Implantación de OpenLDAP y medición de su rendimiento

Luis Gerardo de la Fraga, Axel Ernesto Moreno Cervantes
y Guillermo Morales Luna

Sección de Computación

Departamento de Ingeniería Eléctrica

CINVESTAV-IPN

Av. Instituto Politécnico Nacional 2508. 07300 México, D.F.
E-mail: {fraga,gmorales}@cs.cinvestav.mx

Resumen

El Protocolo Ligero de Acceso a Directorio (Lightweight Directory Access Protocol) puede ser visto como un repositorio donde podemos colocar información para depués consultarla para su procesamiento. El repositorio se asemeja a una base de datos, pero en LDAP ha sido diseñada y optimizada para realizar operaciones de consulta. Con el uso de LDAP podemos centralizar información que pueden ser consultada por todos los clientes. En este trabajo se presenta la implantación de un servidor con OpenLDAP para autenticar a todos los usuarios de una red. También se presentan las medidas de su rendimiento que se han obtenido con distintos clientes y en redes con IPv4 e IPv6.

Palabras clave: LDAP, seguridad en redes, administración de redes.

Índice

1. Introducción

El Protocolo Ligero de Acceso a Directorio (LDAP en inglés) fue desarrollado en la Universidad de Michigan en 1993 para remover parte de la carga excesiva del acceso de X.500 desde los clientes del directorio. [1]. A LDAP se le modificaron muchas operaciones de X.500, retirando características poco usadas y emulando algunas operaciones con otras.

Las principales características de LDAP son:

- Está basado en el modelo cliente-servidor
- Organiza la información de modo jerárquico, utilizando directorios.
- Es capaz de propagar sus directorios a otros servidores LDAP
- Tiene un API de programación bien definido

Un directorio LDAP puede contener cualquier tipo de información, desde imágenes, direcciones de correo electrónico, contraseñas y referencias html, hasta certificados digitales, direcciones IP, etc.

La gran diversidad de información que puede ser almacenada en estos directorios los hace aptos para utilizarse en aplicaciones como:

- Directorios de páginas blancas o amarillas
- Servidores de correo electrónico
- Servidores de nombres de dominio (DNS)
- Repositorio para certificados digitales
- Repositorios de cuentas de usuario

por sólo mencionar algunas.

En este trabajo se presentará la utilización de LDAP como un repositorio de la información de cuentas de usuario. Por ejemplo, en una computadora GNU/Linux la lista de usuarios (en /etc/passwd), la de contraseñas (en /etc/shadow), y la de grupos (en /etc/shadow) puede ser manejada por un servidor LDAP y además todas las computadoras de los usuarios pueden encontrar estos datos en el servidor. Desde este punto de vista, LDAP facilita la administración de una red al centralizar la información usada tanto por el servicio de autenticación para todos los usuarios como por el Servicio de Conmutación de Nombres (NSS, Nameservice Switch).

La desventaja de LDAP es que resulta complicado de configurar [2] ya que es un sistema complejo. De hecho, la puesta a punto del sistema que se presentará llevó varias semanas y tuvo que activarse el archivos de autoría (log) durante este periodo de configuración para revizar donde estaban las fallas. La documentación primaria de LDAP puede encontrarse en [3] y [4].

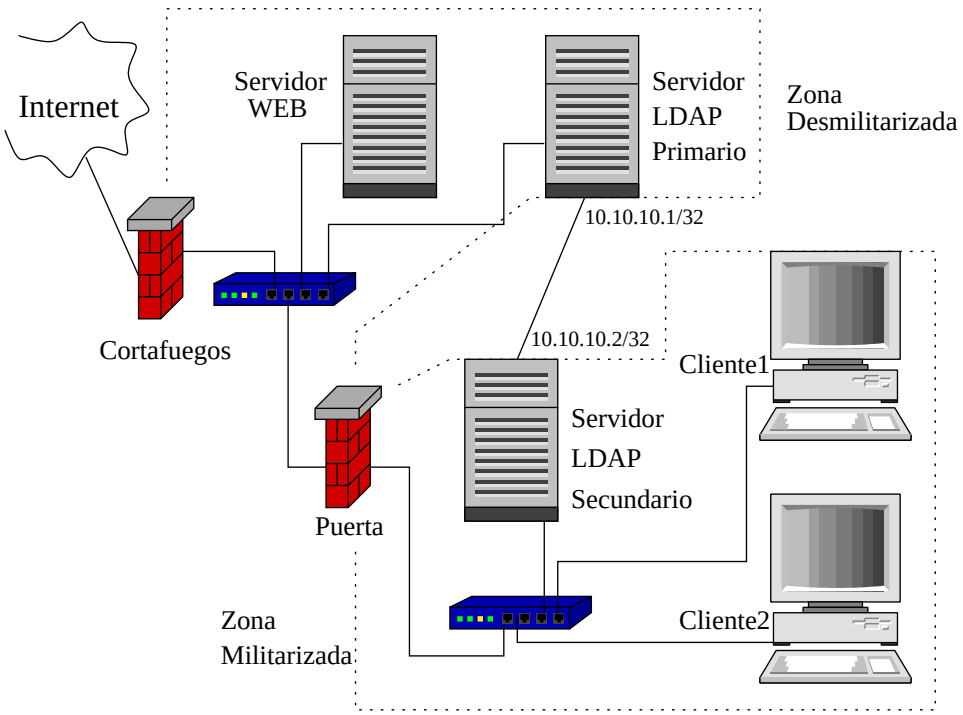

Figura 1: Implantación de LDAP en la red de la Sección de Computación del CINVESTAV. Detalles en el texto.

Primero se describirá el sistema que se implantó en la Sección de Computación del CINVESTAV y a continuación de describirán todos los detalles de la configuración y administración del sistema.

2. Sistema desarrollado

En la Fig. 1 se observa el sistema que se desea desarrollar. Los laboratorios y salas de estudiantes de la Sección de Computación se encuentran dentro de *zonas militarizadas* [5] (redes con direcciones IP privadas). Se cuentan con varias salas de estudiantes, la mayor con 30 computadoras con GNU/Linux; en la Fig. 1 sólo se muestran dos maquinas cliente dentro de una sola zona militarizada. La *puerta* de la Fig. 1 es una máquina GNU/Linux que realiza la traducción de dirección IP con IPTables, más detalles pueden encontrarse en la referencia [5].

Hasta aquí puede entenderse por qué es necesario un *servidor de autenticación*: tener actualizados los archivos `/etc/passwd`, `/etc/shadow` y `/etc/group` para todas la máquinas cliente en todos los laboratorios y salas de estudiantes es una tarea super ardúa si se quiere realizar a mano. Y no se tiene una solución estándar para mantener las contraseñas de los usuarios en todas las máquinas cliente si alguno de ellos desease cambiar su contraseña. Es aquí donde se justifica el uso del servidor de autenticación que nos permitirá:

- Administrar mejor la red. Se centraliza el dar de alta, baja o cambiar las cuentas y contraseñas de usuarios
- Un usuario podrá cambiar su contraseña desde cualquier máquina cliente.

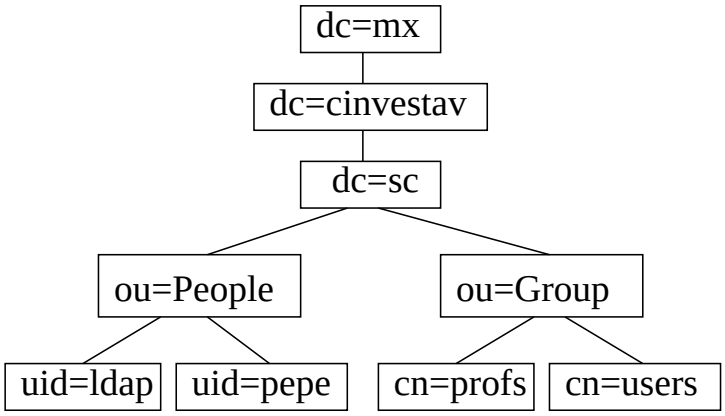

Figura 2: Estructura del directorio LDAP usado en el sistema propuesto

El servidor de autenticación primario se muestra en la Fig. 1 dentro de una *zona desmilitarizada* [5]. Este servidor LDAP primario permitirá el acceso al servidor para los estudiantes (no mostrado en la Fig. 1, pero debe estar dentro de la ZDM) desde Internet. En la Sección este servidor primario también se usa para autenticar a los usuarios de la red inalámbrica y es parte de un *punto caliente* realizado con NoCat [6].

Para no sobrecargar al servidor LDAP primario, que da servicio a la red inalámbrica y al servidor de los estudiantes, se hace una *replicación* hacia servidores LDAP secundarios para cada laboratorio. Para ello se ha realizado una nueva red, esquematizado en el cable más a la izquierda en la Fig. 1 siendo la red 10.10.10.0/24. La razón de esta nueva red será explicada en la sec. 7

El cortafuegos de la Fig. 1 tiene habilitada la entrada del servicio LDAP sobre SSL, puerto 636, para la IP de la puerta (y del servidor de estudiantes) y tiene bloqueada cualquier otra entrada (o tiene bloqueada la entrada desde Internet). La puerta tiene habilitada la entrada del servicio LDAP sobre SSL sólo desde el servidor LDAP primario.

El servidor de autenticación también podría realizarse con *radius* o *kerberos*. Un trabajo a futuro será comparar las distintas realizaciones del servidor.

3. Organización de la base de datos

Lo primer paso que tiene que realizarse para implantar LDAP es diseñar la forma que tendrá el directorio. La estructura que se maneja es de forma semejante a la estructura de árbol jerárquica usada en el DNS. En la Fig. 2 se muestra la estructura usada en nuestro sistema.

La primera decisión que hay que tomar en el nombre del *reino*, en nuestro caso es "dc=sc,dc=cinvestav,dc=mx". "dc" son las siglas en inglés de *componente de dominio*. Como estamos usando el servidor para guardar información de las cuentas de usuario de la Sección de Computación, bastaría con poner el nombre del reino a "dc=sc". Se ha puesto como "dc=sc,dc=cinvestav,dc=mx" por si alguna vez se extiende el servicio a toda la red del CINVESTAV.

El resto de la estructura de la Fig. 2, muestra que existen dos subdominios: "ou=People" y "ou=Group". "ou" son las siglas en inglés de *unidades organizacionales*. Estos subdominios fueron generadas automáticamente con las herramientas de migración que serán

explicadas en la sec. 5, por lo tanto no debemos preocuparnos más en como realizarlos. Dentro del subdominio "ou=People" existen los identificadores de usuario para "ldap" y "pepe" y existen dos grupos "profs" y "users".

4. Configuración del servidor de LDAP

Una vez que tenemos decidido el nombre del reino hay que configurar el servidor LDAP primario. El software necesario se instaló de los RPMS de RedHat (openldap, openldap-clients y openldap-servers).

El archivo que mantiene la configuración del servidor LDAP es `/etc/openldap/slapd.conf` que fue llenado como sigue:

```
1 include         /etc/openldap/schema/core.schema
2 include         /etc/openldap/schema/cosine.schema
3 include         /etc/openldap/schema/inetorgperson.schema
4 include         /etc/openldap/schema/nis.schema
5 include         /etc/openldap/schema/redhat/rfc822-MailMember.schema
6 include         /etc/openldap/schema/redhat/autofs.schema
7 include         /etc/openldap/schema/redhat/kerberosobject.schema
8
9 idletimeout 60
10
11 TLSCertificateFile /usr/share/ssl/certs/slapd.pem
12 TLSCertificateKeyFile /usr/share/ssl/certs/slapd.pem
13
14 # Control de acceso
15
16 access to dn=".*,dc=sc,dc=cinvestav,dc=mx" attr=userPassword
17         by dn="cn=Manager,dc=sc,dc=cinvestav,dc=mx" write
18         by dn="cn=replicator,dc=sc,dc=cinvestav,dc=mx" write
19         by dn="cn=proxyuser,dc=sc,dc=cinvestav,dc=mx" auth
20         by self write
21         by anonymous auth
22         by * none
23
24 access to *
25         by dn="cn=Manager,dc=sc,dc=cinvestav,dc=mx" write
26         by dn="cn=replicator,dc=sc,dc=cinvestav,dc=mx" write
27         by * read
28
29 ########################################
30 # Definiciones de la base de datos
31 ########################################
32
33 database        ldbm
34 lastmod         off
35 suffix          "dc=sc,dc=cinvestav,dc=mx"
36 rootdn          "cn=Manager,dc=sc,dc=cinvestav,dc=mx"
37 # rootpw                secret
```

```
38 rootpw           {SSHA}CipB6+P3iqIeV+XVZ2Tn8xbmhUk/MOnc
39
40 # The database directory MUST exist prior to running slapd AND
41 # should only be accessible by the slapd/tools. Mode 700 recommended.
42 directory       /var/lib/ldap
43
44 # Indices to maintain
45 index   objectClass,uid,uidNumber,gidNumber,memberUid   eq
46 index   cn,mail,surname,givenname                       eq,subinitial
```

Como el servidor se usa para autenticar las contraseñas de los usuarios, es muy importante que la comunicación entre los clientes y el servidor esté encriptada para que la información no pase en claro (y cualquiera en la red pueda *olerla*). Para gererar los certificados para la comunicación encriptada con TLS, se cambia uno al directorio `/usr/share/ssl/certs` y se ejecuta los comandos `make slapd.pem; chmod 640 slapd.pem; chgrp ldap slapd.pem`. Las líneas 11 y 12 del archivo de configuración indican donde se encuentras los certificados generados.

Una parte importantísima del archivo de configuración para el servidor es el control de acceso al directorio LDAP, viene en las líneas 16 a 27. Como se ha puesto significa lo siguiente:

- El acceso al subdirectorio `dn=".*,dc=sc,dc=cinvestav,dc=mx"` en el atributo `userPassword` se permite para los usarios `Manager` y `replicator` para escritura. El usuario `proxyuser` sólo puede autenticar sobre el mismo atributo `userPassword`. La línea `by self write` significa que un usuario ya registrado también puede cambiar su contraseña. El usuario `anonymous` sólo puede autenticar sobre el atributo `userPassword` y cualquier otro acceso es denegado.

- El acceso a todo el directorio sólo se permite para los usarios `Manager` y `replicator`, esto es, sólo ellos dos pueden crear, borrar y/o modificar al directorio. Cualquier otro usuario sólo puede leer el directorio (pero no el atributo `userPassword` que se controla con la primera lista de acceso.

A continuación en el archivo de configuración viene la parte que identifica al servidor en sí. La línea 33 indica el tipo de base de datos (`ldbm` por defecto). La opción en la línea 34, 'lastmod off', indica que no se use la información para dar seguimiento a los cambios que se efectúen sobre los objetos; esos cambios usan los atributos modifiersName, modifyTimestamp, creatorsName y createTimestamp [7] La línea 35 indica la raíz del directorio LDAP. La línea 38 tiene la contraseña para el usuario `Manager`, que es quien puede controlar todo el directorio. Esta contraseña fue generada con el comando `/usr/sbin/slappasswd`. Se puede comentar la línea 38 y descomentar la línea 37 si la contraseña se almacena dentro de la misma base de datos. El archivo de configuración tiene el modo 600 y pertenece al usuario LDAP.

Realizada las modificaciones se puede levantar el servidor de LDAP con la instrucción

```
/etc/rc.d/init.d/ldap start|stop|restart
```

Para probar el funcionamiento del servidor se realiza alguna consulta a él. Para ello ahora veremos como configurar un cliente y realizar algunas consultas.

4.1. Configuración de un cliente

El archivo de configuración `/etc/ldap.conf` es usado para aplicar los valores por defecto para los clientes de ldap. Este archivo tiene el contenido siguiente:

```
1 host 10.200.100.1
2
3 base dc=sc,dc=cinvestav,dc=mx
4
5 URI      ldaps://servidor.sc.cinvestav.mx
6
7 pam_login_attribute uid
8
9 # Specify a minium or maximum UID number allowed
10 pam_min_uid 100
11 pam_max_uid 9000
12
13 pam_password md5
14
15 nss_base_passwd     ou=People,dc=sc,dc=cinvestav,dc=mx?one
16 nss_base_shadow     ou=People,dc=sc,dc=cinvestav,dc=mx?one
17 nss_base_group      ou=Group,dc=sc,dc=cinvestav,dc=mx?one
18
19 ssl start_tls
20 ssl on
```

La línea 1 tiene la dirección IP del servidor LDAP. La línea 3 tiene al reino. La línea 5 es la forma de identificar al servidor (usa el protocolo ldap seguro, sobre TLS). Para activar la comunicación vía TLS son necesarias las líneas 13, 19 y 20. Las líneas 15-17 se usan por el Servicio de Conmutación de Nombres (NSS, Nameservice Switch) que será explicado en la sec. 6.

Ahora bien, para probar el servidor podemos agregar un usuario, que no aparezca en `/etc/passwd`, a la base de datos y realizar la consulta. Cómo agregar un usuario se explicará en la sec. 5. Supongamos ahora que este usuario de llama *prueba*. La consulta a la base de datos se realiza con el comando `ldapsearch` de la siguiente forma:

```
ldapsearch -x uid=prueba
```

y esto debe funcionar si tenemos bien configurado el archivo `/etc/ldap.conf` que usan los clientes de LDAP. La opción `-x` indica que se use el método de autenticación simple y es la única opción que debe utilizarse si está bien realizada la configuración hasta ahora.

5. Creación y mantenimiento de la base de datos

Se usaron las *herramientas de migración*, que en realidad son scripts hechos en perl, que vienen en la distribución en `/usr/share/openldap/migration`. Se usaron los scripts `migrate_base.pl`, `migrate_group.pl` y `migrate_passwd.pl`. Usando el primer primer script se generan los primeros registros:

```
./migrate_base.pl > base.ldif
```

y el contenido del archivo `base.ldif` debe de editarse ya que no decesitamos toda la información. Para que se refleje la información que se diseño en la Fig. 2 el contenido del archivo `base.ldif` debe ser:

```
dn: dc=sc,dc=cinvestav,dc=mx
dc: sc
objectClass: top
objectClass: domain

dn: ou=People,dc=sc,dc=cinvestav,dc=mx
ou: People
objectClass: top
objectClass: organizationalUnit

dn: ou=Group,dc=sc,dc=cinvestav,dc=mx
ou: Group
objectClass: top
objectClass: organizationalUnit
```

Este archivo tiene el formato LDIF que maneja LDAP.

El script `migrate_group.pl` cambia la información en `/etc/group` a formato LDIF. El script `migrate_passwd.pl` cambian la información de los archivos `/etc/passwd` y `/etc/shadow` a formato LDIF. Si las salidas de la ejecución respectiva de los scripts se manda a los archivos `group.ldif` y `usuarios.ldif`, necesitamos también editar estos archivos ya que no necesitamos todos los grupos y usuarios. Por ejemplo, no debe existir un registro para el usuario root.

Ahora podemos agredar los archivos `%.ldif` creados para llenar la base de datos. Esto se realiza con el comando `ldapadd`:

```
ldapadd -x -W -D 'cn=Manager,dc=sc,dc=cinvestav,dc=mx' -f base.ldif
ldapadd -x -W -D 'cn=Manager,dc=sc,dc=cinvestav,dc=mx' -f group.ldif
ldapadd -x -W -D 'cn=Manager,dc=sc,dc=cinvestav,dc=mx' -f usuarios.
    ldif
```

Se invita a que se revise las páginas man de este comando para identificar el significado de sus opciones.

Para agregar otro usuario lo que se ha realizado es editar un nuevo archivo `u.ldif` con un solo registro copiado del archivo `usuarios.ldif` y agregar el registro a la base de datos con el comando `ldapadd`. Para cambiar la contraseña el nuevo usuario se usa el comando `ldappasswd` de la siguiente forma:

```
ldappasswd -x -W -D 'cn=Manager,dc=sc,dc=cinvestav,dc=mx' 'uid=pepito
    ,ou=People,dc=sc,dc=cinvestav,dc=mx'
```

El comando anterior genera automáticamente una contraseña para el usuario 'pepito'.

Un usuario puede borrarse usando el comando `ldapdelete`. Por ejemplo, para borrar el usuario 'pepito' la línea sería:

```
ldapdelete -x -v -W -D 'cn=Manager,dc=sc,dc=cinvestav,dc=mx' 'uid=
    pepito,ou=People,dc=sc,dc=cinvestav,dc=mx'
```

Se pueden crear fácilmente nuevos scripts en perl para manipular listas de usuarios.

6. Autenticación

Ya que tenemos funcionando el servidor LDAP y haber configurado los clientes, es tiempo de configurar el servicio de autenticación de los clientes con el servifor LDAP.

La información del usuario consiste del mapeo entre los números de identificación de usuario y los nombres de usuario (que se usa, por ejemplo, al realizar el comando `ls -l`) ó la localizaciñon de los directios de casa (usado, por ejemplo, al realizar `cd ~`. La consulta a tal información es manejada por el subsistema del Servicio de Conmutación de Nombres (NSS, Nameservice Switch). La autenticación, que es el chequeo de las contraseñas, es manejada por el subsistema PAM (plugable authentication module). Estos dos subsistemas se configuran de forma separada, pero los requerimos para trabajar con LDAP.

El NSS se configura en el archivo `/etc/nsswitch.conf`. Deben de modificarse las líneas siguientes:

```
1 # /etc/nsswitch.conf
2 # con permisos 644
3 passwd:       files ldap
4 shadow:       files ldap
5 group:        files ldap
```

que indica que la información de usuario sea buscada en el servidor LDAP. Para probar que funciona, para un usuario que no existe en `/etc/passwd`, se puede hacer un `finger` o se puede crear cualquier archivo, asignale los uid y gid del usuario y hacer un `ls -l` sobre el archivo creado; los números del udi y gid deben de estar cambiados por los nombres respectivos.

La forma fácil de configurar la autenticación en RedHat es usando el comando `authconfig`. Este comando abre una interfaz gráfica y modifica los archivos `/etc/ldap.conf` y `/etc/pam.d/system-auth`. El contenido de este último archivo es el siguiente:

```
1 #%PAM-1.0
2 # This file is auto-generated.
3 # User changes will be destroyed the next time authconfig is run.
4 auth        required      /lib/security/pam_env.so
5 auth        sufficient    /lib/security/pam_unix.so likeauth nullok
6 auth        sufficient    /lib/security/pam_ldap.so use_first_pass
7 auth        required      /lib/security/pam_deny.so
8
9 account     required      /lib/security/pam_unix.so
10 account    [default=bad success=ok user_unknown=ignore service_err=
    ignore system_err=ignore] /lib/security/pam_ldap.so
11
12 password   required      /lib/security/pam_cracklib.so retry=3 type=
13 password   sufficient    /lib/security/pam_unix.so nullok
    use_authtok md5 shadow
14 password   sufficient    /lib/security/pam_ldap.so use_authtok
15 password   required      /lib/security/pam_deny.so
16
17 session    required      /lib/security/pam_limits.so
18 session    required      /lib/security/pam_unix.so
19 session    optional      /lib/security/pam_ldap.so
```

Para un usuario que queremos ingrese en los laboratorios pero no en el servidor LDAP, se puede agregar sus datos en el archivo `/etc/passwd`:

```
1 pepito:x:3106:540:Usuario pepito:/nohome:/bin/false
```

como no existe tanto su casa como el shell de inicio, no puede 'pepito' ingresar al servidor LDAP pero si usarlo para autenticarse para ingresar en las máquinas cliente.

Como el número de consultas puede ser grande en el servidor primario, se le activó el servicio de caché para el NSS. Ya estaba instalado, tuvo que activarse su ejecución cada vez que se inicia la máquina con el comando `chkconfig` y echarlo a andar con `/etc/rc.d/init.d/nscd start`.

7. Replicación del servidor LDAP

La configuración del servidor secundario LDAP se realiza de la misma forma que la configuración del servidor primario. Al final de la configuración (en el archivo `/etc/openldap/slapd.conf`) de debe agregar lo siguiente:

```
1 updatedn "cn=replicator,dc=cs,dc=cinvestav,dc=mx"
2 updateref ldaps://servidorldap.sc.cinvestav.mx
```

Éstas líneas del archivo de configuración para el servidor secundario tienen las directivas 'updatedn' y 'updateref'. La primera indica el nombre de identificación con que se harán las replicas y la segunda envía la replica al servidor primario cuando un cliente realiza una modificación en el servidor secundario.

A la configuración del servidor primario hay que agregarle además la siguiente información:

```
1 # Replicas to which we should propagate changes
2 replogfile      /var/lib/ldap/replog
3
4 replica host=replicador:389 tls=yes
5         binddn="cn=replicator,dc=cs,dc=cinvestav,dc=mx"
6         bindmethod=simple credentials=a890linux
```

Aunque parece que se está usando TLS, la implantación sólo funcionó a través del puerto 389. No se ha verificado aún si la información hacia el replicador viaja encriptada. Además, como la información solamente puede modificarse en el servidor primario y este propaga los cambios a los secundarios, la contraseña del administrador de los secundarios debe estar en claro en el archivo `/etc/openldap/slapd.conf` (como puede verse en la última línea del listado de arriba). Por ello el archivo `/etc/openldap/slapd.conf` debe tener el modo de sólo lectura y escritura para el usuario ldap. Por estas mismas razones y dado que el servidor secundario está dentro de una zona militarizada, se decidió poner una red alterna, la 10.10.10.0/24 (ver Fig. 1; de esta forma nos evitamos configurar la puerta para dejar pasar las conexiónes para realizar la replicación y aseguramos que nadie puede "oler" éstas conexiones.

En 'replicador' se usó tambien el tcp wrapper (en el archivo `/etc/hosts.allow` para permitir sólo el acceso del servidor primario a éste:

```
1 slapd:    10.10.10.1
```

En el servidor primario se configuró el tcp wrapper para permitir el acceso sólo a la puerta.

8. LDAP sobre IPv6 y otras plataformas

Se probó el rendimiento de un servidor LDAP en una red IPv6 contra clientes Windows XP, Solaris 9 y Max OSX 10.2. Se utilizó el mecanismo de transición *dual stack* en la red IPv6, lo que permite que el tratamiento de los paquetes sea el nativo de IPv6.

Se utilizaron 100 entradas en el directorio LDAP y cuatro máquinas cliente. Para el servidor se usó RedHat 9. Se probaron de 100 hasta 1000 transacciones por cada cliente. los resultados se muestran en la Fig. 3

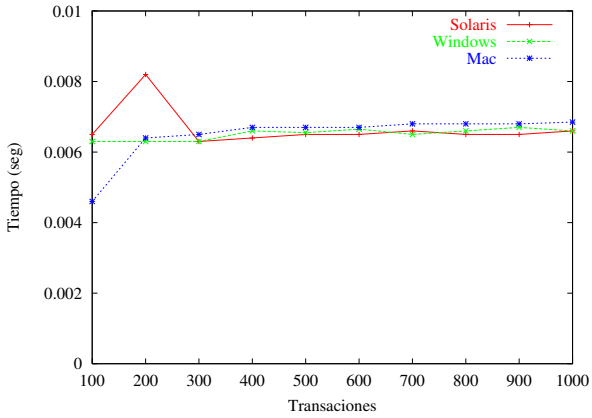

Figura 3: Rendimiento de un servidor LDAP

Los resultados de las pruebas en la Fig. 3 muestran que el tiempo de acceso al servidor LDAP por los distintos clientes no es afectado por la plataforma. Se presenta la misma eficiencia para los clientes Window, Mac y Solaris y así mismo se verifica que el servicio de LDAP funciona correctamente en IPv6 y que podría moverse toda nuestra red a IPv6 (al menos para este servicio).

9. Conclusiones

Se ha presentado la configuración de un servicio para centralizar la información y autenticación de usuarios usando LDAP. El proceso de configuración resultó una tarea árdua de varias semanas, conllevó la activación y revisión de los archivos de auditoría para la verificación del funcionamiento de cada parte de la configuración. El funcionamiento en IPv6 es eficiente por lo que se podría migrar este servicio a una red IPv6 sin ningún problema.

Como trabajo a futuro se probarán las realizaciones del servidor de autenticación con RADIUS y kerberos.

Agradecimientos

Este trabajo ha sido apoyado parcialmente por el proyecto 45306 de CONACyT.

Referencias

[1] W. Wahl, T. Howes, and S. Kille. Lightweight directory access protocol (v3), rfc 2251, 1997. www.faqs.org/rfcs/rfc2251.html.

[2] M. Bauer. Authenticate with ldap, part iii. *Linux Journal*, (113), September 2003. Disponible en www.linuxjournal.com.

[3] Sitio de openldap. En http://www.openldap.org.

[4] Information about installing, configuring, running and maintaining a ldap (lightweight directory access protocol) server on a linux machine. http://www.tldp.org/HOWTO/LDAP-HOWTO/.

[5] E. Bonilla and L.G. de la Fraga. Seguridad y configuración de redes de computadoras con GNU/Linux. *Congreso Nacional de Software Libre 2004 (CONSOL2004)*, Feb. 10-13 2004. http://www.consol.org.mx/2004.

[6] M. Kershaw. Linux-powered wireless hot spots. *Linux Journal*, (113), Sep 2003.

[7] S. Vugt. InDepth: The Lightweight Directory Access Protocol. *LinuxJournal*, 2001.