Universidad de las Ciencias Informáticas

Facultad 2

Título: **Integración de servicios usando software libre.**

Trabajo de Diploma para optar por el título de

Ingeniero en Ciencias Informáticas

Autores: Manuel Cheong Gómez

Dionner Polanco Noy

Tutor: Ing. Eduard Palomo Gené

2 de Julio del 2007

Declaración de autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

_____ Firma de Autores

_____ Firma del Tutor

Pensamiento

"A un plan obedece nuestro enemigo: de enconarnos, dispensarnos, dividirnos, ahogarnos. Por eso obedecemos nosotros a otro plan: enseñarnos en toda nuestra altura, apretarnos, juntarnos, burlarlo, hacer por fin a nuestra patria libre. Plan contra plan. Sin plan de resistencia no se puede vencer un plan de ataque."

José Martí

Agradecimientos

Primero que todo a nuestras familias, quienes nos han brindado todo su apoyo estando la mayor parte del tiempo lejos de ellos. A toda la confianza y seguridad que nos transmiten con sus consejos, enseñanzas y constante preocupación.

Al tutor y amigo Ing. Eduard Palomo Gené por sus buenos consejos y revisiones.

A nuestro amigo el Ing. Alexander Fernandez quien nos inicio en muchos de los temas que tratamos y de quien aprendimos muchísimo.

Al colectivo de Seguridad y Administración de Redes de la Universidad de Ciencias Informáticas, en especial al Ing. Raydel Montesino por la confianza depositada en nosotros.

A la comunidad de Software Libre de la UCI, donde formamos nuestro espíritu de investigación y colaboración.

A todos los que sin querer olvidamos, muchísimas gracias.

Dedicatoria

A nuestros padres

A todos nuestros familiares queridos

A nuestros amigos

Y al movimiento de Software Libre

Resumen

Con el modelo de integración de servicios usando software libre se logra tener un mismo usuario para disfrute de todos los servicios, con la fiabilidad y rapidez solicitada para los grandes volúmenes de datos, como lo requiere la infraestructura de nuestra universidad, con alrededor de 15 000 usuarios. Esta información es distribuida y replicada automáticamente para garantizar su fiabilidad y rápido acceso a la misma. Toda esta infraestructura se despliega sin costo alguno, por pago de licencia y da la posibilidad de seguirla mejorando con total libertad.

Este documento recoge las pautas a seguir y aplicaciones a utilizar para la migración hacia software libre de los servicios de red de la universidad de las Ciencias Informáticas, actualmente implementados sobre software privativo. Se propone una solución integral, con un servicio de directorio de alta disponibilidad, compatibilidad y libre de licencias, proporcionado por OpenLdap. Se implementa un sistema de correo electrónico con Postfix, servidor muy seguro y escalable, un sistema de navegación para Internet con Squid y un sistema de mensajería instantánea basado en el protocolo Jabber; todo lo anterior usa como mecanismo de autenticación al servicio de directorio antes mencionado. Con esta arquitectura basada en un directorio central, se tiene una solución hecha a la medida, sistemática, para unificar identidades, recursos, dispositivos y políticas tales como: direcciones de correo electrónico, aplicaciones, personas o grupos, sistemas automatizados, dispositivos periféricos (como impresoras) y otros componentes de red. Por lo que su implementación en una red extensa como la de nuestra universidad, permitirá una gestión de los recursos informáticos más eficiente y segura sin gasto adicional.
.

Indice

Introducción

Teniendo en cuenta el desarrollo de la informática que esta llevando a cabo nuestro país, una de las vertientes de desarrollo de software que se están tomando es la soportada en una plataforma libre. La administración de Sistemas Operativos GNU/Linux y la realización de labores de administración en redes heterogéneas, en las cuales existan múltiples clientes, y cada uno de ellos pueda tener un sistema operativo distinto, sobre el cual puedan operar una infinidad de usuarios, conlleva cierto grado de complejidad. Se hace necesario buscar un método que facilite, en la medida de lo posible, la labor de administración, en una red extensa como la de La Universidad de la Ciencias Informáticas.

Siempre concentrados en las posibles amenazas de Estados Unidos, se debe ver como un asunto de seguridad nacional, podemos comprobarlo en el discurso pronunciado por nuestro ministro de la Informática y las Comunicaciones Ramiro Valdés Menéndez en el Acto Inaugural de la XII Convención y Expo Internacional, Informática 2007

"*Durante toda su historia la Revolución Cubana se ha visto obligada a sortear los más perversos planes, por lo cual han resultado piezas claves las medidas y acciones que en materia de seguridad ha tomado en todos los frentes….*"

Es conocido por todos que Microsoft en el campo de los sistemas operativos posee un monopolio casi absoluto, el alto costo de las licencias de software y los problemas de seguridad, hacen de estos un objetivo de ataque por parte de hackers en internet. Como software propietario oculta su código impidiendo ser estudiadas cada una de sus funciones, por lo que no tenemos forma de saber lo que éste hace o qué aberturas en la "parte trasera" del programa hayan podido dejar abiertas los diseñadores para tener acceso en un futuro. Por otra parte nuestro país tiene problemas con las actualizaciones de su software de patente, debido a nuestro lento enlace satelital con el mundo exterior hace que las descargas de actualizaciones se vuelvan lentísimas. Muchas de nuestras computadoras operan con programas copiados. Y las compañías estadounidenses, presuntamente preocupadas sobre las leyes de Estados Unidos que restringen el comercio con Cuba, están bloqueando cada vez más las descargas de software a la isla. Por lo que muchas computadoras no alcanzan a descargar los "parches" (que sirven para corregir fallas en el programa original) y son vulnerables. Por las razones

antes expuestas podemos decir que más que un problema de seguridad es un problema económico y político.

LDAP (*Lightweight Directory Access Protocol)* aparece como el estándar de los servicios de directorios. 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 control de acceso (access control lists) y replicación de directorios.

Los servicios de directorio proporcionan una manera consistente de nombrar, describir, localizar, acceder, administrar y asegurar la información. La posibilidad de utilización de credenciales únicas facilita al usuario el acceso a la red e implica para los administradores una mayor seguridad y control del funcionamiento de la misma. Máquinas, usuarios y servicios pueden ser fácilmente creados, modificados y eliminados si se dispone de un punto de administración único. Un servicio de directorio permite además distribuir y replicar un directorio en muchas computadoras de una red para tenerlo más disponible a los usuarios y hacerlo resistente a las fallas. La implementación de este modelo de integración permitirá una mejor administración del directorio y de los servicios que de él se complementan.

Problema:

La alternativa de migración de los servicios de red de la Universidad de Ciencias Informáticas a Software Libre, requiere un modelo adaptable a las condiciones que presenta la misma y de una integración que permita la fácil administración de cuentas de usuarios junto a los servicios de red.

Idea a defender:

Con la utilización de este modelo de integración de servicios sobre LDAP y soportado en una plataforma de Software Libre, que ofrezca mayor disponibilidad en un medio tan heterogéneo y que facilite las labores gestión y administración de cuentas en la red, entonces se alcanzaría un sistema libre de pago por licencia de software que ofrece servicios como los de un sistema privativo y con las facilidades de gestión requeridas por la infraestructura de la Universidad de Ciencias Informáticas.

Objetivo General:

Implementar un modelo de integración de servicios sobre OpenLDAP que permita usar los servicios de red, libre de pago de licencias, de una manera óptima, eficiente y segura.

Para alcanzar este objetivo se desarrollaron las siguientes tareas:

- Desarrollo de una arquitectura de directorios con OpenLDAP, donde se guarden las referencias a los diferentes dispositivos de la red, así como usurarios y contraseñas.

- Implementación de un mecanismo de replica entre varios directorios para mantener siempre actualizada y disponible la información de los usuarios.

- Instalación de un Controlador de Dominio Primario con samba, para permitir el acceso al mismo por cualquier tipo de plataforma que usen los clientes, ya sea Windows o Linux.

- Configuración de un Servidor Postfix como solución optima para la distribución del correo electrónico. Así como lograr una arquitectura distribuida de servidores de correo Postfix donde todos manejen el mismo dominio y cuentas específicas.

- Instalación de un servidor EJabberd para proveer la mensajería instantánea a todos los usuarios del directorio.

- Configuración de un servidor proxy para manejar la navegación a internet de los usuarios.

Objeto de estudio:

Para llevar a cabo el desarrollo de este modelo integrador se emplearon las siguientes herramientas; Red Hat Enterprise 4 AS como sistema operativo, OpenLDAP como implementación del directorio, Samba para permitir compartir ficheros con sistemas operativos Windows, Squid como servidor Proxy, para la mensajería instantánea EJabberd y Postfix como servidor de correo.

Capítulo 1: Fundamentación Teórica

1.1 El software Libre

El software libre surgió envuelto en una filosofía de derechos y de libertad que pretendía elevar el concepto de la creación de software a los niveles más altos de las virtudes humanas como son el altruismo y la cooperación. Esta ideología se vería englobada en puntos fundamentales, las libertades que este proyecto ofrecía a los usuarios y desarrolladores de software libre: La primera, que es la que resulta mas obvia, es la libertad de utilizar el software, ya que seria incongruente restringir su uso a cualquier sector de la población tratándose de un proyecto de esta índole; la segunda es la libertad de estudiar y en su caso adaptar el software a las necesidades especificas que se presentan, para ello una característica muy importante del software libre es la liberación del código fuente, lo cual puede ser la mayor ventaja que ofrece este concepto ya que tanto los usuarios como el proyecto se ven beneficiados por esta característica debido a que así se realizan constantes mejoras a las versiones de los programas y se siguen distribuyendo de manera libre ya que la verdadera importancia del Software libre no es que pueda ser gratuito, (y no siempre es así, actualmente se vende una gran diversidad de versiones y distribuciones de programas libres, pues los desarrolladores requieren una fuente de ingresos para continuar con el proyecto, aunque generalmente no limitan la elaboración de copias), sino que brinda la posibilidad realizar aportaciones y adaptaciones a quien lo desee.[1]

1.2 Libertades del Software Libre.

De acuerdo con tal definición, el software es "libre" si garantiza las siguientes libertades:

- *"libertad 0", ejecutar el programa con cualquier propósito (privado, educativo, público, comercial, etc.)*
- *"libertad 1", estudiar y modificar el programa (para lo cuál es necesario poder acceder al código fuente)*
- *"libertad 2", copiar el programa de manera que se pueda ayudar al vecino o a cualquiera.*

- *"libertad 3", mejorar el programa, y hacer públicas las mejoras, de forma que se beneficie toda la comunidad.*

Es importante saber que las *“libertades 1 y 3”* obligan a que se tenga acceso al código fuente. La *"libertad 2"* hace referencia a la libertad de modificar y redistribuir el software libremente licenciado bajo algún tipo de licencia de software libre que beneficie a la comunidad. [2]

El término *Software no Libre* se emplea para referirse al software distribuido bajo una licencia de software más restrictiva que no garantiza estas cuatro libertades. Las leyes de la propiedad intelectual reservan la mayoría de los derechos de modificación, duplicación y redistribución para el dueño del “copyright”; el software dispuesto bajo una licencia de software libre rescinde específicamente la mayoría de estos derechos reservados. [3]

La definición de Software Libre no contempla el asunto del precio; un eslogan frecuentemente usado es "libre como en libertad, no como en cerveza gratis" o en inglés *"Free as in freedom, not as in free beer"* (aludiendo a la ambigüedad del término inglés "*free*"), y es habitual ver a la venta CDs de *Software Libre* como distribuciones Linux. Sin embargo, en esta situación, el comprador del CD tiene el derecho de copiarlo y redistribuirlo. El software gratis puede incluir restricciones que no se adaptan a la definición de Software Libre, por ejemplo, puede no incluir el código fuente, puede prohibir explícitamente a los distribuidores recibir una compensación a cambio. [4]

Para evitar la confusión, alguna gente utiliza los términos *"libre"* (Libre Software) y *"gratis"* (Gratis Software) para evitar la ambigüedad de la palabra inglesa *"Free"*. [5] Sin embargo, estos términos alternativos son usados únicamente dentro del movimiento del software libre, aunque están extendiéndose lentamente hacia el resto del mundo. Otros defienden el uso del término **Open Source Software** (OSS, S*oftware de Código Abierto, también llamado de Fuentes Abiertas*). La principal diferencia entre los términos “*Open Source*” y “*Free Software*" es que éste último tiene en cuenta los aspectos éticos y filosóficos de la libertad, mientras que el "*Open Source*" se basa únicamente en los aspectos técnicos. [6]

1.3 Significación Política.

Una vez que un producto de *Software Libre* ha empezado a circular, rápidamente está disponible a un coste muy bajo o sin coste. Al mismo tiempo, su utilidad no decrece. Esto significa que el *Software Libre* se puede caracterizar como un bien público en lugar de un bien privado. Aunque realmente no lo es en ningún momento. [7]

Puesto que el *Software Libre* permite el libre uso, modificación y redistribución, a menudo encuentra un hogar en los países del tercer mundo para los cuales el coste del software no libre es a veces prohibitivo. También es sencillo modificarlo localmente, lo que permite que sean posibles los esfuerzos de traducción a idiomas que no son necesariamente rentables comercialmente.

La mayoría del *Software Libre* se produce por equipos internacionales que cooperan a través de la libre asociación. Los equipos están típicamente compuestos por individuos con una amplia variedad de motivaciones. Existen muchas posturas acerca de la relación entre el Software Libre y el actual sistema económico capitalista:

- Algunos consideran el *Software Libre* como un competidor del capitalismo, una forma de anarquismo práctico.
- Algunos consideran el *Software Libre* como otra forma de competición en el mercado libre, y que el “copyright” es una restricción gubernamental sobre el mercado; dependiendo de la concepción de libre mercado que se tenga esto puede significar incluso una perspectiva librecambista de anticapitalismo o no.
- Algunos comparan el *Software Libre* a una economía del regalo, donde el valor de una persona está basado en lo que ésta da a los demás.

En gran parte de las implicaciones políticas y económicas del Software Libre se hace alusión a varios conceptos y principios anarquistas; cuestión que para muchos es notoria y para otros de alguna manera es leve o le restan importancia.

1.4 ¿Qué es un servicio de directorio?

Un directorio es una agrupación de archivos de datos, atendiendo a su contenido, a su propósito o a cualquier criterio que se decida y se usa para almacenar información acerca de

objetos de interés. En un directorio telefónico, por ejemplo, se almacena información sobre suscripciones de teléfono. En un sistema de archivo, el directorio almacena información acerca de los archivos que contiene. En este trabajo, los términos directorio y servicio de directorio hacen referencia a los directorios que existen en una red. Un servicio de directorio difiere de un directorio en que incluye tanto la fuente de información de directorios como los servicios que hacen que la información esté disponible y al alcance de los usuarios.

Un directorio es como 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. Los directorios están afinados para proporcionar una respuesta rápida a operaciones de búsqueda o consulta. Pueden tener la capacidad de replicar información de forma amplia, con el fin de aumentar la disponibilidad y la fiabilidad, y a la vez reducir el tiempo de respuesta. Cuando se duplica (o se replica) la información del directorio, pueden aceptarse inconsistencias temporales entre la información que hay en las réplicas, siempre que finalmente exista una sincronización.

Existen muchas maneras distintas 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, etc. Algunos servicios de directorio son locales, proporcionando servicios a un contexto restringido, otros servicios son globales, proporcionando servicio en un contexto mucho más amplio.

1.5 ¿Para qué sirve un servicio de directorio?

Un servicio de directorio es uno de los componentes más importantes de un sistema de cómputo extendido. Los usuarios y administradores con frecuencia no saben el nombre exacto de los objetos en que están interesados. Quizá conozcan uno o más atributos de los objetos y puedan consultar el directorio para obtener una lista de objetos que concuerden con los atributos. Un servicio de directorio permite que un usuario encuentre cualquier objeto conociendo sólo uno de sus atributos.

Los servicios de directorio son almacenes de información acerca de entidades de red, como

aplicaciones, archivos, impresoras y usuarios. Los servicios de directorio son importantes porque proporcionan una manera consistente de nombrar, describir, localizar, acceder, administrar y asegurar información acerca de esos recursos. Muchas organizaciones crean almacenes especializados o servicios de directorio dentro de sus aplicaciones para permitir una funcionalidad específica que requieren sus clientes. Así, los directorios empresariales dan un paso importante hacia la consolidación de directorios corporativos ofreciendo interfaces estándar que permiten la interoperabilidad y una administración de directorio centralizada. Un servicio de directorio puede:

1. Aplicar la seguridad definida por los administradores para mantener a salvo la información de intrusos.
2. Distribuir un directorio en muchas computadoras de una red.
3. Replicar un directorio para tenerlo disponible a más usuarios y hacerlo resistente a las fallas.
4. Particionar un directorio en varios almacenes que permitan guardar un número muy grande de objetos.

1.6 ¿Qué facilidades puede proporcionar la gestión de usuarios con OpenLDAP?

- **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.

1.7 ¿Cuáles servicios de red facilitan la integración?

1.7.1 Samba

Una pieza software de red es, Samba, que implementa a la perfección las funciones necesarias para que las máquinas con sistema operativo Windows se sienta como en su casa, en su dominio.

La principal característica de Samba es ofrecer servicios de impresión y de ficheros a clientes Windows. Pero Samba puede funcionar también como controlador de dominio para los clientes Windows, implementando todas las características de un dominio Windows NT.

1.7.2 Postfix

Postfix es un Agente de Transporte de Correos (MTA) de código abierto, un programa informático para el enrutamiento y envío de correo electrónico, creado con la intención de que sea una alternativa más rápida y fácil de administrar.

Algunas de las virtudes de Postfix son:

- Diseño modular (no es un único programa monolítico)
- La seguridad ha sido un condicionante desde el comienzo de su diseño.
- Lo mismo cabe decir del rendimiento (seguramente Sendmail no se diseñó pensando que algún día habría sitios que necesitaran procesar cientos de miles o millones de mensajes al día).
- Soporte para las tecnologías más usadas hoy día: LDAP, Bases de datos (MySQL), autentificación mediante SASL, LMTP, etc.
- Estricto cumplimiento de los estándares de correo-e.
- Soporte para dominios virtuales.
- Facilidad de configuración.
- Abundante documentación, y de calidad.
- Fácil integración con antivirus.

- Uso sencillo de listas negras.
- Se pueden lanzar varias instancias de Postfix en la misma máquina con distintas configuraciones, usando distintas direcciones IP, distintos puertos, etc.
- Filtrado de cabeceras y cuerpos de mensajes por expresiones regulares.

Por último, pero no menos importante, hay que decir que el código fuente de Postfix (por supuesto de dominio público) es un ejemplo de diseño, claridad y documentación, lo cual facilita su mantenimiento (por su autor o, en el futuro, por otros) así como la incorporación de nuevas capacidades, corrección de errores, etc.

El sistema Postfix está compuesto de varios procesos que se comunican entre sí, aparte de varias utilidades que puede usar el administrador para influir en el sistema u obtener información de él. Utiliza técnicas desarrolladas para los modernos servidores Web y, una PC puede recibir y entregar un millón de mensajes distintos al día.

1.7.3 Ejabberd

Con la introducción de una herramienta de mensajería instantánea basada en el protocolo Jabber. Gracias a ella, los usuarios podrán disfrutar de la opción de mantener conversaciones entre ellos de una forma mucho más dinámica y fluida de la que se proporcionaba actualmente con el uso de los foros y listas de correo.

Esta mejora consta de dos partes:

- Por un lado, la integración de un servidor de mensajería instantánea basado en el protocolo Jabber. Éste se encarga de la gestión del servicio y de la comunicación entre los usuarios.
- La integración de un cliente web basado en Jabber que sirva de interfaz a los usuarios para comunicarse entre ellos sin la necesidad de instalar ningún software externo.

Soporte de la información persistente sobre Base de datos MySQL en la que guarda la información de los usuarios y con como método de autenticación en soporte de OpenLDAP.

Protocolo de comunicaciones: Jabber

Protocolo abierto: Con todas las ventajas que supone el uso de un protocolo estándar de comunicaciones para mensajería instantánea.

Descentralizado: Se puede utilizar un servidor para Jabber, el cual, puede interoperar o no con el resto de la red Jabber. Es decir que podemos configurado como un servidor privado para el uso exclusivo de este servicio para el grupo de usuarios de la red, puede pasar a formar parte de la red de Jabber e interactuar con cualquier usuario que use este protocolo y se gestione su servicio en un servidor de Jabber.

Seguro: Permite la autenticación y el establecimiento de comunicaciones usando protocolos de seguridad (SSL, GPG, etc.).

Multiredes: Mediante el uso de pasarelas se puede comunicar con otros clientes como el MSN, el ICQ o el Yahoo!.

Después de realizar un estudio teniendo en cuenta un extenso grupo de servidores Jabber (todos los cuales son *"open source"*) que se encuentran en el mercado actual, se ha decidido utilizar para su integración con LDAP el servidor **Ejabberd**. Este servidor ha sido el escogido por ser el más completo en lo que se refiere a características que posee y por encontrarse en el momento de su elección en continua evolución por parte de su equipo de desarrolladores (es un servidor en continua evolución y expansión).

Es un servidor con licencia GNU GPL escrito principalmente en Erlang. Desde el punto de vista de la instalación y administración resulta muy cómodo, posee una interfaz web y una herramienta en la línea de comandos que le dota de una gran flexibilidad. A parte, existe bastante documentación sobre el mismo (también en castellano).

Está provisto de un gran número de módulos de que le permiten desde realizar estadísticas y salas de conversación, hasta enviar anuncios y crear reglas privadas. Estos módulos pueden ser cargados y descargados "en caliente" sin reiniciar el servidor.

La fiabilidad y escalabilidad ha sido probadas en entornos de hasta 25000 usuarios lo cual le da voto de confianza, aun más cuando se tiene en cuenta que es un proyecto todavía en activo y en continua labor de depuración.

El soporte de la información que debe ser persistente es por defecto Mnesia aunque es posible utilizar un servidor LDAP u otras bases de datos ODBC como son PostgreSQL, MySQL, etc...

1.7.4 Squid

Es un popular programa de software libre que implementa un servidor Proxy y un *demonio* para Web caché, publicado bajo licencia GPL. Tiene una amplia variedad de utilidades, desde acelerar un Servidor Web. Guardando en caché peticiones repetidas, hasta caché de Web, DNS y otras búsquedas para un grupo de gente que comparte recursos de la red, además de añadir seguridad filtrando el tráfico. Está especialmente diseñado para ejecutarse bajo entornos Unix-Linux.

Squid ha sido desarrollado durante muchos años y se le considera muy completo y robusto. Soporta muchos protocolos, aunque se usa principalmente para HTTP y FTP. Se añade soporte también a TLS, SSL, Internet Gopher y HTTPS. [8]

Capítulo 2: Características de OpenLDAP

2.1 Introducción

Cuando una organización posee múltiples ordenadores la opción de administración más adecuada suele ser la creación de un dominio o conjunto lógico de sistemas que admita gestionarse de forma centralizada, típicamente mediante un esquema cliente/servidor. Para ello, es imprescindible que en dicho sistema operativo exista alguna tecnología que permita esa centralización de aspectos tales como cuentas de usuarios y grupos, autenticación, políticas de seguridad, etc. En la actualidad, tanto en el caso de Windows como el de Linux, esta tecnología existe y está basada en un servicio de directorio compatible con el estándar LDAP. [9]

2.2 Que es LDAP?

LDAP como su propio nombre indica, es un protocolo ligero para acceder al servicio de directorio, especialmente al basado en X.500. LDAP se ejecuta sobre TCP/IP o sobre otros servicios de transferencia orientados a conexión. La definición detallada de LDAP está disponible en el RFC2251“The Lightweight Directory Access Protocol (v3)” y en otro documento que comprende las especificaciones técnicas, RFC3377. [10]

2.3 Servidores LDAP.

Una vez que sabemos qué es un directorio, podemos explicar en qué consiste LDAP. Existen diferentes estándares que especifican servicios de directorio, siendo el denominado X.500 tal vez el más conocido. El estándar X.500 define de forma nativa un protocolo de acceso denominado DAP (Directory Access Protocol). Este protocolo de acceso resulta muy complejo (y computacionalmente pesado) porque está definido sobre la pila completa de niveles del modelo OSI. Como alternativa a DAP para acceder a directorios de tipo X.500, LDAP

(Lightweight Directory Access Protocol) ofrece un protocolo ligero casi equivalente, pero mucho más sencillo y eficiente, diseñado para operar directamente sobre TCP/IP.

LDAP define una serie de operaciones para la consulta del directorio y el método de hacer y obtener los datos de estas consultas o actualizaciones por red. Actualmente, la mayoría de los servidores de directorio X.500 incorporan LDAP como uno de sus protocolo de acceso.[11] LDAP, al igual que X.500, proporciona un modelo de datos/espacio de nombres para el directorio y el protocolo. La información se guarda en el directorio de forma jerárquica (árbol) mediante entradas de objetos con una serie de atributos donde se almacena la información.

2.4 ¿Cómo funciona 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 del 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 dónde puede el cliente hallar más información (normalmente otro servidor LDAP). No importa con qué servidor LDAP se conecte el cliente: 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. Es ésta una característica importante de un servicio de directorios universal como LDAP. [12]

2.5 ¿Para qué se utiliza LDAP?

LDAP puede ser utilizado con varios propósitos, como por ejemplo la administración centralizada de usuarios, manteniendo todas las cuentas de usuario en una única ubicación LDAP (lo que no significa que esté albergada en un único servidor, puesto que LDAP soporta alta disponibilidad y redundancia). Sin embargo LDAP puede utilizarse igualmente para otros fines, como:

• Infraestructura de clave pública.
• Calendario compartido.
• Libreta de direcciones compartida.
• Almacenamiento para DHCP, DNS, etc.
• Directivas de configuración para las clases del sistema (guardando registro de las configuraciones de varios servidores).

La integración de sistemas es la pieza angular que debe afrontar un administrador de sistemas. Cuando en una organización deben convivir diferentes sistemas operativos, el administrador debe facilitar a los usuarios la forma de acceder a los recursos independientemente de la plataforma que éstos decidan utilizar. LDAP resuelve estas dificultades de manera eficiente, provee soporte para integración de cualquier sistema operativo, debido a su carácter estándar. [13]

2.6 ¿Cuando se puede 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.

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?

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 IMAP 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.

2.7 Diferencias entre el directorio LDAP y una base de datos relacional.

Las características de una base de datos relacional 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)

Que tipo de información se puede almacenar en un directorio.

El modelo de información de LDAP está basado en entradas. Una entrada es una colección de atributos que tienen un único y global Nombre Distinguido (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 “Sergio González”. Un atributo email puede contener un valor “sergio@ejemplo.com”. El atributo *jpegPhoto* ha de contener una fotografía en formato JPEG.

2.8 ¿Cómo se almacena la información?

En LDAP, las entradas están organizadas en una estructura jerárquica en árbol. Tradicionalmente, esta estructura reflejaba los límites geográficos y organizacionales. Las entradas que representan países aparecen en la parte superior del árbol. Debajo de ellos, están las entradas que representan los estados y las organizaciones nacionales. Debajo de estás, pueden estar las entradas que representan las unidades organizacionales, empleados, impresoras, documentos o todo aquello que pueda imaginarse. [14]

El árbol también se puede organizar basándose en los nombres de dominio de Internet. Este tipo de nombramiento se está volviendo muy popular, ya que permite localizar un servicio de directorio haciendo uso de los DNS.

Además, LDAP permite controlar que atributos son requeridos y permitidos en una entrada gracias al uso del atributo denominado *objectClass*. El valor del atributo *objectClass* determina que reglas de diseño (*schema rules*) ha de seguir la entrada.

2.9 ¿Cómo es referenciada la información?

Una entrada es referenciada por su nombre distinguido, que es construido por el nombre de la propia entrada (llamado *Nombre Relativo Distinguido* o RDN) y la concatenación de los nombres de las entradas que le anteceden. Por ejemplo, la entrada para *Nuno Gonçalves* en el ejemplo del nombramiento de Internet anterior tiene el siguiente RDN: *uid=manuel* y su DN sería: *uid=manuel,ou=usuarios,dc=bsd,dc=cu*. El formato completo para los DN está descrito en el RFC2253, “Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names.”

2.10 ¿Cómo se accede a la información?

LDAP define operaciones para interrogar y actualizar el directorio. Provee operaciones para añadir y borrar entradas del directorio, modificar una entrada existente y cambiar el nombre de una entrada. La mayor parte del tiempo, sin embargo, LDAP se utiliza para buscar información almacenada en el directorio. Las operaciones de búsqueda de LDAP permiten buscar entradas que concuerdan con algún criterio especificado por un filtro de búsqueda. La información puede ser solicitada desde cada entrada que concuerda con dicho criterio.

Por ejemplo, imagínese que quiere buscar en el subárbol del directorio que está por debajo de *dc=bsd,dc=cu* a personas con el nombre *Dionner Polanco*, obteniendo la dirección de correo electrónico de cada entrada que concuerde. LDAP permite hacer esto fácilmente. La siguiente sección describe con mayor detalle que se puede hacer con LDAP y como puede serle útil.

2.11 Seguridad y control de accesos.

LDAP provee de un complejo nivel de instancias de control de acceso, o ACIs. A causa de que el acceso puede ser controlado en el lado del servidor, es muchos más seguro que los métodos de seguridad que trabajan haciendo seguro a través del software cliente. [15]

Con LDAP ACIs, puedes hacer cosas como:

- Conceder a los usuarios la capacidad de cambiarse su número de teléfono del apartamento y su domicilio, mientras que se les restringe el acceso a solo lectura para otro tipo de datos (como título de trabajo o login de usuario).
- Conceder a cualquiera en el grupo "Domain Admin" (administradores de Dominio) la capacidad de modificar la información de los usuarios para los siguientes campos: número ID del usuario, nombre del departamento, y número del departamento. No habrán permisos de escritura para otros campos.

- Denegar el acceso de lectura a cualquiera que intente consultar al LDAP por la contraseña de un usuario, mientras que se seguirá permitiendo al usuario cambiar su propia contraseña.
- Conceder permisos solo de lectura a los usuarios para datos tales como los números de teléfono de casa, mientras que se deniega este privilegio a cualquier otro.
- Conceder a cualquiera en el grupo "host-admins" crear, borrar, y editar todos los aspectos de información del hosts almacenados en LDAP.
- A través de una página Web, permitir a la gente en "administradores" selectivamente conceder o denegarse a ellos mismos el acceso de lectura a subsets de la base de datos de contactos de los usuarios.
- A través de una página Web, permitir a cualquier propietario de grupo añadir o eliminar entradas de sus grupos. Las listas de distribución designadas como "pública" pueden permitir que los usuarios se añadan o se eliminen ellos mismos (pero solo a ellos mismos) de o a esos alias de correo. Las restricciones pueden basarse también en direcciones IP o nombres de máquina. Por ejemplo, los campos pueden hacerse legibles solo si la dirección IP del usuario empieza por 10.*.*.*, o si la resolución inversa del nombre de máquina del usuario por DNS se mapea a *.uci.cu.

Esto te dará una idea de lo que es posible utilizando el acceso controlado con directorios LDAP, una correcta implementación requiere mucha más información que la mostrada aquí.

Capítulo 3: Integración de Servicios sobre LDAP

3.1 Introducción.

La integración de los Servicios sobre LDAP hace uso del Software Libre y las diferentes tecnologías disponibles en un sistema GNU/Linux. De forma muy resumida se mostrará una pequeña presentación de las tareas a realizar por cada una de dichas tecnologías:

Se procede en las siguientes secciones con las instalaciones y configuraciones de las respectivas tecnologías y servicios de red expuestos anteriormente.

3.2 Conexión segura OpenSSL.

Creación de certificados.

En esta sección se realiza la creación de la entidad certificadora y los certificados necesarios para hacer uso de una conexión segura, característica que provee la versión 3 del protocolo LDAP por defecto, y por tanto, la versión de OpenLDAP que se está utilizando. [16]

Para habilitar las conexiones SSL/TLS hacia el servidor, se necesita la presencia de un certificado en el servidor por parte de los protocolos SSL/TLS. Además, en el establecimiento de una conexión SSL, el certificado del servidor sólo proporciona una conexión segura y encriptada al servidor. Si se desea autentificar al cliente, se ha de presentar al servidor LDAP el certificado certificado y el par de llaves del cliente.

Hay dos formas de crear e instalar un certificado en el servidor. Ambos métodos requieren la creación de un certificado para el servidor, enviárselo a los clientes OpenLDAP y realizar los cambios apropiados a los archivos de configuración de OpenLDAP. Ambos métodos necesitan el uso de comandos OpenSSL que solicitarán información para la creación del certificado.

Certificado autofirmado.

La primera forma para la creación del certificado del servidor emplea OpenSSL y genera un certificado autofirmado para el servidor.

OpenLDAP sólo trabaja con llaves no encriptadas, por lo que se ha de emplear el parámetro "-nodes" de OpenSSL para evitar la encriptación de la llave privada.

$ openssl req -newkey rsa:1024 -x509 -nodes -out slapd.pem -keyout slapd.pem -days 365
Generating a 1024 bit RSA private key
writing new private key to 'server.pem'
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
*Country Name (2 letter code) [AU]:**CU***
*State or Province Name (full name) [Some-State]:**UCI***
*Locality Name (eg, city) []:**UCI***
*Organization Name (eg, company) [Internet Widgits Pty Ltd]:**UCI***

*Organizational Unit Name (eg, section) []:**UCI***

*Common Name (eg, YOUR name) []:**ldap.nodo.cu***

*Email Address []:**soporte@nodo.cu***

Esto creará un archivo server.pem en el directorio donde haya ejecutado el comando.

Ahora sólo falta indicar a OpenLDAP que utilice el certificado anteriormente creado. Para ello se han de añadir las siguientes líneas al archivo de configuración de slapd, /etc/ldap/slapd.conf, utilizado en la próxima sección:

TLSCACertificateFile /etc/ldap/tls/slapd.pem

TLSCertificateFile /etc/ldap/tls/slapd.pem

TLSCertificateKeyFile /etc/ldap/tls/slapd.pem

3.3 Servidor de Directorio LDAP (OpenLDAP)

Ahora que están creados los certificados, sólo queda configurar OpenLDAP. El primer paso para instalar OpenLDAP, es instalar los paquetes slapd y ldap-utils.

Recuerde que los procedimientos expuestos a continuación fueron hechos sobre Debian GNU/Linux y Red Hat.

Configuración del Servidor LDAP.[17]

Instalamos el software con en commando: *apt-get install slapd ldap-utils*

Observaciones a la instalación.

Dependiendo de como se encuentre el sistema y los paquetes que tenga instalados en el mismo, se instalarán y sugerirán más o menos dependencias a la hora de instalar OpenLDAP. [18]

Comprobar la instalación con los siguientes comandos:

server-deb:/# ps aux|grep ldap

openldap 4636 0.3 0.8 56988 4328 ? Ssl 01:51 0:03 /usr/sbin/slapd -g openldap -u openldap

root 4732 0.0 0.1 2880 748 pts/0 R+ 02:10 0:00 grep ldap

server-deb:/# netstat –tln | grep 389

```
tcp        0      0 0.0.0.0:389             0.0.0.0:*               LISTEN
tcp6       0      0 :::389                  :::*                    LISTEN
```

Configuración por defecto de OpenLDAP (/etc/default/slapd).

En este archivo se configuran los aspectos relativos a la ejecución del demonio slapd: parámetros pasados en el arranque, usuario y grupo de ejecución del demonio, etc. En los siguientes pasos se verán las diferentes alteraciones. [19]

Especificación de las interfaces donde escuchar.

La configuración por defecto del demonio slapd hace que escuche en todas las interfaces de red presentes en el sistema. Esta característica no es deseable, por este motivo se verá la forma de modificarla. [20]

La especificación de las interfaces de red, así como el protocolo utilizado en cada una de ellas (ldap, ldaps), se realiza en el archivo /etc/default/slapd. Dentro de este, la variable SLAPD_SERVICES poseerá las interfaces donde se desea que escuche slapd. Ejemplo:

SLAPD_SERVICES="ldap://ldap.nodo.cu:389/ ldaps://ldap.nodo.cu:636/"

El protocolo "ldap" especifica las interfaces y los puertos donde escuchará slapd con la característica de que las conexiones que se establezcan a la misma no harán uso de encriptación. [21]

El protocolo "ldaps" especifica las interfaces y los puertos donde escuchará slapd con la característica de que las conexiones que se establezcan a la misma harán uso de encriptación.

Una vez se han asignado las interfaces necesarias, se ha de reiniciar el demonio slapd y comprueba su ejecución:

 # /etc/init.d/slapd restart

Acto seguido ejecutamos los comandos de anteriores para comprobar el funcionamiento del servidor. [22]

Configuración de Servidor LDAP

```
# Global Directives:
# Features to permit
#allow bind_v2

# Schema and objectClass definitions [23]

include         /etc/ldap/schema/core.schema
include         /etc/ldap/schema/cosine.schema
include         /etc/ldap/schema/nis.schema
include         /etc/ldap/schema/inetorgperson.schema
include         /etc/ldap/schema/samba.schema

# Where the pid file is put. The init.d script
# will not stop the server if you change this.
pidfile         /var/run/slapd/slapd.pid

# List of arguments that were passed to the server
argsfile        /var/run/slapd/slapd.args

# Read slapd.conf(5) for possible values
loglevel        0

# Where the dynamically loaded modules are stored
modulepath      /usr/lib/ldap
moduleload      back_bdb

# The maximum number of entries that is returned for a search operation
sizelimit 500

# The tool-threads parameter sets the actual amount of cpu's that is used
```

```
# for indexing.
tool-threads 1

#######################################################################
# Specific Backend Directives for bdb:
# Backend specific directives apply to this backend until another
# 'backend' directive occurs
backend         bdb
checkpoint 512 30



#######################################################################
# Specific Directives for database #1, of type bdb:
# Database specific directives apply to this databasse until another
# 'database' directive occurs
database        bdb

# The base of your directory in database #1
suffix          "dc=nodo,dc=cu"

# rootdn directive for specifying a superuser on the database. This is needed
# for syncrepl.
# rootdn          "cn=admin,dc=nodo,dc=cu"

# Where the database file are physically stored for database #1
directory       "/var/lib/ldap"

# For the Debian package we use 2MB as default but be sure to update this
# value if you have plenty of RAM
dbconfig set_cachesize 0 2097152 0

# Sven Hartge reported that he had to set this value incredibly high
```

```
# to get slapd running at all. See http://bugs.debian.org/303057
# for more information.

# Number of objects that can be locked at the same time.
dbconfig set_lk_max_objects 1500
# Number of locks (both requested and granted)
dbconfig set_lk_max_locks 1500
# Number of lockers
dbconfig set_lk_max_lockers 1500

# Indexing options for database #1
index           objectClass eq

# Save the time that the entry gets modified, for database #1
lastmod         on

# Where to store the replica logs for database #1
#replogfile     /var/lib/ldap/replog
replogfile      /etc/ldap/log/replica

# The userPassword by default can be changed
# by the entry owning it if they are authenticated.
# Others should not be able to see it, except the
# admin entry below
# These access lines apply to database #1 only
access to attrs=userPassword,shadowLastChange
        by dn="cn=admin,dc=nodo,dc=cu" write
        by anonymous auth
        by self write
        by * none

# Ensure read access to the base for things like
# supportedSASLMechanisms.  Without this you may
```

```
# have problems with SASL not knowing what
# mechanisms are available and the like.
# Note that this is covered by the 'access to *'
# ACL below too but if you change that as people
# are wont to do you'll still need this if you
# want SASL (and possible other things) to work[24]
# happily.
access to dn.base="" by * read

# The admin dn has full write access, everyone else
# can read everything.
access to *
        by dn="cn=admin,dc=nodo,dc=cu" write
        by * read

# userPassword por defecto puede ser cambiado
# solamente para el dueño
# Otros no están habilitados para ver, excepto admin

access to attrs=userPassword,shadowMax,shadowExpire
by dn="cn=admin,dc=nodo,dc=cu" write
by anonymous auth
by self write
by * none

# Las claves Samba 3 por defecto pueden ser cambiadas
# por el usuario si se ha autentificado.
access to attrs=sambaPwdLastSet,sambaPwdMustChange,sambaPwdCanChange
by dn="cn=admin,dc=nodo,dc=cu" write
by anonymous auth
by self write
by * none

access to attrs=sambaLMPassword,sambaNTPassword
by dn="cn=admin,dc=nodo,dc=cu" write
by dn="uid=.*,ou=people,dc=nodo,dc=cu" write
by anonymous auth
by self write
by * none
#######################################################################

#Configuracion de La replica del Servidor
```

```
updatedn cn=admin,dc=nodo,dc=cu
updateref ldaps://10.128.50.32


# slurpd replication

replica host=10.128.50.32:389
        suffix="dc=nodo,dc=cu"
        binddn=cn=Manager,dc=nodo,dc=cu
        credentials=passwd
        bindmethod=simple
       tls=yes
       starttls=yes

# SSL/TLS Certificados del Servidor [25] [26]


TLSCipherSuite HIGH:MEDIUM:+SSLv2

TLSCACertificateFile /etc/ldap/tls/cacert.pem

TLSCertificateFile /etc/ldap/tls/server.pem

TLSCertificateKeyFile /etc/ldap/tls/server.pem
```

3.4 Autentificación usando PAM y NSS.

Se verá como configurar una máquina para que sus usuarios se autentifiquen a través de un servidor LDAP. Para ello se han de modificar dos aspectos del comportamiento del sistema:

El mapeado entre los números de identificación de los usuarios y sus nombres. La búsqueda de este tipo de información es responsabilidad del servicio de nombres (NSS), cuyo archivo de configuración es: */etc/nsswitch.conf*. [27]

La autentificación (comprobación de claves) es responsabilidad del subsistema PAM, cuya configuración se hace a través del directorio */etc/pam.d/.*

Ambos subsistemas se han de configurar separadamente, pero en este caso, ambos se van a configurar de tal forma que hagan uso de LDAP.

Instalación de pam_ldap y nss_ldap.

Antes de poder autentificar a los usuarios a través de un servidor LDAP, es necesario instalar algunas utilidades en el cliente, como *pam_ldap* y *nss_ldap*. Esta sección mostrará la forma de instalación de estas utilidades.

El paquete pam_ldap permite hacer uso de un servidor LDAP para la autentificación de usuarios (comprobación de claves) a aquellas aplicaciones que utilicen PAM.

En Debian GNU/Linux el paquete libpam-ldap provee esta funcionalidad, por lo que será instalado en la máquina:

apt-get install libpam-ldap libnss-ldap

El método de encriptación elegido para almacenar las claves ha sido "SSHA", de ésta forma pam-ldap utilizará el algoritmo de hash especificado en el archivo /etc/ldap/slapd.conf, en lugar de realizar la operación hash localmente y escribir el resultado en la base de datos directamente.

La configuración del módulo pam_ldap.so se almacena en el archivo /etc/pam_ldap.conf.

El archivo /etc/pam_ldap.conf se ha de poder leer por todos los usuarios del sistema, para asegurarse de que es legible por todo el mundo:

 # chmod 644 /etc/pam_ldap.conf.

En estos momentos el sistema ya está listo para la configuración de los distintos servicios que utilizan PAM, de forma que estos utilicen LDAP para la comprobación de la clave. Cada servicio que hace uso de PAM para la autentificación, posee su propio archivo bajo el directorio */etc/pam.d/*. Para que dicho servicio utilice LDAP en la comprobación de claves, se ha de modificar su archivo de configuración.

El paquete nss-ldap permite a un servidor LDAP actuar como un servidor de nombres. Esto significa que provee la información de las cuentas de usuario, los IDs de los grupos, la información de la máquina, los alias, los grupos de red y básicamente cualquier cosa que normalmente se obtiene desde los archivos almacenados bajo /etc o desde un servidor NIS.

En Debian GNU/Linux el paquete libnss-ldap provee esta funcionalidad. La configuración del paquete nos muestra información adicional sobre el mismo. Si se quiere ver el ejemplo del archivo */etc/nsswitch.conf* que provee libnss-ldap, acceda a: */usr/share/doc/libnss-ldap/examples/nsswitch.ldap*.

La configuración de nss-ldap se almacena en el archivo /etc/libnss-ldap.conf. El archivo /etc/libnss-ldap.conf se ha de poder leer por todos los usuarios del sistema, para asegurarse de que es legible por todo el mundo, puede ejecutar:

 # chmod 644 /etc/libnss-ldap.conf

Configuración de los archivos necesarios

Se debe tener en cuenta que va a modificar archivos de configuración utilizados para el ingreso al sistema. Sería recomendable que tuviese en todo momento una consola de root abierta, por si deja de funcionar la autentificación.

nsswitch.conf es el fichero de configuración de las Bases de Datos del Sistema y del sistema de Conmutación de los Servicios de Nombres (Name Service Switch).

En otras palabras, es un archivo que indica el orden y el procedimiento a seguir para la búsqueda de la información requerida, por ejemplo, para hacer búsquedas de hosts y/o usuarios.

La forma de configurar este archivo es muy simple: primero se especifica la base de datos sujeta a la búsqueda (primera columna) seguida del procedimiento que se va a emplear para realizar una búsqueda sobre la misma (columnas siguientes).

De esta forma, basta con configurar el procedimiento de búsqueda para que haga uso de LDAP en algún momento como se muestra en esta modificación:

passwd: ***(1)*** *compat ldap* ***(2)***

group: ***(3)*** *compat ldap* ***(4)***

shadow: ***(5)*** *compat ldap* ***(6)***

hosts: ***(7)*** *files dns* ***(8)***

(1)(3)(5)(7) Bases de datos de búsqueda.

(2)(4)(6) Procedimiento de búsqueda: primero se mira en los archivos locales y luego en el directorio LDAP.

(8) Procedimiento de búsqueda: primero se mira en los archivos locales, luego en el directorio LDAP y finalmente se realiza una consulta al servidor DNS.

Se muestra que no se ha eliminado el uso de los ficheros locales, "files", ya que algunos usuarios y grupos de usuarios (como por ejemplo root) permanecerán de forma local. Si su sistema no utiliza la entrada "files", y el servidor LDAP se cae, nadie, ni siquiera root, podrá entrar al sistema.

nss-ldap espera que las cuentas sean objetos con los siguientes atributos: uid, uidNumber, gidNumber, homeDirectory y loginShell. Estos atributos están permitidos por la clase objeto (objectClass) posixAccount.

Una vez realizada la configuración, se puede comprobar que todo funciona bien con el comando *getent* seguido de la base de datos de búsqueda deseada (por ejemplo: *getent hosts*). Como resultado se mostrará la base de datos consultada por pantalla.

Configuración de PAM.

PAM permite configurar el método de autentificación que van a utilizar las aplicaciones que hagan uso de él. Gracias a esto, se pueden añadir fácilmente distintas opciones de autentificación, como el uso de una base de datos LDAP.

En las siguientes secciones se mostrarán los archivos que se han de modificar para lograr la autentificación a través de LDAP.

pam-ldap asume que las cuentas del sistema son objetos con los siguientes atributos: *uid* y *userPassword*. Los atributos están permitidos por la clase objeto *(objectClass) posixAccount*.

Se editan los correspondientes ficheros:

Configuración correcta para usar el servicio del LDAP, en Debian, ya en Red Hat esto se configura con la ultilidad system-config-authentication

/etc/pam.d/common-account

Estos archivos han de tener únicamente estas entradas:

```
account     required      pam_unix.so broken_shadow
account     sufficient    pam_succeed_if.so uid < 100 quiet
account     [default=bad success=ok user_unknown=ignore] pam_ldap.so
account     required      pam_permit.so
```

/etc/pam.d/common-auth

```
auth        required      pam_env.so
auth        sufficient    pam_unix.so likeauth nullok
auth        sufficient    pam_ldap.so use_first_pass
auth        required      pam_deny.so
```

/etc/pam.d/common-session

```
session     required      pam_limits.so
session     required      pam_unix.so
session     optional      pam_ldap.so
```

Si se desea que el sistema sea capaz de crear directorios *home* auntomaticamente (piense en el siguiente caso: acaba de añadir un usuario en la base de datos LDAP, pero no ha creado un directorio *home* para este usuario en el sistema), puede utilizar el módulo *pam_mkhomedir es* para este propósito. Para ello se añade la siguiente línea al principio del archivo *common-session*:

```
session     required      pam_mkhomedir.so skel=/etc/skel/ umask=0022
```

El módulo *pam_mkhomedir* sólo crea directorios de un nivel. Es importante tener esto en cuenta para planificar la estructura del *home* de los usuarios.

/etc/pam.d/common-password

```
password    requisite     pam_cracklib.so retry=3
password    sufficient    pam_unix.so nullok use_authtok
password    sufficient    pam_ldap.so use_authtok
password    required      pam_deny.so
```

Comprobando que todo funciona.

Ya está el sistema preparado para hacer uso de LDAP en la autentificación de los usuarios, sería recomendable hacer algunas pruebas con la nueva configuración para ver si todo funciona correctamente.

El comando pamtest puede ayudar a realizar estas pruebas. pamtest acepta dos parámetros: el primero es el nombre del servicio al cual se va a conectar para realizar la autentificación, el segundo es el nombre del usuario que se va a autentificar sobre dicho servicio.

El comando *pamtest* se encuentra en el paquete libpam-dotfile, si no está disponible en su sistema, ha de ejecutar:

 # apt-get install libpam-dotfile

server-deb:/# pamtest passwd dpolanco

Trying to authenticate <dpolanco> for service <passwd>.

Password: *[Clave del usuario]*

Authentication successful.

server-deb:/# pamtest ssh dpolanco

Trying to authenticate <dpolanco> for service <ssh>.

Password: *[Clave del usuario]*

Authentication successful.

Una vez se ha llegado a este punto, el sistema ya está preparado para autentificar a los usuarios a través de LDAP usando el procedimiento normal de autentificación y OpenSSH.

En el próximo apartado dedicado a Samba veremos, entre otras cosas, como añadir usuarios a la base de datos LDAP.

3.5 Servidor de Correo (Postfix)

Algunas definiciones

El servicio de correo electrónico consta de dos partes bien diferenciadas: aquella con la que trata el usuario (CLIENTE), y aquella que se encarga de transportar los mensajes del origen al destino (SERVIDOR). A menudo hay un componente adicional encargado de distribuir o transportar el correo que llega a la máquina destino a una ubicación especial dentro de ésta, propia de cada usuario y programas adicionales anti-virus, anti-spam, etc.

MTA (Mail Tranfer Agent): es el programa encargado de recoger mensajes y enviarlos, y de comunicarse con otros MTA si es necesario. Lo normal es que funcione como servicio (es decir, de modo continuo, esperando peticiones de los clientes o de otros MTAs y atendiéndolas). En Unix/Linux se implementan como uno o más demonios. Un MTA instalado y corriendo en un equipo, lo convierte en un servidor de correo.

MUA (Mail User Agent): es un programa cliente que solicita la descarga de correos de un servidor y el envío hacia él mismo para su futura entrega a otros servidores. Los MUAs suelen tener muchas funcionalidades que superan la estricta lectura y composición de mensajes, como el mantenimiento de libretas de direcciones, gestión de adjuntos (attachments), gestión de múltiples carpetas para organizar el correo, filtros de correo para borrarlo, responderlo, o redirigirlo a carpetas determinadas, todo ello automáticamente y en función de las

características del mensaje, etc. Los MUAs son entonces programas cliente, encargados de establecer comunicación con el servidor de correo.

MDA (Mail Delivery Agent): es el programa encargado de recibir los correos del MTA para la entrega local y colocarlos adecuadamente en los buzones individuales de cada usuario (si el usuario tiene una cuenta en el servidor local). Un MTA no necesariamente cuenta con un MDA, aunque hay los que si cuentan con uno. Postfix puede ser configurado para usar distintos MDA y procmail es también el usado por descarte en primeras versiones de postfix en Red Hat; sin embargo, actualizaciones posteriores incorporan a postfix su propio agente MDA que puede interactuar con otros como Cyrus-imap.

SMTP (Simple Mail Transfer Protocol): es el protocolo estándar de transmisión de información para el envío de e-mail entre servidores de correo. Adicionalmente, los clientes usan el protocolo SMTP para el envío de correo a un servidor que cuenta con un MTA instalado. La comunicación SMTP usualmente es a través del puerto 25 de comunicaciones.

IMAP (Internet Message Access Protocol) : IMAP es un protocolo de acceso a mensajes de correo electrónico localizados y mejor organizados en buzones en un servidor de correo, que a diferencia de POP3 es más rápido y eficiente, y permite la consulta de un cliente desde múltiples equipos sin descargas. Por ejemplo, un usuario que desea acceder a sus mensajes acumulados en un servidor IMAP, lo puede hacer desde su computadora de escritorio en casa, su estación de trabajo en la oficina, de viaje desde una laptop, desde un café internet o una computadora no propia en casa de un amigo, etc. sin que exista transferencia de archivos a la máquina desde la que accede (sólo consulta). La comunicación se da por el puerto 143.

SASL (Simple Authentication and Security Layer): es un método para proveer a un protocolo de comunicación el soporte de autentificación. Con el empleo de SASL, un protocolo usa un comando para identificar y autentificar la conexión del usuario ante el servidor y negocia la protección a la subsecuente interacción del protocolo.

TLS (Transport Layer Security): es un nivel (Layer) que garantiza seguridad (conexión segura) y privacidad a la interacción de un protocolo ya que provee mecanismos de encriptación de la comunicación. Su antecesor es SSL (Security Socket Layer). Los protocolos en nivel TLS cambian su nomenclatura y el puerto de comunicaciones que utilizan. Así http (puerto 80) sobre TLS cambia a https (puerto 443). Los protocolos mencionados cambian a: SMTPs puerto 465, POP3s puerto 995, IMAPs puerto 993. [28]

Uso de SMTP AUTH

Autentificación.- cuando un cliente, con cuenta activa, desea enviar un correo a través del servidor, el software cliente MDA debe acceder al servidor mediante el protocolo SMTP. El servidor verifica si la dirección IP del cliente que hace la solicitud, pertenece a la red autorizada. Si pertenece, el correo se envía, si no (que puede deberse a que el cliente se conecta desde cualquier sitio en el mundo a través de internet); para que el servidor autorice al cliente al envío de dicho correo, el cliente debe autentificarse como usuario del servicio mediante un login y una contraseña. A este mecanismo se le llama SMTP AUTH. Por otro lado, cuando el cliente, desea consultar los correos que le han enviado y que se encuentran en sus buzones, el acceso al servidor se realiza mediante el protocolo IMAP o POP3. El servidor debe autorizar al cliente (IMAP AUTH) mediante su login y contraseña. Este software en ambos casos es Cyrus-SASL.

¿Por qué Postfix?

Postfix es un proyecto de software libre robusto, confiable y altamente seguro, que surge como alternativa amigable al tradicional sendmail, siendo más rápido, fácil de administrar y sobre todo mucho más seguro. Postfix fue implementado en compatibilidad a sendmail en el sentido de una fácil asimilación y migración para los administradores que usan sendmail. Su alto grado de confiabilidad aún en situaciones de stress (uso al límite de espacio en disco y memoria) lo convierten en opción favorable capaz de enviar un millón de correos diferentes cada día.

Algunas de las características principales de Postfix son:

- Amplia difusión.

Postfix debe ser usado por muchas personas para causar un impacto significativo en el funcionamiento y seguridad del correo electrónico en Internet. Por lo tanto, el software es ofrecido gratis.

- Funcionamiento.

Una PC de escritorio corriendo Postfix puede recibir y enviar un millón de mensajes distintos al día. Postfix usa procedimientos de servidores web para reducir la creación de procesos sin perder fiabilidad.

- Compatibilidad.

Postfix está diseñado para ser compatible con sendmail y así hacer una migración hacia el primero de una manera sencilla.

- Seguridad y robustez.

Postfix está diseñado para comportarse racionalmente bajo situaciones complicadas. Cuando el sistema local se queda sin espacio en el disco o sin memoria libre, Postfix se desactiva en vez de agravar el problema. Por características de diseño, el programa Postfix no se colapsa cuando el número de mensajes aumenta; está diseñado para permanecer controlado.

- Flexibilidad.

Postfix está construido sobre varios pequeños programas que realizan sólo una específica tarea cada uno: recibir un mensaje vía SMTP, entregar un mensaje vía SMTP. Entregar un mensaje localmente, reescribir una dirección.

- Seguridad.

Postfix usa múltiples capas de defensa para proteger el sistema local contra intrusos. Casi todos los demonios de Postfix pueden ejecutarse con bajos privilegios. No hay un camino directo desde la red hasta los programas de entrega local sensibles a la seguridad; un intruso tiene que atravesar muchos otros programas primero. Incluso Postfix no confía en los contenidos de su propia cola de archivos.

- Dominios virtuales.

En la mayoría de los caso, añadiendo el soporte para un dominio virtual requiere el cambio a sólo una tabla lookup de Postfix. Otros agentes de correo normalmente necesitan niveles múltiples de aliases o redirecciones para conseguir el mismo resultado.

- Control de UCE (Unsolicited Commercial Email; correo publicitario no solicitado).

Postfix admite restricciones en cuanto al correo entrante. Implementa los aspectos habituales (listas negras, tablas de búsqueda DNS HELO/sender,…).

Postfix no necesita ningún hardware específico adicional para funcionar. Tampoco tiene limitaciones por las características técnicas (velocidad, memoria,…) de la máquina donde se esté ejecutando.

Configuracion

El fichero */etc/cyrus.conf*

Este fichero de configuración consta de tres partes claramente diferenciadas:

1. **START**: esta sección lista los scripts que se ejecutarán antes de que se arranquen los servicios. Su uso más característico es inicializar las bases de datos y lanzar los servicios de larga ejecución.
2. **SERVICES**: esta sección es el corazón del fichero */etc/cyrus.conf*, pues describe los procesos que deberán lanzarse para atender las conexiones que los clientes hagan a ciertos sockets, bien sean tipo TCP o UNIX.
3. **EVENTS**: esta sección lista los procesos que deberían ejecutarse a intervalos específicos, de modo similar a los trabajos del *cron*. Típicamente se usa para llevar a cabo tareas programadas de limpieza y mantenimiento.

START: esta sección lista los scripts que se ejecutarán antes de que se arranquen los servicios. Su uso más característico es inicializar las bases de datos y lanzar los servicios de larga ejecución.

SERVICES: esta sección es el corazón del fichero /etc/cyrus.conf, pues describe los procesos que deberán lanzarse para atender las conexiones que los clientes hagan a ciertos sockets, bien sean tipo TCP o UNIX.

EVENTS: esta sección lista los procesos que deberían ejecutarse a intervalos específicos, de modo similar a los trabajos del cron. Típicamente se usa para llevar a cabo tareas programadas de limpieza y mantenimiento. [29]

Es suficiente con realizar un único cambio, y es comentar la línea donde se declara la ejecución del servicio pop3. Asimismo, de momento no vamos a utilizar IMAP sobre SSL, así es que no tocaremos la línea que hace referencia a ese servicio. Por lo tanto, nos quedará un fichero de configuración tal que:

```
START {
    recover     cmd="/usr/sbin/ctl_cyrusdb -r"
    delprune    cmd="/usr/sbin/ctl_deliver -E 3"
    tlsprune    cmd="/usr/sbin/tls_prune"
```

```
}
SERVICES {
    imap        cmd="imapd -U 30" listen="imap" prefork=0 maxchild=100
    lmtpunix    cmd="lmtpd" listen="/var/run/cyrus/socket/lmtp" prefork=0 maxchild=20
    sieve       cmd="timsieved" listen="localhost:sieve" prefork=0 maxchild=100
    notify      cmd="notifyd" listen="/var/run/cyrus/socket/notify"
    proto="udp" prefork=1
}
EVENTS {
    checkpoint  cmd="/usr/sbin/ctl_cyrusdb -c" period=30
    delprune    cmd="/usr/sbin/ctl_deliver -E 3" at=0401
    tlsprune    cmd="/usr/sbin/tls_prune" at=0401
}
```

En estos momentos entra en escena una cuestión importante en la configuración: usar sockets TCP o sockets UNIX. En la configuración que se acaba de presentar se ha optado por la segunda opción debido a que se considera que van a ejecutarse todos los servicios en la misma máquina. En un caso como éste, muy habitual, es mejor usar sockets UNIX debido, principalmente, al mejor rendimiento que ofrecen y a que simplifican la configuración en general. En cambio, en un contexto donde los servidores Postfix y Cyrus IMAP estén en máquinas diferentes, será necesario usar sockets TCP.

El fichero /etc/imapd.conf

/etc/imapd.conf es el fichero de configuración del servidor Cyrus IMAP y en él se definen los parámetros locales para IMAP. Cada una de las líneas debe tener el formato opción: valor, donde opción es el nombre de la opción a configurar y valor el valor al cual se está estableciendo esa opción. Las líneas en blanco o que empiecen por # son ignoradas. A continuación se detallan algunas de las opciones más relevantes y sus valores recomendados:

Altnamespace: esta opción viene por defecto con el valor no, forzando que las subcarpetas de usuario se creen debajo de inbox; si se cambia a yes, las subcarpetas del usuario se crearán a la misma altura que inbox.

lmtp_downcase_rcpt: esta opción sirve para forzar que el nombre de usuario se convierta a minúsculas, viene por defecto comentada, es decir, con valor no. Debido a que Cyrus diferencia mayúsculas y minúsculas, es una buena idea trabajar con los nombres de usuario siempre en minúsculas.

Admins: esta opción permite definir los usuarios que tendrán permisos de administrador (flag a de la ACL de un buzón) sobre todos los buzones del sistema. El usuario cyrus, y únicamente él, es la opción más recomendable, por lo que bastará con descomentar la línea del fichero. Este usuario se va a autenticar mediante el método SASL.

Allowanonymouslogin: esta opción permite el acceso anónimo a los buzones en cuyas ACLs se haya añadido al usuario anonymous. Carece de sentido a menos que se quieran implementar grupos de noticias, por lo que se dejará su valor por defecto no.

sasl_mech_list: esta es la lista de los mecanismos de autenticación que se van a soportar. Es útil para evitar que se prueben todos los plugings existentes y para definir el orden de los mismos.

sasl_pwcheck_method: Esta opción nos permite especificar el método de autenticación de los usuarios, saslauthd en este caso.

Podemos acceder a la interfaz de comandos para la gestión del servidor Cyrus IMAP mediante una llamada a *cyradm*, tal que:

```
$cyradm --user cyrus localhost
Password:
localhost>
```

Una vez dentro, el comando *help* nos mostrará una descripción de los comandos disponibles y sus alias. De entre todos, los de uso más frecuente se comentan acto seguido:

Comando	**Alias**	**Función**	**Sintaxis**	**Ejemplos**
createmailbox	cm	Crear buzones de correo	cm <buzon>	cm user.mcheong
deletemailbox	dm	Borrar buzones de correo	dm <buzon>	dm user.mcheong
listacl	lam	Listar las ACL de un buzón	lam <buzon>	lam user.mcheong
setacl	sam	Establecer las ACL en un buzón	sam <buzon> <usuario> <permisos>	sam user.mcheong mcheong lrs sam user.mcheong mcheong all sam user.suppliers group:suppliers lrswipd
deleteacl	dam	Borrar las ACL de un buzón	dam <buzon> <usuario>	dam user.mcheong mcheong

El fichero /etc/postfix/master.cf

Usaremos el protocolo LMTP para la comunicación entre el MTA Postfix y Cyrus, por lo que debemos asegurarnos de que el servicio *lmtpunix* está habilitado en el fichero */etc/cyrus.conf* y que Postfix tiene acceso a ese fichero (un socket, a efectos de permisos, funciona igual que un fichero).

Por lo tanto, nos aseguraremos de que el fichero */etc/postfix/master.cf* contenga esta línea:

```
# service   type    private  unpriv   chroot   wakeup  maxproc command + args
#                     (yes)   (yes)    (yes)    (never)  (100)
#
==========================================================================
==
lmtp        unix    -        -        n        -        -        lmtp
```

El fichero /etc/postfix/main.cf

Para conseguir que Postfix entregue los correos a Cyrus a través de LMTP deberemos configurar un transporte en el primero. No es aconsejable usar cyrdeliver. La configuración del transporte en Postfix puede hacerse de diversas maneras (default_transport, transport_maps o mailbox_transport). Usaremos mailbox_transport debido a que Postfix 2.x no pasa a minúsculas los destinatarios en las entregas por LMTP, por lo que es aconsejable usar la opción lmtp_downcase_rcpt: yes en el fichero /etc/imapd.conf. Para el uso de sockets Unix, el transporte de Postfix se especifica como lmtp:unix:/var/run/cyrus/socket/lmtp (en este ejemplo se usa la localización por defecto del socket de Cyrus en Debian, que se define en /etc/cyrus.conf).

Se necesita también un servicio lmtpd de Cyrus escuchando en ese socket, luego es conveniente asegurarse de que exista una línea como esta:

```
lmtpunix    cmd="lmtpd" listen="/var/run/cyrus/socket/lmtp" prefork=0 maxchild=20
```

en la sección SERVICES del fichero /etc/cyrus.conf. Asimismo, es imprescindible asegurarse de que tanto Cyrus como Postfix pueden hablarse a través de ese socket. Los sockets Unix funcionan igual que los ficheros, por lo que esto se traduce en que tanto el usuario cyrus como el usuario postfix pueden leer y escribir en ese fichero. Aviso: debido a que Cyrus preautentica cualquier cosa que proceda del socket Unix, cualquiera que pueda escribir en él será capaz de inyectar correo directamente en Cyrus.

A continuación vamos a activar el uso de SASL en Postfix. El objetivo es autenticar los clientes smtp para que puedan hacer relay a través del servidor de correo. Para ello se modifica la opción smtpd_recipient_restrictions del fichero /etc/postfix/main.cf:

```
smtpd_sasl_auth_enable = yes
smtpd_sasl_local_domain = nodo.cu
smtpd_recipient_restrictions =
   permit_mynetworks,
```

permit_sasl_authenticated,
reject_unauth_destination
smtpd_sasl_security_options = noanonymous

De este modo, se permite hacer relay a los clientes sin autenticar que pertenezcan a las redes indicadas en mynetworks (habitualmente la red local) y a los clientes autenticados mediante el método SASL.

Para que Postfix sepa qué mecanismo usar a través de SASL hay que crear el fichero /etc/postfix/sasl/smtpd.conf con las siguientes líneas:

pwcheck_method: saslauthd

mech_list: plain login

En algún caso, nos veremos en la necesidad de utilizar los servicios de un proveedor para procesar el envío de correo. En este caso, igualaremos el parámetro relayhost a cada uno de los proveedores de servicios que nos interesen. [30]

- *Configuraciones avanzadas*

Servidores Virtuales

Los servidores virtuales son realmente todos los dominios que gestiona nuestro servidor. Es decir, que un solo servidor de correo puede recibir e-mails para muchos dominios diferentes. La configuración de los servidores virtuales sería

mydestination = mihost.dominio.com, localhost.dominio.com, localhost, hash:/etc/postfix/virtual

Veamos lo que contiene el fichero /etc/postfix/virtual

dominiovirtual1.com ok
dominiovirtual2.com ok

Cada vez que modifiques este fichero debes ejecutar el comando

cd /etc/postfix && postmap virtual && postfix reload

Medios de transporte

Los medios de transporte sirven para desviar el correo entrante a otros servidor de correo en funcion del dominio o de donde se encuentra la cuenta de usuario. Esto es util para servidores que manejan cantidades grandes de correo. Una configuración tıpica serıa:

transport_maps = hash:/etc/postfix/transport

y el fichero /etc/postfix/transport contiene

dominio1.com smtp:servidor2.dominio2.com
dominio2.com smtp:servidor3.dominio3.com
dominio3.com smtp:servidor4.dominio4.com

En caso de que se quiera enviar el correo directamente al servidor donde se encuentra la cuenta del usuario habría que usar alguna característica del directorio que nos permita diferenciar. Esto podría ser algún atributo (ejemplo: mailHost) de los objetos del directorio o la ubicación de la cuenta dentro del mismo.
Si por ejemplo, los servidores de correo van a agrupar las cuentas por facultades y se designa un servidor por facultad, el reenvío seria:

transport_maps = ldap://etc/postfix/fac2

En el fichero fac2 iría:

server_host = ucidc3

version = 3

timeout = 10

search_base = ou=FAC02,ou=Students,ou=UCI Domain Users,dc=uci,dc=cu

server_port = 389

domain = hash:/etc/postfix/searchdomains

query_filter = (mailNickname=%u)

result_attribute = smtp:serverfac2

bind = yes

bind_dn = usuario@uci.cu

bind_pw = password

scope = sub

Cada vez que se modifique este fichero debe ejecutarse postmap /etc/postfix/fac2

Este fichero buscaría dicho usuario y desviaría el correo al servidor serverfac2. De este mismo modo se haría para las otras facultades, permitiendo aprovechar la base de búsqueda del directorio LDAP para dirigir los correos directamente a los buzones.

3.6 Servidor PDC (Samba)

El servidor Samba se instalará y configurará para que actúe como PDC de la red local. La información de las cuentas de los usuarios se almacenará en el directorio LDAP. Proveerá servicios de perfiles de usuarios. [31]

En este apartado se dará soporte, en la estructura del directorio LDAP, para almacenar los datos relativos a una cuenta de usuario Samba.

Una vez se haya incorporado esta estructura en el directorio LDAP, los usuarios que ahí se almacenen tendrán la posibilidad de autentificarse en cualquier sistema GNU/Linux y/o Microsoft© Windows que haga uso del servidor LDAP para la autentificación de usuarios. La particularidad es que tendrán la misma cuenta de acceso para los todos sistemas, tanto en GNU/Linux como en Windows, de toda la red.

Instalación del servidor.

El paquete principal del servidor Samba es "samba", a continuación con la configuración:

.

Configuración del Servidor.

Se configurará Samba como un Controlador Primario de Dominio que almacena su base de datos SAM en un servidor OpenLDAP.

Para la configuración se asumirá que:

El nombre del dominio será: NODO

El nombre del servidor NetBIOS será: Server-RH

Fichero de configuración del Samba se encuentra en */etc/samba/smb.conf:*

```
[global]

        workgroup = NODO
        netbios name = Server-RH
        security = user
        interfaces = 10.128.50.32
        enable privileges = yes
         username map = /etc/samba/smbusers
         server string =  Server-RH
         encrypt passwords = Yes
         pam password change = yes
         obey pam restrictions = yes
         ldap passwd sync = Yes
         unix password sync = Yes
         passwd program = smbldap-passwd -u %u
         passwd chat = "Changing password for*\nNew password*" %n\n "*Retype new password*"
%n\n"
         log file = /var/log/samba/log.%m
         max log size = 100000
         time server = Yes

         socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192

         mangling method = hash2
         Dos charset = 850
         Unix charset = ISO8859-1

         logon script = logon.bat
         logon drive = H:
         logon home =
         logon path =

         bind interfaces only = yes
         domain logons = Yes
         domain master = Yes
```

```
        os level = 99
        preferred master = Yes
        wins support = yes
        name resolve order = wins bcast hosts
        passdb backend = ldapsam:ldap://10.128.50.32/
        ldap admin dn = cn=Manager,dc=nodo,dc=cu
        ldap suffix = dc=nodo,dc=cu
        ldap group suffix = ou=grupos
        ldap user suffix = ou=usuarios
        ldap machine suffix = ou=maquinas
        ldap idmap suffix = ou=idmap
        ldap delete dn = Yes
        ldap ssl = On
        add user script = smbldap-useradd -m "%u"
        delete user script = smbldap-userdel "%u"
        add machine script = smbldap-useradd -t 0 -w "%u"
        add group script = smbldap-groupadd -p "%g"
        delete group script = smbldap-groupdel "%g"
        add user to group script = smbldap-groupmod -m "%u" "%g"
        delete user from group script = smbldap-groupmod -x "%u" "%g"
        set primary group script = smbldap-usermod -g '%g' '%u'
       admin users = Administrador

       idmap uid = 100000-200000
       idmap gid = 100000-200000
       template shell = /bin/false
       winbind use default domain = no

[homes]
        comment = Carpeta de %U, %u
        valid users = %U
        read only = No
        create mask = 0664
        directory mask = 0775
        browseable = No

[netlogon]
        comment = Network Logon Service
        path = /home/netlogon
        browseable = No
        read only = Yes
        guest ok = Yes

[profiles]
        comment = Perfiles de Usuarios
        path = /home/profiles
        create mask = 0600
        directory mask = 0700
        browseable = No
        writeable = yes
```

force user = %U
valid users = %U, @"administradores_dominio"
write list = %U
read only = No
profile acls = Yes
csc policy = disable
nt acl support = No
map acl inherit = Yes
case sensitive = No
dont descend = /proc,/dev,/etc,/lib,/lost+found,/initrd

Usar la herramienta ofrecidas por el mismo Samba para comprobar la configuración.

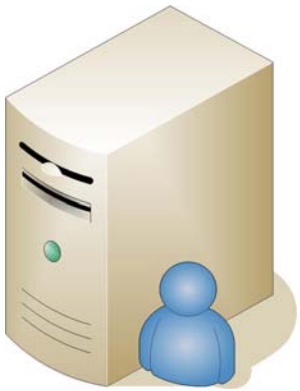
Ejemplo: usar el comando [testparm]

Server-deb: ~# testparm

Load smb config files from /etc/samba/smb.conf
Processing section "[profiles]"
Processing section "[netlogon]"
Processing section "[Homes]"
Loaded services file OK.
Server role: ROLE_DOMAIN_PDC

Esta herramienta nos chequea la configuración y no avisa de algunas anomalías en ella ya sea errores o warning.

El resultado de la comprobación a la configuración, muestra lo deseado. Ya está instalado y configurado un Controlador de Dominio Primario con soporte LDAP que permite establecer políticas de seguridad con usuarios, grupos y computadoras; semejante con lo que ofrece un Windows® Advanced Server 2000 y 2003, pero usando completamente Software Libre.



3.7 Servidor de mensajería instantánea (jabber)

De acuerdo a la existencia de un JID (Identificación Jabber, por su siglas en inglés) se realiza la instalación y configuración para hacer coincidir con la cuenta de correo, y así ofrecer una mayor integridad. Se debe realizar la modificación pertinente en el DNS para lo siguiente después al @ responda también el servidor Jabber. En esta tesis estamos usando el software EJabberd.

Hay que mencionar que el Ejabberd es una versión mejorada del anterior jabberd2. Entre algunas de las ventajas esta la fácil instalación y configuración, además el paquete de software incluye todos lo modulos para conexión con otros transportes ya sea Yahoo, ICQ, IRC, etc, soporte tls, integración con LDAP incorporada, entre otras ventajas.[32]

Se procede con la configuración donde del paquete EJabberd con soporte LDAP.

Configuración del Servidor jabber

La configuración del servidor se encuentra en el directorio /etc/ejabberd/ejabberd.cfg

1ro:

Configuramos un usuario para las tareas de administración del servidor

%% Admin user

{acl, admin, {user, "Administrador", "jabber.nodo.cu"}}.

2do:

Luego le ponemos el nombre DNS.

%% Hostname

{hosts, ["jabber.nodo.cu"]}.

3ro:

Comentamos la línea [auth_method, internal] para que no use los usuarios internos del servidor y use los que están creados en el servidor LDAP.

% Authentication method. If you want to use internal user base, then use

% this line:

%{auth_method, internal}.

4to

Descontentamos el soporte LDAP pues por defecto viene deshabilitado.

% For LDAP authentication use these lines instead of above one:

{auth_method, ldap}.

{ldap_servers, ["ldap.nodo.cu"]}. % List of LDAP servers

{ldap_uidattr, "cn"}. % LDAP attribute that holds user ID

{ldap_base, "ou=usuarios,dc=nodo,dc=cu"}. % Search base of LDAP directory

{ldap_rootdn, "cn=admin,dc=nodo,dc=cu"}. % LDAP manager

{ldap_password, "password"}. % Password to LDAP manager

De esta forma ya tenemos correctamente configurada la autenticación con el servidor LDAP

3.8 Servidor Proxy (Squid)

La autentificación de LDAP sobre el Servidor Proxy se basa en un software llamado *squid_ldap_auth y squid_ldap_group, estos demonios "ayudan" a autenticar los usuarios cuando necesitan navegar a través del proxy.[33]*

Configuración de Servidor Proxy.

La configuración del servidor se encuentra en el directorio /etc/squid/squid.conf

auth_param basic program /usr/lib/squid/squid_ldap_auth

-h ldap.nodo.cu -b "ou=usuarios,dc=nodo,dc=cu" -f "uid=%s"

#Para grupos LDAP

external_acl_type ldap_group %LOGIN /usr/lib/squid/squid_ldap_group

-h ldap.nodo.cu -b "ou=grupos,dc=nodo,dc=cu" -f "(&(objectClass=posixGroup)(cn=%g)(memberUid=%u))"

Esto indica [squid_ldap_auth] toma el usuario/contraseña y verifica si existe en el servidor LDAP *ldap.nodo.cu* buscando el [uid] del usuario en la base *ou=People,dc=nodo,dc=cu*,

además creamos una ACL donde le “decimos” al Proxy que verifique mediante el demonio [squid_ldap_group] que para poder navegar el usuario debe pertenecer al un determinado grupo, por ejemplo creamos un grupo llamado “Internet” , todo usuario que pertenezca a dicho grupo podrá acceder a internet.[34][35]

Conclusiones

El software libre surge hace algún tiempo envuelto en una filosofía de derechos y de libertad que pretende elevar el concepto de la creación de software a los niveles mas altos de las virtudes humanas como son el altruismo y la cooperación. Su implantación en países en vía de desarrollo, como el nuestro, es una medida más para lograr la independencia de esos modelos de software que nos tratan de imponer las principales empresas capitalistas del primer mundo, así como una vía para el avance en una rama tan importante y con tanto cambio como la informática.

Con este trabajo se demostraron las facilidades que brinda el Protocolo de Acceso a Directorios Ligeros LDAP, gracias a su estructura jerárquica, capacidad de distribución de la información y gran disponibilidad. También como facilita las labores de administración permitiendo el uso de una única cuenta de usuario/contraseña para todos los servicios de red, gracias a que hoy en día la mayoría de las aplicaciones ofrecen soporte para LDAP,

Se demostraron las ventajas del uso de Postfix como herramienta gestora del correo electrónico, por ser un sistema seguro, fiable y muy adaptable al entorno. A su vez se garantiza la navegación de los usuarios a internet por medio del proxy Squid, de una manera confiable, eficiente y controlada.

Con la utilización de Samba para confeccionar el Controlador de Dominio Primario los usuarios Windows se sentirán como en un dominio Windows NT facilitando los inicios de sesión interactivos y la compartición de recursos en la red. Por último se garantiza la mensajería instantánea con la implementación del servidor Jabber, el cual provee este servicio de una manera rápida y confiable.

Luego de este análisis, se puede afirmar que la investigación alcanzó los objetivos propuestos y quedo elaborado un modelo de integración de servicios sobre LDAP con el uso del Software Libre en la Universidad de Ciencias Informáticas.

Recomendaciones

Aplicar los resultados de este estudio en la red de nuestra universidad permitiéndole dar los primeros pasos en la migración de los servicios a Software Libre. Profundizar e incorporar en métodos y herramientas que garanticen mejor la confiabilidad, accesibilidad e integridad de la información.

Dicho estudio debe ser mucho mas profundo debido a la importancia que presenta en la actualidad.

Referencias bibliográficas

[1] Ana Molina**.** [Citado: 25/05/2007.] http://www.rcci.net/globalizacion/2004/fg435.htm.

[2] Proyecto GNU y Fundación de Software Libre: “Filosofía del Proyecto GNU”. [Citado el: 1/11/2006.] http://www.gnu.org/philosophy/philosophy.es.html.

[3] Stallman, Richard: “Porqué "Software Libre" es mejor que software de "Código Fuente Abierto". [Citado el: 24/11/2006.] http://www.gnu.org/philosophy/free-software-for-freedom.es.html

[4] Raymond, Eric S.: “La Catedral y el Bazar”. [Citado el: 04/11/2006.] http://www.sindominio.net/biblioweb/telematica/catedral.html.

[5] Proyecto GNU y Fundación de Software Libre: “La Definición de Software Libre”. [Citado el: 17/11/2006.] http://www.gnu.org/philosophy/free-sw.es.html.

[6] Junco, Andrés Ricardo Almanza. 2004. Junio de 2004. [Citado el: 05/05/2007.] http://72.14.205.104/search?q=cache:w2ZlYeNWn0IJ:www.acis.org.co/memorias/JornadasSeguridad/IVJNSI/AndresAlmanza-IVJNSI.pdf+ldap+%2B+importancia&hl=es&ct=clnk&cd=20&gl=cu&lr=lang_es&client=firefox-a.

[7] Wikipedia, la Enciclopedia Libre: “Software Libre”. [Citado el: 02/03/2007.] http://es.wikipedia.org/wiki/Software_libre.

[8] BULMA: Como configurar SQUID, el Proxy-Cache de Internet [Citado el: 14/04/2007.] http://bulma.net/body.phtml?nIdNoticia=441

[9] Wikipedia: La Enciclopedia Libre: “LDAP”. [Citado el: 11/10/2006.] http://es.wikipedia.org/wiki/LDAP.

[10] "Productos Linux. Servicio de Directorio," Desarrollo & Webhosting Enet Ltda, 2004. [Citado el: 10/10/2006.] http://www.enetchile.cl/productos/productoslinux/directorio.php

[11] Wikipedia, la Enciclopedia Libre: “Mapa Conceptual de Software Libre”. [Citado el: 04/11/2006.] http://es.wikipedia.org/wiki/Imagen:Mapa_conceptual_software_libre.png. Fecha Visita.

[12] L. E. Pinheiro-Malere, "LDAP-Linux-Cómo es," 1999. [Citado el: 10/10/2006.] Http://www.insflug.org/COMOs/LDAP-Linux-Como/LDAP-Linux-Como-1.html

[13] R. González, "LDAP: un protocolo para Servicio de Directorio," Revista Telemática, vol. Año I, pp. 3-5,2002. [Citado el: 17/12/2006.] http://www.cujae.edu.cu/revistas/telematica/

[14] B. Coles, "Guía Gentoo para la autenticación con OpenLDAP," Gentoo Linux, 2005. [Citado el: 13/12/2006.] http://www.gentoo.org/doc/es/ldap-howto.xml

[15] RFCs: “An Approach for Using LDAP as a Network Information Service”. [Citado el: 21/04/2007.] http://www.faqs.org/rfcs/rfc2307.html

[16] Dierkes, BJ: “Howtos Self Signed SSL Certificates”. [Citado el: 17/12/2006.] http://en.wikipedia.org/wiki/howtos_self_signed_ssl_certificates.

[17] Roncero, Jesús: “Autentificación de un cliente linux a través de LDAP”. [Citado el: 05/12/2006.] http://bulmalug.net/body.phtml?nIdNoticia=1371.

[18] CAFELUG: “Entendiendo LDAP”. Universidad Tecnológica Nacional Regional. 2002.

[19] RFCs: “Lightweight Directory Access Protocol (v3)”. [Citado el: 24/03/2007.] http://www.faqs.org/rfcs/rfc2251.html.

[20] Van Meer, Roel: “LDAP Implementation HOWTO”. [Citado el: Visita: 05/03/2006.] http://www.tldp.org/HOWTO/LDAP-Implementation-HOWTO.

[21] Donnelly, Michael: “Una Introducción a LDAP”. [Citado el: 15/12/2006.] http://ldapman.org/articles/sp_intro.html.

[22] RFCs: “A Summary of the X.500 (96) User Schema for use with LDAPv3”. [Citado el: 27/03/2007.] http://www.faqs.org/rfcs/rfc2256.html.

[23] RFCs: “The LDAP URL Format” [Citado el: 22/02/2007.] .http://www.faqs.org/rfcs/rfc2255.html.

[24] RFCs: “The String Representation of LDAP Search Filters”. [Citado el: 22/02/2007.] http://www.faqs.org/rfcs/rfc2254.html.

[25] RFCs: “Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions”. [Citado el: 24/01/2007.] http://www.faqs.org/rfcs/rfc2252.html.

[26] RFCs: “Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names”. ”. [Citado el: 29/01/2007.] http://www.faqs.org/rfcs/rfc2253.html.

[27] González González, Sergio: “Integración de redes con OpenLDAP, Samba, CUPS y PyKota”. [Citado el: 05/10/2006.] http://es.tldp.org/Tutoriales/doc-openldap-samba-cupspython/ html/ldap%2Bsamba%2Bcups%2Bpykota.html. Portugal. 2004.

[28] SMTP Authentication with Postfix using files or MySQL [Citado el: 29/04/2007.]

http://small.dropbear.id.au/myscripts/postfixmysql.html

[29] Postfix SASL Howto [Citado el: 25/04/2007.] http://www.postfix.org/SASL_README.html

[30] Postfix SMTP AUTH (and TLS) HOWTO [Citado el: 26/04/2007.] http://postfix.state-ofmind. de/patrick.koetter/smtpauth/

[31] “Servidores LDAP," Linux Works. Información, Productos y Servicios para Linux, 2005. [Citado el: 13/12/2006.] http://www.linux-works.com.ar/servidores.shtml

[32] Protocolo de comunicaciones: Jabber. [Citado el: 10/05/2007.] http://forge.morfeoproject.org/wiki/index.php/Mejora_GForge_7:_Incluir_mejoras_en_la_comunicaci%C3%B3n_de_los_usuarios.

[33] Configuración de SQUID [Citado el: 13/04/2007.] http://es.tldp.org/Tutoriales/doc-servir-webescuela/doc-servir-web-escuela-html/x518.html

[34] Configure squid for LDAP authentication using squid_ldap_auth helper: [Citado el: 21/04/2007.] http://www.cyberciti.biz/tips/howto-configure-squid-ldap-authentication.html.

[35] Squid - Wikipedia, la enciclopedia libre [Citado el: 13/04/2007.] http://es.wikipedia.org/wiki/Squid

Bibliografía

Ana Molina**.** [Citado: 25/05/2007.] http://www.rcci.net/globalizacion/2004/fg435.htm.

BULMA: Como configurar SQUID, el Proxy-Cache de Internet [Citado el: 14/04/2007.] http://bulma.net/body.phtml?nIdNoticia=441

B. Coles, "Guía Gentoo para la autenticación con OpenLDAP," Gentoo Linux, 2005. [Citado el: 13/12/2006.] http://www.gentoo.org/doc/es/ldap-howto.xml

Configuración de SQUID [Citado el: 13/04/2007.] http://es.tldp.org/Tutoriales/doc-servir-webescuela/doc-servir-web-escuela-html/x518.html

Configure squid for LDAP authentication using squid_ldap_auth helper: [Citado el: 21/04/2007.] http://www.cyberciti.biz/tips/howto-configure-squid-ldap-authentication.html.

CAFELUG: “Entendiendo LDAP”. Universidad Tecnológica Nacional Regional. 2002.

Dierkes, BJ: “Howtos Self Signed SSL Certificates”. [Citado el: 17/12/2006.] http://en.wikipedia.org/wiki/howtos_self_signed_ssl_certificates.

Donnelly, Michael: “Una Introducción a LDAP”. [Citado el: 15/12/2006.] http://ldapman.org/articles/sp_intro.html.

González González, Sergio: “Integración de redes con OpenLDAP, Samba, CUPS y PyKota”. [Citado el: 05/10/2006.] http://es.tldp.org/Tutoriales/doc-openldap-samba-cupspython/html/ldap%2Bsamba%2Bcups%2Bpykota.html. Portugal. 2004.

Junco, Andrés Ricardo Almanza. 2004. Junio de 2004. [Citado el: 05/05/2007.] http://72.14.205.104/search?q=cache:w2ZlYeNWn0IJ:www.acis.org.co/memorias/Jornad

asSeguridad/IVJNSI/AndresAlmanza-IVJNSI.pdf+ldap+%2B+importancia&hl=es&ct
=clnk&cd=20&gl=cu&lr=lang_es&client=firefox-a.

L. E. Pinheiro-Malere, "LDAP-Linux-Cómo es," 1999. [Citado el: 10/10/2006.]
Http://www.insflug.org/COMOs/LDAP-Linux-Como/LDAP-Linux-Como-1.html

Proyecto GNU y Fundación de Software Libre: “Filosofía del Proyecto GNU”. [Citado el: 1/11/2006.] http://www.gnu.org/philosophy/philosophy.es.html.

"Productos Linux. Servicio de Directorio," Desarrollo & Webhosting Enet Ltda, 2004. [Citado el: 10/10/2006.] http://www.enetchile.cl/productos/productoslinux/directorio.php

Proyecto GNU y Fundación de Software Libre: “La Definición de Software Libre”. [Citado el: 17/11/2006.] http://www.gnu.org/philosophy/free-sw.es.html.

Postfix SASL Howto [Citado el: 25/04/2007.] http://www.postfix.org/SASL_README.html

Postfix SMTP AUTH (and TLS) HOWTO [Citado el: 26/04/2007.] http://postfix.state-ofmind.
de/patrick.koetter/smtpauth/

Protocolo de comunicaciones: Jabber. [Citado el: 10/05/2007.] http://forge.morfeoproject.
org/wiki/index.php/Mejora_GForge_7:_Incluir_mejoras_en_la_comunicaci%C3%B3n_de_los_usuarios.

Raymond, Eric S.: “La Catedral y el Bazar”. [Citado el: 04/11/2006.]
http://www.sindominio.net/biblioweb/telematica/catedral.html.

R. González, "LDAP: un protocolo para Servicio de Directorio," Revista Telemática, vol. Año I, pp. 3-5,2002. [Citado el: 17/12/2006.] http://www.cujae.edu.cu/revistas/telematica/

RFCs: “An Approach for Using LDAP as a Network Information Service”. [Citado el: 21/04/2007.]
http://www.faqs.org/rfcs/rfc2307.html

Roncero, Jesús: “Autentificación de un cliente linux a través de LDAP”. [Citado el: 05/12/2006.] http://bulmalug.net/body.phtml?nIdNoticia=1371.

RFCs: “Lightweight Directory Access Protocol (v3)”. [Citado el: 24/03/2007.] http://www.faqs.org/rfcs/rfc2251.html.

RFCs: “A Summary of the X.500 (96) User Schema for use with LDAPv3”. [Citado el: 27/03/2007.] http://www.faqs.org/rfcs/rfc2256.html.

RFCs: “The LDAP URL Format” [Citado el: 22/02/2007.] .http://www.faqs.org/rfcs/rfc2255.html.

RFCs: “The String Representation of LDAP Search Filters”. [Citado el: 22/02/2007.] http://www.faqs.org/rfcs/rfc2254.html.

RFCs: “Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions”. [Citado el: 24/01/2007.] http://www.faqs.org/rfcs/rfc2252.html.

RFCs: “Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names”. ”. [Citado el: 29/01/2007.] http://www.faqs.org/rfcs/rfc2253.html.

SMTP Authentication with Postfix using files or MySQL [Citado el: 29/04/2007.]

http://small.dropbear.id.au/myscripts/postfixmysql.html

“Servidores LDAP," Linux Works. Información, Productos y Servicios para Linux, 2005. [Citado el: 13/12/2006.] http://www.linux-works.com.ar/servidores.shtml

Squid - Wikipedia, la enciclopedia libre [Citado el: 13/04/2007.] http://es.wikipedia.org/wiki/Squid

Stallman, Richard: “Porqué "Software Libre" es mejor que software de "Código Fuente Abierto". [Citado el: 24/11/2006.] http://www.gnu.org/philosophy/free-software-for-freedom.es.html

Van Meer, Roel: “LDAP Implementation HOWTO”. [Citado el: Visita: 05/03/2006.] http://www.tldp.org/HOWTO/LDAP-Implementation-HOWTO.

Wikipedia, la Enciclopedia Libre: “Software Libre”. [Citado el: 02/03/2007.] http://es.wikipedia.org/wiki/Software_libre.

Wikipedia: La Enciclopedia Libre: “LDAP”. [Citado el: 11/10/2006.] http://es.wikipedia.org/wiki/LDAP.

Wikipedia, la Enciclopedia Libre: “Mapa Conceptual de Software Libre”. [Citado el: 04/11/2006.] http://es.wikipedia.org/wiki/Imagen:Mapa_conceptual_software_libre.png. Fecha Visita.

ANEXOS

Figura #1 Diagrama de Replicación multimaster en Servidores LDAP.

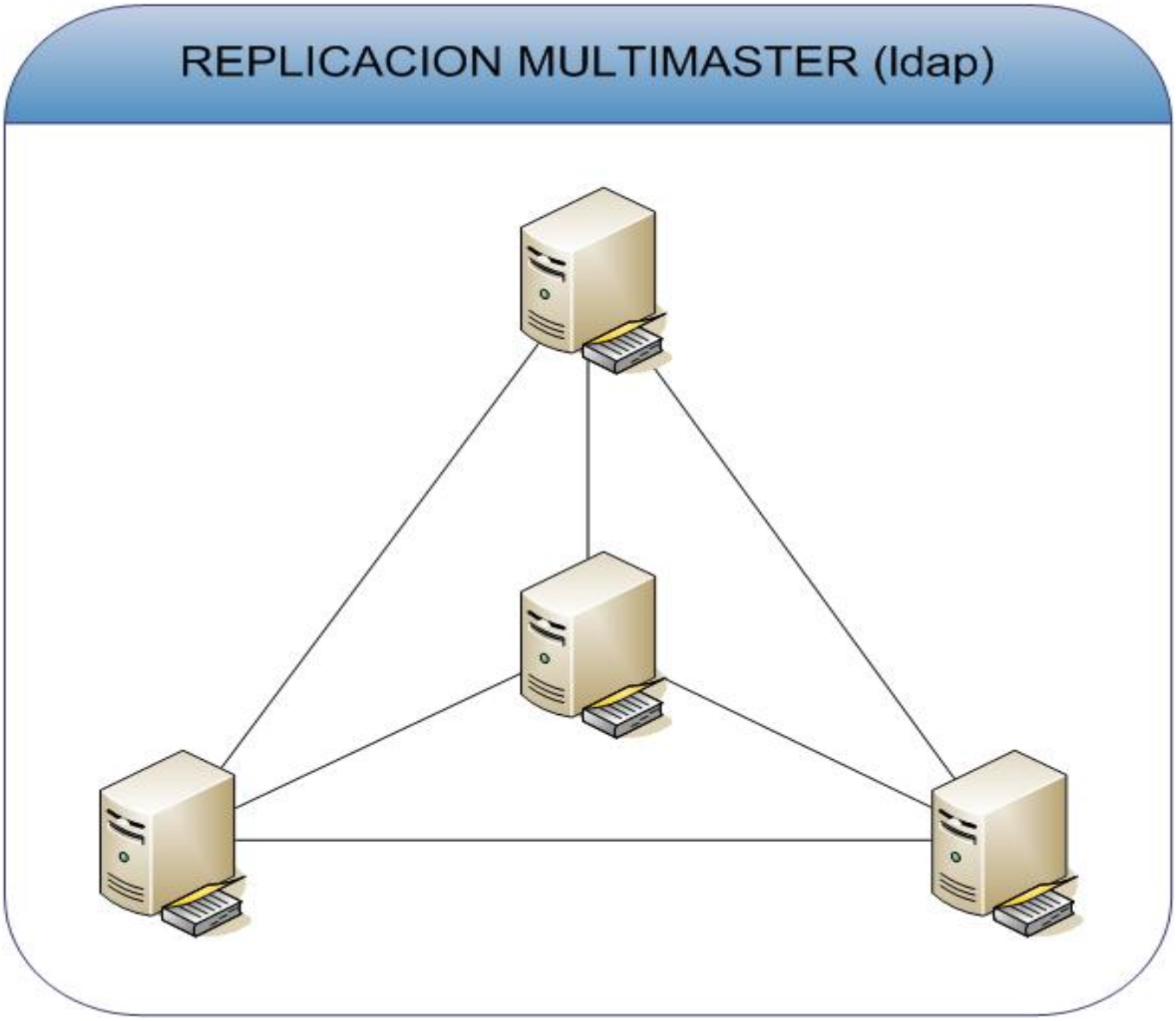


Figura # 2 Diagrama de Balanceo de carga en servidores de correos.

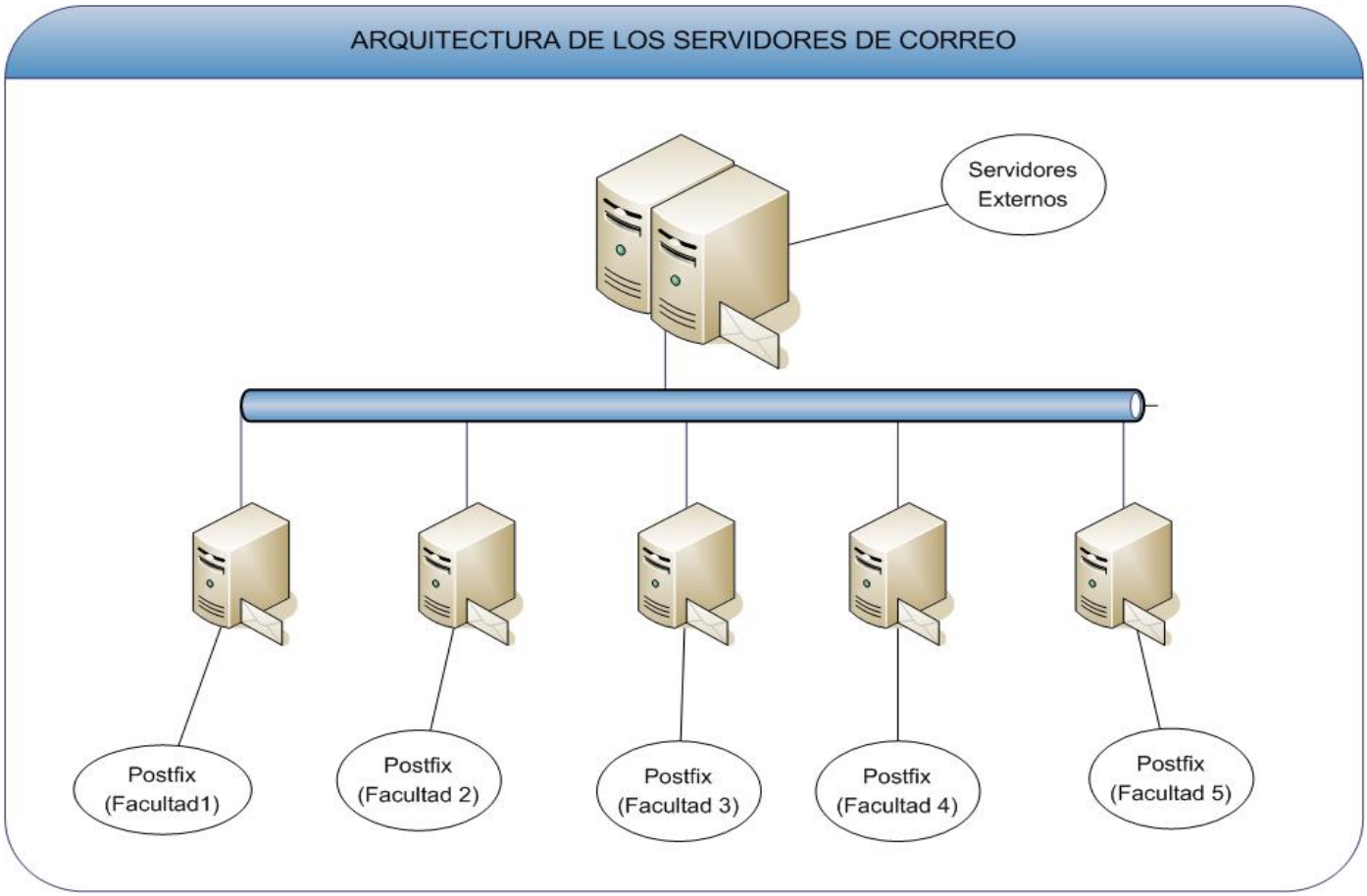


Figura # 3 Diagrama de Integración de servicios sobre LDAP.

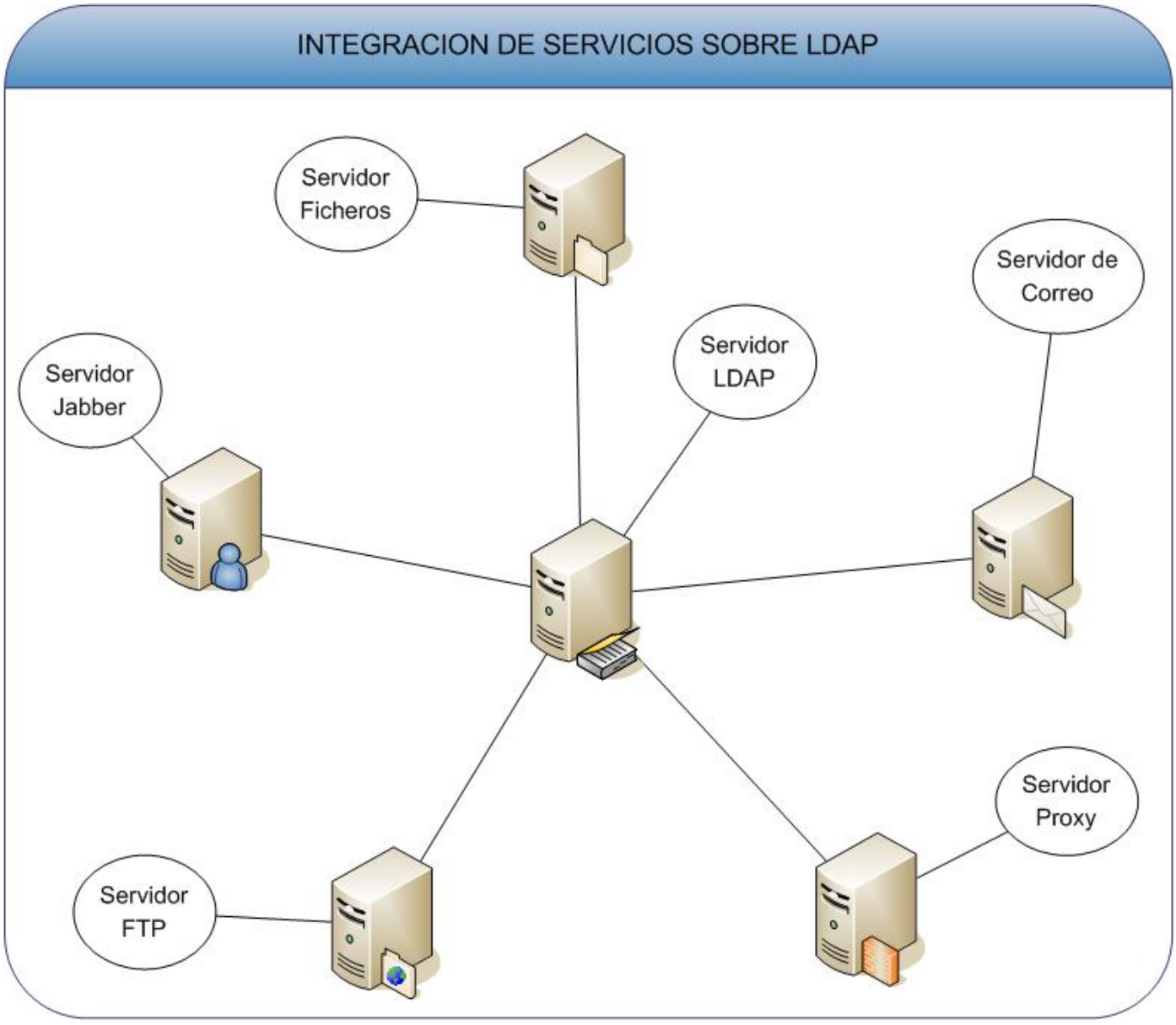
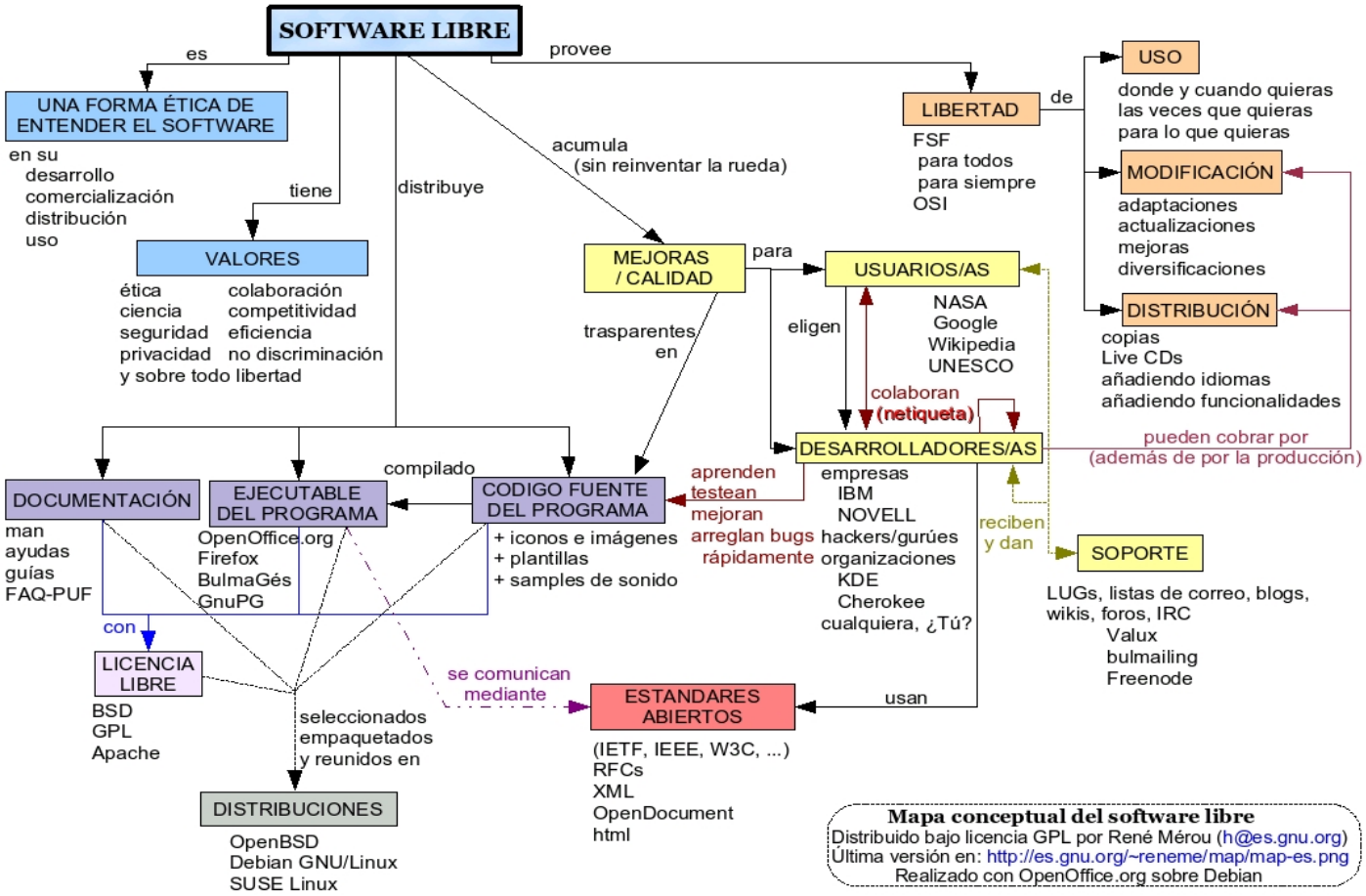


Figura # 4 Mapa conceptual del software libre. [17]



GLOSARIO

1. PAM (Pluggable Authentication Module) es una biblioteca de autentificación genérica que cualquier aplicación puede utilizar para validar usuarios, utilizando por debajo múltiples esquemas de autentificación alternativos (ficheros locales, Kerberos, LDAP, etc.). Esta biblioteca es utilizada por el proceso de "login" para averiguar si las credenciales tecleadas por el usuario (nombre y contraseña) son correctas.
2. NSS (Name Service Switch) presenta una interfaz genérica para averiguar los parámetros de una cuenta (como su UID, GID, shell inicial, directorio de conexión, etc.), y es utilizada por el proceso de "login" para crear el proceso de atención inicial del usuario.
3. OpenSSL contiene herramientas de administración y librerías para suministrar funciones criptográficas a otros (acceso seguro a sitios Web HTTPS://).
4. OpenLDAP. es una implementación Open Source del protocolo LDAP.
5. Samba es una suite que permite la interconexión, a través de la red, de sistemas Windows, Unix y otros sistemas operativos, haciendo uso de los protocolos de red nativos de Windows.
6. OpenSSH una implementación libre protocolo SSH/SecSH para la comunicación segura en las redes, una solución de seguridad que se ha ganando la confianza de los usuarios de Internet.
7. Postfix es un Agente de Transporte de Correos (MTA), que tiene la intención de ser una alternativa más rápida, fácil de administrar y segura del ampliamente utilizado Sendmail.
8. Cyrus IMAP implementa el protoco IMAPv4 ofreciendo un rápido, eficiente y robusto servicio.
9. Squid implementa un servidor proxy y un demonio para web cache.
10. Ejabberd implementa el protocolo libre Jabber para mensajería instantánea.
11. Apache es un servidor HTTP multiplataforma, implementa el protocolo HTTP/1.1 (RFC 2616) y la noción de sitio virtual.
12. PhpLDAPAdmin es programa de administración LDAP basado en Web. Permite Gestionar los accesos a los servicios relacionados con POSIX, SHADOW, SAMBA.