INDICE

1 Que es LDAP?
- 1.1 Descripción de LDAP
- 1.2 Ventajas en el uso de LDAP
- 1.3 Usos prácticos de LDAP
- 1.4 cuando resulta interesando usar LDAP
- 1.5 Diferencias con una base de datos relacional
- 1.6 Historia de LDAP
- 1.7 Servidores LDAP disponibles en el mercado

2 Administración de LDAP
- 2.1 Introducción a la estructura de árbol
- 2.2 Definición de términos
- 2.3 Integración de LDAP con otros sistemas

3 Presentación de OpenLDAP
- 3.1 Presentación de OpenLDAP
- 3.2 Requisitos para instalar OpenLDAP

4 Administración de OpenLDAP
- 4.1 Calculo del dimensionamiento del Servidor / Servidores
- 4.2 Nomenclatura
- 4.3 Descarga del software

1. Que es LDAP?

1.1 Descripción de LDAP

LDAP ("Lightweight Directory Acces Protocol", en español Protocolo Ligero de Acceso a Directorios) es un protocolo de tipo cliente-servidor para acceder a un servicio de directorio.

Se usó inicialmente como un Front-end o interfaz final para x.500, pero también puede usarse con servidores de directorio únicos y con otros tipos de servidores de directorio.

¿Qué es un directorio?

Un directorio es una base de datos, pero en general contiene información más descriptiva y más basada en atributos.

La información contenida en un directorio normalmente se lee mucho más de lo que se escribe. Como consecuencia los directorios no implementan normalmente los complicados esquemas para transacciones o esquemas de reducción que las bases de datos utilizan para llevar a cabo actualizaciones complejas de grandes volúmenes de datos, Las actualizaciones en un directorio son usualmente cambios sencillos de todo o nada, si es que permiten algo.

Los directorios están para proporcionar una respuesta rápida a operaciones de búsqueda o consulta.

Pueden tener capacidad de replicar información de forma amplia, con el fin de aumentar la disponibilidad y fiabilidad, y a la vez reducir tiempo de respuesta. Cuando se duplica la información de un directorio, pueden aceptarse inconsistencias temporales entre la información que hay en las réplicas, siempre que finalmente exista una sincronización.

Hay muchas formas de proporcionar un servicio de directorio. Los diferentes métodos permiten almacenar en el directorio diferentes tipos de información, establecer requisitos diferentes para hacer referencias a la información, consultarla y actualizarla, la forma en que protege al directorio de accesos no autorizados. Algunos servicios de directorios son locales, proporcionando servicios a un contexto restringido. Otros servicios son globales, proporcionando servicio en un contexto mucho más amplio.

¿Un directorio LDAP es una base de datos?

El sistema gestor de una base de datos (Database Management System ó DBMS) de Sybase, Oracle, Informix ó Microsoft es usado para procesar peticiones (queries) ó actualizaciones a una base de datos relacional. Estas bases de datos pueden recibir cientos o miles de órdenes de inserción, modificación o borrado por segundo.

Un servidor LDAP es usado para procesar peticiones (queries) a un directorio LDAP. Pero LDAP procesa las órdenes de borrado y actualización de un modo muy lento.
En otras palabras, LDAP es un tipo de base de datos, pero no es una base de datos relacional. No está diseñada para procesar cientos o miles de cambios por minuto como los sistemas relacionales, sino para realizar lecturas de datos de forma muy eficiente.

Funcionamiento de LDAP

El servicio de directorio LDAP se basa en un modelo cliente-servidor.

Uno o más servidores LDAP contienen los datos que conforman el árbol de directorio LDAP o base de datos troncal, el cliente LDAP se conecta con el servidor LDAP y le hace una consulta. El servidor contesta con la respuesta correspondiente, o bien con una indicación de donde puede el cliente hallar más información. No importa con que servidor LDAP se conecte el cliente ya que siempre observará la misma vista del directorio; el nombre que se le presenta a un servidor LDAP hace referencia a la misma entrada a la que haría referencia en otro servidor LDAP.

1.2 Ventajas en el uso de LDAP

Un directorio LDAP destaca sobre los demás tipos de bases de datos por las siguientes características:

- es muy rápido en la lectura de registros

- permite replicar el servidor de forma muy sencilla y económica

- muchas aplicaciones de todo tipo tienen interfaces de conexión a LDAP y se pueden integrar fácilmente

- Dispone de un modelo de nombres globales que asegura que todas las entradas son únicas

- Usa un sistema jerárquico de almacenamiento de información.

- Permite múltiples directorios independientes

- Funciona sobre TCP/IP y SSL

- La mayoría de aplicaciones disponen de soporte para LDAP

– La mayoría de servidores LDAP son fáciles de instalar, mantener y optimizar.

1.3 Usos prácticos de LDAP

Dadas las características de LDAP sus usos más comunes son:

- **Directorios de información.** Por ejemplo bases de datos de empleados organizados por departamentos (siguiendo la estructura organizativa de la empresa) ó cualquier tipo de páginas amarillas.

- **Sistemas de autenticación/autorización centralizada.** Grandes sistemas donde se guarda gran cantidad de registros y se requiere un uso constante de los mismos.
Por ejemplo:

o Active Directory Server de Microsoft, para gestionar todas las cuentas de acceso a una red corporativa y mantener centralizada la gestión del acceso a los recursos.

o Sistemas de autenticación para páginas Web, algunos de los gestores de contenidos más conocidos disponen de sistemas de autenticación a través de LDAP.

o Sistemas de control de entradas a edificios, oficinas….

- **Sistemas de correo electrónico**. Grandes sistemas formados por más de un servidor que accedan a un repositorio de datos común.

- **Sistemas de alojamiento de páginas web y FTP**, con el repositorio de datos de

usuario compartido.

- **Grandes sistemas de autenticación basados en RADIUS**, para el control de accesos de los usuarios a una red de conexión o ISP.
- **Servidores de certificados públicos y llaves de seguridad.**
- **Autenticación única ó "single sign-on" para la personalización de aplicaciones.**
- **Perfiles de usuarios centralizados, para permitir itinerancia ó "Roaming"**
- **Libretas de direcciones compartidas.**

1.4 Cuando resulta interesante usar LDAP

Como hemos visto LDAP es una base de datos optimizada para entornos donde se realizan muchas lecturas de datos y pocas modificaciones o borrados.

Por lo tanto es muy importante saber elegir dónde es conveniente usarlo. No será conveniente como base de datos para sitios que realicen constantes modificaciones de datos (por ejemplo en entornos de e-commerce)

Normalmente el tipo de preguntas que debes hacerte para saber si LDAP es conveniente para tus aplicaciones son:

• ¿Me gustaría que los datos fueran disponibles desde distintos tipos de plataforma?

• ¿necesito acceso a estos datos desde un número muy elevado de servidores y/o aplicaciones?

• Los datos que almaceno ¿son actualizados muchas veces?, o por el contrario ¿son sólo actualizados unas pocas veces?

• ¿tiene sentido almacenar este tipo de datos en una base de datos relacional? Si no tiene sentido, ¿puedo almacenar todos los datos necesarios en un solo registro?

Pongamos algunos ejemplos:

Sistema de correo electrónico

Cada usuario se identifica por su dirección de correo electrónico, los atributos que se guardan de cada usuario son su contraseña, su límite de almacenamiento (quota), la ruta del disco duro donde se almacenan los mensajes (buzón) y posiblemente atributos adicionales para activar sistemas anti-spam o anti-virus.

Como se puede ver este sistema LDAP recibirá cientos de consultas cada día (una por cada email recibido y una cada vez que el usuario se conecta mediante POP3 o webmail). No obstante el número de modificaciones diarias es muy bajo, ya que solo se puede cambiar la contraseña o dar de baja al usuario, operaciones ambas que no se realizan de forma frecuente.

Sistema de autenticación a una red

Cada usuario se identifica por un nombre de usuario y los atributos asignados son la contraseña, los permisos de acceso, los grupos de trabajo a los que pertenece, la fecha de caducidad de la contraseña…

Este sistema recibirá una consulta cada vez que el usuario acceda a la red y una más cada vez que acceda a los recursos del grupo de trabajo (directorios compartidos, impresoras…) para comprobar los permisos del usuario.

Frente a estos cientos de consultas solo unas pocas veces se cambia la contraseña de un usuario o se le incluye en un nuevo grupo de trabajo.

1.5 Diferencias con una base de datos relacional.

Las **características de una base de datos relacional** (RDBMS o Relation Database Management Systems) son:

- Realizan **operaciones de escritura intensivas**: las bases de datos relacionales estás preparadas para hacer un uso constante de operaciones orientadas a transacciones, que implican la modificación o borrado constante de los datos almacenados.

- **Esquema específico** para cada aplicación: las bases de datos relacionales son creadas para cada aplicación específica, siendo complicado adaptar los esquemas a nuevas aplicaciones.

- **Modelo de datos complejo**: permiten manejar complejos modelos de datos que requieren muchas tablas, foreign keys, operaciones de unión (join) complejas…

- **Integridad de datos**: todos sus componentes están desarrollados para mantener la consistencia de la información en todo momento. Esto incluye operaciones de rollback, integridad referencial y operaciones orientadas a transacciones.

- Además las **transacciones se efectúan siempre aisladas** de otras transacciones. De tal forma que si dos transacciones están ejecutándose de forma concurrente los efectos de la transacción A son invisibles a la transacción B y viceversa, hasta que ambas transacciones han sido completadas.

- Disponen de **operaciones de roll-back** (vuelta atrás). Hasta el final de la transacción ninguna de las acciones llevadas a cabo pasa a un estado final. Si el sistema falla antes de finalizar una transacción todos los cambios realizados son eliminados (roll-back)

Las **características de un servidor LDAP** son:

- **Operaciones de lectura muy rápidas**. Debido a la naturaleza de los datos almacenados en los directorios las lecturas son más comunes que las escrituras.

- **Datos relativamente estáticos**. Los datos almacenados en los directorios no suelen actualizarse con mucha frecuencia.

- **Entorno distribuido**, fácil replicación

- **Estructura jerárquica**. Los directorios almacenan la información de forma jerárquica de forma nativa.

- **Orientadas a objetos**. El directorio representa a elementos y a objetos. Los objetos son creados como entradas, que representan a una colección de atributos.

- **Esquema Standard**. Los directorios utilizan un sistema standard que pueden usar fácilmente diversas aplicaciones.

- **Atributos multi-valor**. Los atributos pueden almacenar un valor único o varios.

- **Replicación multi-master**. Muchos de los servidores LDAP permiten que se realicen escrituras o actualizaciones en múltiples servidores.

1.6 Historia de LDAP

LDAP aparece con el estándar de los directorios de servicios. La versión original fue desarrollada por la Universidad de Michigan. La primera versión no se usó y fue en 1995 cuando se publicaron los RFC (Request For Comments) de la versión LDAPv2. Los RFC para la versión LDAPv3 fueron publicados en 1997. La versión 3 incluía características como las listas de acceso (control access lists) y replicación de directorios.

LDAP RFCs

Los RFCs asociados con LDAP son:

RFC1777 - Lightweight Directory Access Protocol. (Obsoletes RFC1487)
RFC1778 - The String Representation of Standard Attribute Syntaxes
RFC1779 - A String Representation of Distinguished Names (Obsoletes RFC1485)
RFC1823 - The LDAP Application Program Interface
RFC1960 - A String Representation of LDAP Search Filters (Obsoletes RFC1558)
RFC 2251 - Lightweight Directory Access Protocol (v3)
RFC 2252 - LDAPv3 Attribute Syntax Definitions
RFC 2253 - UTF-8 String Representation of Distinguished Names
RFC 2254 - The String Representation of LDAP Search Filters
RFC 2255 - The LDAP URL Format
RFC 2256 - A Summary of the X.500(96) User Schema for use with LDAPv3
RFC2829 Authentication Methods for LDAP.
RFC2830 - Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security.

RFCs relacionados

RFC1274 - The COSINE and Internet X.500 Schema
RFC1279 - X.500 and Domains
RFC1308 - Executive Introduction to Directory Services Using the X.500 Protocol
RFC1309 - Technical Overview of Directory Services Using the X.500 Protocol
RFC1617 - Naming and Structuring Guidelines for X.500 Directory Pilots (Obsoletes RFC1384)
RFC1684 - Introduction to White Pages services based on X.500
RFC2079 - Definition of an X.500 Attribute Type and an Object Class to Hold Uniform

1.7 Servidores LDAP disponibles en el mercado

Los más usados son:

- OpenLDAP – http://www.openldap.org

- Sun SunONE 5.2 – http://www.sun.com/software/products/directory_srvr/home_directory.html

- Siemens DirX Server 6.0 – http://www.siemens.com/directory

- Syntegra Intrastore Server 2000 - http://www.syntegra.com/us/directory_messaging/

- Computer Associates eTrust Directory 3.6 - http://www3.ca.com/Solutions/Product.asp?ID=160

- Novell NDS Corporate Edition 8.7.1 - http://www.novell.com/coolsolutions/nds/

-Microsoft ADS - Windows 2000 server edition - http://www.microsoft.com/windows2000/technologies/directory/ad/default.a...

Comparativa

Requirements	OpenLDAP	SunONE	DirX Server	Intrastore	eTrust	NDS	ADS

K4 (UMich) bind	yes	no	?	?	?	no	no
LDAPv2 protocol	yes	yes	?	?	?	?	maybe
LDAPv3 protocol	yes	yes	yes	yes	yes	?	yes
K5 bind	yes	no	?	?	?	no	yes
SASL GSSAPI (Krb5) Auth	yes	yes	?	?	?	?	yes
Scalable > 200K entries	yes	yes	yes	?	yes	?	?
Solaris 8	yes	yes	yes	yes	yes	yes	no
Search limit tied to binddn	yes	?	?	?	?	?	?
Multi-Master	yes	yes	?	?	?	?	?
Supports "<text> *(space)<text>" search	no	?	?	?	?	?	?
Support avail.	yes*	yes	yes	yes	yes	yes	yes

- Se realiza a través de listas de correo, se puede contratar soporte técnico con empresas como Symas Corp., Mind NV, or Inter7.

2. Administración de LDAP.

2.1 Introducción a la estructura de árbol.

Tradicionalmente se han usado las estructuras de árbol para jerarquizar la información contenida en un medio. El ejemplo más claro es la estructura de carpetas (directorios) de un sistema operativo. Esta organización nos permite ordenar la información en subdirectorios que contienen información muy específica.

Otro ejemplo muy común son los servidores DNS que nos permiten acceder a distintos servicios concretos que representan un dominio, por ejemplo

www.empresa.com – servidor www principal de la empresa
www.admin.empresa.com – servidor de administración
mail.empresa.com – servidor de mail de la empresa
us.mail.empresa.com – servidor secundario de correo en USA
es.mail.empresa.com – servidor secundario de correo en España

2.2 Definición de términos.

Entradas

El modelo de información de LDAP está basado en entradas. Una entrada es una

colección de atributos que tienen un único y global Nombre Distintivo (DN). El DN se utiliza para referirse a una entrada sin ambigüedades. Cada atributo de una entrada posee un tipo y uno o más valores. Los tipos son normalmente palabras nemotécnicas, como “cn” para common name, o “mail” para una dirección de correo.

La sintaxis de los atributos depende del tipo de atributo. Por ejemplo, un atributo cn puede contener el valor “Jose Manuel Suarez”. Un atributo email puede contener un valor “jmsuarez@ejemplo.com”. El atributo jpegPhoto ha de contener una fotografía en formato JPEG.

Atributos

Los datos del directorio se representan mediante pares de atributo y su valor.
Por ejemplo el atributo commonName, o cn (nombre de pila), se usa para almacenar el nombre de una persona. Puede representarse en el directorio a una persona llamada José Suarez mediante:

• cn: José Suarez

Cada persona que se introduzca en el directorio se define mediante la colección de atributos que hay en la clase de objetos person.

Otros atributos:
• givenname: José
• surname: Suarez
• mail: jmsuarez@ejemplo.com

Los atributos requeridos son aquellos que deben estar presentes en las entradas que utilicen en la clase de objetos. Todas las entradas precisas de los atributos permitidos son aquellos que pueden estar presentes en las entradas que utilicen la clase de objetos.

Por ejemplo, en la clase de objetos person, se requieren los atributos cn y sn. Los atributos description (descripción), telephoneNumber (número de teléfono), seealso (véase también), y userpassword (contraseña del usuario) se permiten pero no son obligatorios.

Cada atributo tiene la definición de sintaxis que le corresponde. La definición de sintaxis describe el tipo de información que proporciona ese atributo:

1. bin binario
2. ces cadena con mayúsculas y minúsculas exactas (las mayúsculas y minúsculas son significativas durante las comparaciones)
3. cis cadena con mayúsculas y minúsculas ignoradas (las mayúsculas y minúsculas no son significativas durante las comparaciones)
4. tel cadena de número de teléfono (como cis, pero durante las comparaciones se ignoran los espacios en blanco y los guiones"_")
5. dn "distinguished name" (nombre distintivo)

Tipos de Atributos

Una definición de tipo de atributo especifica la sintaxis de un atributo y cómo se ordenan y comparan los atributos de ese tipo.

Los tipos de atributos en el directorio forman un árbol de clases. Por ejemplo, el tipo de atributo "commonName" es una subclase del tipo de atributo "name".

Hay atributos obligatorios y opcionales listados en la siguiente tabla:

Identificador de Atributo Descripción del Valor de Atributo
NUMERICOID (obligatorio) Identificador de Objeto Único (OID)
NAME Nombre del Atributo

DESC Descripción del Atributo
OBSOLETE "true" si es obsoleto; "false" o ausente si no lo es
SUP Nombre del tipo de atributo superior del que se deriva el tipo de atributo
EQUALITY Nombre ó OID de la regla de correspondencia si la igualdad de correspondencia está permitida; ausente si no lo está
ORDERING Nombre o OID de la regla de correspondencia si está permitida la ordenación; ausente si no lo está.
SUBSTRING Nombre o OID de la regla de correspondencia si está permitida la correspondencia de sub-string ausente si no lo está.

SYNTAX OID numérico de la sintaxis de los valores de este tipo
SINGLE-VALUE "true" si el atributo no es multi-valor; "false" o ausente si lo es
COLLECTIVE "true" si el atributo es colectivo; "false" o ausente si no lo es
NO-USER-MODIFICATION "true" si el atributo no es modificable por el usuario; "false" o ausente si lo es
USAGE Descripción del uso del atributo

Estos atributos corresponden a la definición de "AttributeTypeDescription" en la RFC 2252.

LDIF

Para importar y exportar información de directorio entre servidores de directorios basados en LDAP, o para describir una serie de cambios que han de aplicarse al directorio, se usa en general el fichero de formato conocido como LDIF (formato de intercambio de LDAP).

Un fichero LDIF almacena información en jerarquías de entradas orientadas a objeto. Todos los servidores LDAP que incluyen una utilidad para convertir ficheros LDIF a formato orientadas a objeto. Normalmente es un fichero ASCII.

EJEMPLO:
Un fichero LDIF corriente tiene este aspecto:

dn: uid=jmsuarez,ou=People,dc=empresa,dc=com
uid: jmsuarez
cn: Jose Manuel Suarez
objectclass: account
objectclass: posixAccount
objectclass: top
loginshell: /bin/bash
uidnumber: 512
gidnumber: 300
homedirectory: /home/jmsuarez
gecos: Jose Manuel Suarez,,,,
userpassword: {crypt}LPnaOoUYN57Netaac

Como se puede notar, cada entrada está identificada por un nombre distintivo:
DN (“distinguished name”, nombre distintivo) esta compuesto por el nombre de la entrada en cuestión, más la ruta de nombres que permiten rastrear la entrada hacia atrás hasta la parte superior de la jerarquía del directorio.

Objetos

En LDAP, una clase de objetos define la colección de atributos que pueden usarse para definir una entrada. El estándar LDAP proporciona estos tipos básicos para las clases de objetos:

1. Grupos en el directorio, entre ellos listas no ordenadas de objetos individuales o de grupos de objetos.

2. Emplazamientos, como por ejemplo el nombre del país y su descripción.
3. Organizaciones que están en el directorio.
4. Personas que están en el directorio.

Una entrada determinada puede pertenecer a más de una clase de objetos.
Por ejemplo, la entrada para personas se define mediante la clase de objetos person, pero también puede definirse mediante atributos en las clases de objetos inetOrgPerson, groupOfNames y organization. La estructura de clases de objetos del servidor determina la lista total de atributos requeridos y permitidos para una entrada concreta.

2.3 Integración de LDAP con otros sistemas.

Una vez que hayamos configurado e instalado LDAP lo podemos usar como repositorio de datos para multitud de aplicaciones que disponen de soporte

- Radius
- Samba
- DNS
- Mail Transfer Agents
- Libretas de direcciones
- Servidores FTP
- Servidores de certificados de seguridad

3 Presentación de OpenLDAP.

3.1 Presentación de OpenLDAP.

El proyecto OpenLDAP nació como la continuación de la versión 3.3 del servidor LDAP de la Universidad de Michigan cuando dejaron de desarrollarlo.

OpenLDAP es un servidor LDAP que se distribuye bajo licencia GNU (OpenSource), que permite que el software se pueda usar de forma gratuita tanto de forma educativa como profesional. Además disponemos del código fuente para poder realizar nuestras propias modificaciones.

Se puede descargar de forma gratuita en la siguiente dirección

http://www.openldap.org/software/download/

A la hora de descargarte OpenLDAP verás que hay varias versiones disponibles:

- OpenLDAP Release. Las últimas versiones de OpenLDAP para uso general. OpenLDAP-2.2.15 es la última versión disponible.

- OpenLDAP Stable Release. Es la última versión que ha sido intensamente probada y suele ser la más fiable de las versiones disponibles.

- OpenLDAP Test Releases. Ocasionalmente los programadores de OpenLDAP hacen disponible una versión beta o gamma. Estas versiones son sólo para pruebas y no son para uso general.

En este momento OpenLDAP-2.2.13, es la versión considerada más estable.

Las versiones OpenLDAP 2.x funcionan con la versión 3 de LDAP (RFC 3377). LDAPv3 es es standar actual para todos los servidores LDAP.

Los paquetes que incluyen las distribuciones de OpenLDAP son:

• servidor LDAP (slapd)
• servidor de replicación LDAP (slurpd)
• Software Development Kit (ldap)
• Utilidades, herramientas, ejemplos…

Toda la documentación sobre el producto puede consultarse en

http://www.openldap.org/doc/

El coordinador del proyecto OpenLDAP se llama **Kurt D. Zeilenga** y es fácil contactar con él a través de las listas de correo.

Además de desarrollar OpenLDAP Kurt trabaja en IBM donde es Ingeniero de investigación de Servicios de Directorio y desarrollador de IBM Linux Technology Center.

3.2 Requisitos para instalar OpenLDAP.

Sistema operativo. OpenLDAP funciona en los siguientes sistemas operativos:

Apple Mac OS X
Linux: Debian, RedHat, Suse, Fedora, Mandrake…
FreeBSD
IBM AIX
Microsoft Windows 2000/NT
NetBSD
Solaris

No obstante por mi experiencia puedo decir que la versión de Windows funciona muy mal, no es nada recomendable su instalación. En Solaris, FreeBSD e Irix lo he probado y funcionaba bastante bien. Pero donde mejor funciona sin lugar a dudas es en Linux.

4. Administración de OpenLDAP.

4.1 Calculo del dimensionamiento del servidor / servidores.

Elige bien tu plataforma hardware:

- **Procesador:** Normalmente servidores multiporcesador.

- **Discos duros:** Para OpenLDAP los más óptimo es que uses un disco duro para el sistema operativo (preferiblemente en RAID) y un disco separado para la base de datos (normalmente sin RAID)
Elige discos duros muy rápidos, esta es la optimización más importante para OpenLDAP.

- **Tamaño de la memoria:** Dependerá del número de entradas que quieras almacenar y del número de atributos que use cada entrada. También de las pruebas de carga que realices y sus resultados. Normalmente necesitarás entre 1 GB y 4 GB.

Instalación del sistema operativo

- Elegir una instalación simple, sólo con los complementos imprescindibles.

- Actualizar el sistema operativo con los últimos parches o service packs (ej.: sunsolve.sun.com, redhat.com, windowsupdate.microsoft.com….)

- Elegir un sistema de archivos adecuado, normalmente:
o Ext3 para Linux
o UFS con LOGGING para Solaris

- Parar todos los servicios y demonios que no se vayan a usar.

- Securizar el servidor.

- Optimizar los parámetros del sistema operativo (hay diversos métodos de hacerlo que no se incluyen en este manual)

- Optimizar la configuración de la pila TCP.

4.2 Nomenclatura.

Antes de instalar el servidor elige una nomenclatura de directorios para todos tus trabajos (debes pensar siempre en las actualizaciones posteriores a la instalación actual)

Un ejemplo es usar un directorio como /opt/apps

/opt/source/openldap-2.1.25 directorio con el código fuente
/opt/apps/openldap-2.1.25 directorio para tu aplicación
/opt/apps/openldap es un link a la aplicación
/opt/data/openldap es el directorio para la base de datos
/opt/backup es el backup diario

Con una nomenclatura como esta es muy fácil implementar actualizaciones de la aplicación.

Elige además una nomenclatura para todos los objetos, atributos, usuarios….

4.3 Descarga del software.

Es preferible descargar y compilar OpenLDAP frente a instalarlo desde un paquete (RPM, deb o similar), porque estos paquetes no suelen venir muy optimizados.

Descarga la última versión estable de OpenLDAP (en este momento la 2.2.15) desde http://www.openldap.org/

Verifica la firma MD5 del paquete que te has descargado usando el siguiente comando:

[root@dep tmp]# md5sum openldap-2.2.15.tgz

Ahora verifica que la firma es exactamente la misma que la contenida en un archivo llamado openldap-2.2.15.md5 que te puedes descargar desde el servidor FTP de OpenLDAP.

Desconfía de servidores FTP que no sean los oficiales y de paquetes cuya firma MD5 no coincida con los de las páginas oficiales de OpenLDAP.