de nuestra política de contraseñas:**





**Actualizar la clave del usuario:**

**Hacer click en Actualizar Objeto**

****

**Hacer click en Actualizar Objeto**

**ADMINISTRACION DESDE LINEA DE COMANDOS**

Hay algunas tareas que vamos a tener que realizar desde la línea de comandos, como por ejemplo luego de crear un usuario las únicas dos tareas que debemos realizar son, forzarlo a que cambie su clave y también asignarle las políticas de claves.

**Aplicar políticas de claves:**

Una vez que creamos del usuario desde la herramienta web , lo primero que debemos hacer es aplicarle las políticas de claves, esto lo hacemos de la siguiente manera:

[root@slnxldap1 ldap]# cat agrego-user-policy.ldif

dn: cn=Pablo Morales,ou=Users,dc=coto,dc=com,dc=ar

changetype: modify

add: pwdPolicySubentry

pwdPolicySubentry: cn=default,ou=Policies,dc=coto,dc=com,dc=ar

**ldapadd -Wx -D "cn=Manager,dc=coto,dc=com,dc=arg" -f agrego-user-policy.ldif**

**Forzar cambio de clave:**

Una vez que se le aplican las políticas de claves al usuario, debemos forzarlo a que cambie su clave en su primer intento de autenticación:

[root@slnxldap1 Policies]# cat force-password-reset.ldif

dn: cn=Pablo Morales,ou=Users,dc=coto,dc=com,dc=ar

changetype: modify

add: pwdReset

**ldapadd -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f force-password-reset.ldif**

El usuario cuando quiera conectarse le va a aparecer el siguiente mensaje:

adm01/home/pmorales>ssh 128.2.107.40

pmorales@128.2.107.40's password:

You are required to change your LDAP password immediately.

Last login: Wed Feb 13 17:43:57 2013 from adm01.redcoto.com.ar

WARNING: Your password has expired.

You must change your password now and login again!

Changing password for user pmorales.

Enter login(LDAP) password

Debido a la política de claves aplicada, si el usuario requiere cambiar nuevamente su clave no lo va a poder hacer hasta pasado los 7 días y le va a aparece el siguiente mensaje:

[pmorales@sucxxx ~]$ passwd

Changing password for user pmorales.

Enter login(LDAP) password:

New UNIX password:

Retype new UNIX password:

LDAP password information update failed: Can't contact LDAP server

Password is too young to change

passwd: Permission denied

Cuando la clave estè por vencer aparecerà el siguiente mensaje:

pmorales@128.2.107.127's password:

Your LDAP password will expire in 3 days.

**Cambiar la clave a un usuario:**

Para cambiar la clave a un usuario usamos un archivo con el siguiente formato, en el siguiente ejemplo le cambiamos la clave a 123456.

[root@slnxldap1 ldap]# cat cambio-password.ldif

dn: cn=pablo morales,ou=Usuarios,dc=coto,dc=com,dc=ar

changetype: modify

replace: userPassword

userPassword: 123456

**ldapmodify -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f cambio-password.ldif**

**Desbloquear un usuario:**

En base a nuestra política de claves si un usuario no ingresa correctamente su clave al quinto intento el id se le va a bloquear, automáticamente al usuario se le creará un atributo llamado **pwdAccountLockedTime** el cual podemos ver dentro de las propiedades del usuario de la siguiente manera:

**ldapsearch -x "(uid=pmorales)" pwdAccountLockedTime**

# extended LDIF

#

# LDAPv3

# base <dc=coto,dc=com,dc=ar> (default) with scope subtree

# filter: (uid=pmorales)

# requesting: pwdAccountLockedTime

#

# Pablo Morales, Usuarios, coto.com.ar

dn: cn=Pablo Morales,ou=Usuarios,dc=coto,dc=com,dc=ar

**pwdAccountLockedTime: 20130305120124Z**

# search result

search: 2

result: 0 Success

Como se puede observar el usuario tiene el atributo **pwdAccountLockedTime** dentro de sus propiedades, lo que indica que no se va a poder autenticar en el servidor hasta que no pasen dos cosas, o pase el tiempo de bloqueo, **pwdLockoutDuration** que según nuestra política es de 15 minutos, o que el administrador le blanquee la clave como ilustramos más arriba. Para desbloquearlo usamos el siguiente formato de archivo:

[root@slnxldap1 ldap]# cat desbloqueo-usuario.ldif

dn: uid=pmorales,ou=Usuarios,dc=coto,dc=com,dc=ar

changetype: modify

delete: pwdAccountLockedTime

**ldapmodify -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f desbloqueo-usuario.ldif**

**Agregar un usuario:**

Utilizamos un archivo con la siguiente estructura, le generamos una clave 123456 con slappasswd y pegamos el hash:

[root@slnxldap1 Estructura]# cat usuario-nuevo.ldif

dn: cn=Osama Bin Laden,ou=Usuarios,dc=coto,dc=com,dc=ar

givenName: Osama

sn: Bin Laden

cn: Osama Bin Laden

uid: osama

userPassword:{SSHA}FPZHXatXVEGp7mEc7IQ0r3HJAEUDtMNO

gidNumber: 300

homeDirectory: /home/osama

loginShell: /bin/ksh

objectClass: inetOrgPerson

objectClass: posixAccount

objectClass: top

uidNumber: 999

**ldapadd -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f usuario-nuevo.ldif**

Enter LDAP Password:

**adding new entry "cn=Osama Bin Laden,ou=Usuarios,dc=coto,dc=com,dc=ar"**

Listamos el Usuario:

**[root@slnxldap1 Estructura]# ldapsearch -x "(uid=osama)"**

# extended LDIF

#

# LDAPv3

# base <dc=coto,dc=com,dc=ar> (default) with scope subtree

# filter: (uid=osama)

# requesting: ALL

# Osama Bin Laden, Usuarios, coto.com.ar

dn: cn=Osama Bin Laden,ou=Usuarios,dc=coto,dc=com,dc=ar

givenName: Osama

sn: Bin Laden

cn: Osama Bin Laden

uid: osama

gidNumber: 300

homeDirectory: /home/osama

loginShell: /bin/ksh

objectClass: inetOrgPerson

objectClass: posixAccount

objectClass: top

uidNumber: 999

# search result

search: 2

result: 0 Success

# numResponses: 2

# numEntries: 1

**Borrar un usuario:**

Utilizamos un archivo con la siguiente estructura:

[root@slnxldap1 Estructura]# cat borro-usuario.ldif

dn: cn=Osama Bin Laden,ou=Usuarios,dc=coto,dc=com,dc=ar

changetype: delete

**ldapmodify -Wx -D "cn=Manager,dc=coto,dc=com,dc=ar" -f borro-usuario.ldif**

Enter LDAP Password:

deleting entry "cn=Osama Bin Laden,ou=Usuarios,dc=coto,dc=com,dc=ar"

**[root@slnxldap1 Estructura]# ldapsearch -x "(uid=osama)"**

# extended LDIF

#

# LDAPv3

# base <dc=coto,dc=com,dc=ar> (default) with scope subtree

# filter: (uid=osama)

# requesting: ALL

**Generación de Índices:**

Para incrementar la velocidad de búsqueda de los usuarios, debemos generar los índices correspondientes, para ello algo importante a tener en cuenta es que para la generación de los mismos la base debe estar baja, esto es, bajando el servicio, una vez hecho eso, ejecutamos el siguiente comando:

[root@slnxldap1 ldap]# service slapd stop

Stopping slapd: [ OK ]

[root@slnxldap1 ldap]# slapindex -vv

The first database does not allow slapindex; using the first available one (2)

indexing id=00000001

indexing id=00000002

indexing id=00000003

indexing id=00000004

indexing id=00000005

indexing id=00000006

indexing id=00000007

indexing id=00000008

indexing id=00000009

indexing id=0000000a

[root@slnxldap1 ldap]# service slapd start

**Utilidades:**

Para facilitarnos la administración traté de ordenar los scripts según su función y que he utilizado para este tutorial, dentro del home de /root/temp/ldap hay diferentes directorios que contienen algunos archivos, a continuación paso describir el contenido de los mismos y su utilidad.

**Directorio Backup:**

Contiene un vuelco de la base de datos, esto es todos los usuarios y la configuración de la misma, la misma se genera con el comando slapcat –n0 y –n2 > nombre\_de\_archivo

n0 y n2 corresponde al número de la base de datos, recordemos que tenemos dos base de datos, uno para el DIT y otra para el cn=config.

[root@slnxldap1 Backups]#

backup.1-n0

backup.1-n2

backup.2-n0

backup.2-n2

**Directorio Configuración:**

Contiene los archivos de configuración inicial de la base de datos que usamos durante este tutorial, ellos son:

[root@slnxldap1 Configuracion]#

acl.ldif Access List

certificates.ldif Archivo para configurar certificados si queremos.

loglevel.ldif Archivo que utilizamos para cambiar el nivel de registro

manager.ldif Archivo que utilizamos para agregar el Usuario Manger al DIT

module.ldif Archivo que usamos para cargar el overlay de la Policy

rootDN.ldif Archivo que usamos para agregar el usuario Manager de la cn=config

rootDNPW.ldif Archivo que usamos para generar la clave para el usuario Manager

**Directorio Estructura:**

Contiene los archivos de configuración que utilizamos para la creación de la estructura de nuestra base de datos, o sea el DIT.

[root@slnxldap1 Estructura]#

estructura.ldif Archivo que define los grupos de la estructura

grupos.ldif Archivo que define grupos adicionales, java, sfw, etc.

usuarios.ldif Contiene un volcado de la base con los usuarios.

**Directorio Policies:**

Contiene los archivos de configuración de las políticas de claves de nuestra estructura.

agrega-todos-ppolicy.ldif Archivo para agregar masivamente a todos los usuarios

agrego-user-ppolicy.ldif Archivo para aplicar la política de claves al usuario

force-password-reset.ldif Archivo para forzar a un Usuario a que cambie su clave

modifico-ppolicy.ldif

ppolicy.ldif Contiene la definición de la política en si

todos-password-reset.ldif Para forzar a todos los usuarios a cambiar la clave.

unlock-user.ldif Archivo para desbloquear un usuario

**Directorio Raiz /root/temp/ldap:**

Contiene algunos archivos para el manejo de usuarios, alta, baja, modificaciones, etc.

borro-usuario.ldif Archivo para borrar un usuario

cambio-password.ldif Archivo para cambiar la clave a un usuario

crea-grupo.ldif Archivo para crear un grupo

desbloqueo-usuario.ldif Archivo para debloquear un usuario

fuerza-cambio-passowrd.ldif Archivo para forzar cambio de clave

usuario-nuevo.ldif Archivo para crear un usuario nuevo

**Servicios:**

Verificar que los siguientes servicios se encuentren en los siguientes estados de ejecución en ambos servidores:

**[root@slnxldap1 etc]# chkconfig --list**

drbd 0:off 1:off 2:on 3:on 4:on 5:on 6:off

httpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off

ip6tables 0:off 1:off 2:off 3:off 4:off 5:off 6:off

iptables 0:off 1:off 2:off 3:off 4:off 5:off 6:off

nscd 0:off 1:off 2:off 3:off 4:off 5:off 6:off

nslcd 0:off 1:off 2:on 3:on 4:on 5:on 6:off

ntpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off

ntpdate 0:off 1:off 2:off 3:off 4:off 5:off 6:off

slapd 0:off 1:off 2:off 3:off 4:off 5:off 6:off

**SELinux:**

SElinux debe estar deshabilitado, podemos comprobarlo de la siguiente manera:

[root@slnxldap1 sysconfig]# getenforce

Disabled

Para deshabilitarlo editar el siguiente archivo **/etc/sysconfig/selinux** y verificar que la etiqueta SELINUX contenga disabled como mostramos a continuación: **SELINUX=disabled**, luego reiniciar el servidor.

**Registro:**

Para poder registrar toda la actividad del servicio en /var/log/slapd.log debemos configurar rsyslog.conf de la siguiente manera, luego de modificarlo reiniciar el servicio con service rsyslog restart:

local4.emerg /var/log/slapd.log

local4.alert /var/log/slapd.log

local4.crit /var/log/slapd.log

local4.err /var/log/slapd.log

local4.warning /var/log/slapd.log

local4.notice /var/log/slapd.log

local4.info /var/log/slapd.log

local4.debug /var/log/slapd.log

**REPLICA CON DRBD**

El siguiente link contiene un tutorial de DRBD basado en una implementación similar que se realizó anteriormente, el documento tiene sus correcciones pertinentes dependiendo de la versión de DRBD que usemos, el procedimiento de creación de los logical volumes y puesta en funcionamiento es igual salvo donde se especifique que varíe dependiendo de la versión de DRBD que estemos utilizando, en este caso vamos a estar replicado el recurso /var/lib/ldap.

<http://intranet/SiteDirectory/gerenciadesistemas/unix/Documentos%20compartidos/Linux/DRBD.docx>

La lógica del funcionamiento es que DRBD como trabaja a bajo nivel , toma posesión del dispositivo, en nuestro caso el logical volumen donde se encuentran los datos de LDAP, entonces y por ese motivo en /etc/fstab vemos que tenemos comentado lo siguiente:

#/dev/drbd0 /var/lib/ldap ext4 rw,noatime,noauto 1 2

Como no tenemos un cluster, vamos a tener que montar manualmente el dispositivo en un caso que tengamos que reiniciar el servidor o tengamos que hacer failover al secundario, en cualquier caso el procedimiento es el siguiente:

**En el secundario vamos a ver lo siguiente si reiniciamos el primario:**

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: PingAck did not arrive in time.

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: peer( Primary -> Unknown ) conn( Connected -> NetworkFailure ) pdsk( UpToDate -> DUnknown )

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: asender terminated

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: Terminating asender thread

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: Connection closed

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: conn( NetworkFailure -> Unconnected )

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: receiver terminated

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: Restarting receiver thread

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: receiver (re)started

Mar 5 16:39:05 slnxldap2 kernel: d-con ldap: conn( Unconnected -> WFConnection )

**Cuando se vuelve a conectar vemos lo siguiente:**

Mar 5 16:39:15 slnxldap2 kernel: d-con ldap: Handshake successful: Agreed network protocol version 101

Mar 5 16:39:15 slnxldap2 kernel: d-con ldap: conn( WFConnection -> WFReportParams )

Mar 5 16:39:15 slnxldap2 kernel: d-con ldap: Starting asender thread (from drbd\_r\_ldap [6305])

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: drbd\_sync\_handshake:

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: self C4AE73F88F3E5CF8:0000000000000000:CAA685405438495E:CAA585405438495E bits:0 flags:0

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: peer 4EF81E5D7A86B0CC:C4AE73F88F3E5CF9:CAA685405438495E:CAA585405438495E bits:19456 flags:2

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: uuid\_compare()=-1 by rule 50

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: peer( Unknown -> Secondary ) conn( WFReportParams -> WFBitMapT ) disk( UpToDate -> Outdated ) pdsk( DUnknown -> UpToDate )

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: receive bitmap stats [Bytes(packets)]: plain 0(0), RLE 73(1), total 73; compression: 100.0%

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: send bitmap stats [Bytes(packets)]: plain 0(0), RLE 73(1), total 73; compression: 100.0%

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: conn( WFBitMapT -> WFSyncUUID )

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: updated sync uuid C4AF73F88F3E5CF8:0000000000000000:CAA685405438495E:CAA585405438495E

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: helper command: /sbin/drbdadm before-resync-target minor-0

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: helper command: /sbin/drbdadm before-resync-target minor-0 exit code 0 (0x0)

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: conn( WFSyncUUID -> SyncTarget ) disk( Outdated -> Inconsistent )

Mar 5 16:39:15 slnxldap2 kernel: block drbd0: Began resync as SyncTarget (will sync 77824 KB [19456 bits set]).

Mar 5 16:39:16 slnxldap2 kernel: block drbd0: Resync done (total 1 sec; paused 0 sec; 77824 K/sec)

Mar 5 16:39:16 slnxldap2 kernel: block drbd0: updated UUIDs 4EF81E5D7A86B0CC:0000000000000000:C4AF73F88F3E5CF8:C4AE73F88F3E5CF9

Mar 5 16:39:16 slnxldap2 kernel: block drbd0: conn( SyncTarget -> Connected ) disk( Inconsistent -> UpToDate )

Mar 5 16:39:16 slnxldap2 kernel: block drbd0: helper command: /sbin/drbdadm after-resync-target minor-0

Mar 5 16:39:16 slnxldap2 kernel: block drbd0: helper command: /sbin/drbdadm after-resync-target minor-0 exit code 0 (0x0)

**Una vez que volvemos a conectarnos al primario vemos el estado de la conexión:**

[root@slnxldap1 ~]# service drbd status

drbd driver loaded OK; device status:

version: 8.4.2 (api:1/proto:86-101)

GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6, 2012-09-06 08:16:10

m:res cs ro ds p mounted fstype

0:ldap Connected **Secondary/Secondary** UpToDate/UpToDate C

**Volvemos a ser primarios:**

Recordemos como explicamos en el documento al cual hago referencia sobre la configuración de DRBD que solo un nodo puede ser primario, al menos en este tipo de implementación que tenemos nosotros:

[root@slnxldap1 ~]# drbdadm primary ldap

[root@slnxldap1 ~]#

[root@slnxldap1 ~]# service drbd status

drbd driver loaded OK; device status:

version: 8.4.2 (api:1/proto:86-101)

GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6, 2012-09-06 08:16:10

m:res cs ro ds p mounted fstype

0:ldap Connected **Primary/Secondary** UpToDate/UpToDate C

[root@slnxldap1 ~]#

**Montamos el dispositivo:**

[root@slnxldap1 ~]# mount /dev/drbd0 /var/lib/ldap/

[root@slnxldap1 ~]#

**Iniciamos el servicio LDAP:**

[root@slnxldap1 ~]# mount /dev/drbd0 /var/lib/ldap/

[root@slnxldap1 ~]# service slapd start

Starting slapd: [ OK ]

**Archivos de configuración:**

[root@slnxldap1 drbd.d]# pwd

/etc/drbd.d

**global\_common.conf**

**ldap.res**

resource ldap {

# A que velocidad va a syncronizar, depende de la tarjeta de red

syncer {

rate 100M;

}

on slnxldap1 {

# Dispositivo que vamos a usar para montar luego el recurso

device /dev/drbd0;

# El volume que vamos a replicar

disk /dev/rootvg/ldap;

address 128.2.1.42:7789;

# Volumen para los metadatos

meta-disk /dev/rootvg/drbdldap[0];

}

on slnxldap2 {

device /dev/drbd0;

disk /dev/rootvg/ldap;

address 128.2.1.43:7789;

meta-disk /dev/rootvg/drbdldap[0];

}

}

FIN