<u>LDAP</u>

<u>Lightweight Directory Access Protocol</u>

Servidor LDAP

Darío Martín Lorca
Administración de Sistemas y
Servicios Telemáticos
Junio 2000

Índice

Objetivos y descripción del servicio

¿Qué es un servicio de directorio?

Un directorio es como una base de datos, pero que tiende a contener información más descriptiva y basada en atributos. La información contenida generalmente se lee mucho más a menudo que se escribe. Un directorio, por tanto, debe ser capaz de dar respuestas rápidas a altos volúmenes de operaciones de búsqueda.

Los servicios de directorio suelen estar distribuidos a lo largo de varias máquinas. Típicamente se define un nombre de dominio que agrupa toda la información relacionada entre sí bajo un mismo nombre.

¿Qué es LDAP?

LDAP fue desarrollado en la Universidad de Michigan (el Umich LDAP) a principios de los años 90 como un protocolo de servicio de directorio más universal que el críptico y complejo protocolo OSI X.500. De éste se extrajeron sus mejores virtudes y, sobre todo, se orientó su uso a redes basadas en TCP/IP. Todos los detalles sobre LDAP están definidos en la RFC 1777 "*The Lightweight Directory Access Protocol*".

Para reducir el tiempo de búsqueda y aumentar la disponibilidad y fiabilidad de los datos, la información contenida en el servidor LDAP puede replicarse en varios puntos o, incluso, estar distribuida en varios servidores.

Se abandona el modelo de un solo servidor con un solo superusuario en favor de un modelo distribuido, multiservidor y multiadministrador.

¿Para qué sirve LDAP?

Hoy en día la mayoría de las aplicaciones de primer nivel orientadas a red implementan el soporte LDAP de alguna forma. Entre algunas de ellas, se encuentra Apache, Sendmail, IExplorer, Netscape y otras de las más importantes casas de software como HP, Novell, IBM, etc.

Elegir una aplicación que soporte el estándar LDAP para nuestra red nos asegura la ventaja de poder integrarla dentro de un sistema centralizado de administración y accesibilidad de la información.

Un ejemplo de lo que ya se puede conseguir con LDAP es el siguiente:

Existe una red de ordenadores en la que cualquier usuario puede trabajar en cualquier PC y que, tan solo validándose adecuadamente, tiene automáticamente disponible todo su entorno de escritorio personalizado a su gusto, con todas las aplicaciones que usa y sus ficheros de configuración.

La mayor facilidad con que provee LDAP a esa red es la facilidad de adaptarse a los cambios que se hagan en ella. Por ejemplo, cambiar una impresora de sitio solo conlleva actualizar los datos en el servidor LDAP para que todas las aplicaciones sepan acceder a ella.

Tan sólo con añadir un usuario al servidor LDAP ya lo deja listo para sentarse a trabajar y compartir información con otros trabajadores. Incluso un cambio en la dirección de correo electrónico de una persona, el lugar donde residan sus ficheros personales, la IP de un ordenador o un nombre de dominio, no supone ningún esfuerzo extra de configuración, ni siquiera es necesario informar a nadie de los nuevos cambios.

Es más, ese directorio LDAP puede publicarse en el web. Así, cualquier persona desde Internet es capaz de contactar con la persona de un departamento concreto, todo mediante niveles de permisos que restringen la información a mostrar. Ese mismo directorio puede servir a la vez a las aplicaciones de correo para localizar el *email* de una persona, autentificar usuarios y PCs en la red, etc.

¿Qué tipo de información se puede almacenar?

El modelo de servicio de directorio de LDAP está basado en entradas. Una entrada es un conjunto de atributos que tiene un nombre único que lo identifica unívocamente (DN: *Distinguished Name*). Cada atributo se define con un tipo y uno o más valores. Cada tipo es una cadena de caracteres, como *cn* (*common name*), o *mail* para dirección de mail. Los valores dependen del tipo de atributo. Por ejemplo *cn* puede tener "Pedro Pérez".

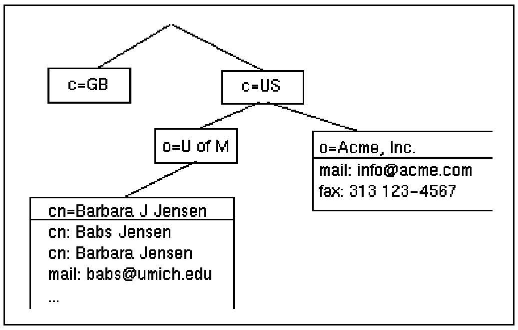
Los tipos de datos en la mayoría de los casos están definidos de forma estándar; éstos son los llamados *objectclass*. Es muy importante que todas las aplicaciones se basen en las mismas definiciones de clases, para que el directorio sea universal y no tenga que almacenar datos replicados. Por el momento, en los RFCs se han definido multitud de clases y se está trabajando en la definición de nuevas clases que puedan extender la funcionalidad de LDAP. Las clases se refieren a objetos reales, cuya definición y atributos es normalmente difícil de describir. Un buen sitio para encontrarlas puede ser

http://www.hklc.com/ldapschema
http://home.netscape.com/eng/server/directory/schema

que son unas clases promovidas y desarrolladas en conjunto por Netscape y el *Internet Engineering Task Force* (IETF).

Las entradas de directorio se agrupan en una estructura en árbol conocida como *Directory Information Tree* (DIT). Esta estructura jerárquica debería asemejarse a la estructura real de la empresa, es decir, dividirla según la localización de sus sucursales, unidades departamentales, etc. El nodo central del árbol se denomina sufijo (*suffix*). Es posible desarrollar un directorio LDAP con múltiples sufijos, muy útil para proveedores de Internet o grandes empresas.

Las entradas que representan países aparecen en la parte de arriba del árbol. Debajo, las que representan estados u organizaciones nacionales. Debajo, las que representan departamentos, personas, impresoras, documentos…



Además, cada entrada debe pertenecer al menos a dos clases de objetos, a *top* y a la clase que mejor lo defina. *Top* es una clase especial que debe estar presente por defecto en todas las entradas DN. En el fichero `/etc/openldap/slapd.oc.conf` se encuentran las clases, con sus atributos obligatorios y opcionales, que trae OpenLDAP.

Una entrada se referencia a través de su DN, que se construye tomando el nombre de la propia entrada (RDN: *Relative Distinguished Name*) y concatenando los nombre de las entradas anteriores en la jerarquía, hasta llegar a la raíz, como en el caso de un sistema de archivos donde hay una ruta o *path* hasta un nombre de fichero. Por ejemplo, la entrada de Barbara Jensen del ejemplo anterior, tiene un RDN "cn=Barbara J Jensen" y un DN "cn=Barbara J Jensen, o=U of M, c=US". El formato de DN se describe en la RFC 1779, "*A String Rrepresentation of Distinguished Names*".

LDAP define una serie de operaciones para actualizar y solicitar entradas dentro del directorio. También para añadir y quitar entradas, cambiar una entrada existente o cambiar su nombre. También existen métodos de autentificación para el acceso de los clientes a la base de datos.

¿Qué es slapd*?*

Slapd es un servidor de directorio LDAP capaz de correr en numerosas plataformas UNIX. Se puede utilizar para un servicio de directorio propio, privado, o conectarlo al servicio global de LDAP. Sus principales características son:

- Elección de bases de datos: *slapd* soporta tres tipos diferentes: LDBM, una base de datos de alto rendimiento; SHELL, un interfaz de base de datos para comandos UNIX o *scripts* de *shell*; y PASSWD, un simple fichero de *passwords*.

- Múltiples instancias de bases de datos. Significa que un simple servidor LDAP puede responder a peticiones de diferentes porciones del árbol LDAP usando la propia u otras bases de datos diferentes.

- API de bases de datos genérica. Gracias a la API de C de *slapd* se pueden escribir *backends* de bases de datos para *slapd* personalizados.

- Control de acceso a las entradas basadas en datos de autentificación, dirección IP, nombre de dominio, etc.

- Hebras. *Slapd* utiliza hebras para asegurar un alto rendimiento.

- Replicación. El esquema que utiliza es el de maestro-esclavo, vital para entornos con altos volúmenes de información.

- Altamente configurable a través de un archivo de texto de configuración.

Slapd también tiene limitaciones. No soporta alias. El principal *backend* de bases de datos, LDBM, no soporta peticiones de rangos o de negaciones muy bien, etc.

¿Qué es slurpd*?*

Slurpd es un demonio de UNIX que ayuda a proporcionar a *slapd* el servicio de replicación. Es responsable de distribuir los cambios hechos en la base de datos maestra de *slapd* a las distintas réplicas. Libera a *slapd* de preocuparse de que algunas réplicas puedan estar caídas o inalcanzables cuando sea necesario actualizar un cambio; *slurpd* se encarga de reintentarlo automáticamente. *Slapd* y *slurpd* se comunican a través de un simple archivo de texto de *log*, que se usa para archivar los cambios.

Requisitos

El desarrollo del Umich LDAP se congeló después de que Netscape, en un alarde de visión de futuro, contratara a sus principales desarrolladores para llevárselos a trabajar en su *Directory Server*. Después de un tiempo de confusión, con parches y versiones no oficiales, se creó la *OpenLDAP Foundation* para asegurar la continuidad del desarrollo de una plataforma *Open Source* para LDAP. Para más información y obtener los paquetes de binarios consultar OpenLDAP:

http://www.openldap.org/

OpenLDAP 1.2 se deriva de la distribución de LDAP v3.3 de la Universidad de Michigan. Se puede encontrar más información acerca de LDAP v3.3 en:

http://www.umich.edu/~dirsvcs/ldap/ldap.html

En el caso de la distribución de Debian, la instalación de OpenLDAP se realiza instalando el paquete `openlapd` con

```
apt-get install openlapd
```

En ese paquete se encuentran el servidor *slapd,* el demonio *slurpd,* las utilidades para actualizar y mantener el directorio, el cliente LDAP, los archivos de configuración, etc.

Dentro del contenido del paquete no existe ningún archivo que indique los requisitos software para instalarlo, cosa bastante rara en una documentación de un paquete en el mundo Linux. De todos modos todos los problemas de dependencias de paquetes las trata *apt-get*, siendo bastante automático el proceso. En cuanto a problemas de con qué versión de *glibc* ha sido compilado y algún otro requisito especial lo mejor es ir al sitio *oficial* y consultar la documentación en el área de descarga, existiendo paquetes de binarios para casi todas las configuraciones y distribuciones y en formato *tgz*, con las fuentes, en caso de que sea necesario compilarlo.

Una extensa documentación sobre administración de *slapd* y *slurpd* puede encontrarse en el paquete `openldap-guide`. También está disponible en postscript, pdf y html en la página de OpenLDAP.

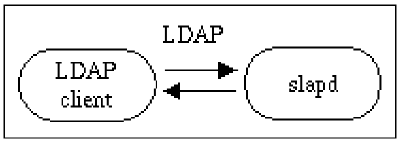
Arquitectura del servicio LDAP

LDAP presenta una arquitectura basada en el modelo cliente-servidor. Uno o más servidores LDAP contienen los datos que forman el árbol de directorio. Un cliente LDAP se conecta a un servidor LDAP y solicita una información. El servidor responde con los datos requeridos o con un puntero a donde el cliente puede obtener más información. No importa a qué servidor se conecte el cliente, el DN que solicita a un servidor LDAP se refiere a la misma entrada que si estuviera en otro servidor LDAP.

Formas de configuración

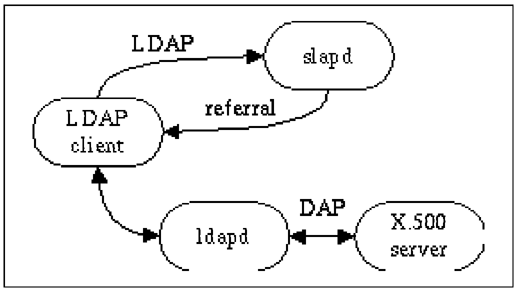
1. LDAP como servidor local.

En esta configuración, *slapd* sólo proporciona un servicio de directorio para el dominio local, no interactúa con otros servidores. Es fácilmente escalable a otras configuraciones si se desea.



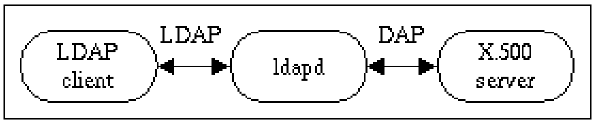
2. Servicio local con referencias a X.500.

En esta configuración, *slapd* proporciona servicio de directorio para el dominio local y *ldapd* proporciona acceso al resto del mundo X.500. No es necesario tener ejecutándose *ldapd*, sólo se necesita encontrar uno al que se pueda apuntar.



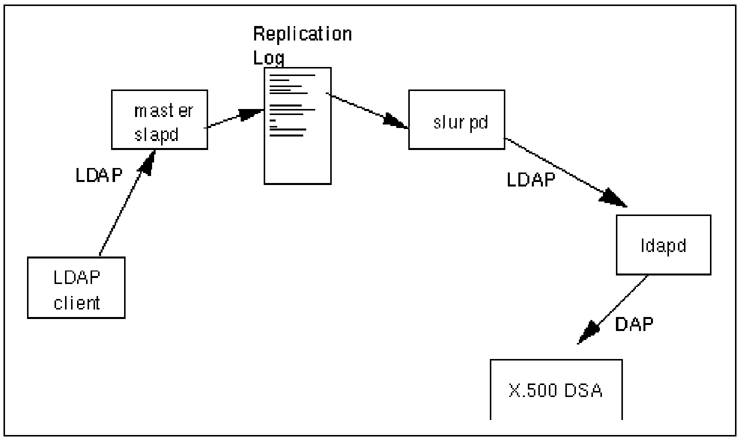
3. LDAP como un *front-end* de X.500.

En esta configuración se ejecuta un servicio X.500 que proporciona un servicio de directorio para el dominio local y servicios de pasarela (*gateway*) al resto del mundo X.500. Los clientes LDAP consiguen acceder al directorio a través de *ldapd*. *Slapd* no se necesita en esta caso.



4. Servicio de replicación de *slapd*

El demonio *slurpd* se usa para propagar los cambios del servidor *slapd* maestro a otros esclavos. Se puede usar en conjunción con las dos primeras configuraciones cuando un solo *slapd* no proporciona la suficiente fiabilidad y disponibilidad.



Instalación y configuración de OpenLDAP

El fichero de configuración de slapd: slapd.conf

Una vez instalados los paquetes, se nos proporciona un fichero de configuración de *slapd* de ejemplo en `/etc/openldapd/slapd.conf.example`. Éste incluye las opciones más comunes, incluyendo los ficheros de atributos (`/etc/openldap/slapd.at.conf`) y definiciones de clases (`/etc/openldap/slapd.oc.conf`), donde se guarda el pid de *slapd*, donde se va a buscar si no se puede procesar una petición no local, el tipo de base de datos que se va a utilizar, el sufijo, el directorio de trabajo, la entrada del administrador, su *password*, el modo de acceso a los datos, los servidores de réplicas, los modos de indexación de los atributos por parte de la base de datos, el control de acceso a los datos, etc. Para obtener ayuda acerca de las etiquetas consultar la página de manual de *slapd.conf(5).*

Ejecutando slapd

Slapd puede ejecutarse de dos modos diferentes: solo (*single*) o desde *inetd(8).* Si se está utilizando una base de datos LDBM se recomienda ejecutarlo solo, de esta forma se permite a la base de datos acelerar la *caché* y evitar problemas de concurrencia con los ficheros de índices de LDBM. Si se está utilizando PASSWD o SHELL como *backends* se puede ejecutar desde *inetd.*

En Debian se ejecuta en modo *single*. Se adjunta un fichero de arranque/parada/reinicio del servidor *slapd* en `/etc/init.d/`

Creación de la base de datos

Existen dos maneras de crear una base de datos:

- Creando la base de datos *on-line* usando LDAP. De esta forma se añaden entradas usando un cliente LDAP. Es un método bastante cómodo para pequeñas bases de datos, de unos pocos cientos o miles de entradas.

- Creando la base de datos *off-line*, usando utilidades de generación de índices: *ldif2ldmb*, *ldif2index*, *ldbmcat*, *ldif*. Este método es el mejor cuando se tienen miles de entradas que crear o para asegurar que no se puede acceder a los datos hasta que esté completamente creada.

Aquí explicaremos el primer método. Antes de ejecutar *slapd* es necesario configurar algunas opciones del *slapd.conf* como *suffix*, *directory* y *rootdn* y *rootpw*. Además hay que ajustar el valor de *index*. Una vez hecho esto hay que arrancar *slapd*, conectar un cliente LDAP y añadir las entradas. El formato de intercambio de datos *Data Interchange Format* (LDIF) se usa para representar las entradas LDAP en formato texto. Por ejemplo, para añadir unas entradas de una organización "ficticia" con *ldapadd* se crea un fichero `/tmp/mientrada.ldif`. Cada entrada comienza por dn:

```
dn:o=ficticia.com
o: ficticia.com
objectclass: organization
dn: c=ES, o=ficticia.com
c: ES
objectclass: country
dn: ou=Tech Lab, c=ES, o=ficticia.com
ou: Tech Lab
objectclass: organizationalUnit
dn: uid: pedrog, ou=Tech Lab, c=ES, o=ficticia.com
uid: pedrog
cn: Pedro Gutiérrez
sn: Gutiérrez
mail: pedrog@ficticia.com
mobile: 600111111
preferredLanguage: ES
userpassword: {CRYPT} Oury2XbQerCrtT
loginShell: /bin/bash
uidNumber: 502
gidNumber: 101
homeDirectory: /home/pedrog
gecos: Pedro Gutiérrez
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: account
```

Ahora es necesario introducirlo en el servidor:

```
ldapadd -f /tmp/mientrada.ldif -D "cn=root, o=ficticia.com" -w secret
```

donde *secret* corresponde a la *password* del administrador del servidor, que está en *slapd.conf*.

Replicación con slurpd

En ciertas configuraciones, una sola instancia de *slapd* puede resultar insuficiente para manejar un gran número de clientes accediendo al directorio vía LDAP. Puede ser necesario ejecutar más de una instancia. De este modo, una sola entrada de DNS hacia el servidor LDAP devuelve las IP de los servidores maestros y esclavos, repartiéndose la carga entre ellos.

Slurpd proporciona la capacidad para el *slapd* maestro de propagar los cambios a servidores esclavos de *slurpd*, implementando un esquema de replicación maestro/esclavo. *Slurpd* se ejecuta en la misma máquina en que corre el servidor *slapd* principal.

Configurando slurpd *y un servidor esclavo de* slapd

Para crear una réplica *slapd* se necesita configurar los servidores *slapd* esclavos y maestro para replicación y apagar el servidor maestro para copiar la base de datos. Finalmente, se arrancan las instancias maestras y esclavas. Se pueden crear tanto servidores *slapd* esclavos como se quiera.

1. Configurar el servidor *slapd* maestro. Añadir una directiva `replica` para cada réplica. Añadir una directiva `replogfile` que grabará un registro de cambios que leerá *slurpd*.

2. Configurar el servidor *slapd* esclavo. Su configuración debe ser igual que la del maestro excepto:
   - No incluir una directiva `replica`.
   - No incluir una directiva `replogfile`.
   - No incluir una línea `updatedn`.
   - Estar seguro de que el DN de la directiva `updatedn` tiene permiso de escritura en la base de datos.

3. Para que el esclavo comience con la copia exacta de los datos del maestro, se debe resetear el *slapd* maestro con kill -TERM <pid>, donde <pid> es el pid del *slapd* maestro.

4. Copiar la base de datos del maestro al esclavo. Se deben copiar todos los archivos del sufijo que estén en el directorio de índices indicado en el fichero de configuración de *slapd*.

5. Para configurar *slapd* para generar un fichero de log, se debe añadir la opción de configuración `replica` al fichero de configuración de *slapd* maestro. Por ejemplo para mandar los cambios al host truelies.rs.itd.umich.edu:389, ligándolo al *slapd* esclavo a través de *slurpd* "cn=Replicator, o=U of M, c=US", usando la autentificación "secret" sería

```
replica  host=truelies.rs.itd.umich.edu:389
binddn="cn=Replicator, o=U of M, c=US"
bindmethod=simple credentials=secret
```

6. Resetear el *slapd* maestro. Comprobar que genera los *logs*, que se puede modificar cualquier entrada de la base de datos y que queda registrado en el fichero de cambios.

7. Comenzar la ejecución de *slurpd*. Inmediatamente *slurpd* manda la modificación de prueba anterior. Comprobar en el archivo de log del esclavo de que lo ha mandado:

```
slurpd -f <mastersldapdconfigfile>
```

Pruebas y monitorización

Pruebas

Ahora los datos de la empresa pueden empezar a ser consultados por cualquier persona que tenga acceso al servidor y que disponga de una aplicación que soporte el protocolo LDAP.

Una comprobación rápida de que el servidor funciona correctamente, puede hacerse usando el cliente LDAP que viene con OpenLDAP:

```
ldapsearch -b "o=ficticia.com" "objectclass=*"
```

Mediante el BaseDN (*Base Distinguished Name*), se define desde qué parte del árbol se quieren efectuar las búsquedas. Por ejemplo, si sólo se quieren buscar los *emails* de la gente del departamento *Tech Lab*, el BaseDN sería "ou=Tech Lab, c=ES, o=ficticia.com", quedando:

```
ldapsearch –b "ou=Tech Lab, c=ES, o=ficticia.com" "mail=*"
```

También se puede usar, por ejemplo, el *addressbook* de Netscape Communicator para ver las entradas del servidor. Para ello se debe abrir la libreta de direcciones y seleccionar "Archivo", "Directorio Nuevo". Se debe introducir el nombre del servidor LDAP y el BaseDN (que Netscape llama "Buscar en Raíz" a partir del que buscar. Usando el "*" en el campo de búsquedas se pueden ver todas las entradas, aunque también se pueden realizar búsquedas más avanzadas escribiendo las primeras letras de los nombres de los empleados, o buscando por departamentos.

Con Netscape se puede hacer algo todavía mejor: usar el *Roaming Access*. Con esta facilidad se puede almacenar en el servidor LDAP los *bookmarks*, *address book*, *cookies*, preferencias, certificados, etc. Así, cada vez que se arranque Netscape se descargarán automáticamente todos esos ficheros, dejándolo al gusto del usuario. Si además se utiliza una cuenta de correo IMAP, se pueden tener todos los mensajes de correo accesibles desde cualquier PC con Netscape. Consultar el Apéndice A.

Esto es la punta del iceberg de lo que se puede hacer con LDAP. Muchas veces se necesita control sobre los objetos o atributos a los que cada persona puede acceder, y si puede modificarlos o sólo consultarlos. Esto se hace mediante reglas de control de acceso (consultar el manual de *slapd.conf(5)*). Un buen sitio donde se puede encontrar más información al respecto es dentro de FAQ-O-Matic de OpenLdap, en concreto:

http://www.openldap.org/faq/data/cache/62.html

Además se puede encontrar información sobre cómo acelerar el rendimiento de las búsquedas utilizando índices.

También se pueden desarrollar con relativa facilidad entornos web a servidores LDAP usando PerlLDAP o PHP. Incluso es posible incluir soporte *roaming* al estilo Netscape a muchas aplicaciones, mediante sencillos *scripts* que se encarguen de bajar las preferencias de cada usuario antes de lanzar un programa.

Otro proyecto muy interesante es el iniciado por la empresa PADL, con su `nss_ldap`. El *Name Service Switch* engloba un conjunto de extensiones a la librería C, para poder usar un servidor LDAP como fuente primaria para el manejo de diversos aspecto de nuestra máquina como los *alias* de correo, los interfaces de red, *hosts*, protocolos, *shadow passwords*, incluso como sistema de reemplazo a NIS. También han creado un módulo LDAP para PAM, que consigue que cualquier aplicación que soporte el API de PAM pueda usar autentificación LDAP y una serie de *scripts* en *perl* para realizar la migración del tradicional archivo de *passwords* o servidor NIS a LDAP.

Monitorización de slapd

Slapd proporciona un interfaz de monitorización en el que se puede encontrar gran cantidad de información acerca del funcionamiento del servidor. Se puede acceder a esta función haciendo una búsqueda de SLPAD_MONITOR_DN del fichero include/ldapconfig.h con cualquier clase de filtro (por ejemplo “objectclass=*”). Por defecto este DN es “cn=monitor”. Se obtiene una entrada con los siguientes atributos:

- version: versión del servidor *slapd*.
- threads: número de hebras actualmente en ejecución.
- connection: resumen de una conexión abierta.
- currentconnections: número de conexiones abiertas.
- totalconnections: número total de conexiones realizadas desde que se arrancó *slapd*.
- dtablesize: tamaño de la tabla de descriptores de archivos.
- writewaiters: número de hebras bloqueadas esperando a escribir datos en un cliente.
- readwaiters: número de hebras bloqueadas esperando a leer datos de un cliente.
- opsinitiated: número total de operaciones iniciadas desde que se arrancó el servidor.
- opscompleted: número total de operaciones completadas desde que se arrancó *slapd*.
- entriessent: número total de entradas enviadas a clientes desde que se inició *slapd*.
- bytessent: número total de bytes enviados a los clientes.
- currenttime: hora actual según *slapd*.
- starttime: hora en que se inició *slapd*.
- backends: número de *backends* actualmente siendo servidas por *slapd*.

Se puede usar cualquier cliente LDAP para recoger esta información. Por ejemplo, se puede hacer con el cliente *ldapsearch(1):*

```
ldapsearch –s base –b cn=monitor ‘objectclass=*’
```

Gestión diaria

La gestión del servicio de LDAP consistirá en mantener actualizado el directorio mediante las utilidades de OpenLDAP para añadir, borrar, editar una entrada, de tal forma que la base de datos quede actualizada.

El administrador del servidor debe comprobar que el sistema funciona, que las réplicas están sincronizadas con el maestro, etc. También es necesario que mantenga un control de acceso a los datos.

Si se está utilizando una base de datos LDBM es necesario hacer un buen mantenimiento de la misma, de los índices, tener bien dimensionada la caché, etc. Es importante porque de la base de datos depende el rendimiento, la fiabilidad y la disponibilidad de todo el sistema.

Apéndice A: Roaming con Netscape Communicator 4.5

En primer lugar es necesario haber configurado correctamente *openldapd*. Para añadir los usuarios puede usar la plantilla que se muestra más abajo. Notar que hay dos entradas para cada usuario. La primera es una entrada estándar de usuario y la segunda es la entrada de *roaming* para ese usuario. Si ya se han introducidos los usuarios en la base de datos, sólo hay que añadir la segunda entrada para cada uno.

```
# Roaming LDIF file

# This is for normal user entries, not LDAP-NS style ones
dn: cn=Full Name, ou=People, o=Your Company, c=US
objectClass: top
objectClass: person
cn: Full Name
sn: Last Name
userPassword: {crypt}<encrypted password>
# you can copy the crypt from /etc/shadow if you want,
# or encrypt a new one

# This is the Roaming entry
dn: nsLIProfile=Full Name, ou=Roaming, o=Your Company, c=US
objectClass: top
objectClass: nsLIProfile
owner: cn=Full Name, ou=People, o=Your Company, c=US
# Note, the owner is the DN from the entry above.

# End LDIF file
```

Esto puede añadirse al directorio LDAP usando *ldapadd* como administrador del directorio.

Ahora sólo falta configurar Netscape. Es necesario disponer del paquete completo del Netscape Communicator versión 4.5 o superior.

- Seleccionar del menú Edición->Preferencias.
- Seleccionar Acceso móvil de la lista de la izquierda.
- Activar el Acceso móvil para ese perfil.
- Introducir el nombre de usuario. Netscape usará ese nombre para reemplazar el $USERID en otros campos.
- Seleccionar Información del Servidor
- Asegurarse de que Servidor del directorio LDAP está activado.
- En la Dirección poner

```
ldap://ldapserver/nsLIProfileName=$USERID,ou=Roaming,o=Your Company,c=US
```

ldapserver es el nombre de la máquina donde el servidor *slapd* se está ejecutando

- En el DN del usuario poner:

```
cn=$USERID, ou=People, o=Your Company, c=US
```

Este es el DN completo de la entrada principal del usuario (el mismo DN que se ha usado para el atributo *owner* del listado LDIF anterior).

Ahora ya se puede salir de Netscape y volver a entrar. Entonces preguntará por el *password* de LDAP (el que estaba encriptado en el listado LDIF) y ¡ya se está haciendo *roaming*!.

Apéndice B: Direcciones de interés

Servidores que LDAP que soportan Linux

- **OpenLDAP:** www.openldap.org
- **Netscape Directory Server:** http://home.netscape.com/eng/server/directory. Desde www.iplanet.com se puede descargar una versión de evaluación de 60 días.
- **Innosoft Directory Server**: http://www3.innosoft.com/ids-products.hml
- **IntraStore**: http://intrastore.cdc.com/www. Es gratuito hasta 250 usuarios en Linux.
- **Umich**: www.umich.edu/~dirsvcs/ldap/

Clientes LDAP

- **Cliente LDAP para proyecto KDE:** www.mountpoint.ch/oliver/kldap/
- **Cliente LDAP GTK:** www.iit.edu/~gawojar/ldap

DBM backends para OpenLDAP

- **DBM2:** www.sleepycat.com
- **GDBM:** ftp://ftp.etsimo.uniovi.es/pub/gnu/gdbm

Otros enlaces de interés

- **Cómo habilitar soporte LDAP para Sendmail:** www.standford.edu/~bbense/inst.html, http://freshmeat.net/search.php3?query=ldap

Aplicaciones Linux relacionadas con LDAP

- **NSS_LDAP:** www.rage.net/ldap/ldapns
- **PADL:** www.padl.com/projects.html

Literatura recomendada

- **LDAP Netcape:** www.devedge.netscape.com/docs/manuals/communicator/ldap45.html
- **Why LDAP:** http://people.netscape.com/bjm/whyLDAP.html
- **Understanding LDAP:** http://www.redbooks.ibm.com/abstracts/sg244986.html
- **Perl y LDAP:** http://www.mozilla.org/directory/perldap.html
- **LinuxWorld:** www.linuxworld.com