Universidad Central “Marta Abreu” de Las Villas

Facultad de Matemática, Física y Computación

Trabajo para optar por Título de

Licenciada en Ciencia de la Computación

Propuesta de servicios básicos de redes para una Empresa basada en software libre

Autora

Navine Alavia Dixon

Tutores

M.Sc. Manuel Castro Artiles

Ing. Alberto Rodríguez Carvajal

Santa Clara

Curso 2011-2012

"Año 54 de la Revolución"

Universidad Central “Marta Abreu” de Las Villas

Facultad de Matemática, Física y Computación

Trabajo para optar por Título de

Licenciada en Ciencia de la Computación

Propuesta de servicios básicos de redes para una Empresa basada en software libre

Autora

Navine Alavia Dixon (ndixon3126@yahoo.com)

Tutores

M.Sc. Manuel Castro Artiles (mcastro@uclv.edu.cu)

Ing. Alberto Rodríguez Carvajal (alberto@vcl.uci.cu)

Santa Clara

Curso 2011-2012

"Año 54 de la Revolución"

Hago constar que el presente Trabajo para optar por Título de Licenciada en Ciencia de la Computación ha sido realizado en la facultad de Matemática-Física y Computación de la Universidad Central “Marta Abreu” de Las Villas (UCLV) como parte de la culminación de los estudios de Licenciatura en Ciencia de la Computación, autorizando a que el mismo sea utilizado por la institución para los fines que estime conveniente, tanto de forma total como parcial y que además no podrá ser presentado en eventos ni publicado sin la previa autorización de la UCLV.

_____

Firma del Autor

Los abajo firmantes, certifican que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y que el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

_____	_____
Firma del Tutor	Firma del Tutor
M.SC. Manuel Castro Artiles	Ing. Alberto Rodríguez Carvajal

PENSAMIENTO

Y sabemos que Dios ordena todas las cosas para bien de los que le aman, de los que han sido elegidos según su designio.

Romanos 8 vs. 28

And we know that all things work together for good to them that love God, to them who are the called according to his purpose.

Romans 8 vs. 28

KJV

DEDICATORIA

I wish to dedicate this thesis:

- To my parents, Phillip and Hazel Dixon for being wonderful and supportive throughout my studies here in Cuba.
- To my husband Nathanael Clairveau for his love and continuous support, and for being my personal tower of strength in some of the darkest moments I faced as a student.
- To my cousin Ryan Kristopher O´Brian for always encouraging and believing in me.
- To my brothers and sisters: Joan, Maurah, Alrick, and Horatio for their love and support in helping me get this far and for being my motivation for finishing this career.
- To all my friends in Cuba whom have been like my family away from home.

AGRADECIMIENTOS

- Thanks to the Almighty God for his grace and mercy that has kept me both as a student and as an individual. Lord I have learnt the true meaning of the scripture that says "I can do all things through Christ who strengthens me"; for it was only through your strength I have made it thus far.
- Thanks to my mother, Hazel Dixon, my father, Phillip Dixon, my brothers, Alrick and Horatio, my sisters, Joan and Maurah, my cousins, Ryan O´Brian, Sudan Ramsey and Sheldon Cordoza, and the rest of my family, uncle Shara, Aunty Vita for all your love, guidance and support.
- Special thanks to my husband Nathanael Clairveau for being my personal tower of strength and for his genuine love and support in my darkest moments.
- Thanks to my tutors Manuel Castro Artiles and Alberto Rodríguez Carvajal for their dedication and support in the realization of this work.
- Thanks to my professors from the MFC faculty of the Central University of Las Villas for imparting unto me the priceless gift of knowledge.
- Special thanks to my professor Guillermo Sosa Gomez for his support towards me over the years.
- Thanks to my dear pastor Rev. Conrad Pitkin and the members of my home church, Faith Temple Assembles of God for their prayers and support during my stay here in Cuba.
- Special thanks to my mothers in Christ: Aunty June, Aunty Evadney and Aunty Trisha, for their spiritual support and prayers over the years.
- Special thanks to my mother-in –law Matant Clairveau whose love, care and prayers are a never ending shower of blessing over my life.
- Thanks to my roommate Kamuni Katjimbari for her friendship and moral support over the years.

- Thanks to my good friend Richard Kerr whose faith in me has allowed me to aim for the stars in all I do.
- Thanks to my friends: Kimoya Henry, Kimiesha Stenneth, Nicole White, Debbie Malcolm, Laurna Raul, Natalia Godfrey, George Peters, Kishauna Baptist, Tercy Josephs, Laurna Raoul, Laura Perez, Yvette Alfonzo Lopez, Fidelis Alexander, Antonio Bouza, and Ravi Lamontagne, whose companionship and friendship over the years has served as a source of strength and motivation making me a better person and more dedicated student and I count myself privileged to be called your friend.

Thanks to you all.

Navine A. Dixon-Clairveau

RESUMEN

En la actualidad la mayoría de los administradores de red cubanos utilizan el servidor de directorio Active Directory del Windows Server que es una solución propietaria. Sin embargo desde abril del 2004 el Ministerio de la Informática y de Comunicaciones (MIC) dispuso la migración progresiva hacia software libre. Uno de los servidores de directorio en software libre más utilizados y que brinda una gran variedad de utilidades es el OpenLDAP. Existen diferentes herramientas que permiten la administración del servidor de directorio OpenLDAP de forma visual, sin embargo no cuentan con gran cantidad de facilidades y prestaciones, provocando rechazo a la migración a software libre. En el presente trabajo se muestra como puede introducirse OpenLDAP en un ambiente heterogéneo así como la instalación y configuración de los servicios básicos de redes facilitando la migración hacia el software libre.

Además, se hace una propuesta de integración de recursos en un dominio con diferentes servicios utilizando software libre que puede ser utilizado en cualquier empresa. Esa propuesta se hace sobre los servicios de directorios, correo electrónico, acceso a internet y servicios de infraestructura de redes.

ABSTRACT

Today most Cuban network administrators utilize ***Active Directory for Windows Servers*** as solution for providing directory service with their organizations. However, in Cuba, since April 2004, the Ministry of Information and Communications (MIC) has ordered a gradual migration towards the utilization of free software within the country. Unfortunately there is a tendency to reject this migration process due to a general lack of functional features in the many existing directory service tools today. OpenLDAP however is one of the most efficient and commonly used free directory service tools available and is well known to provide a wide range of utilities and functionality for its support. This project therefore shows how one can introduce OpenLDAP in a heterogeneous environment, facilitating a fully functional domain. It shows how to realize the installations and configuration necessary for the basic network services, all this facilitating an easy transition towards free software.

A proposal is made for the integration of the resources within a domain and its basic network services utilizing free software. This proposal serves as a viable guide for any company in need of them. It includes implementation of services such as directory services, electronic mail, internet access and services of network infrastructure.

ÍNDICE

INTRODUCCIÓN

La información en todas sus formas se ha convertido en un activo de muy alto valor, es el recurso económico básico que genera mayor capital por lo que es necesario protegerla y asegurarla para garantizar su integridad, confidencialidad y disponibilidad. La utilización de computadoras en el manejo de la información como elemento indispensable en la actualidad ha permitido incrementar el uso de aplicaciones electrónicas como correo, comercio electrónico, transacciones y dinero electrónico, firmas y certificados digitales, acceso a bancos de datos y otras aplicaciones.

El conocimiento y manejo de distintas técnicas, procedimientos y herramientas de administración es una necesidad mundial y en especial de las empresas cubanas que cada día se incorporan con mayor fuerza al trabajo con sistemas informáticos en redes ofreciendo diversos servicios a sus usuarios. Una técnica muy utilizada actualmente consiste en utilizar máquinas virtuales para implementar los sistemas y probarlos antes de que entren en producción. Esto permite familiarizar a los administradores con los servicios que se ofrecen y ajustarlos a los requerimientos de la empresa antes de que los usuarios tengan acceso a los mismos, de igual manera se prueban diferentes herramientas y configuraciones que se desea utilizar.

El costo del software propietario obliga a que muchos países procuren vías alternativas en su desarrollo realizando la migración hacia el software libre y Cuba se encuentra enfrascada en este proceso que se desarrolla de manera progresiva dando pasos cada vez más orientados a la sustitución de aplicaciones y sistemas completos con software libre a la vez que incentiva el desarrollo de soluciones propiamente cubanas para lo cual se han creado diversos centros que se encargan de este proceso a diferentes niveles y con diferentes responsabilidades.

Planteamiento del problema

En la actualidad se ha orientado el proceso de migración hacia software libre como la política para el país, este proceso se ve limitado por la falta de preparación de gran parte del personal dedicado a manejo de redes. Existen administradores calificados para realizar este trabajo de manera natural pero hay un alto porcentaje de administradores que no están aptos para enfrentar este proceso debido a su preparación por lo que es importante ofrecer una guía que le sirva como ayuda para poder tener operativo un dominio y sus servicios básicos en el menor tiempo posible.

Para dar respuesta a este problema es necesario definir qué servicios se ofrecen normalmente en una empresa, qué software es más utilizado y más confiable para cada aplicación, y qué sistema operativo pudiera utilizarse, además, es necesario definir las políticas de control y acceso de la red.

Objetivo general

Proponer un grupo de servicios básicos necesarios, empleando herramientas de software libre, que garanticen el correcto funcionamiento y seguridad de una red de computadoras en el ámbito empresarial.

Objetivos específicos

Para el cumplimiento de este objetivo se ha definido un conjunto de objetivos específicos que se relacionan a continuación.

a) Definir los servicios de red básicos necesarios para una empresa.
b) Definir una estrategia para la distribución de los servicios básicos de redes según el equipamiento disponible.
c) Implementar los servidores por medio de maquinas virtuales, realizando la simulación del sistema completo y comprobando el correcto funcionamiento de los servicios.

Preguntas de investigación

1) ¿Qué tipos de sistema operativos deben utilizarse de acuerdo a las necesidades y el hardware disponible en la empresa?
2) ¿Qué servicios de red mínimos se necesitan para comenzar la migración a software libre?
3) ¿Cómo se garantiza un nivel mínimo adecuado de seguridad?

Justificación y viabilidad de la investigación

Este trabajo es necesario dada la situación general en el país con el proceso de migración al software libre.

Para realizar este trabajo se necesita montar el sistema completo y probarlo antes de instalar los servidores de producción, esto puede hacerse utilizando máquinas virtuales con configuraciones semejantes a los actuales servidores de la empresa para poder monitorear el comportamiento y determinar si cumple con los requerimientos.

Hipótesis de investigación

Con los equipos disponibles generalmente en la empresa destinados a la administración es posible lograr una red propia con un ambiente de trabajo heterogéneo que satisface los requerimientos de la empresa.

DISEÑO DE LA TESIS

Este trabajo está estructurado en tres capítulos, a continuación se presenta un resumen de los mismos:

En el primer capítulo se hace referencia al desarrollo de Linux y se ofrece una definición de los servicios básicos que se ofrecen regularmente en una red.

El Capitulo II aborda propuestas de implementación de los servicios que garanticen la interoperabilidad en un ambiente heterogéneo utilizando OpenLDAP como centro vital de organización y se realiza la caracterización de los servicios con diferentes herramientas.

Además define las aplicaciones que se necesitan instalar, la distribución de los servicios en los servidores y la seguridad de las mismas.

El Capítulo III muestra como se hace la integración de sistemas heterogéneos en el dominio y se valora el proceso de migración. También muestra en forma de ejemplo el servicio de correo electrónico funcionando.

En los Anexos se exponen los detalles de algunas aplicaciones y sus correspondientes archivos de configuración.

CAPÍTULO I: ADMINISTRACIÓN DE RED Y EL USO DEL LINUX PARA SERVIDORES

1.1 Antecedentes de la administración de redes de computadoras

La administración de redes se ha convertido en un aspecto crítico, especialmente en redes de computadores con sistemas operativos heterogéneos. El modelo Cliente – Servidor, con una gran cantidad de estaciones de trabajo, necesita de la administración de redes para manejar y controlar los servicios brindados así como los componentes asociados al hardware y al software.

1.1.1 Cambios en la Administración de Redes

Las computadoras se conectan de forma distinta a como lo hacian en un principio (una gran computadora central a la cual estaban conectadas estaciones de trabajo homogéneas). Con el advenimiento de las LANs (Local Area Networks), existen dos escenarios de redes de computadoras:

- Modelo Cliente – Servidor: un cliente requiere un servicio de un servidor que está preparado para proporcionar dichos servicios a los clientes que lo necesitan.
- Modelo peer – to – peer: no existen roles fijos como cliente y servidor, cualquier computadora puede, en un determinado momento, ser un cliente o un servidor.

La evolución de una red de computadoras a redes de tipo LAN heterogéneas fue gradual. La transformación al ambiente LAN es más complicado por la existencia de aplicaciones y protocolos de diferentes grupos de estándares y fabricantes. Sin embargo, las limitaciones de la tecnología, protocolos y topologías imponen restricciones acerca del número de computadores que se pueden conectar a la LAN. Por todos estos aspectos, la conexión y administración de redes, así como también de sus componentes, se está volviendo más y más importante.

1.1.2 Administración de la Empresa, Administración de Redes y Administración de Sistemas

Se define como **administración** al monitoreo, control y coordinación de los recursos de la computadora, los recursos usados en la conexión y comunicación de las mismas, y las aplicaciones usadas en esas computadoras. [1]
Actualmente existen dos modelos de administración de redes dependiendo del ambiente computacional. Una LAN es un típico **ambiente distribuido**, y la administración de su ambiente puede ser hecho con una administración peer-to-peer, también conocida como **administración de redes distribuida**.

El otro modelo es la **administración jerárquica** o centralizada. En este modelo la administración es realizada desde un solo punto, conocido como administrador. Pueden haber casos en donde exista un administrador que controle el funcionamiento de varios administradores y se lo conoce como administrador del administrador (*MOM, Manager of Manager*).

Para una mejor comprensión es necesario definir algunos conceptos: ***monitoreo****:* los recursos usados en redes de computadoras tienen que ser continuamente "vigilados", y cualquier comportamiento desfavorable lleva al deterioro del funcionamiento de un recurso o de la red, y por lo tanto debe ser corregido. Esto es más una acción preventiva que una reactiva. Los recursos tienen que ser ***controlados***. Esto significa que se debe permitir controlar cómo los recursos se comportan a fin de que su función se realice apropiadamente. Cuando los recursos tienen que ser monitoreados y controlados, existe un factor necesario: **coordinación.** Si no hay coordinación, la situación es del tipo de contienda general y surge el caos. [2]

1.1.3 Objetivos de la Administración de Redes:

- ***Alta disponibilidad de la red:*** Proveer eficiencia operacional, reduciendo los *downtime (tiempo en que el sistema este afuera de uso)* de la red y del sistema con tiempos de respuesta aceptables. Los problemas de la red deben ser rápidamente detectados y corregidos.
- ***Reducción de costos operacionales de red***: este es uno de los motivos primarios detrás de la administración de redes. Como las tecnologías cambian rápidamente, es deseable la administración de sistemas heterogéneos y múltiples protocolos.
- ***Reducción de cuellos de botella en la red*****:** dependiendo de cada caso en particular, puede ser deseable un monitor centralizado para administración y en otros casos esta tarea debe ser distribuida.
- ***Incrementar flexibilidad de operación e Integración***: las tecnologías de redes están cambiando a velocidades mayores que los cambios de requerimientos y necesidades. Cuando se usa una nueva aplicación, los protocolos usados en redes deberán cambiar también. Debe ser posible absorber la nueva tecnología con un costo mínimo y adicionar el nuevo equipamiento sin mucha dificultad. Además, debe permitir lograr una fácil migración de un software de administración de redes a otra versión.
- ***Alta eficiencia:*** debemos incrementar la eficiencia en detrimento de otros objetivos de la administración pero dependerá de otros factores tales como utilización, costo operacional, costo de migración y flexibilidad.
- ***Facilidad de uso***: las interfaces de usuario son críticas para el éxito de un producto. El uso de aplicaciones de administración de redes no debe incrementar la curva de aprendizaje.
- ***Seguridad:*** existen casos en donde la seguridad es un aspecto a tener en cuenta, en casos como la información de contaduría, información gerencial, etc.

1.2 Conceptos básicos

Servidor: En informática, un servidor es una computadora que, formando parte de una red, provee servicios a otros computadores denominados clientes. Algunos servicios habituales son los servicios de archivos, que permiten a los usuarios almacenar y acceder a los archivos de una computadora y los servicios de aplicaciones, que realizan tareas en beneficio directo del usuario final, es posible que un ordenador cumpla simultáneamente las funciones de cliente y de servidor.

LDAP: *Lightweight Directory Access Protocol* (Protocolo Ligero de Acceso a Directorios) es un protocolo de tipo cliente-servidor para acceder a un servicio de directorio. Se usó inicialmente como *front-end* o interfaz final para X.500, pero también puede usarse con servidores de directorios únicos y con otros tipos de servidores de directorio.

LDIF: *LDAP Data Interchange Format* (formato de intercambio de data LDAP) de forma general se usa para importar y exportar información de directorio entre servidores de directorios basados en LDAP, o para describir una serie de cambios para aplicarse al directorio. Un fichero LDIF almacena información en jerarquías de entradas orientadas a objeto. El paquete de software LDAP que se aborda en este trabajo incluye una utilidad para convenir ficheros LDIF a formato **LDBM (LDAP Data Base Management)**.

PAM: *Pluggable Authentication Modules* es un mecanismo de autenticación flexible que permite abstraer las aplicaciones y otro software del proceso de identificación.
Para identificar a un usuario que desea ingresar en su cuenta existen varios mecanismos, desde la clásica contraseña hasta diversos sistemas de identificación biométrica o claves de un solo uso. Muchos *softwares* necesitan identificar a los usuarios (servidores de correo, web, bases de datos y otros), antiguamente el programa debía ser modificado para usar cada sistema de identificación. Sin embargo, al igual que un programa de retoque fotográfico no necesita ser compilado para cada tarjeta gráfica (TG), escáner y ningún otro dispositivo, en cambio se usa un modo estándar para usar una TG u otro dispositivo apuntador y por otro lado se usa un modelo que permite manejar el dispositivo (Controlador de dispositivo).

Cuando una aplicación se prepara para usar **PAM**, esta se encarga de la autenticación y puede usar diversos métodos sin modificar la aplicación (contraseña, *token*, biometría y otros). Además, permite otras opciones como admitir el acceso en horarios preestablecidos.

1.3 El comienzo del GNU/Linux

Linux es un sistema operativo que fue creado inicialmente como un hobby por un joven estudiante, Linus Torvalds, en la Universidad de Helsinki en Finlandia. Linus tenía un interés en Minix, un pequeño sistema UNIX, y decidió desarrollar un sistema que superara los estándares de Minix. Comenzó su trabajo en 1991, cuando se lanzó la versión 0.02 y trabajó constantemente hasta 1994, cuando la versión 1.0 del núcleo Linux fue liberada. El núcleo es el corazón de todos los sistemas Linux y es desarrollado y liberado bajo la *GNU General Public License*, estando su código fuente disponible libremente para todo el mundo. En la actualidad hay cientos de empresas y organizaciones y un número igual de individuos que han lanzado sus propias versiones de sistemas operativos basados en el kernel de Linux.

Aparte del hecho de que es de distribución gratuita, la funcionalidad de Linux, la adaptabilidad y robustez, se ha convertido en la principal alternativa para sistemas propietarios como Unix y los sistemas operativos de Microsoft. IBM, Hewlett-Packard y otros gigantes del mundo de la computación han adoptado Linux y su desarrollo en curso. Así en su segunda década de existencia, Linux ha sido adoptado en todo el mundo principalmente como una plataforma para servidores de aplicaciones.

A lo largo de la mayor parte de la década de 1990, los expertos de tecnología, ignoraron en gran medida el potencial de Linux, se desestimó como un proyecto de aficionado a los ordenadores, no aptos para el público en general y las necesidades de computación. Gracias a los esfuerzos de los desarrolladores de los sistemas de administración de escritorios como KDE y GNOME, proyectos de oficina como Libre Office y el navegador web del proyecto Mozilla (por nombrar sólo unos pocos), ahora hay una amplia gama de aplicaciones que se ejecutan en Linux y puede ser utilizado por cualquier persona independientemente de sus conocimientos de informática [3].

1.3.1 Estudio No. 1 Titulado: “Tendencias de la adopción de Linux en el 2012

Según informe ***Titulado “Tendencias de la adopción de Linux en el 2012***: publicado en enero del año 2012, por *Linux Foundation* la situación actual ofrece un panorama aún más prometedor para Linux (una copia del informe completo se puede descargar desde el sitio de la *Linux Foundation*) [4] debido a que:

- Los problemas económicos mundiales pueden seguir amortiguando las previsiones de gasto para los departamentos de Tecnología Informática (TI) de todo el mundo, pero eso no impide que las grandes empresas añadan más servidores Linux para sus operaciones**.**

De hecho, un 80% de los entrevistados en una encuesta reciente de la Linux Foundation dijeron que no sólo han añadido servidores Linux en los últimos 12 meses, sino que también planean agregar más en los próximos 12 meses y durante los próximos cinco años.

Sólo un 21,7% de los encuestados planea un aumento en los servidores de Windows y más de un cuarto en realidad están planeando reducir el número de servidores Windows que utilizan en los próximos cinco años, informa la Linux Foundation. Vea los gráficos 1.1-1.3 para resultados de este estudio.

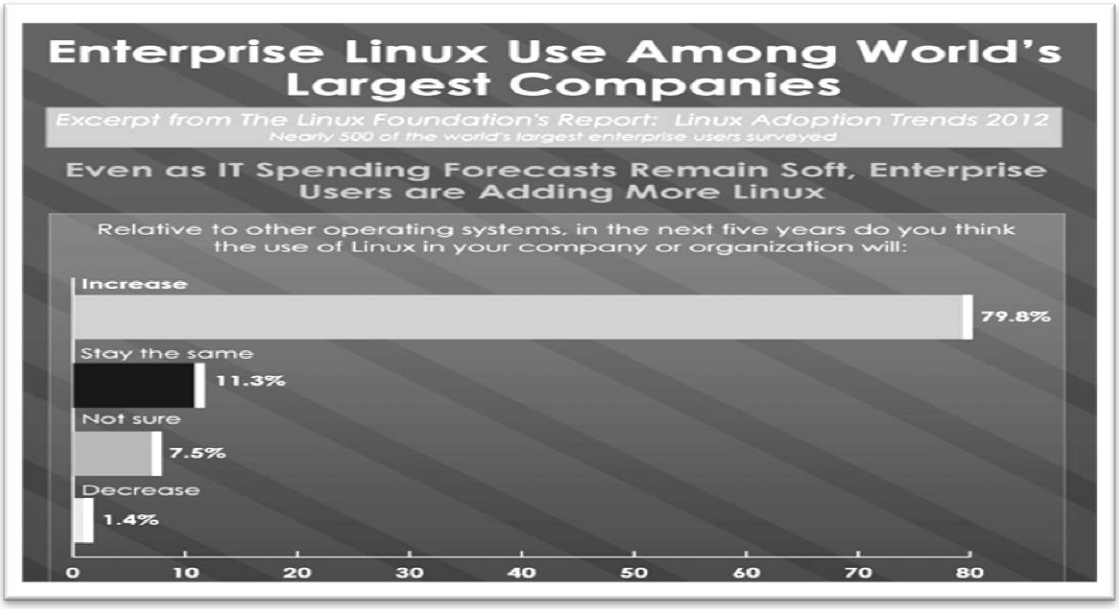


Gráfico 1.1: Resultados a la pregunta de investigación ¿"Relativo a otros sistemas operativos, en los próximos cinco años usted piensa que el uso de Linux en su compañía seria"? a) Aumentado, b) se mantiene igual c) no está seguro o d) disminuir".

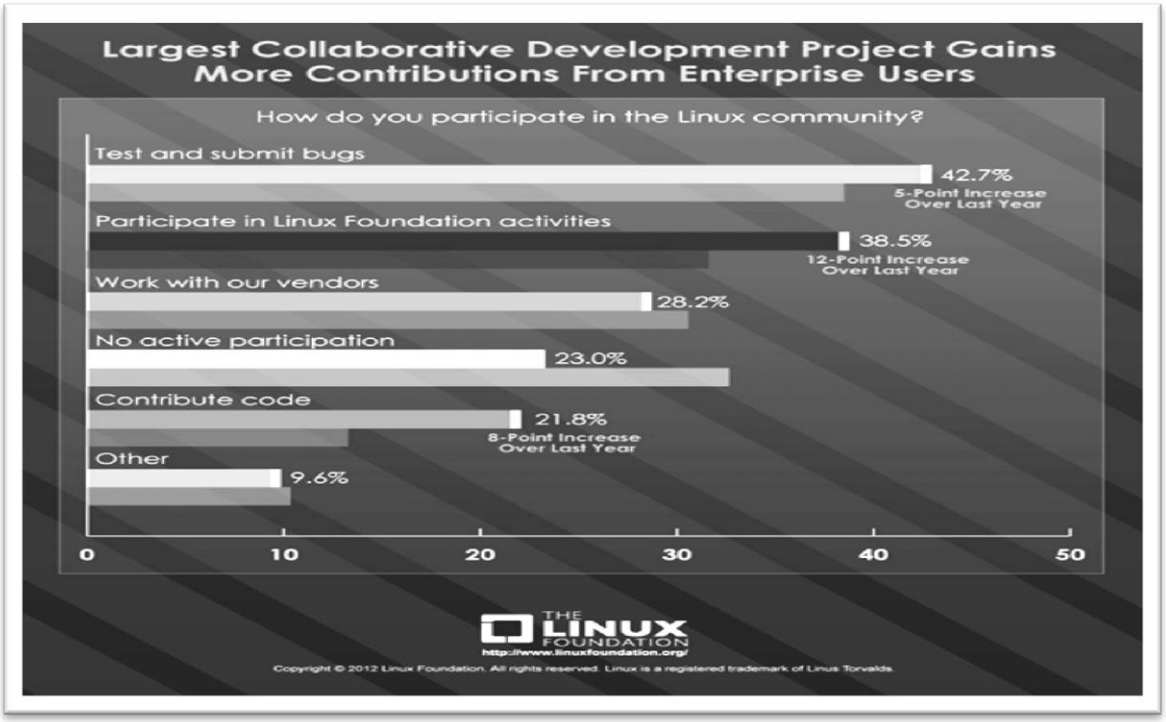


Gráfico 1.2: Resultados a la pregunta de investigación ¿"Como usted participa en la Comunidad de Linux"?.

Uso creciente de misiones críticas

La encuesta de *la Linux Foundation* se llevó a cabo en colaboración con *Yeoman Technology Group* el año pasado. Aunque se recibieron respuestas de casi 1.900 personas, el informe se centra en los datos de las empresas más grandes del mundo y las organizaciones de gobierno, representadas por 428 encuestados en organizaciones con ingresos anuales de US$ 500 millones o más, y con más de 500 empleados.

Más del 71% de estos usuarios dijo que la mayoría de sus nuevas implementaciones Linux en los últimos dos años fueron de aplicaciones y servicios nuevos, mientras que la migración a Linux fue principalmente a expensas del legado de los sistemas Windows y Unix.

Casi el 70% indicó que tiene previsto aumentar el uso de Linux en el próximo año para cargas de trabajos críticos.

Un papel clave en el "Big Data"

Una gran parte del crecimiento de Linux en las grandes empresas se debe a la computación en altos niveles y a las tendencias de "*Big Data*" (significando el gran incremento de data), indica la Linux Foundation.

Cuando se trata de los llamados Big Data, por ejemplo, más del 75% de los encuestados expresaron su preocupación por que los soporten, y casi el 72% eligen Linux por ello. Por el contrario, sólo el 35,9% dijo que utilizará Windows para satisfacer las demandas de este nuevo entorno. Ver gráfico 1.3 para resultado.

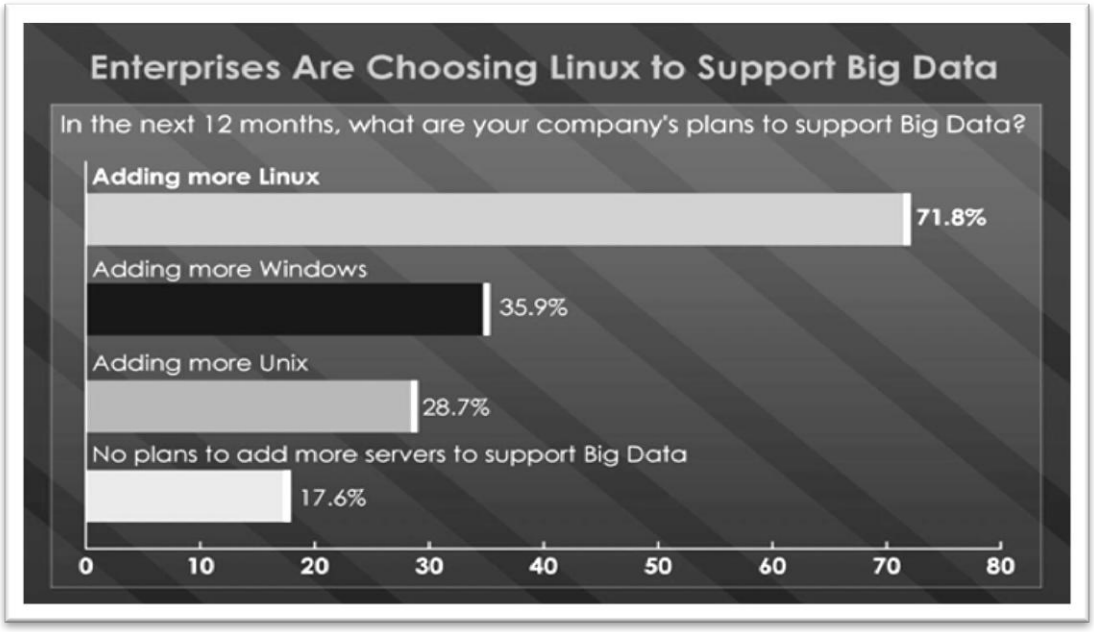


Gráfico 1.3: Resultados a la pregunta de investigación ¿"En los próximos 12 meses, cuáles son los planes para soportar grandes volúmenes de datos"?.

Al ser consultados por sus razones para elegir Linux, los encuestados citaron el costo total de propiedad (TCO), set de características, y la seguridad global como las tres principales ventajas que ofrece. De hecho, hubo un amplio acuerdo sobre la seguridad de primer nivel de Linux, con más de dos tercios de los encuestados indicando que ellos consideran que Linux es más seguro que otros sistemas operativos.

La falta de proveedores cerrados, la apertura del código, y la viabilidad a largo plazo fueron otras razones principales que citaron los encuestados.

Comparación con los mismos estudios hechos en el año 2010

En comparación con la primera encuesta de la Linux Foundation sobre este tema, los encuestados de este segundo sondeo dijeron que también ven menos problemas en el camino del éxito de Linux. Los problemas técnicos fueron citados por sólo el 12,2% – 40% menos desde el 2010, mientras que el número de los encuestados que citó la percepción de la administración como un problema se redujo en un 22%, según la fundación.

1.3.2 Comparación del uso de Servidores de Sistema Operativo Linux y los de Windows

Actualmente el sistema operativo Linux se ha extendido bastante en cuanto a implementación de servidores. A nivel de usuario, en instalaciones desktop, no es muy aceptado, ya que en ocasiones hay problemas de compatibilidad de archivos, programas, *drivers*, dispositivos, etc. A pesar de esta realidad hoy en día, las últimas versiones de Linux son tan fáciles de usar y personalizables como Windows, lo que lleva a una mejor aceptación por la comunidad.

Con esto no se pretende demostrar que Windows o Linux sea el mejor Sistema Operativo (SO), hay personas que prefieren Windows por su accesibilidad, facilidad de uso y potente soporte grafico pero en cambio hay gente que prefiere Linux por su estabilidad y optimización de recursos, sin olvidarse de la gestión de la seguridad de ficheros y que es software libre. Tabla 1.1 muestra la comparación de algunos aspectos entre los sistemas Linux y el Windows.

Tabla 1.1: Comparación de SO Linux y SO Windows

Comparativa GNU/Linux vs. Microsoft Windows **Tecnología**		
Aspecto	**GNU/Linux**	**Windows**
Filosofía	Es un sistema al que cualquiera puede acceder. Se puede distribuir, usar y modificar libremente.	Pertenece a Microsoft, que es la única autorizada tanto de realizar modificaciones como de distribuirlo.
Precio	Es software libre, de uso gratuito con tantas licencias como se deseen.	Dependiendo de las versiones, cientos de dólares por cada licencia.
Desarrollo	Mantenido por miles de voluntarios en todo el mundo, pertenece a una comunidad en la que cualquiera puede participar.	Desarrollado por Microsoft, que vende los datos técnicos imprescindibles más relevantes y oculta otros.
Código fuente	Abierto a todo el mundo.	Cerrado, secreto empresarial.

Comparativa GNU/Linux vs. Microsoft Windows Tecnología		
Estabilidad	Muy estable, cuando una aplicación se bloquea es fácil e inmediato terminar ese proceso, sin que afecte a la estabilidad del resto del sistema.	Para muchas tareas administrativas es necesario reiniciar la máquina. Cuando una aplicación se queda bloqueada repercute en el resto, llegando a comprometer la estabilidad de todo el sistema.
Seguridad	Seguro. Existen pocos virus desarrollados para este sistema.	Seguro. Para este sistema existen miles de virus desarrollados.
Difusión	Poco extendido en hogares. Utilizado generalmente para servidores.	Ocupa el 90% del mercado de ordenadores domésticos.

A continuación se exponen resultados de investigaciones realizadas sobre el costo de instalación y mantenimiento de instalaciones con servidores Windows y Linux. Según CNY Support, LLC©2012 los resultados de este estudio [6], se muestran en la tabla 1.2

Tabla 1.2: Comparación de costo de licencia e instalaciones en Servidores de Linux y de Windows

Linux®/Windows® Comparación de Costo de Licencia			
Linux®		**Windows®**	
Users	**20**		**20**
Linux Server Licence	$0.00	Windows® Server Licence	$3,999.00
Postfix/Dovecot Server License	$0.00	Exchange Server License	$3,999.00
MySQL Server License	$0.00	SQL Server Client License	$8,487.00
LibreOffice License	$0.00	MS Office Suite License	$9,999.00
MySQL Database Client License	$0.00	SQL Server Client License	$3,240.00
Linux	$0.00	Windows	$5,999.00
Linux Server CAL	$0.00	Windows Server CAL	$0.00

Postfix/Dovecot CAL	$0.00	Exchange CAL	$1,340.00
Total License Cost	**$0.00**	**Total License Cost**	**$37,063.00**

Teniendo en cuenta estos valores actuales, un sistema básico con Linux no tiene costo alguno mientras que el mismo sistema con Windows cuesta $37,063 dólares americanos, por lo que esta es una de las razones que motiva a muchas empresas a la migración.

1.4 Los sistemas operativos de la distribución GNU/Linux

A continuación se detalla una breve descripción de algunas de las distribuciones de GNU/Linux:

Slackware: una de las primeras distribuciones que aparecieron. Fue creada por Patrick Volkerding y tuvo un gran éxito en sus primeros años de existencia.

Debian GNU/Linux: una de las primeras distribuciones de GNU/Linux que aparecieron y aún siguen existiendo y evolucionado. Está desarrollada por un grupo de colaboradores distribuidos por todo el mundo y no cuenta con el respaldo de ninguna empresa. Aunque es de las más estables y seguras que existen, su sistema de instalación y configuración necesita de conocimientos previos.

RedHat Linux: junto con SuSE, es una de las distribuciones de mayor popularidad. Está creada por una empresa de EUA, aportando software de gran calidad. Tiene un entorno muy intuitivo que facilita mucho su instalación y configuración.

SuSE Linux: aunque es una distribución creada bastante recientemente, ha tenido una gran difusión. Está desarrollada por una empresa alemana, aportando mucho software propietario de calidad. Es muy completa y fácil de instalar y mantener, aunque en algunos aspectos no se siguen algunos de los estándares de la comunidad.

Fedora es una de las distribuciones más robustas y más longevas de las mencionadas, estando con nosotros desde 2003. Cuenta con el apoyo y promoción

de Red Hat y está basada en RPM (*RPM Package Manager*, antes *Red Hat Package Manager*). Fedora es el proyecto comunitario de Red Hat, una de las distribuciones comerciales más exitosas.

Ubuntu es la más famosa de todas las distribuciones Linux, Ubuntu se ha ganado este lugar por la manera en que trabaja y de haber sabido combinar flexibilidad, estabilidad, usabilidad y solidez. Está basada en Debían y esto le da de entrada una ventaja por la gran cantidad de paquetes disponibles. Con el paso de los años se ha convertido en el Sistema Operativo libre más usado, han ganado experiencia y se encuentra en una etapa de madurez que les ha permitido experimentar un poco. Marcan muchas pautas y su influencia en otras distribuciones es innegable.

1.5 Servicios básicos de redes.

De acuerdo a las dimensiones y requerimientos de una empresa su red debe proporcionar un conjunto de servicios básicos. Por ejemplo, para una red asociada a la PYME (Pequeña Y Mediana Empresa), son necesarios un grupo de servicios mínimos (básicos) que posibiliten la conectividad dentro y fuera de la red local, el acceso al correo e internet.

En este grupo de servicios tenemos:

a. Servicios de Infraestructura: DNS, DHCP
b. Servicios de Directorio: Samba y LDAP integrados.
c. Servicio de correo electrónico.
d. Servicio de acceso a internet.

1.5.1 Servidor DNS (Domain Name System)

Domain Name System (Sistema de Nombre de Dominio) es un sistema de nomenclatura jerárquica para computadoras, servicios o cualquier recurso conectado al internet o a una red privada. Este sistema asocia información variada con nombres de dominios asignados a cada uno de los participantes. Su función más importante, es traducir (resolver) nombres inteligibles para los humanos en identificadores binarios asociados con los equipos

conectados a la red, esto con el propósito de poder localizar y direccionar estos equipos mundialmente.

El DNS es una base de datos distribuida y jerárquica que almacena información asociada a nombres de dominio en redes como Internet. Aunque como base de datos el DNS es capaz de asociar diferentes tipos de información a cada nombre, los usos más comunes son la asignación de nombres de dominio a direcciones IP y la localización de los servidores de correo electrónico de cada dominio (vea la figura 1.1). [8]

La asignación de nombres a direcciones IP es ciertamente la función más conocida de los protocolos DNS. Por ejemplo, si la dirección IP del sitio HTTP de softlib.uclv.edu.cu es 10.12.1.103, la mayoría de la gente llega a este equipo especificando http://softlib.uclv.edu.cu y no la dirección IP. Además de ser fácil de recordar, el nombre es más fiable. La dirección numérica podría cambiar por muchas razones, sin que tenga que cambiar el nombre.

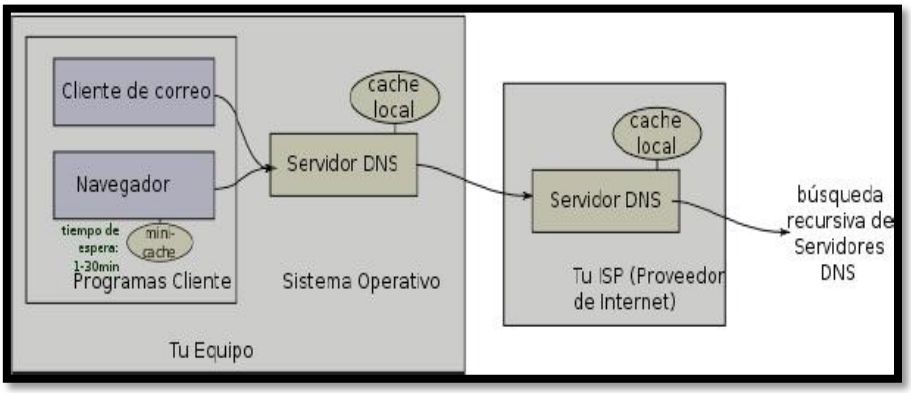


Figura 1.1: Servidor DNS

Roles de Servidor de DNS:

La tarea principal del servidor DNS es obtener el número IP de un ordenador o dominio a partir de su nombre. A continuación se muestra los tres roles fundamentales del servidor DNS:

- Resolución de nombres.
- Resolución inversa de direcciones.
- Resolución de servidores de correo.

Existen varios tipos de servidores de DNS como Bind, PowerDNS, djbdns y todos trabajan sobre el puerto 53 protocolo TCP/UDP.

Existen cuatro formas de implementar un servidor DNS:

- **Maestro:** Es el servidor responsable para determinada zona DNS y se encarga de la resolución de nombres dentro de esa zona donde es la autoridad.

- **Esclavo:** Este tipo de servidor sirve como espejo de un servidor DNS Maestro, recibe sus actualizaciones del maestro y se utiliza para aliviar la carga de trabajo de los servidores maestros.

- **Caché:** Este tipo de servidor se utiliza dentro de una red local, cuando se hace una consulta a un servidor DNS Caché y no contiene la resolución envía una petición a un DNS Maestro y la resolución quedará guardada en el caché del DNS local hasta que expire el tiempo de vida.

- **Reenvío:** Reenvía las peticiones a una lista específica de servidores DNS para la resolución de nombres.

Un servidor DNS puede ser de varios tipos configurados en el mismo servidor DNS.

1.5.2 Servidor DHCP

Un servidor *Dynamic Host Configuration Protocol* (DHCP) asigna dinámicamente las direcciones IP y otras configuraciones de una red determinada a otros ordenadores clientes que están conectados a la red. Esto simplifica la administración de la red y hace que la conexión de nuevos equipos a la red sea mucho más fácil. Todas las direcciones IP de todos los equipos se almacenan en una base de datos que reside en un servidor (vea la figura 1.2) [9].

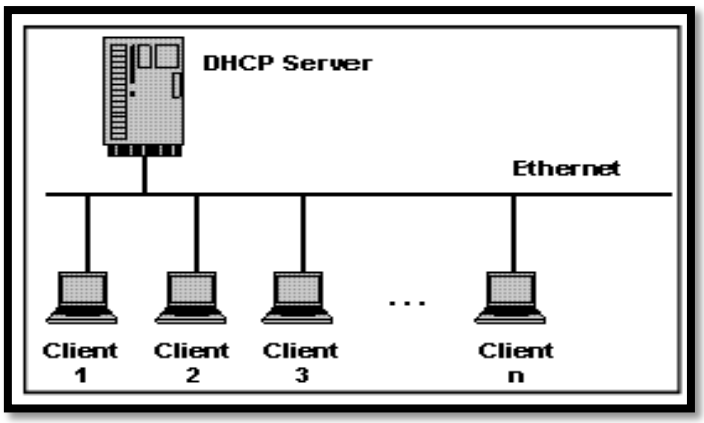


Figura 1.2: Red con servidor DHCP

Un servidor DHCP puede proporcionar los ajustes de configuración utilizando dos métodos:

- **Rango De Direcciones**:

  Este método se base en la definición de un grupo de las direcciones IP para los clientes DHCP (también llamado IP address pool) que suministran sus propiedades de configuración de forma dinámica según lo soliciten los ordenadores clientes. Cuando un cliente DHCP ya no está en la red durante un periodo determinado, la configuración vence y la dirección IP se libera del pool para su uso por otros clientes DHCP.

- **Dirección MAC**:

  Este método se basa en utilizar el protocolo DHCP para identificar la dirección de hardware única de cada tarjeta de red conectada a la red y luego asignarle una configuración fija, cada vez que el cliente realiza una petición al servidor DHCP recibe la misma asignación.

1.5.3 Servicio De Directorio

Un **servicio de directorio** es una aplicación o un conjunto de aplicaciones que almacena y organiza la información sobre los usuarios de una red de ordenadores, sobre recursos de red, y permite a los administradores gestionar el acceso de usuarios a los recursos sobre

dicha red. Además, los servicios de directorio actúan como una capa de abstracción entre los usuarios y los recursos compartidos.

El servicio de directorio proporciona la interfaz de acceso a los datos que se contienen en unos o más espacios de nombre de directorio. La interfaz del servicio de directorio es la encargada de gestionar la autenticación de los accesos al servicio de forma segura, actuando como autoridad central para el acceso a los recursos de sistema que manejan los datos del directorio. Como base de datos, un servicio del directorio está altamente optimizado para lecturas y proporciona alternativas avanzadas de búsqueda en los diferentes atributos que se puedan asociar a los objetos de un directorio. Los datos que se almacenan en el directorio son definidos por un esquema extensible y modificable. Los servicios de directorio utilizan un modelo distribuido para almacenar su información y esa información generalmente está replicada entre los servidores que forman el directorio.[10]

1.5.3.1 Desarrollos de servicios de directorio

La gran mayoría de implementaciones están basados en el estándar X.500, que posteriormente fue la base de LDAP, pero utilizando la pila TCP/IP en vez de usar el modelo OSI, adquiriendo especial relevancia en internet.

Existen numerosas formas de implementación de servicios de directorio de diferentes compañías. Algunos de estos ejemplos son[11]:

- NIS *Network Information Service* protocolo, nombrado originalmente como Páginas Amarillas, implementación de Sun Microsystems' en un servicio de directorio para redes de entorno UNIX. (Sun, a principios del 2000, se unió a iPlanet, alianza de Netscape y desarrolló la base de LDAP, servicio de directorio que formó parte de Sun ONE, la empresa que es ahora Sun Java Enterprise).

- eDirectory, desarrollado por Novell, es un servicio de directorio que soporta múltiples arquitecturas incluyendo Windows, NetWare, Linux, e incluyendo algunas distribuciones de Unix. Se ha utilizado durante tiempo para la administración de usuarios, gestión de configuraciones y gestión de software. eDirectory se ha desarrollado como componente central en una gama más amplia de productos para la gestión de identidad. Fue conocido previamente como servicios de directorio de Novell.

- Servidor de directorio de Red Hat: Red Hat lanzó un servicio de directorio, que adquirió de "*Netscape Security Solutions* de AOL", el cual funcionaba como producto comercial, bajo Red Hat Enterprise Linux denominado como servidor de directorio de Red Hat, como parte del núcleo de Fedora.

- CentOS Directory Server: CentOS Directory Server está basado en Red Hat Directory Server, posee similares características, y está disponible para instalar vía yum, sin necesidad de tener un contrato de por medio. La base del software está licenciada bajo GNU/GPL 2, y se incluye una excepción para ser integrado con software no libre, la cual proviene de RedHat.

- Active Directory (AD): El servicio del directorio de Microsoft, es el directorio que se incluye en las versiones de los sistemas operativos Windows Server 2000 y sus sucesores. AD es una implementación propietaria (creada por Microsoft) de los Servicios de Directorio, y proporciona una manera de compartir información entre recursos y usuarios de la red. Además de proporcionar una fuente centralizada para esa información, AD también funciona como autoridad de seguridad centralizada de Autenticación para la red.

  AD combina capacidades que tradicionalmente se hallaban en sistemas separados y especializados de directorio, como integración simplificada, gestión y seguridad de los recursos de la red. El paquete SAMBA puede configurarse para usar los servicios del AD desde un controlador de dominio de Windows.

- Open Directory: El servidor del Mac OS X de Apple ofrece un servicio del directorio llamado Open Directory que integra muchos protocolos estándares abiertos tales como LDAP y Kerberos así como soluciones propietarias de directorio como Active Directory y eDirectory.

- Servidor de directorio de Apache: Apache Software Fundation ofrece un servicio del directorio llamado ApacheDS.

- Directorio de Internet de Oracle: (OID) es el servicio del directorio de *Oracle Corporation*, que es compatible con la versión tres de LDAP.

- Directorio CA: El directorio CA contiene un motor de caché previo que puede indexar todos los atributos que se usan en los filtros de búsqueda de LDAP, y poner en caché aquellos atributos devueltos en tales búsquedas. Teniendo bastante memoria, el directorio CA es el directorio más rápido del planeta.

- OpenDS: La nueva generación de servicio de directorio abierto ofrecido por Sun Microsystems.

1.5.4 Servidor de Correo Electrónico

El servicio de correo electrónico consta de tres agentes o componentes bien diferenciadas (vea la figura 1.3). Estos son:

1.5.4.1 MUA (Mail User Agent)

MUA es un programa que permite leer y escribir correos. Suelen tener muchas funcionalidades que superan la estricta lectura y composición de mensajes, como el mantenimiento de libretas de direcciones, gestión de anexos (*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. Nombres habituales de MUAs son: mail, elm, pine, kmail (entorno KDE), Netscape Messenger, Microsoft Outlook Express, Qualcomm Eudora (en Windows), PegasusMail (en Windows) etc.

1.5.4.2 MTA (Mail Transport Agent)

Es un programa encargado de recoger mensajes y enviarlos, comunicando para ello con otros MTA según sea preciso. Lo normal es que funcione como servicio (es decir, de modo continuo, esperando peticiones de los MUAs o de otros MTAs y atendiéndolas). En Unix/Linux se implementan como uno o más demonios. El MTA más famoso y utilizado es sendmail; otros MTAs son Postfix, QMail etc. Además, productos de groupware como Microsoft Exchange, Lotus Domino Server, Novell Groupwise o Netscape Messaging Server incluyen MTAs.

1.5.4.3 MDA (Mail Delivery Agent)

Se encarga de copiar los mensajes desde el servidor de correo hasta el buzón de usuario. MDA es el encargada de realizar la entrega de correos a los MUA. Algunos de los más usados son: Qpopper, Courier, Cyrus, Maildrop (Unix) y Dovecot.

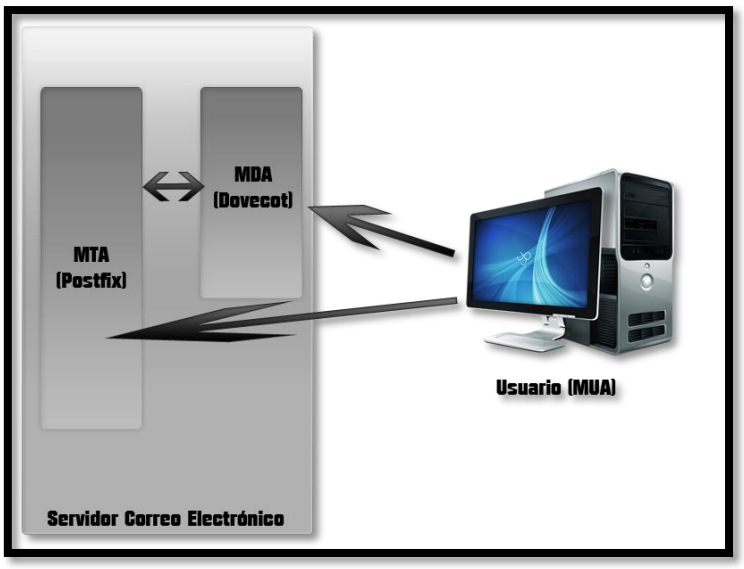


Figura 1.3: Los tres componentes del servicio de correo

1.5.4.4 Protocolos de Correo Electrónico

1.5.4.4.1 Protocolo SMTP

Simple Mail Transfer Protocol (SMTP) en español, Protocolo Simple de Transferencia de Correo, es un protocolo de la capa de aplicación. Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos.

SMTP se basa en el modelo cliente-servidor, donde un cliente envía un mensaje a uno o varios receptores. La comunicación entre el cliente y el servidor consiste enteramente en líneas de texto compuestas por caracteres ASCII. El tamaño máximo permitido para estas líneas es de 1000 caracteres. Las respuestas del servidor constan de un código numérico de tres dígitos, seguido de un texto explicativo. El número va dirigido a un procesado automático de la respuesta por autómata, mientras que el texto permite que un humano interprete la respuesta. En el protocolo SMTP todas las órdenes, réplicas o datos son líneas de texto, delimitadas por el carácter <CRLF>. Todas las réplicas tienen un código numérico al comienzo de la línea. En el conjunto de protocolos TCP/IP, el SMTP va por encima del TCP, usando normalmente el puerto 25 en el servidor para establecer la conexión.

1.5.4.4.2 Protocolo POP

Post Office Transport Protocol (POP), se utiliza para obtener/descargar los mensajes guardados en el servidor al usuario. POP3 está diseñado para recibir correo, no para enviarlo; le permite a los usuarios con conexiones intermitentes o muy lentas (tales como las conexiones por módem), descargar su correo electrónico mientras tienen conexión y revisarlo posteriormente incluso estando desconectados. Cabe mencionar que la mayoría de los clientes de correo incluyen la opción de dejar los mensajes en el servidor, de manera tal que, un cliente que utilice POP3 se conecta, obtiene todos los mensajes, los almacena en la computadora del usuario como mensajes nuevos, los elimina del servidor y finalmente se desconecta. En contraste, el protocolo IMAP permite los modos de operación conectado y desconectado.

Los clientes de correo electrónico que utilizan IMAP dejan por lo general los mensajes en el servidor hasta que el usuario los elimina directamente. Esto y otros factores hacen que la operación de IMAP permita a múltiples clientes acceder al mismo buzón de correo. La mayoría de los clientes de correo electrónico soportan POP3 ó IMAP; sin embargo, solo unos cuantos proveedores de internet ofrecen IMAP como valor agregado de sus servicios.

1.5.4.4.3 Protocolo IMAP

Internet Message Access Protocol, o su acrónimo IMAP, es un protocolo de red de acceso a mensajes electrónicos almacenados en un servidor. Mediante IMAP se puede tener acceso al correo electrónico desde cualquier equipo que tenga una conexión a Internet. IMAP tiene varias ventajas sobre POP, que es el otro protocolo empleado para obtener correo desde un servidor. Por ejemplo, es posible especificar en IMAP carpetas del lado servidor. Por otro lado, es más complejo que POP ya que permite visualizar los mensajes de manera remota y no descargando los mensajes como lo hace POP. IMAP, tiene la misma finalidad que POP aunque funciona de forma diferente; de este protocolo se pueden observar algunas ventajas, como tiempos de respuesta más rápidos, acceso remoto a los mensajes,

accesos simultáneos a múltiples clientes, vigilancia en el estado del mensaje, agilidad en las búsquedas, entre otras ventajas sobre el protocolo POP.

1.5.4.5 Funcionamiento de los Servidores de Correo Electrónico

Un servidor de correo electrónico debe constar en realidad de dos servidores el SMTP encargado de enviar y recibir mensajes, y un servidor POP/IMAP, que será el que permite a los usuarios obtener sus mensajes; para esto los usuarios se sirven de clientes, es decir, programas que implementan un protocolo POP/IMAP (Vea la figura 1.4), los que en algunas ocasiones se ejecutan en la máquina del usuario (como son los casos de, Evolution, Outlook). Sin embargo existe otra posibilidad: que el cliente de correo no se ejecute en la máquina del usuario; es el caso de los clientes vía web. [12]

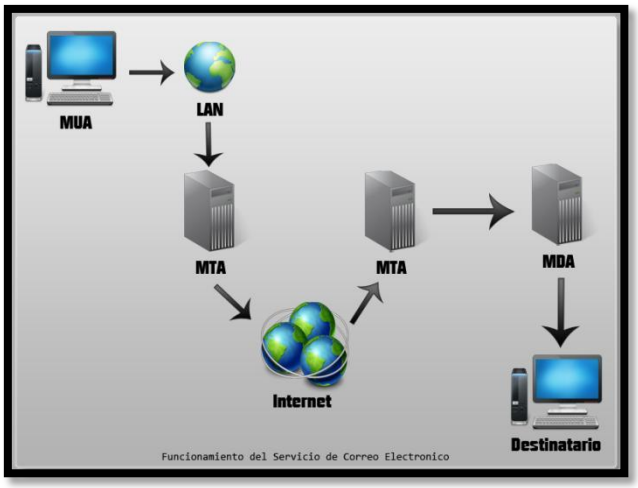


Figura 1.4: Proceso en el envió de un correo Electrónico

1.5.4.6 Aspectos a tener en cuenta al instalar un servidor de correo electrónico:

- ✓ Volumen de correos que van a gestionarse.
- ✓ Configurar otros servicios como:
  - DNS: Es esencial definir los registros MX para definir cuál es el servidor SMTP.
  - Definir reglas en el firewall (permitir el tráfico en MDA/MTA).
- ✓ Crear buzones de correo siguiendo determinadas políticas (nombre de la cuenta).
- ✓ Disponer de una máquina de *backup.*

1.5.5 Servidor de acceso a internet (Proxy)

El término en ingles «Proxy» tiene un significado muy general y al mismo tiempo ambiguo, aunque invariablemente se considera un sinónimo del concepto de «Intermediario». Se suele traducir, en el sentido estricto, como delegado o apoderado.

Un **Servidor Intermediario** se define como un dispositivo o software que ofrece un servicio de red que consiste en permitir a los clientes realizar conexiones de red indirectas hacia otros servicios de red (vea la figura 1.5). [13, 14] Durante el proceso ocurre lo siguiente:

- Cliente se conecta a un Servidor Proxy.
- Cliente solicita una conexión, archivo u otro recurso, disponible en otro servidor.
- Servidor Intermediario proporciona el recurso ya sea conectándose hacia el servidor especificado o sirviendo éste desde un caché.
    - En algunos casos el Servidor Intermediario puede alterar la solicitud del cliente o bien la respuesta del servidor para diversos propósitos.

Los Servidores Proxy generalmente trabajan simultáneamente como muro cortafuegos, operando en el Nivel de Red actuando como filtro de paquetes, como en el caso de iptables, o bien operando en el Nivel de Aplicación, controlando diversos servicios, como es el caso de TCP Wrapper. Dependiendo del contexto, el muro cortafuegos también se conoce como BPD o Border Protection Device o simplemente filtro de paquetes.

Una aplicación común de los Servidores Proxy es funcionar como caché de contenido de Red (principalmente HTTP), proporcionando en la proximidad de los clientes, un caché de páginas y archivos disponibles a través de la Red en servidores HTTP remotos, permitiendo a los clientes de la red local acceder hacia éstos de forma más rápida y confiable.

Cuando se recibe una petición para un recurso de Red especificado en un **URL** (*Uniform Resource Locator*), el Servidor Intermediario busca el resultado del URL dentro del caché. Si éste es encontrado, el Servidor Intermediario responde al cliente proporcionado inmediatamente el contenido solicitado. Si el contenido solicitado no estuviera disponible

en el caché, el Servidor Intermediario lo traerá desde un servidor remoto, entregándolo al cliente que lo solicitó y guardando una copia en el caché. El contenido en el caché es eliminado luego a través de un algoritmo de expiración de acuerdo a la antigüedad, tamaño e historial de respuestas a solicitudes (hits) (ejemplos: **LRU**, **LFUDA** y **GDSF**).

Los Servidores Proxy para contenido de Red (Web Proxies) también pueden actuar como filtros del contenido servido, aplicando políticas de censura de acuerdo a criterios arbitrarios.

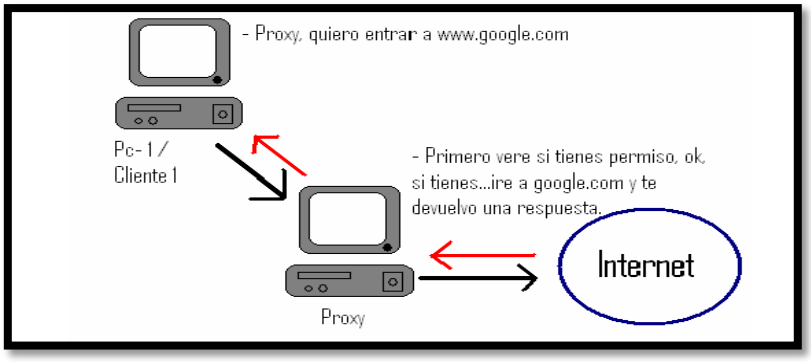


Figura 1.5: Como funciona el proxy

1.5.5.1 Ventajas de un servidor proxy: Un Proxy hace posible:

- **Control**: sólo el intermediario hace el trabajo real, por tanto se pueden limitar y restringir los derechos de los usuarios, y dar permisos sólo al proxy.
- **Ahorro**. Por tanto, sólo *uno* de los usuarios (el proxy) ha de estar equipado para hacer el trabajo real.
- **Velocidad**. Si varios clientes van a pedir el mismo recurso, el proxy puede hacer caché: guardar la respuesta de una petición para darla directamente cuando otro usuario la pida. Así no tiene que volver a contactar con el destino, y acaba más rápido.
- **Filtrado**. El proxy puede negarse a responder algunas peticiones si detecta que están prohibidas.

- **Modificación**. Como intermediario que es, un proxy puede falsificar información, o modificarla siguiendo un algoritmo.
- **Anonimato**. Si todos los usuarios se identifican como uno sólo, es difícil que el recurso accedido pueda diferenciarlos.

1.6 Conclusiones parciales del capítulo

En este capítulo hemos presentado algunos servicios de redes que consideramos básicos, necesarios y minimales. Básicos porque son la base de muchos otros servicios. Necesarios porque garantizan la conectividad y consistencia de la red y minimales porque son el conjunto mínimo de servicios que garantiza la conectividad dentro de la LAN y el acceso a correo electrónico e internet como forma imprescindibles de comunicación y colaboración para la empresa moderna. Se hace un resumen de cada servicio y sus características más relevantes.

CAPÍTULO II: IMPLEMENTACIÓN DE LOS SERVICIOS EN MÁQUINAS VIRTUALES

2.1 Introducción

Muchos países, Cuba entre ellos, han adoptado el uso de las distribuciones Linux, específicamente Debian y Ubuntu, debido a su estabilidad y la disponibilidad de sus repositorios, para facilitar la migración hacia software libre. Por este motivo, en este trabajo se ha seleccionado como sistema operativo a la versión de Linux, Ubuntu Server 10.04 LTS.

El trabajo en la red con el sistema Ubuntu junto con clientes Windows implica la oferta e integración de los servicios comunes a los entornos Windows. Estos servicios ayudan en la compartición de datos e información entre los equipos y usuarios implicados en la red, y pueden clasificarse en tres grandes categorías de funcionalidades:

- Compartir impresoras y archivos.
- Servicios de Directorio.
- Autentificación y acceso.

Afortunadamente, un sistema Ubuntu puede proporcionar tales facilidades a clientes Windows y compartir recursos de red con ellos. Una de las principales piezas de software que incluye su sistema Ubuntu para trabajar con redes Windows es el paquete de herramientas y aplicaciones de servidor Samba (SMB).

2.2 Ubuntu Server 10.04 LTS

Ubuntu es un sistema operativo mantenido por Canonical y la comunidad de desarrolladores. Utiliza un núcleo Linux, y su origen está basado en Debian. Ubuntu está orientado al usuario novel y promedio, con un fuerte enfoque en la facilidad de uso y mejorar la experiencia de usuario. Está compuesto de múltiples software normalmente distribuidos bajo una licencia libre o de código abierto. Las estadísticas web sugieren que el porcentaje de mercado de Ubuntu dentro de "distribuciones Linux" es de aproximadamente

49%, y con una tendencia a subir como servidor web. Y un importante incremento activo de 20 millones de usuarios hacia fines de 2011. [7]

Para realizar este trabajo se usa como sistema operativo la versión de Linux Ubuntu 10.04 server LTS. Cada dos años el proyecto Ubuntu libera una versión con soporte técnico extendido a la que se añade la terminación LTS. Los lanzamientos LTS contarán con actualizaciones de seguridad de paquetes de software por un periodo de tiempo extendido. En versiones anteriores, era de tres años en entorno de escritorio y cinco años en servidor por parte de Canonical, a diferencia de los lanzamientos de cada seis meses de Ubuntu que sólo cuentan con dieciocho meses de soporte.

La primera LTS fue la versión 6.06 de la cual se liberó una remasterización (la 6.06.1) para la edición de escritorio y dos remasterizaciones (6.06.1 y 6.06.2) para la edición servidor, ambas incluían actualizaciones de seguridad y corrección de errores. La segunda LTS fue la versión 8.04, de la cual ya va por la cuarta y última revisión de mantenimiento (la 8.04.4). El Ubuntu 10-04 LTS es la tercera LTS y fue liberada en abril de 2010, y cuya última versión de mantenimiento es la 10.04.4.

Ya que uno de los servicios básicos más importantes es el servicio de directorio, es necesario contar al menos con un servidor que sirva como controlador principal de dominio (PDC). Es recomendable tener al menos otro servidor que sirva como respaldo (backup) del PDC,a ese se le denomina controlador secundario del dominio (BDC).

PDC (*Primary Domain Controller*)

Para establecer un dominio, se precisa al menos de un sistema como controlador principal que es el encargado de mantener la base de datos de cuentas de usuario, recursos del dominio, así como de las listas de control de acceso. Es el servidor más importante al tener la copia del directorio donde pueden realizarse escrituras.

BDC (*Backup Domain Controller*)

Por cuestiones de seguridad y rendimiento es aconsejable tener al menos otro servidor que sirva de *backup* del PDC. Este servidor llamado *Backup Domain Controller* mantiene una copia del árbol de directorio del dominio de solo lectura. El concepto de Controlador secundario o *backup* fue eliminado en el ambiente Windows a partir del lanzamiento del

SO Windows Server 2003, pero aún se mantiene con las actuales implementaciones de Samba en el ambiente Linux.

2.3 Herramientas utilizadas para la virtualización de los servidores

En el trabajo se utilizó *ORACLE VirtualBox versión 4.1.14* (Vea figuras 2.1 y 2.2). Los servidores fueron virtualizados considerando 512 MB de memoria RAM y discos de ocho gigabytes lo que fue suficiente para la implementación con fines de probar y mostrar los servicios. Para un ambiente de producción se recomienda como mínimo que los servidores sean maquinas con al menos un gigabyte de memoria y el tamaño de los discos debe ser lo mayor posible sobre todo en el BDC teniendo en cuenta que el correo y el Squid necesitan gran cantidad de estos recursos.

En la medida de sus posibilidades se sugiere que las empresas dispongan de los computadores con mejores características de hardware que tengan para estos roles.

Figura 2.1: Virtual Box versión 4.1.14

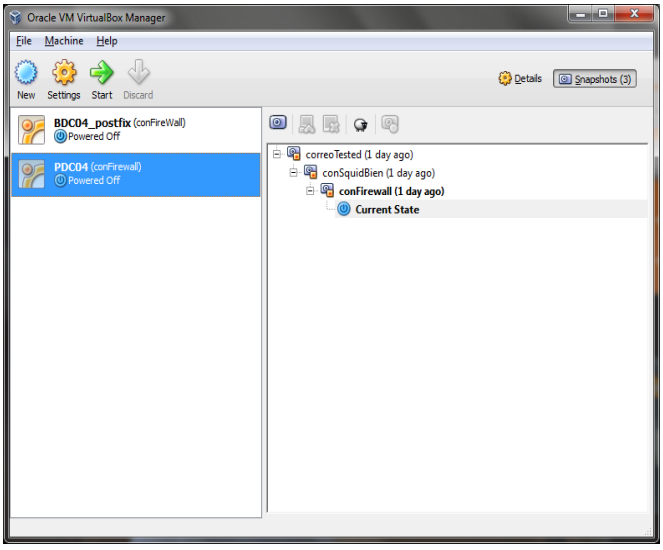


Figura 2.2: Servidores instalados en el Virtual Box

2.4 Distribución de servicios básicos sobre servidores PDC y BDC

Como servicios básicos de cualquier red empresarial están los servicios de correo y de acceso a internet, la distribución de estos servicios en uno o más servidores depende de la situación real de la empresa donde se desee su instalación, dependiendo mucho de los recursos materiales y económicos con lo que cuente la empresa. Es decir, depende del equipamiento en existencia y/o el respaldo monetario que se tenga.

Aunque se recomienda tener estos servicios en computadores separados no siempre esto es posible. En este trabajo asumimos que solo hay disponibles dos servidores por lo que utilizaremos el BDC para su instalación, de esa manera los servicios básicos de directorio e infraestructura se mantienen en el servidor principal. También al tener separados los servicios del PDC se muestra cómo es posible la vinculación de ellos con el PDC.

2.4.1 Orden de instalación de los servicios

Es muy importante que el administrador de cualquier empresa siga el orden lógico que se muestra a continuación para la instalación de los servicios en servidores PDC y BDC. Primeramente se instala el sistema operativo deseado para el servidor, en este caso **Ubuntu Server 10.04 LTS**, después hay que configurar la red para reflejar una dirección IP estática en el archivo /etc/network/interfaces. Posteriormente se comprueba la conectividad con el repositorio y se actualiza el sistema operativo.

Luego, es necesario instalar el DNS pues otros servicios dependen de su correcto funcionamiento de resolución de nombres.

Una vez que se tiene el DNS bien configurado, se debe instalar y configurar el servicio de DHCP para poder asignar automáticamente la configuración de red a los ordenadores que realicen la petición correspondiente. Suponiendo que tanto su red como su servidor DHCP están correctamente configurados, la red local no necesitaría más configuración adicional para poder funcionar. El servidor DHCP proporcionará a quienes lo soliciten, la pasarela (*gateway*) predeterminada, la dirección IP que usará el dispositivo de red, la máscara de red, los servidores DNS usados en la red, entre otros.

Como servicio de directorio se instalará OpenLDAP integrado con Samba como controlador del directorio. Es muy importante al instalar OpenLDAP tener otro servidor con OpenLDAP que sirva de *backup* logrando la actualización través de mecanismos de replicación que pueden o no estar encriptados (TLS/SSL). El proceso de instalación de Samba permitirá crear el dominio.

A partir de este momento comienzan a instalarse las diferentes aplicaciones como correo, proxy, etc. Se recomienda instalar y configurar el cortafuego en este momento antes de realizar las instalaciones por ejemplo del correo que es una aplicación que es accedida desde el exterior. Se ha supuesto que cualquier servidor que sea instalado lo hace a partir de un repositorio local y después de configurado se le permite el acceso a internet.

La configuración básica para tener acceso a los repositorios pasa por configurar la dirección IP correctamente, para eso Linux tiene varios ficheros predefinidos como:

- /etc/**hostname** :Indica el nombre del computador
- /etc/**hosts:** Es usado para la resolución de los nombres de los ordenadores en la red local. En el mismo se indica el nombre dado al ordenador durante el proceso de instalación.
- /etc/**host.conf**: Indica al Sistema Operativo el método empleado para la resolución de nombres de ordenadores o nombres de dominio.
- /etc/**host.conf** : Muestra el orden en que se procesará cada consulta por parte del servidor:**order hosts,bind** o **order bind,hosts**
- /etc/**resolv.conf:** En este archivo se indica el servidor DNS empleado por el ordenador para la resolución de nombres.
- /etc/network/**interfaces:** En este archivo se indica el IP empleado por el ordenador (uno por cada interfaz de red).

2.5 Servidor DNS (Bind9)

2.5.1 BIND (*Berkeley Internet Name Domain*)

Anteriormente conocido como *Berkeley Internet Name Daemon* es el servidor de DNS más comúnmente usado en Internet, especialmente en sistemas Unix, en los cuales es un estándar. Es patrocinado por la *Internet Systems Consortium*. BIND fue creado originalmente por cuatro estudiantes de grado en la Universidad de California, Berkeley y liberado por primera vez en el 4.3BSD. Paul Vixie comenzó a mantenerlo en 1988 mientras trabajaba para la DEC.

Una nueva versión de BIND **(BIND9)** fue escrita desde cero en parte para superar las dificultades arquitectónicas presentes anteriormente: entre ellas auditar el código en las primeras versiones de BIND, y también para incorporar DNSSEC (DNS Security Extensiones). BIND 9 incluye entre otras características importantes: TSIG, notificación DNS, nsupdate, IPv6, rndc flush, vistas, procesamiento en paralelo, y una arquitectura mejorada en cuanto a portabilidad. Es comúnmente usado en sistemas GNU/Linux.

2.5.2 Paquetes de Instalación de Bind9

BIND9 está disponible en el repositorio principal. Para instalar el servidor sólo tiene que instalar el paquete bind9. Un paquete muy útil para probar y solucionar los problemas de DNS es el paquete de dnsutils. Asimismo, la documentación BIND9 se pueden encontrar en el paquete bind9-doc. Se instalan ejecutando la línea de comando: apt-get install bind9 dnsutils bind9-doc

2.5.3 Archivos principales de configuración

Los archivos de configuración del DNS están guardados en el directorio /etc/bind. Los archivos que deben ser configurados dentro /etc/bind son:

- named.conf
- named.conf.options

- named.conf.local

  Vea Ubuntu Server Guide, año 2012, capítulo 8 "*Domain Name Service*" para ver como se instalar y configurar el servidor DNS.

2.5.4 Comandos para comprobar el correcto funcionamiento del DNS

- nslookup nombre_de_dominio
- dig nombre_de_dominio
- dig –x IP_servidordns
- ping
- host –t a nombre_del_servidor
- host –t cname alias_del_servidor
- named-checkzone

2.5.5 Logs del servicio bind

- /var/log/syslog
- /var/log/messages

2.6 Servidor DHCP

DHCP asigna automáticamente la configuración de red a los ordenadores que realicen la petición correspondiente.

2.6.1 Paquetes de Instalación de DHCP

El paquete que se instala para este servicio es dhcp3-server que se instala ejecutando la línea de comando: apt-get install dhcp3-server.

2.6.2 Archivo de configuración del servicio

El archivo principal de configuración es **/etc/dhcpd.conf**.

2.6.3 Comando para comprobar el correcto funcionamiento del DHCP

- cat /var/lib/dhcp3/dhcpd.leases

2.6.4 Logs del servicio DHCP

- /var/log/dhcp3

2.7 Servidor OpenLDAP

2.7.1 Historia de OpenLDAP

El proyecto OpenLDAP se inició en 1998 por Kurt Zeilenga. El proyecto comenzó como un clon de la implementación LDAP de la Universidad de Michigan, entidad donde se desarrolló originalmente el protocolo LDAP y que también actualmente trabaja en la evolución del mismo.

En abril de 2006, el proyecto OpenLDAP incorpora tres miembros principales: Howard Chu (Arquitecto jefe), Pierangelo Masarati, y Kurt Zeilenga. Hay otros importantes y activos contribuyentes incluyendo Luke Howard, Hallvard Furuseth, Quanah Gibson-Mount, y Gavin Henry.

Históricamente la arquitectura del servidor OpenLDAP (slapd, Standalone LDAP Daemon) fue dividida entre una sección frontal que maneja las conexiones de redes y el procesamiento del protocolo, y una base de datos dorsal o de segundo plano (backend) que trata únicamente con el almacenamiento de datos. La arquitectura es modular y una variedad de backends está disponible para interactuar con otras tecnologías, no sólo bases de datos tradicionales.

Nota: En versiones antiguas (1.x), los términos "backend" y "database (base de datos)" podían intercambiarse. Para ser precisos, un "backend" es una clase de interfaz de almacenamiento, y una base de datos es una instancia de un backend. El servidor slapd puede utilizar arbitrariamente varios backends de una sola vez, y puede tener

arbitrariamente muchas instancias de cada backend (por ejemplo varias bases de datos) activas por vez.

2.7.2 Funcionamiento de OpenLDAP

La aplicación del servidor LDAP proporciona funcionalidad de servicio de directorio a ordenadores Windows de una forma muy similar a los servicios de Microsoft Active Directory. Tales servicios incluyen gestionar las identidades y las relaciones entre los ordenadores, usuarios y grupos de ordenadores o usuarios que participan en la red, y proporcionan una forma consistente de describir, localizar y gestionar esos recursos. La implementación de libre distribución disponible se llama OpenLDAP. Los daemonss de servidor responsables de gestionar las peticiones de directorio OpenLDAP, y la propagación de datos de directorio entre un servidor LDAP y otro Linux, son slapd y slurpd. LDAP puede usarse junto con SAMBA para proporcionar servicios de Archivo, Impresión y Directorio prácticamente de la misma forma que un Controlador de Dominio de Windows, si se compila SAMBA con soporte LDAP. Con OpenLDAP uno se puede lograr las mismas funcionalidades que se ofrecen con Microsoft Active Directory pues existen las herramientas.

El servicio de directorio OpenLDAP 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 repuesta 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 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 directorio universal como LDAP.

2.7.3 Ventajas en el uso de OpenLDAP

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

- Es muy rápido en la lectura de registros.
- Permite replicar el servidor de forma muy sencilla y económica.
- Muchas aplicaciones tienen interfaces de conexión a LDAP y se pueden integrar fácilmente.
- Dispone de un modelo de nombres globales que asegura que todas las entradas son únicas.
- Usa un sistema jerárquico de almacenamiento de información.
- Permite múltiples directorios independientes.
- Funciona sobre TCP/IP y SSL.
- La mayoría de las aplicaciones disponen de soporte para LDAP.
- La mayoría de servidores LDAP son fáciles de instalar, mantener y optimizar.
- Manejo centralizado de cuentas y recursos.
- Compatible con X.500 y por lo tanto con los directorios de Microsoft y Novell.
- Base de datos jerárquica basada en *Directory Components* (DC) muy similar a la del DNS. Necesita esquemas.

2.7.4 Los principales componentes del servidor LDAP

- **slapd** es el servidor (demonio) principal de LDAP. Se suministra con tres diferentes bases de datos de *backend* (dorsal, o base de datos de segundo plano). Actualmente hay 18 diferentes *backends* que proporcionados por la distribución de Open LDAP. Los *backends* estándar están organizados de manera imprecisa en tres categorías:
  i. Backends de almacenamiento de datos (*Data Storage backends*) - éstos realmente almacenan información. Algunos ejemplos son:

a. back-bdb
b. back-hdb
c. back-ldif

ii. Proxy backends - actúan como puertas de enlace a otros sistemas de almacenamiento de datos. Algunos ejemplos son:

a. back-ldap
b. back-passwd
c. back-sql

iii. Backends dinámicos - éstos generan datos sobre la marcha. Algunos ejemplos son:

a. back-config
b. back-monitor
c. back-shell

- **Bibliotecas** que implementan el protocolo LDAP.
- **Programas cliente:** ldapsearch, ldapadd, ldapdelete, entre otros.

Vea Anexo I Componentes principal de servidor OpenLDAP

2.7.5 Paquetes de Instalación para el servidor LDAP

Los paquetes de instalación para el servicio OpenLDAP son:

- slapd
- ldap-utils

2.7.6 Archivos de configuración y esquemas

- Generalmente encontrados en /etc/ldap ó /etc/openldap.
- El archivo de configuración primario es slapd.conf, en este archivo se establecen los defaults para todo el sistema (configuración global).
- Esquemas almacenados en archivos de texto separados, usualmente encontrados en /etc/ldap/schema ó /etc/openldap/schema.

A continuación se muestran los esquemas que se utilizarán en el directorio ldap:

- ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/**cosine.ldif**
- ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/**nis.ldif**
- ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/**inetorgperson.ldif**

Vea Ubuntu Server Guide, año 2012, capítulo 7 epígrafe 1 "*OpenLDAP Server*" para como instalar el servidor LDAP.

2.7.7 Comandos para comprobar el correcto funcionamiento de OpenLDAP

- slaptest
- ldapsearch

Vea Anexo II Pruebas Preliminares de servidor LDAP

2.7.8 Logs del OpenLDAP

- /var/log/syslog

2.7.9 Bases de datos

La base de datos del OpenLDAP se puede encontrar en: /var/lib/ldap

2.8 Controlador de dominio (Samba)

2.8.1 Historia de Samba

Samba es una creación de Andrew Tridgell. De acuerdo con información brindada en la web oficial del software**,** Tridgell necesitaba montar un espacio en disco en su computadora para un servidor Unix. En esa computadora corría el sistema operativo DOS e, inicialmente, utilizaba el sistema de archivos NFS (*Network File System*) para el acceso. Sin embargo, la aplicación necesitaba soporte para el protocolo NetBIOS (no soportado por el NFS). La solución encontrada por Tridgell fue la siguiente: escribió un sniffer (pequeño

programa para captura de tráfico de datos en red) que permitiera analizar el tráfico de datos generado por el protocolo NetBIOS, hizo ingeniería reversa en el protocolo SMB (Server Message Block) y lo implementó en el Unix. Eso hizo que el servidor Unix apareciera como un servidor de archivos Windows en su PC con DOS.

Ese código después fue puesto a disposición públicamente por Tridgell en 1992**.** Tiempos después el proyecto fue dejado hasta que un día determinado, Tridgell decidió conectar la computadora de su esposa a su ordenador con Linux, pero no encontró ningún medio mejor que su código para hacer eso y entonces lo utilizó. A través de contactos hechos por correo electrónico, Tridgell descubrió que la documentación de los protocolos SMB y NetBIOS se habían actualizado y así volvió a dedicarse al proyecto. Posteriormente, Andrew Tridgell tuvo la idea de buscar en un diccionario una palabra que tuviera las letras s, m y b (de SMB) y encontró el término **"samba"**. A partir de ahí el proyecto Samba creció y hoy Andrew Tridgell cuenta con un excelente equipo de programadores y miles de usuarios de su solución por todo el mundo.

2.8.2 Definición de Samba

Samba constituye un conjunto (suite) de aplicaciones basadas en el protocolo SMB (Server Message Block).Es empleado por varios Sistemas Operativos entre ellos Windows y OS/2, para dar soporte a las aplicaciones cliente-servidor sobre las redes de comunicaciones. Basándose en Samba los servidores de la familia Unix/Linux pueden desarrollar diferentes roles en la red tal como lo hacen los productos de Microsoft Windows.

Samba es una implementación libre del protocolo de archivos compartidos de Microsoft Windows (antiguamente llamado **SMB**, renombrado recientemente a **CIFS**) para sistemas de tipo UNIX. De esta forma, es posible que con **GNU/Linux**, **Mac OS X** o **UNIX** en general, se vean como servidores o actúen como clientes de redes de **Windows**. SAMBA también permite validar usuarios de cómo Controlador Principal de Dominio (PDC), como miembro de dominio e incluso un dominio de tipo Active Directory para redes basadas en Windows; aparte de ser capaz de servir colas de impresión, directorios compartidos y autentificar con su propio archivo de usuarios.

Entre los sistemas tipo UNIX en los que se puede ejecutar Samba, están las distribuciones GNU/Linux, Solaris y las diferentes variantes BSD entre las se puede encontrar el Mac OS X Server de Apple.

2.8.3 Funcionamiento de SAMBA

Samba permite que los usuarios de Linux puedan lograr un acceso a los sistemas de archivos de Windows de forma transparente. Aun es más comúnmente usado para dar a los usuarios de Windows acceso tranparente a los sistemas de Linux, Unix y sistemas parecidos a Unix. De manera general samba permite:

- Compartir uno o más sistemas de archivos.
- Compartir impresoras, instaladas tanto en el servidor como en los clientes.
- Autentificar clientes iniciando una sesión de trabajo contra un dominio tipo Windows.

2.8.4 Partes de Samba

Samba consta de dos programas:

a) smbd: Demonio que permite compartir archivos o impresoras sobre una red SMB (proporciona autentificación y autorización de acceso para clientes).

- Ofrecer acceso remoto a ficheros e impresoras (implementando el protocolo SMB).
- Autenticar y autorizar usuarios.
- Dos modos de compartición de recursos:
  - ✓ Basado en usuarios (Windows NT ó 2000): Acceso basado en usuario/password en un dominio.
  - ✓ Basado en recursos (Windows 3.11 ó 95): Acceso basado en recurso/password.

b) nmbd: Demonio que busca el IP a través del *Windows Internet Name Service* (WINS).El sistema Unix participa en los mecanismos de resolución de nombres propios de Windows:

- ✓ Anuncio en grupo de trabajo.
- ✓ Listado de ordenadores del grupo de trabajo.

- ✓ Anuncio de recursos compartidos.

2.8.5 Paquetes de Instalación para Samba

Los paquetes necesarios para su instalación son:

a) samba
b) samba-doc
c) smbldap-tools
d) smbclient
e) smbfs
f) libpam-smbpass

Se instalan ejecutando la línea de comando: apt-get install samba samba-doc smbldap-tools smbclient smbfs libpam-smbpass. Vea Ubuntu Server Guide, año 2012, capítulo 7 epígrafe 2 "Samba and LDAP" para cómo se instala, también vea Anexo III Algunos programas asociados a SAMBA.

2.8.6 Los modos de seguridad de Samba

Solamente hay dos tipos de modos de seguridad para Samba, *share-level* y *user-level*, que se conocen de forma colectiva como *niveles de seguridad*. Solamente se puede implementar la seguridad a nivel de recurso compartido de una forma, mientras que la seguridad a nivel de usuario se puede implementar en una de cuatro formas diferentes. Las diferentes formas de implementar un nivel de seguridad se llaman **modos de seguridad**.

2.8.6.1 Seguridad a nivel de usuario (user mode)

La seguridad a nivel de usuario es la configuración predeterminada para Samba. Aún si la directriz security = user no está listada en el archivo smb.conf, es utilizada por Samba. Si el servidor acepta la combinación de nombre de usuario/contraseña del cliente, el cliente puede montar múltiples recursos compartidos sin tener que especificar una contraseña para cada instancia. Samba también puede aceptar solicitudes de nombre de usuario/contraseña

basadas en la sesión. El cliente mantiene múltiples contextos de autenticación usando un único UID por cada inicio de sesión.

En smb.conf, la directiva security = user que configura la seguridad a nivel de los usuarios es:

```
[GLOBAL]
...
security = user
...
```

2.8.6.2 Seguridad a nivel de recurso compartido (share mode)

Con la seguridad a nivel de recurso compartido o servicio, el servidor acepta solamente una contraseña sin un nombre de usuario explícito desde el cliente. El servidor espera una contraseña para cada recurso compartido, independientemente del nombre de usuario. Han surgido informes recientes de clientes Microsoft Windows con problemas de compatibilidad con servidores de seguridad a nivel de recurso compartido. Los desarrolladores de Samba no recomiendan el uso de la seguridad a este nivel.

En smb.conf, la directiva security = share que configura la seguridad a nivel de directorio compartido es:

```
[GLOBAL]
...
security = share
...
```

2.8.6.3 Modo de seguridad de Active Directory (seguridad a nivel de usuario)

Si tiene un entorno Active Directory, es posible unirse al dominio como un miembro nativo de Active Directory. Aún si una política de seguridad limita el uso de protocolos de autentificación compatibles con NT, el servidor Samba se puede unir al ADS utilizando Kerberos. Samba en un modo de miembro de Active Directory puede aceptar *tickets* Kerberos.

En smb.conf, las directivas siguientes hacen a Samba un servidor miembro de Active Directory:

```
[GLOBAL]
...
security = ADS
realm = EXAMPLE.COM
password server = kerberos.example.com
...
```

2.8.6.4 Modo de seguridad de servidor (seguridad a nivel de usuario)

Se utilizó el modo de seguridad de servidor previamente cuando Samba no fue capaz de actuar como un servidor miembro de dominio.

Nota: Se recomienda que no utilice este modo puesto que existen numerosas desventajas desde el punto de vista de la seguridad.

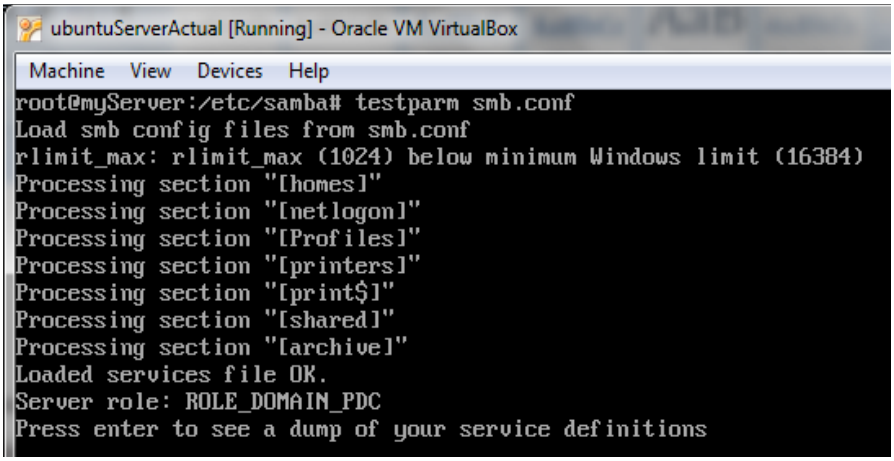
En el archivo **smb.conf**, las directrices siguientes permiten que Samba opere en modo de seguridad de servidor:

```
[GLOBAL]
...
encrypt passwords = Yes
security = server
password server = "NetBIOS_of_Domain_Controller"
...
```

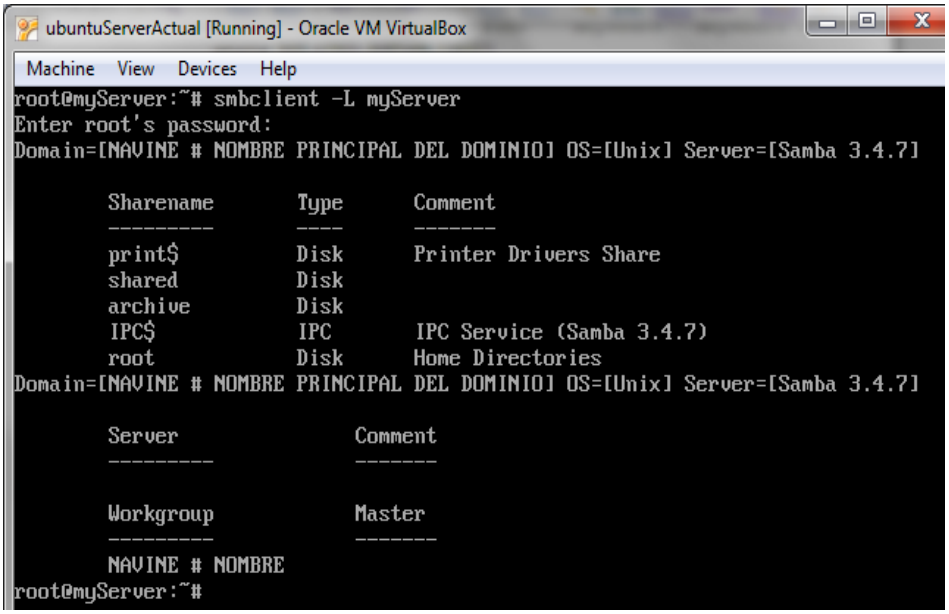
2.8.7 Comandos para comprobar el correcto funcionamiento de samba

A continuación se muestra una lista de pruebas que deben seguirse para validar que el servidor Samba está funcionado bien. Si se logran correr todas las pruebas sin errores significa que la instalación de Samba fue configurada con éxito.

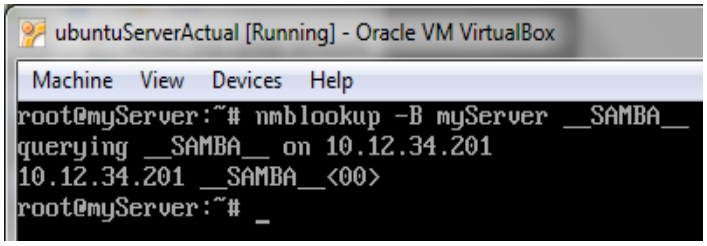
- Ejecuta el comando **testparm. Como resultado** debe mostrar los datos mostrados en la siguiente figura, si no los reporta y sale algún error entonces significa que el archivo smb.conf no está bien configurado



- Ejecuta el comando **ping nombreServidor** desde una maquina conectada a la misma red que el servidor y **ping nombreMaquinaCliente** desde el servidor, si da repuesta válidas entonces el software de su TCP/IP está bien instalado.
- Ejecute el comando **smbclient –L nombreServidor** desde la consola Linux. Debería dar lo siguiente si todo está bien.

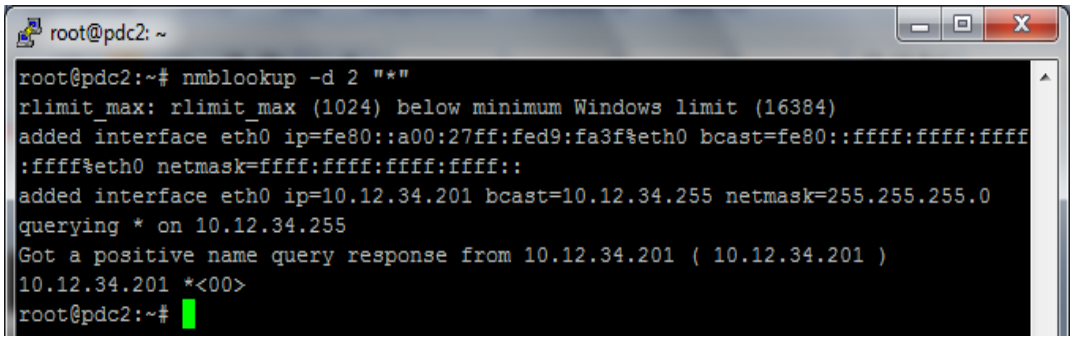


- Si ejecuta el comando nmblookup -B nombreServidor _ _SAMBA _ _. Debe devolver el IP del servidor. Si el nmbd no está bien instalado, hay que revisar el archivo **inetd.conf.** Si fue desde allí que se ejecutó, verifique que el demonio esté corriendo y escuchando sockets UDP en el puerto 137.



- Al ejecutar el comando **nmblookup -B nombreMaquinaCliente ´*´.** Debe devolver la IP de la máquina de cliente.

- Ejecucion del comando **nmblookup -d 2 ´*´.** Este prueba es similar a la prueba anterior sin embargo se realizó a través *broadcast* para la dirección por defecto de *broadcast*. Unos números de NetBIOS/TCP/IP hosts deberían responder a este comando, aun cuando Samba puede no capturar todo en el corto tiempo que tiene para escuchar. Se deben ver los mensajes "*got a possitive name query*" para varios hosts, porque si no, eso significa que el nmblookup no está recibiendo correctamente su dirección de *broadcast*. Si eso ocurre debe experimentar con las opciones de interfaces en el archivo smb.conf para manualmente configurar su dirección IP, broadcast y netmask.



- Ejecución del comando **smbclient //nombreServidor/tmp**. Se pide password, allí debe usar el password de usuario que está actualmente conectado. Si da el error "*invalid network name*", entonces el recurso tmp no está montado bien en smb.conf.

- En el PC del cliente, teclee el comando **net view \\nombreServidor.** Esto se hace desde la ventana *cmd prompt*. Debe devolver una lista de los recursos compartidos (shares) disponibles en el servidor. Si devolvió el mensaje "network name not found" o un mensaje semejante, entonces la resolución de nombres no está funcionando bien. Esto normalmente es resultado de un problema en el nmdb.

- El comando **nmblookup –M testgroup,** donde tesgroup es el nombre del workgroup a quien pertenece Samba y PC´s de Windows,debe devolver el IP del browser maestro para el workgroup.

2.9 TLS (Transport Layer Security)

2.9.1 Historia y desarrollo

Desarrollado por Netscape, SSL versión 3.0 se publicó en 1996, que más tarde sirvió como base para desarrollar TLS versión 1.0, un protocolo estándar IETF definido por primera vez en el RFC 2246. Visa, MasterCard, American Express y muchas de las principales instituciones financieras han aprobado SSL para el comercio sobre Internet.

SSL/TLS opera de una manera modular: sus autores lo diseñaron extensible, con soporte para compatibilidad hacia delante y hacia atrás, y negociación entre las partes (peer-to-peer).

2.9.2 El Protocolo SSL/TLS

SSL (*Secure Sockets Layer*) traducido al español significa Capa de Conexiones Seguras. Es un protocolo que hace uso de certificados digitales para establecer comunicaciones seguras

a través de Internet. Recientemente ha sido sustituido por TLS (*Transport Layer Security*) el cual está basado en SSL y son totalmente compatibles.

El protocolo TLS es una evolución del protocolo SSL, mediante el cual se establece una conexión segura por medio de un canal cifrado entre cliente y servidor.

Esto permite confiar información personal a sitios web, ya que los datos se ocultan a través de métodos criptográficos en sitios con buena seguridad.

Es utilizado ampliamente en bancos, tiendas en línea y cualquier tipo de servicio que requiera el envío de datos personales o contraseñas.

Normalmente el servidor es el único que es autenticado, garantizando así su identidad, pero el cliente se mantiene sin autenticar, ya que para la autenticación mutua se necesita una infraestructura de claves públicas (o PKI) para los clientes.

Estos protocolos permiten prevenir escuchas (eavesdropping), evitar la falsificación de la identidad del remitente y mantener la integridad del mensaje en una aplicación cliente-servidor.

2.9.3 Uso de TLS para asegurar las comunicaciones

TLS/SSL utilizan certificados X.509, los cuales son dados en un formato estándar que ha sido firmado digitalmente por una tercera entidad de confianza conocida como *Certificate Authority (*CA). Una firma digital válida significa que los datos que fueron firmados no han sido falsificados. Si los datos firmados fueran modificados, aunque sea un poco, entonces la firma no sería válida. Las partes independientes, como el cliente y el servidor, pueden validar las firmas porque ambos inician confiando en la CA.

En el certificado del servidor se encuentra la información sobre la propiedad del servidor, incluyendo su nombre en Internet. Por lo tanto se puede estar seguro de que se está conectando al servidor correcto porque el nombre del servidor al que se conectó concuerda exactamente con el nombre que aparece en el certificado, y ha confiado en el CA para validar esto antes de firmar. El certificado también incluye la clave pública del servidor, la

cual puede utilizarse para encriptar datos los que sólo el portador de la clave secreta puede desencriptar.

Las claves públicas y secretas forman la base de la criptografía de las *claves públicas* o *asimétricas*. Es asimétrica porque los datos encriptados por la clave pública pueden ser desencriptados sólo por la clave secreta, y los datos encriptados con la clave secreta pueden sólo ser desencriptados por la clave pública. Una clave se hace pública, y la otra clave se hace secreta. Dado el comportamiento asimétrico de las claves, cualquiera con la clave pública puede encriptar un mensaje, y el receptor del mensaje con la clave secreta puede desencriptarlo.

Después de que el cliente se conecta al servidor y recibe el certificado del servidor, el cliente puede validar que el nombre del servidor es correcto, lo cual evita un ataque *de "man in the middle"*. La clave pública puede utilizarse para ejecutar a través de un protocolo que termina con el cliente y el servidor acordando un secreto compartido que nadie observando la conversación puede determinar. Este secreto es entonces utilizado para codificar el resto de la conversación entre el cliente y el servidor, denominada criptografía simétrica porque la misma clave encripta y desencripta los datos. La división entre criptografía asimétrica y simétrica existe porque la última es una orden de magnitud más rápida. La criptografía de clave pública es utilizada para autentificar y proponer un secreto compartido, y luego la criptografía de clave simétrica toma el control.

2.9.4 Descripción del Protocolo TLS

El protocolo SSL/TSL se basa en tres fases básicas:

- **Negociación**: Los dos extremos de la comunicación (cliente y servidor) negocian que algoritmos criptográficos utilizarán para autenticarse y cifrar la información. Actualmente existen diferentes opciones.
- **Autenticación y Claves**: Los extremos se autentican mediante certificados digitales e intercambian las claves para el cifrado, según la negociación.
- **Transmisión Segura**: los extremos pueden iniciar el tráfico de información cifrada y auténtica.

2.9.5 Objetivos del Protocolo TLS

Los objetivos del protocolo son varios:

- **Seguridad criptográfica**. El protocolo se debe emplear para establecer una conexión segura entre dos partes.
- **Interoperabilidad.** Aplicaciones distintas deben poder intercambiar parámetros criptográficos sin necesidad de que ninguna de las dos conozca el código de la otra.
- **Extensibilidad.** El protocolo permite la incorporación de nuevos algoritmos criptográficos.
- **Eficiencia**. Los algoritmos criptográficos son costosos computacionalmente, por lo que el protocolo incluye un esquema de *caché de sesiones* para reducir el número de sesiones que deben inicializarse desde cero (usando criptografía de clave pública).

2.9.6 Aplicaciones del Protocolo TLS

El protocolo SSL/TLS tiene multitud de aplicaciones en uso actualmente. La mayoría de ellas son versiones seguras de programas que emplean protocolos que no lo son. Hay versiones seguras de servidores y clientes de protocolos como el http, nntp, ldap, imap, pop3, etc. El protocolo SSL/TLS se ejecuta en una capa entre los protocolos de aplicación como **HTTP, SMTP, NNTP** y sobre el protocolo de transporte **TCP**, que forma parte de la familia de protocolos **TCP/IP.** Aunque pueda proporcionar seguridad a cualquier protocolo que use conexiones de confianza (tal como TCP), se usa en la mayoría de los casos junto a HTTP para formar HTTPS.

- HTTPS es usado para asegurar páginas **World Wide Web** para aplicaciones de **comercio electrónico**, utilizando **certificados de clave pública** para verificar la identidad de los extremos. Visa, MasterCard, American Express y muchas de las principales instituciones financieras han aprobado SSL para el comercio sobre Internet.
- SSH utiliza SSL/TLS como mecanismo para crear un canal seguro de comunicación.

- Para asegurar la salida y entrada de correos electrónico. Vea Ubuntu Server Guide, año 2012, página 236 epígrafe 1.3 "*SMPT Authentication*", epígrafe 1.4 "*Configuring SASL*".
- Cifrado IMAPs/POP3s para el servidor Dovecot. Vea Ubuntu Guide, año 2012, página 246 epígrafe 3.3 "*Dovecot SSL Configuration*".
- Uso de SSL y TLS para sesión cifrado con servidor OpenLDAP. Para autentificarse con un servidor OpenLDAP, lo mejor es hacerlo usando una sesión cifrada. Esto se puede conseguirse usando TLS. Vea Ubuntu Guide, año 2012, página 105 epígrafe 1.8 "*TLS*" para como implementar esto.

- Para cifrar el tráfico de la replicación de directorio LDAP entre servidores PDC y BDC. Si tiene configurado **syncrepl** entre servidores, es prudente cifrar el tráfico de replicación usando SSL/TLS. Vea Ubuntu Guide, año 2012, página 107 epígrafe 1.9 "*Replication and TLS*" para como implementar esto.

2.9.7 Implementaciones del Protocolo TLS

Existen diferentes implementaciones, como por ejemplo:

- OpenSSL: Es la implementación de código abierto más utilizada. Es un proyecto desarrollado por la comunidad *Open Source* para libre descarga y está basado en SSL, que ayuda al sistema a implementar el SSL/TLS ofreciéndole un robusto paquete de herramientas de administración y librerías de criptografía que pueden ser usadas para OpenSSH y navegadores web (acceso seguro a HTTPS).
- GnuTLS: es una implementación de código abierto con licencia compatible con GPL. Fue escrito completamente desde cero y se considera equivalente a SSL v3.
- JSSE: es una implementación realizada en el Java incluida en el *Java Runtime Environment.*

2.10 UFW e IPTables en Servidores Linux

2.10.1 Firewall (Cortafuegos)

Un firewall es un dispositivo que filtra el tráfico entre redes. El firewall puede ser un dispositivo físico o un software sobre un sistema operativo. En general se ve como una caja con dos o más interfaces de red en la que se establecen ciertas reglas de filtrado con las que se decide si una conexión determinada puede establecerse o no.

La definición genérica, hoy en día es la siguiente [15]: un firewall es un hardware especifico con un sistema operativo o una IOS que filtra el tráfico TCP/UDP/ICMP/../IP decide si un paquete pasa, se modifica, se convierte o se descarta. Para que un firewall entre redes funcione como tal debe tener al menos dos tarjetas de red. La figura 2.3 muestra la tipología clásica de un firewall.

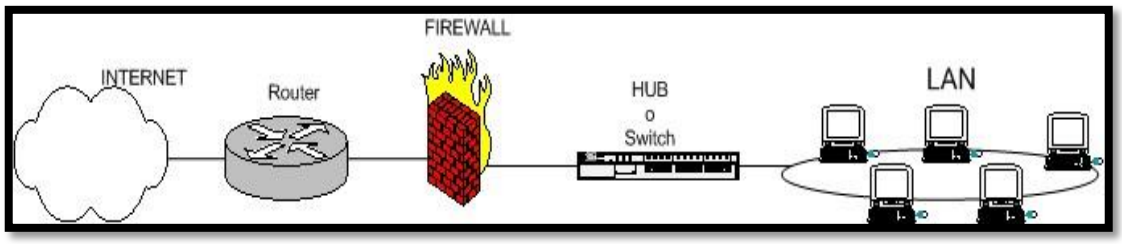


Figura 2.3: Esquema de firewall típico entre red local e internet

En la figura anterior se muestra un esquema típico de firewall para proteger una red local conectada a internet a través de un router. En ella se observa que el firewall debe colocarse entre el router (con un único cable) y la red local (conectado al switch o al hub de la LAN).

Dependiendo de las necesidades de cada red, puede ponerse uno o más firewalls para establecer distintos perímetros de seguridad en torno a un sistema. Es frecuente también que se necesite exponer algún servidor a internet (como es el caso de un servidor web, un servidor de correo, etc.), y en esos casos obviamente en principio se debe aceptar cualquier conexión a ellos. Lo que se recomienda en esa situación es situar ese servidor en lugar aparte de la red, el que se denomina DMZ o zona desmilitarizada. El firewall tendría entonces tres entradas como se muestra en la figura 2.4.

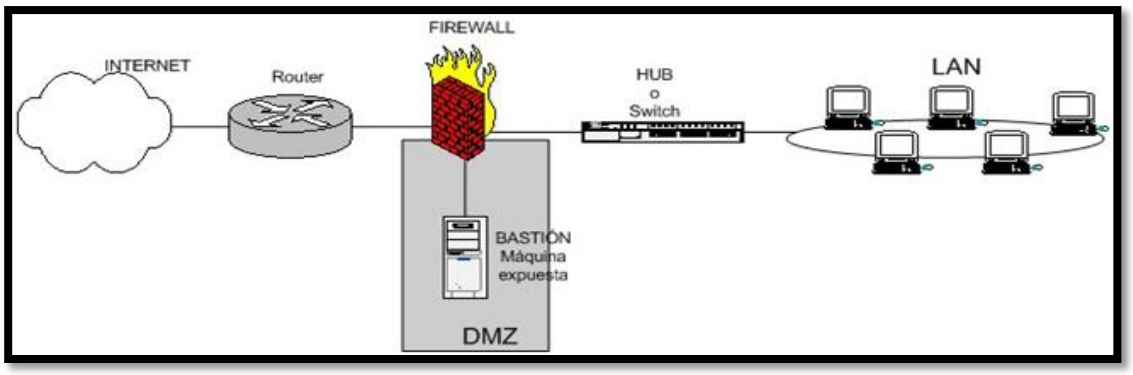


Figura 2.4: Esquema de firewall entre red local e internet con zona DMZ para servidores expuestos

En la zona desmilitarizada se pueden poner tantos servidores como se necesiten. Con esta arquitectura, se permite que el servidor sea accesible desde internet de tal forma que si es atacado y se gana acceso a él, la red local sigue protegida por el firewall. Esta estructura de DMZ puede hacerse también con un doble firewall (se puede usar un único dispositivo con al menos tres interfaces de red). Vea figura 2.5.

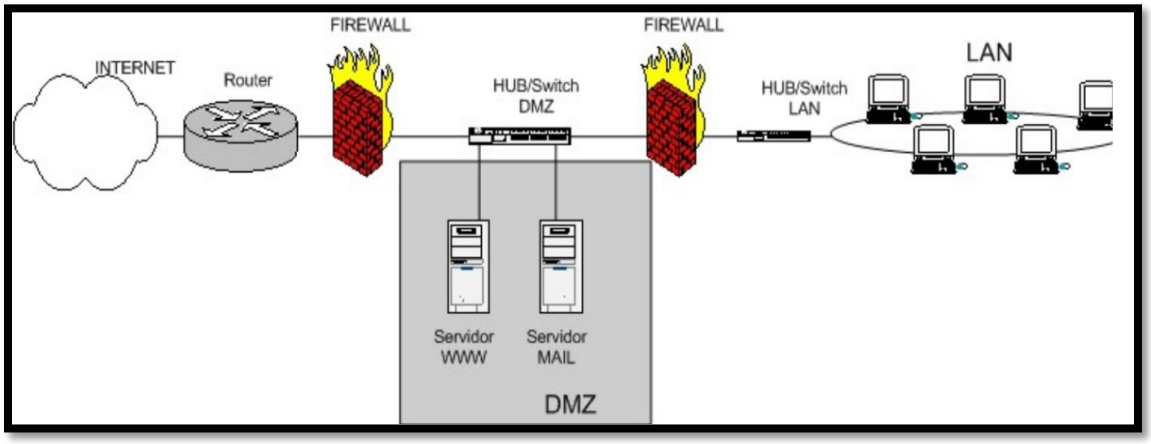


Figura 2.5: Esquema de firewall entre red local e internet con zona DMZ para servidores expuestos creado con doble firewall (perímetro)

Los firewalls se pueden usar en cualquier red. Es habitual tenerlos como protección perimetral en las empresas donde filtran el tráfico de entrada y salida a la red.

Un firewall utiliza un conjunto de reglas que permite filtrar a nivel de paquetes del protocolo TCP/IP y son capaces de filtrar muchos tipos protocolos.

2.10.2 Maneras de implementar un firewall:

i. Política por defecto ACEPTAR: en principio todo lo que entra y sale por el firewall se acepta y solo se denegará lo que se diga explícitamente.

ii. Política por defecto DENEGAR: todo esta denegado, y solo se permitirá pasar por el firewall aquellos que se permita explícitamente.

2.10.3 IPtables

IPtables es un sistema de firewall vinculado al kernel de linux que se ha extendido enormemente a partir del kernel 2.4 de este sistema operativo. Iptables está integrado con el kernel, es parte del sistema operativo. Realmente lo que se hace es aplicar reglas. Para ellos se ejecuta el comando iptables, para añadir, borrar o crear reglas. Por ello un firewall de iptables no es sino un simple script de shell en el que se van ejecutando las reglas de firewall.

2.10.4 UFW

El manejo de las reglas elementales para el firewall en Ubuntu puede hacerse con UFW **(***Uncomplicated Firewall***)** que es muy fácil de utilizar y convierte la regla al formato de IPtables. La aplicación UFW viene integrada desde la versión de Ubuntu 8.04 LTS, desarrollada para facilitar la configuración del firewall IPtables, UFW proporciona una manera fácil de usar para crear un host basado en IPv4 o IPv6 firewall. UFW por defecto está inicialmente deshabilitado.

UFW no pretende proporcionar una completa funcionalidad de firewall vía la interfaz de comandos, pero en su lugar proporciona un modo fácil de añadir y eliminar reglas simples Vea Anexo IV Algunas reglas de UFW. En la actualidad se utiliza principalmente para cortafuegos basados en host. Se propone UFW por ser una solución de administración de firewall sencilla, sin necesidad de entorno gráfico y con plena integración en el sistema de paquetes. Vea Anexo V Ejemplos de cómo utilizar UFW.

2.11 Servidor de correo

En capitulo uno se definió que un servidor de correo debe constar en realidad de dos servidores el SMTP encargado de enviar y recibir mensajes, y un servidor POP/IMAP, que será el que permita a los usuarios obtener sus mensajes.

2.11.1 Postfix

Postfix es un MTA (Mail Transport Agent), escrito originalmente por Wietse Venema, que comenzó siendo una alternativa a Sendmail que es más complicado para configurar. Postfix es una de las alternativas, como lo son también Qmail, Zmailer, etc. Postfix es el MTA por defecto de Ubuntu y hoy en día ha probado ser una alternativa más sencilla, segura y mucho más fácil de administrar que otros MTAs.

2.11.1.1 Paquetes de instalación del Postfix

Los paquetes necesarios para la instalación del servidor de correo son:

- postfix – MTA de alto desempeño.
- postfix-ldap – para la autenticación con el servidor LDAP.
- mailutils – Herramientas para la manejo de correo electrónico.

Se instalan ejecutando la línea de comando: apt-get install postfix postfix-ldap mailutils. Vea Ubuntu Guide, año 2012, pagina 235 capitulo 15 "*Email Services*" epígrafe 1.1 y 1.2

2.11.1.2 Archivos de configuración

Los archivos de mayor importancia de Postfix se encuentran en /etc/postfix/, estos son:

- main.cf
- master.cf

2.11.1.3 Comprobando el funcionamiento del servidor Postfix

Para comprobar el buen funcionamiento del servidor de correo Postfix se pueden emplear algunos comandos:

telnet Nombre_del_Dominio telnet Navine.com

Trying 12.12.34.201...

Connected to navine.com

Escape character is '^]'.

220 navine.com ESMTP Postfix (Ubuntu)

Se ejecutará el siguiente comando para ver las opciones que tiene habilitadas el servidor Postfix:

ehlo hola

250-navine.com

250-PIPELINING

250-SIZE 10240000

250-ETRN

250-STARTTLS

250-ENHANCEDSTATUSCODES

250-8BITMIME

250 DSN

quit

- **Enviar correos desde la consola usando el comando "mailx"**

  mailx jose@navine.com

  Cc: jose@navine.com

  Subject: probando el servidor postfix

  texto

  ctrl +d

Se revisará si el correo llegó correctamente, y si todo está en orden. Si es el caso, se puede afirmar entonces que el servidor Postfix está funcionando correctamente.

2.11.2 Dovecot

Es un servidor de IMAP y POP3 de código abierto para sistemas GNU/Linux / UNIX-like, escrito fundamentalmente enfocando en seguridad. Desarrollado por Timo Sirainen, Dovecot fue liberado por primera vez en julio del año 2002. Dovecot apunta fundamentalmente a ser un servidor de correo de código abierto ligero, rápido, fácil de instalar y por sobre todo seguro.

2.11.2.1 Características de Dovecot

Dovecot puede trabajar con el estándar mbox, maildir y sus propios formatos nativos dbox de alto desempeño. Es completamente compatible con implementaciones de servidores UW IMAP y Courier IMAP, así como con clientes que accedan directamente a los buzones de correo.

Dovecot también incluye un Agente de Entrega de Correo llamado *Local Delivery Agent o LDA* en la documentación de Dovecot con un filtro de apoyo opcional.

2.11.2.2 Paquetes de instalación

Para la instalación del servicio dovecot se instalan los paquetes siguientes:

- dovecot-imappd
- dovecot-pop3d

Se instalan ejecutando la línea de comando: apt-get install dovecot-imapd dovecot-pop3d. Vea Ubuntu Guide, año 2012, pagina 245 capitulo 15 "*Email Services*" epígrafe 3 "*Dovecot Server*"

2.11.2.3 Comprobando el funcionamiento del servidor dovecot

Se inicia Dovecot ejecutando la línea de comando: /etc/init.d/dovecot start

Para comprobar que se está ejecutando, escriba el comando

```
ps -A | grep dovecot
```

Se debe ver el servicio Dovecot funcionando. Si ha habilitado IMAP o POP3, también puede tratar de iniciar sesión con los comandos:

```
telnet localhost pop3 o telnet localhost imap2
```

Si se ve algo como lo siguiente, la instalación ha sido exitosa.

```
telnet localhost pop3

Trying localhost...

Connected to localhost.

Escape character is '^]'.

+OK dovecot ready
```

2.12 Squid (Servidor de Proxy)

Squid es un Servidor Intermediario (Proxy) de alto desempeño que se ha venido desarrollando desde hace varios años y es hoy en día un muy popular y ampliamente utilizado entre los sistemas operativos como GNU/Linux y derivados de Unix (figura 2.6).

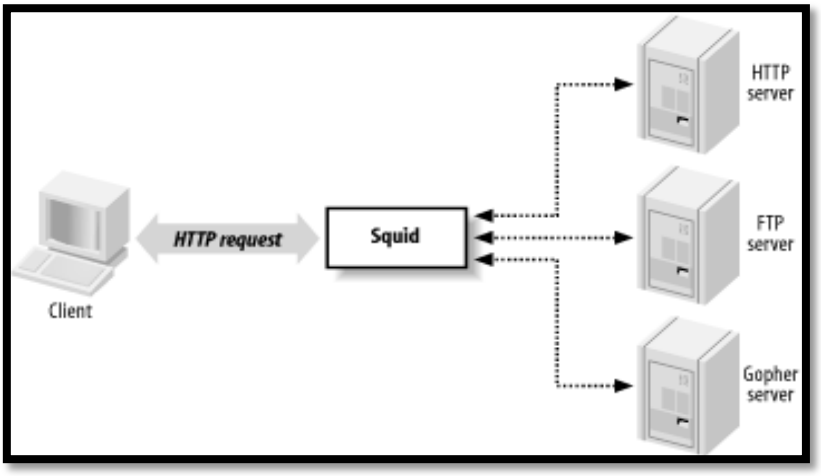


Figura 2.6: Squid como proxy entre clientes y servidores

En el presente trabajo se propone Squid por ser el servidor Proxy más popular y extendido entre los sistemas operativos basados sobre UNIX. Es muy confiable, robusto y versátil. Al ser software libre está libre del pago de costosas licencias de uso o de restricciones en cuanto al número de usuarios.

Entre otras cosas, Squid puede hacer Proxy y caché con los protocolos HTTP, FTP, GOPHER y WAIS, Proxy de SSL, caché transparente, WWCP, aceleración HTTP, caché de consultas DNS y más. El Squid trabaja como un proxy y a la vez una caché. Como un proxy, el Squid es un intermediario en una transacción web. Acepta una demanda de un cliente, procesa tal demanda, y entonces envía la demanda al servidor del origen. La demanda se puede ser anotado, rechazado, e incluso modificado antes de remitir. Como un caché el Squid guarda páginas web recientemente recuperadas para su posible re-uso después. Pueden servirse demandas subsecuentes para los mismas páginas web desde el caché, en lugar de consultar al servidor de origen nuevamente. Si es necesario se puede desactivar la funcionalidad de caché en el Squid, pero la parte del proxy es esencial.

2.12.1 Funcionamiento del Squid

- Mejor uso del ancho de banda.
- Reducir el tiempo de carga de las páginas web que tiene en caché.
- Proteger las máquinas de la red interna.
- Recoger estadística sobre el tráfico web.
- Prevenir el acceso a sitios inapropiados.
- Brindar el servicio a usuarios con autorización.

2.12.2 Ficheros de Eventos (Log)

Squid genera cuatro ficheros de log:

- logs/access.log: Guardan las peticiones que se le hacen al proxy, se puede saber cuántos usuarios usan el proxy, qué páginas son las más visitadas, etc.
- logs/cache.log: Guarda todos los errores generados, mensajes de inicio, etc.
- logs/store.log: Refleja lo que pasa con el caché, que páginas (objetos) se añaden, y cuales se quitan.
- cache/log: Se mapean los objetos (páginas guardadas) al fichero en el que están guardadas físicamente.

2.12.3 Paquete de instalación del Squid

El paquete de instalación en Ubuntu GNU/Linux es **squid3,** que se instala ejecutando la línea de comando: apt-get install **squid3**

Control de su funcionamiento

- /etc/init.d/squid start
- /etc/init.d/squid stop
- /etc/init.d/squid restart

2.12.4 Archivo de configuración

Squid utiliza el archivo de configuración localizado en **/etc/squid/squid.conf.** Vea Ubuntu Server Guide, año 2012, pagina 197 capitulo 13, epígrafe 3 "*Squid-Proxy Server*" para su instalación.

2.12.5 Algoritmos de caché utilizados por Squid.

A través de un parámetro (**cache_replacement_policy**) **Squid** incluye soporte para los siguientes algoritmos para el caché (vea la tabla 2.1).

Tabla 2.1: Algoritmos para el caché del Squid

LRU	Acrónimo de **L**east **R**ecently **U**sed, que traduce como **Menos Recientemente Utilizado**. En este algoritmo los objetos que no han sido accedidos en mucho tiempo son eliminados primero, manteniendo siempre en el caché a los objetos más recientemente solicitados. **Ésta política es la utilizada por Squid de modo predefinido**.
LFUDA	Acrónimo de **L**east **F**requently **U**sed with **D**ynamic **A**ging, que se traduce como **Menos Frecuentemente Utilizado con Envejecimiento Dinámico**. En este algoritmo los objetos más solicitados permanecen en el caché sin importar su tamaño optimizando la **eficiencia** (hit rate) por **octetos** (Bytes) a expensas de la eficiencia misma, de modo que un objeto grande que se solicite con mayor frecuencia impedirá que se pueda hacer caché de objetos pequeños que se soliciten con menor frecuencia.
GDSF	Acrónimo de **G**reedy**D**ual **S**ize **F**requency, que se traduce como **Frecuencia de tamaño *GreedyDual*** (*codicioso dual*), que es el

	algoritmo sobre el cual se basa **GDSF**. Optimiza la **eficiencia** (hit rate) por objeto manteniendo en el caché los objetos pequeños más frecuentemente solicitados de modo que hay mejores posibilidades de lograr **respuesta a una solicitud** (hit). Tiene una eficiencia por **octetos** (Bytes) menor que el algoritmo **LFUDA** debido a que descarta del caché objetos grandes que sean solicitado con frecuencia.

2.13 Replicación LDAP

2.13.1 Introducción

Una organización puede confiar en LDAP para la organización de su dominio pero llega el momento en que la pérdida del servidor LDAP es inaceptable o el volumen de su consulta es muy grande y se necesitan múltiples servidores, o podría ser incluso una combinación de ambos; pero en cualquier caso, se necesita utilizar más de un servidor. Lo más común es tener en adición al servidor PDC que es el servidor máster con LDAP, un servidor BDC o esclavo que tiene una copia del LDAP del PDC.

OpenLDAP suministra dos métodos para realizar la replicación. El primero se realiza a través de slurpd, un demonio separado que observa los cambios en el PDC y conduce los cambios en los esclavos. El segundo utiliza el motor de replicación de sincronización de LDAP, de otro modo conocido como syncrepl. El método slurpd es como considerado obsoleto actualmente.

2.13.2 Replicación por mecanismo syncrepl

Syncrepl permite que los cambios sean sincronizados usando un modelo de consumidor y proveedor donde el proveedor envía cambios de directorio directamente al consumidor.

Syncrepl se crea como una capa insertada entre el núcleo de slapd y la base de datos o backend. Todos los ingresos al árbol son rastreados por el motor syncrepl en lugar de requerir una instancia de servidor distinta. Los ingresos a una réplica son rechazados, con una referencia devuelta al servidor máster. Se inicia desde el esclavo, al cual ahora se le da el nombre de *consumer* y el rol del servidor PDC es denominado *provider*.

2.13.2.1 Funcionamiento de syncrepl

Con syncrepl, el consumidor se conecta al proveedor para obtener actualizaciones del árbol de directorio. En el modo más básico, denominado refreshOnly, el consumidor recibe todas las entradas modificadas desde la última actualización, se solicita una cookie que mantiene el rastro del último cambio sincronizado y luego se desconecta. En la siguiente conexión, la cookie se presenta al proveedor, el cual se envía sólo las entradas que cambiaron desde la última sincronización.

Otro modo syncrepl, se denominado refreshAndPersist, inicia como la operación refreshOnly; pero en lugar de desconectar, el consumidor permanece conectado para recibir cualquier actualización y cualquier cambio que sucede después de que la actualización inicial es enviada inmediatamente al consumidor. Vea Ubuntu Server Guide, año 2012, página 99, sesión 1.6 "*Replication*"

2.14 Conclusiones parciales del capítulo

- En este capítulo se ha definido Ubuntu como un sistema operativo para utilizar en los servidores. Se realizó la instalación y configuración de cada servicio proporcionándose una guía de recomendaciones.
- Se utiliza como plataforma de trabajo para implementar los servidores el *ORACLE VirtualBox versión 4.1.14* que prueba ser muy eficiente para probar las instalaciones en un ambiente controlado antes de pasarlas a los servidores de producción.
- Se planteó una guía lógica a seguir para la instalación y distribución de los servicios que puede ser muy útil para guiar el trabajo de administradores con poca

experiencia en Linux. En esta instalación fueron utilizados servidores que son todos accesibles mediante el repositorio de Ubuntu.

- Se realizó la selección de Postfix como servidor de correo electrónico y de Squid como servidor proxy basándose en el buen desempeño de los mismos y por ser software propios de la distribución de Ubuntu garantizando su soporte para la distribución y mantenimiento.

CAPÍTULO III: ADMINISTRACIÓN DEL DOMINIO

Una tarea administrativa muy importante es la adición de clientes al dominio. Una vez configurado los servidores se comienza el proceso de agregar los clientes al dominio.

3.1 Configuración de los terminales clientes

Se consideran en este caso los clientes que actualmente son más utilizados, siendo estos Microsoft Windows XP, Microsoft Windows 7 y clientes Linux. Por la popularidad que ha tomado Ubuntu como máquina de escritorio, en este capítulo se presenta la adición de un Ubuntu Desktop al dominio aunque el proceso es similar para cualquier otra distribución.

Un aspecto importante a garantizar antes de intentar adicionar los clientes al dominio es que la resolución de nombres este funcionando correctamente y este activada la opción de NetBIOS sobre TCP/IP.

3.1.1 Herramienta gráfica para la administración del directorio OpenLDAP

Existen muchas aplicaciones web que permiten el acceso al directorio LDAP para poder crear y modificar elementos de los cuales PhpLdapAdmin es una de las más utilizadas. Otros posibles exploradores LDAP libres podrían ser LAM (*LDAP Account Manager*) y/o Jexplorer. PhpLdapAdmin necesita de un servidor web, por lo que en caso de no tener ninguno funcionando durante el proceso de su instalación se instalará automáticamente apache2.

La instalación se hace con: apt-get install phpldapadmin

Por defecto la instalación se hace en /usr/share/phpldapadmin, puede copiar esta carpeta completa para /var/www o hacer un enlace simbólico con este directorio para que el servidor web lo muestre.

El archivo de configuración para el phpldapadmin es **config.php** y se encuentra dentro del directorio phpldapadmin (vea Anexo VI Configuración de la aplicación PhpLdapAdmin).

3.1.2 Configuración de red de la máquina cliente Windows XP/Seven

Para configurar la red de las máquinas clientes debe asignar al cliente una dirección IP dentro del segmento de red y se configura la red para que el servidor WINS sea el servidor Samba y el DNS apunte al servidor que corre el demonio bind9. Se debe habilitar también NetBIOS sobre TCP/IP. Antes de intentar unir el cliente al dominio puede utilizar comandos como ping y nslookup para comprobar que la resolución de nombres funciona correctamente. Se asume que es conocido como iniciar el asistente para configurar la red. Vea detalles en la figura 3.1.

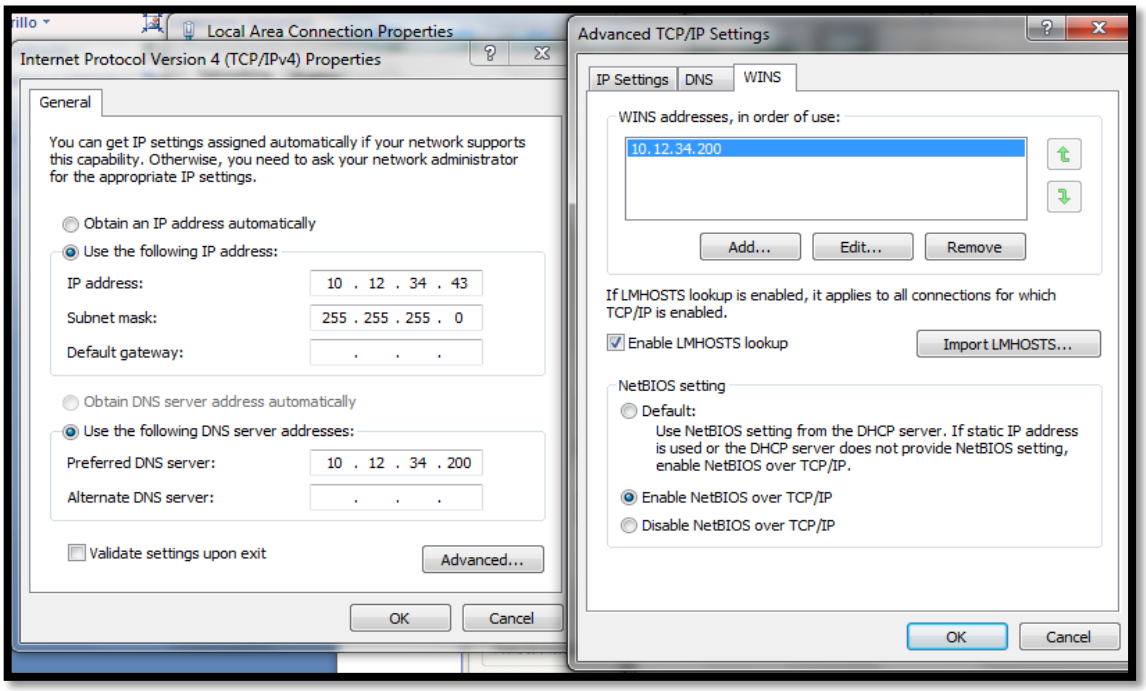


Figura 3.1: Configuración de red de la máquina cliente

3.1.3 Uniendo Windows XP Professional al dominio

Para unir clientes Windows al dominio se siguen los pasos siguientes:

Desde el menú “System Properties”, se accede a la pestaña “Computer Name” se pulsa sobre el botón “Change”. Por defecto la máquina es parte del grupo de trabajo “WORKGROUP” como se ve en figura 3.2. Se pulsa sobre el *radio botton* con etiqueta “Domain”, y al activar el campo de texto se pone el nombre corto del dominio (figura 3.3).

A continuación pulsa el botón OK, y sale un asistente solicitando las credenciales de un usuario con derechos de agregar clientes al dominio (figura 3.4).

Si se completa con éxito el ingreso al dominio se recibe un mensaje de bienvenida y una solicitud de reinicio del sistema.

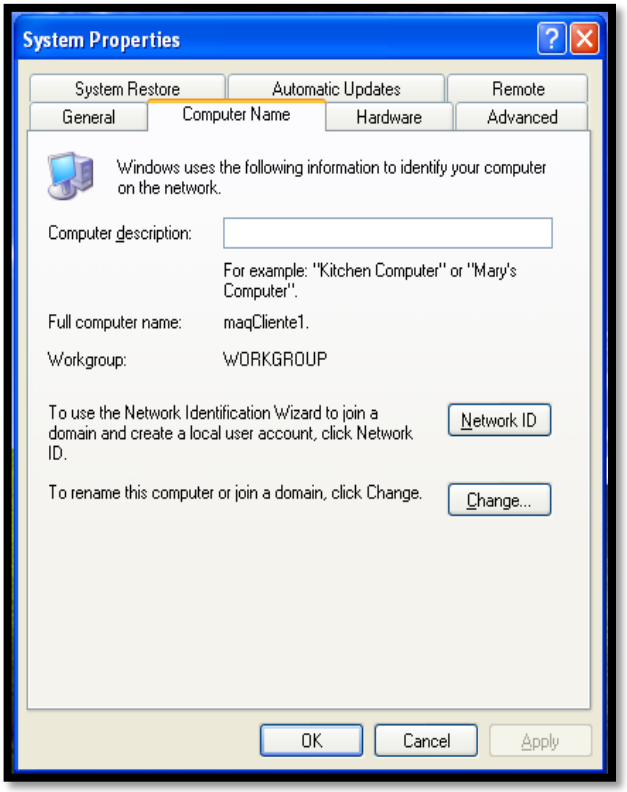


Figura 3.2: Propiedades del sistema Windows XP.

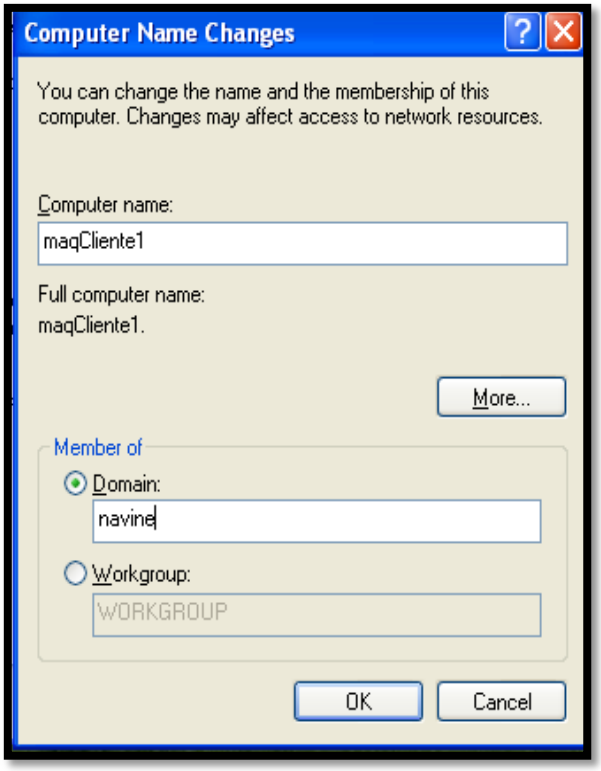


Figura 3.3: Mostrando como añadir maquina a un dominio “NAVINE”

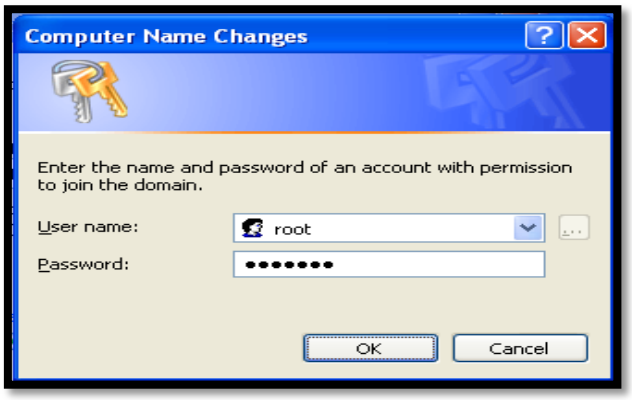


Figura 3.4:Petición por usuario administrativo de LDAP

Puede verificar que la computadora está realmente en el dominio utilizando PhpLadapAdmin (figura 3.5).

Figura 3.5: Windows XP máquina “maqcliente1” como nuevo computadora añadido al dominio

3.1.3.1 Iniciar sesión de un usuario del dominio

Al iniciar la máquina cliente debe mostrar en este momento la opción de entrar el dominio. Se puede ingresar al dominio utilizando el nombre de usuario y su contraseña de un usuario del LDAP (figura 3.6). La figura 3.7 muestra una sesión de usuario “winuser” conectado al dominio.

Figura 3.6: Iniciando sesión con usuario "winuser" miembro del dominio

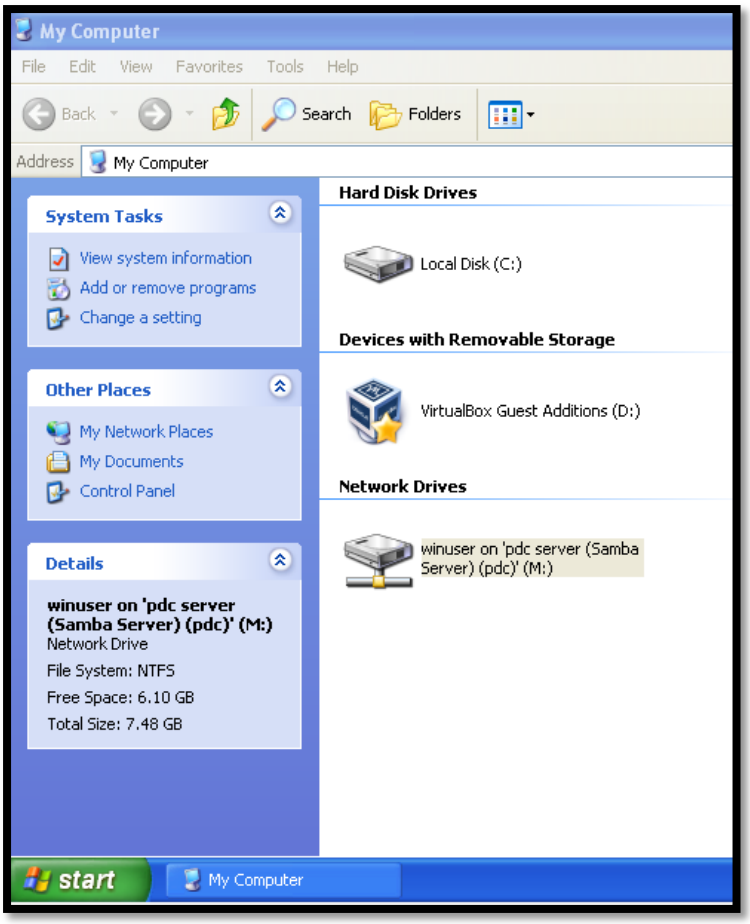


Figura 3.7: Sesión de usuario "winuser" en el dominio.

3.1.4 Uniendo Windows 7 al dominio

Para unir un cliente de Windows 7 al dominio se deben modificar dos campos del registro. Para ello puede guiarse por la información publicada en https://bugzilla.samba.org/attachment.cgi?id=4988&action=view que debe ser bajado de la red y ejecutado.

A continuación se accede a "*System properties*", desde el apartado "*Advanced system settings"* se selecciona, se accede a la pestaña "*Computer Name"* y de aquí en adelante el proceso es similar al realizado para Windows XP Professional. La comprobación de que el cliente fue añadido con éxito es similar a la realizada con Windows XP.

3.1.5 Uniendo clientes Linux al dominio.

. Primero se instala el software necesario para que funcione:

apt-get install auth-client-config libpam-ldap libnss-ldap

Ahora se deberá responder a las preguntas que se hacen en el proceso de instalación (configuración de auth-client-config) de la siguiente forma:

Should debconf manage LDAP configuration? Yes

LDAP server Uniform Resource identifier: LDAP://10.12.34.200/

Distinguished name of the search base: dc=navine,dc=com

LDAP version to use: 3

Make local root Database admin: Yes

Does the LDAP database require login? No

LDAP account for root: cn=admin,dc=navine,dc=com

LDAP root account password: **dejar vacío**

A continuación el archivo /etc/ldap.conf será modificado y se le agregará lo siguiente:

host 10.12.34.200

base dc=navine,dc=com

uri ldap://10.12.34.200/

ldap_version 3

rootbinddn cn=admin,dc=navine,dc=com

port 389

```
bind_policy soft
pam_password md5
```

Ahora se copiará el archivo /etc/ldap.conf a /etc/ldap/ldap.conf con:

```
cp /etc/ldap.conf /etc/ldap/ldap.conf
```

Se creará un nuevo arhivo en /etc/auth-client-config/profile.d con:

```
touch /etc/auth-client-config/profile.d/open_ldap
```

El archivo open_ldap debe ser editado y se le agregará la siguiente sintaxis:

```
[open_ldap]
nss_passwd=passwd:      compat ldap
nss_group=group:        compat ldap
nss_netgroup=netgroup:  compat ldap
nss_shadow=shadow:      compat ldap

pam_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
pam_account=account     sufficient      pam_unix.so
account                 sufficient      pam_ldap.so
account                 required        pam_deny.so
```

```
pam_password=password    sufficient    pam_unix.so    nullok
md5 shadow use_authtok
password        sufficient    pam_ldap.so use_first_pass
password        required      pam_deny.so
pam_session=session        required      pam_limits.so
session         required      pam_mkhomedir.so  skel=/etc/skel/
session         required      pam_unix.so
session         optional      pam_ldap.so
```

Se debe realizar una copia de seguridad de /etc/nsswitch.conf

```
cp  /etc/nsswitch.conf{,.original}
```

Ahora se realizará una copia de seguridad de pam.d de la siguiente forma:

```
cd /etc/pam.d/
mkdir bkup
cp * bkup/
```

Finalmente se activa el nuevo perfil de autentificación LDAP ejecutando el siguiente comando:

```
auth-client-config -a -p open_ldap
```

Ahora se debe reiniciar la PC cliente y después de esto debe dejar iniciar una sesión a los usuarios del dominio.

3.2 Ejemplo del correo electrónico implementado

A continuación se muestra un ejemplo básico del servicio de correo electrónico funcionando. Para realizar este ejemplo se utilizó como MUA el Microsoft Outlook 2007 (figura 3.8) desde una máquina ejecutando el virtual box ORACLE VitualBox versión 4.1.14 que sirvió como plataforma de prueba.

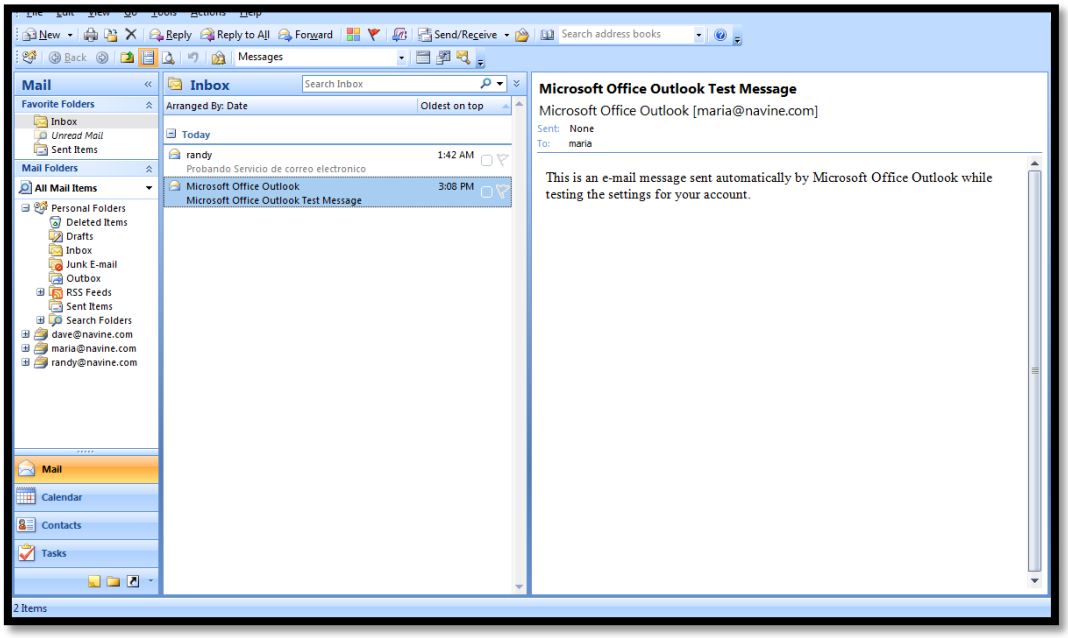


Figura 3.8: Microsoft Outlook 2007 como MUA de correo electrónico

Las figuras 3.9 y 3.10 muestran un ejemplo de la comunicación entre dos usuarios del dominio, nombrados Maria y Randy por medio de correo electrónico. En ella se puede ver el envío de un correo con asunto (subject) "***testing mail server***" desde la cuenta de correo de María (maria@navine.com) hacia la cuenta de Randy (randy@navine.com). Estas figuras prueban que los tres componentes del correo electrónico están en colaboración y funcionan perfectamente bien.

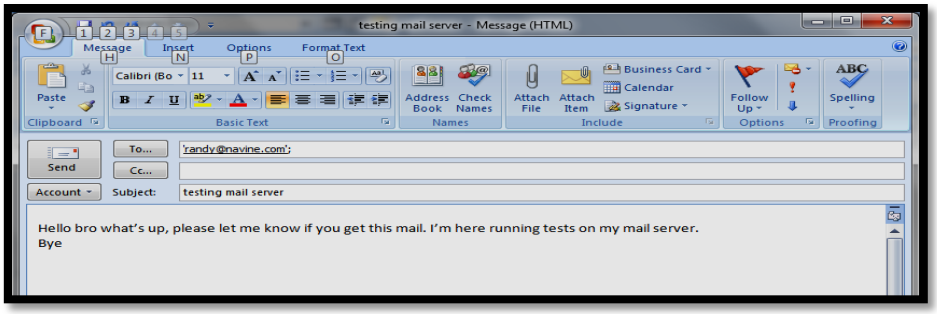


Figura 3.9: Envió de correo desde usuario maría a usuario randy.

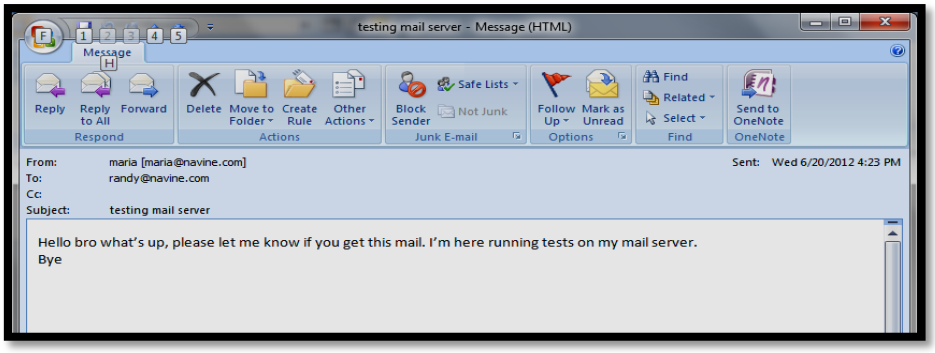


Figura 3.10: Entrada de correo electrónico a usuario randy.

3.3 Errores más comunes en el proceso de migración

Actualmente el país ha definido una política de migración hacia software libre, esta política se ha ido aplicando paulatinamente en diferentes sectores dándose como una regularidad que en la mayoría de ellos no se cuenta con personal capacitado en Linux y administración con Linux.

3.3.1 Errores detectados frecuentemente en este proceso:

- Aplicación de directivas que obligan a migrar en determinada etapa sin tener condiciones reales para enfrentar la migración.
- Contratación de personal no calificado en el área de administración de redes para realizar funciones de administrador.
- Ambientes de trabajo sin acceso a internet para los administradores lo que hace casi imposible la autosuperación y le niega el acceso a comunidades de internet que pueden brindar ayuda ante cualquier problema.
- Ausencia de servicios de redes nacionales a los que tengan acceso todas las empresas.

Esto hace que al enfrentar el proceso técnico de migrar se produzcan dificultades como las mostradas en el siguiente tópico.

3.3.2 Dificultades observadas en el proceso de migración.

- Trabajo con repositorios no actualizados.
- No se configuran las actualizaciones de seguridad al no tener acceso a las mismas.
- Grandes pérdidas de tiempo tratando de obtener información disponible en internet pero a la que no tienen acceso.
- Creación de ambientes de redes donde los servidores se migran a Linux pero no se configura dominio por falta de conocimientos sobre el proceso.
- Malas configuraciones de servicios básicos como correo electrónico, DNS, navegación de internet, etc. por no tener la experiencia e información necesaria. En algunos casos simplemente no se oferta el servicio o se oferta un servicio de mala calidad y pésima seguridad.
- Utilización de distribuciones de Linux inestables para servidores que pueden traer dificultades en su funcionamiento.

3.4 Conclusiones parciales del capítulo

- En este capítulo se muestra que es posible adicionar los clientes al dominio siguiendo un procedimiento sencillo.
- Es posible usar el servicio de de correo para comunicar a diferentes usuarios utilizando Microsoft Outlook 2007.
- El proceso de migración a software libre de los principales servicios de una red no es un procedimiento simple y no está exento de dificultades.

CONCLUSIÓN

Como resultado general se plantea que este trabajo sirve de guía o manual de ayuda para qué personas con pocos conocimientos de administración puedan enfrentar la tarea de implementar un dominio con software libre y servicios básicos de una manera fácil.

RECOMENDACIONES

- Continuar este trabajo de manera que se perfeccionen las configuraciones de los servicios instalados y se adicionen nuevos servicios.
- Realizar un trabajo similar a este pero utilizando Samba 4 que aunque no está disponible como versión estable está siendo muy utilizada.
- Divulgar este trabajo en centros que están migrando al software libre y no cuentan con administradores experimentados.

REFERENCIAS BIBLIOGRÁFICAS

1. Burgess, M., *Principles of Network and System Administration*. Second Edition ed2004.

2. Internet. Available from: http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml.

3. *Qué es GNU/Linux*. Available from: http://www.gnulinuxpy.org/que-es-gnulinux/.

4. Foundation, L. *Linux está alcanzando nuevos niveles en las empresas, según estudio*. 2012; Available from: http://pro.pcworld.pe/noticias/linux-esta-alcanzando-nuevos-niveles-en-empresas-segun-estudio/.

5. Angelverde. 2012; Available from: http://angelverde.info/50-lugares-donde-no-te-imaginabas-que-usan-linux/.

6. CNY Support, L. *The Best Things In Life Are Free*. 2012; Available from: http://www.cnysupport.com/index.php/development-and-support/windows-linux-license-cost-comparison.

7. Wikipedia. Available from: http://es-wikipedia.org/wiki/Ubuntu#mw-head.

8. Ubuntu-Documents, O.S.f. *Ubuntu Server Guide*. 2010.

9. Internet. *Como Instalar y Configurar un Servidor DHCP*. Available from: http://www.guatewireless.org/os/linux/distros/debian/ubuntu/como-instalar-y-configurar-un-servidor-dhcp-en-linux-ubuntu-debian/.

10. Jelmer R. Vernooij, J.H.T., and Gerald Carter, *The Official samba 3.2.x HOWTO and Reference Guide*, 2008.

11. Wikipedia. *Servicio de directorio* 2012; Available from: http://es.wikipedia.org/wiki/Servicio_de_directorio.

12. Dent, K.D., *Postfix, The Definitive Guide*2003: O´Reilly.

13. Wikipedia. *Proxy*. Available from: http://es.wikipedia.org/wiki/Proxy.

14. Wessels, D., *Squid: The Definitive Guide*, 2004, O´Reilly.

15. LZURA, P. X. A. IPtables Manual Práctico, tutorial de iptables con ejemplos. Available: http://pello.info .

BIBLIOGRAFÍA

BURGESS, M. 2004. Principles of Network and System Administration.

CATER, G. 2003. LDAP System Administration. O´Reilly.

DANTE ORDÍN RAMÍREZ LÓPEZ, C. C. M. 2011. El Cifrado Web (SSL/TLS). Available: http://revista.seguridad.unam.mx/numero-10/el-cifrado-web-ssltls.

DENT, K. D. 2003. Postfix, The Definitive Guide, O´Reilly.

DONAHUE, G. A. 2007. Network Warrior. O´Reilly.

H.TERPSTRA, J. 2008. Samba-3 by Example.

HAGEN, W. V. 2007. Ubuntu Linux Bible. Wiley Publishing,Inc.

HUNT, C. Linux Network Servers.

JELMER R. VERNOOIJ, J. H. T., AND GERALD CARTER 2008. The Official samba 3.2.x HOWTO and Reference Guide.

KOETTER, R. H. A. P. 2005. The Book of Postfix State-of -The-Art Message Transaport.

O´REILLY. Available: http://www.oreillynet.com/lpt/a/6849.

SAMS 2000. Maximum Linux Security.

SMITH, R. W. 2001. Linux Samba Server Administration.

SMITH, R. W. 2002. Linux Samba Server Administration. SYBEX Inc.

TANENBAUM, A. S. 2003. Computer Networks.

UBUNTU DOCUMENTATION TEAM, 2010, Ubuntu Server Guide.

UBUNTU DOCUMENTATION TEAM, 2012, Ubuntu Server Guide.

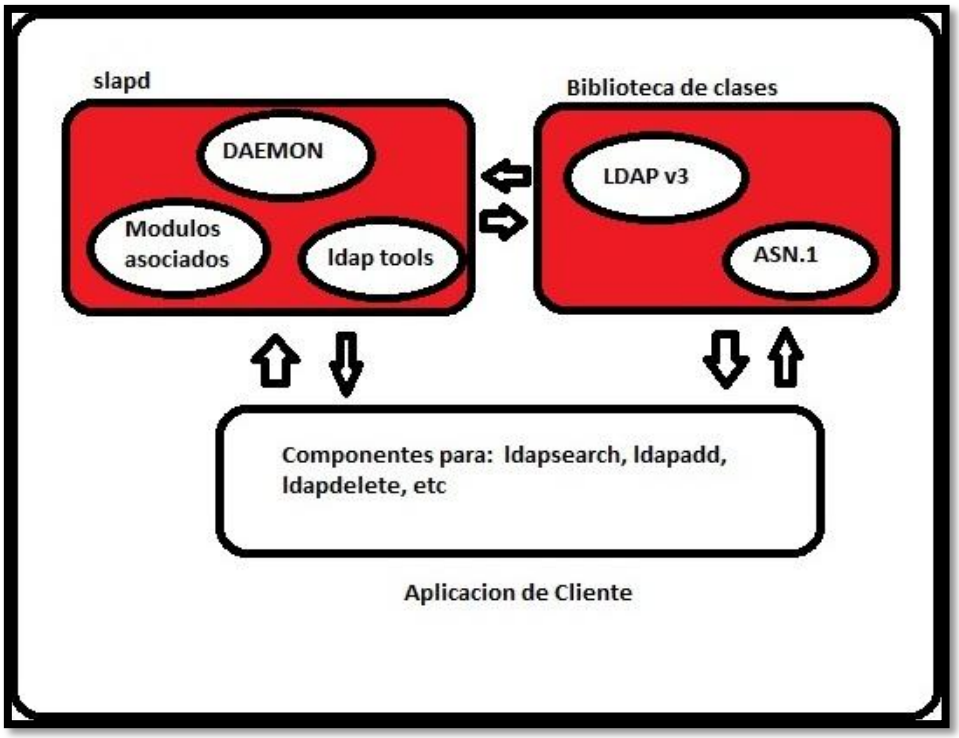
WESSELS, D. 2004. Squid: The Definitive Guide. O´Reilly.

SITIOS WEB CONSULTADOS

- http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml.
- https://launchpad.net/~ubuntu-core-doc
- https://launchpad.net/~ubuntu-server
- https://help.ubuntu.com/community/
- https://code.launchpad.net/serverguide
- https://code.launchpad.net/ubuntu-docs
- http://www.serverworld.net
- http://www.openldap.org
- http://www.serverfault.net
- http://sun.com/software/products/directory_srvr-ee/.
- http://www.blogwindows.com/%C2%BFactive-directory-o-ldap-openldap/122/
- http://www.guatewireless.org/os/linux/distros/debian/ubuntu/como-instalar-y-configurar-un-servidor-dhcp-en-linux-ubuntu-debian/.
- http://es.tldp.org/COMO-INSFLUG/COMOs/LDAP-Linux-Como/.
- https://help.ubuntu.com/11.04/serverguide/C/postfix.html.
- http://www.kriptopolis.org/seguridad-en-servidores-windows-vs-linux.
- http://www.servidoresdedicados.com/.
- http://www.servidoresdedicados.com/linux-windows.asp
- http://www.squid-cache.org/.
- https://launchpad.net/ufw.
- http://es.wikipedia.org
- http://www.tufuncion.ccom/windows-vs-linux/.
- http://www.entmexico.com/hosting/windows-o-linux.html
- http://es.wikipedia.org/wiki/OpenLDAP.
- http://es.wikipedia.org/wiki/Proxy.

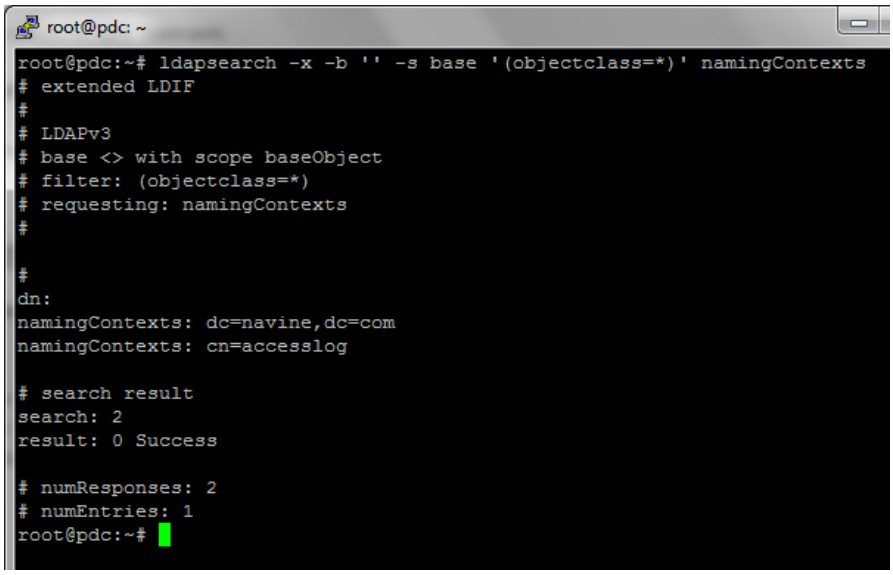
ANEXOS

Anexo I Componentes principales del servidor OpenLDAP



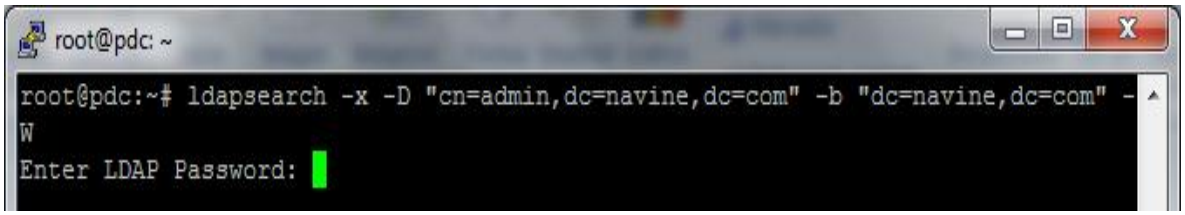
Anexo II Pruebas Preliminares de servidor LDAP

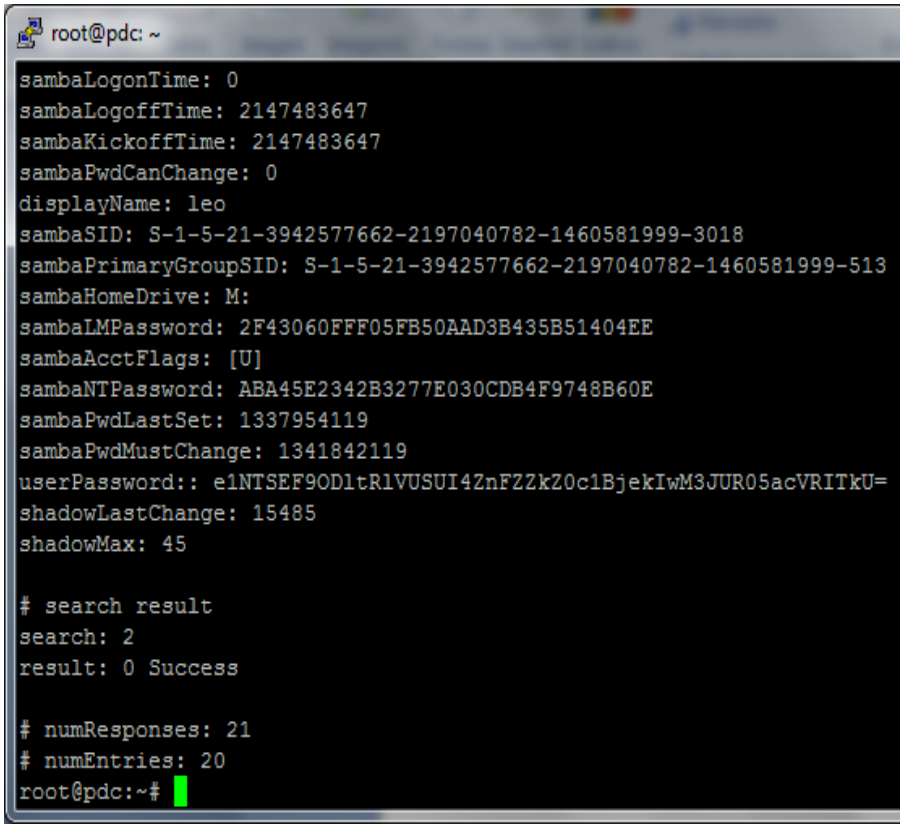
1) Consulta al servidor OpenLDAP para ver si responde correctamente empleando la opción namingContexts



2) Realizar una búsqueda en el directorio ldap autenticado como el usuario administrador del OpenLDAP (admin) y especificando como base dc=navine, dc=com.

   Esta prueba permite comprobar que la autenticación y las ACLs funcionen correctamente, además de comprobar que el directorio se haya inicializado con la estructura básica.




root@pdc: ~
sambaLogonTime: 0
sambaLogoffTime: 2147483647
sambaKickoffTime: 2147483647
sambaPwdCanChange: 0
displayName: leo
sambaSID: S-1-5-21-3942577662-2197040782-1460581999-3018
sambaPrimaryGroupSID: S-1-5-21-3942577662-2197040782-1460581999-513
sambaHomeDrive: M:
sambaLMPassword: 2F43060FFF05FB50AAD3B435B51404EE
sambaAcctFlags: [U]
sambaNTPassword: ABA45E2342B3277E030CDB4F9748B60E
sambaPwdLastSet: 1337954119
sambaPwdMustChange: 1341842119
userPassword:: e1NTSEF9ODltRlVUSUI4ZnFZZkZ0c1BjekIwM3JUR05acVRITkU=
shadowLastChange: 15485
shadowMax: 45
search result
search: 2
result: 0 Success
numResponses: 21
numEntries: 20
root@pdc:~#

Anexo III Algunos programas asociados a SAMBA

- **smbclient:** Cliente FTP Unix que puede ser usado para conectarse a recursos compartidos por Samba.
- **smbtar:** Programa que permite realizar copias de seguridad de datos.
- **nmblookup:** Programa que proporciona búsquedas de nombres NetBIOS sobre TCP/IP.
- **smbpasswd:** Programa que permite a un administrador cambiar las passwords encriptadas usadas por Samba.
- **smbstatus:** Programa para reportar las conexiones de red actuales hacia los recursos compartidos por el servidor Samba.
- **testparm:** Programa para validar el fichero de configuración de Samba.

Anexo IV Algunas reglas de UFW

- ✓ **Habilitando puertos**

Si el servicio openssh está siendo ejecutado en un determinado servidor y está escuchando peticiones en el puerto 59345 y se desea permitir las conexiones, entonces se creará la regla de la siguiente manera.

```
ufw allow 59345 /tcp

Rule added
```

Si se desea activar el servicio apache.

```
ufw allow http

Rule added
```

Permitiendo la conexión al servidor FTP desde el segmento de red 10.12.16.0.

```
ufw allow from 10.12.16.0/24 proto tcp to any port ftp

Rule added
```

Permitiendo la conexión al servidor por el puerto 22 (ssh) desde la PC con dirección 10.12.16.20.

```
ufw allow proto tcp from 10.12.16.20 to any port 22

Rule added
```

- ✓ **Desactivando puertos.**

Para poder bloquear puertos en el firewall sería de la siguiente manera.

```
ufw deny 4025 /tcp

Rule added
```

- ✓ **Eliminando reglas**

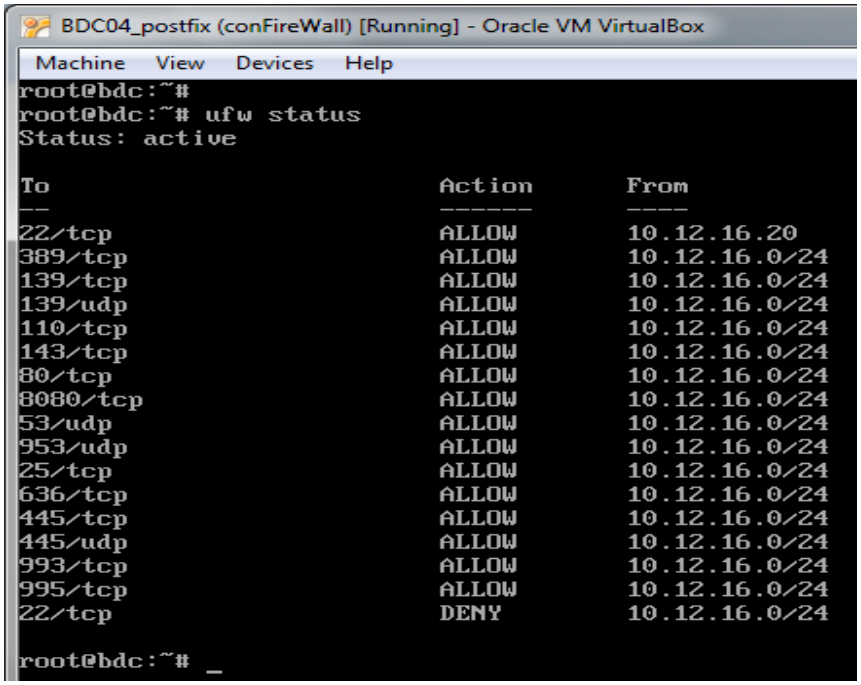
Cuando se tiene alguna regla que permite conectarse a un servicio el cual ya no se está utilizando sería mejor eliminarla para mejor seguridad, por ejemplo:

```
ufw delete allow http

Rule deleted
```

- ✓ **Status UFW**

Para poder ver el estado del firewall se ejecutará:



Como se puede observar son muchas reglas las que han sido creadas.

Anexo V Ejemplos de cómo utilizar UFW

Los siguientes son algunos ejemplos de cómo utilizar UFW:

- ✓ **Primero se debe observar que servicios están escuchando con:**

```
netstat -tanp | grep LISTEN
```

```
tcp    0    0 0.0.0.0:993     0.0.0.0:*     LISTEN   4869/dovecot

tcp    0    0 127.0.0.1:3306  0.0.0.0:*     LISTEN   4698/mysqld

tcp    0    0 0.0.0.0:143     0.0.0.0:*     LISTEN   4869/dovecot

tcp    0    0 0.0.0.0:80      0.0.0.0:*     LISTEN   4945/apache2

tcp    0    0 0.0.0.0:21      0.0.0.0:*     LISTEN   4850/vsftpd

tcp    0    0 10.12.34.201:53 0.0.0.0:*     LISTEN   4578/named

tcp    0    0 127.0.0.1:53    0.0.0.0:*     LISTEN   4578/named

tcp    0    0 0.0.0.0:25      0.0.0.0:*     LISTEN   4833/master

tcp    0    0 0.0.0.0:4025    0.0.0.0:*     LISTEN   4768/partimaged

tcp    0    0 127.0.0.1:953   0.0.0.0:*     LISTEN   4578/named
```

- ✓ **Permitir/bloquear el acceso**

Políticas por defectos

```
ufw default allow

ufw default deny
```

Cuando se crea un firewall se recomienda primero bloquear todos los puertos y conexiones para después abrir solamente los que se requieren.

- ✓ **Para poder iniciar el firewall solamente que hay ejecutar:**

```
ufw enable o /etc/init.d/ufw start
```

- ✓ **Para detener el firewall:**

```
ufw disable o /etc/init.d/ufw stop
```

- ✓ **Activando /desactivando logs**

Para poder activar el log del firewall se debe ejecutar:

```
ufw logging on
```

Para poder desactivarlo:

```
ufw logging off
```

El log del firewall es guardado dentro del archivo /var/log/messages. Pero para poder ver los últimos sucesos del mismo se debe ejecutar:

```
tail -f /var/log/messages
```

Anexo VI Configuración de la aplicación PhpLdapAdmin

Se edita el archivo config.php

```
nano /etc/phpLDAPadmin/config.php
```

Buscar las líneas:

```
$servers->setValue('server','name'.'My LDAP Server');

$servers->setValue('server','host'.'127.0.0.1');

$servers->setValue('server','base',array('dc=example,dc=com'));

$servers->setValue('login','bind_id','cn=admin,dc=example,dc=com');
```

Cambiarlas para reflejar el dominio del servidor

```
$servers->setValue('server','name'.'Navine Server');

$servers->setValue('server','host'.'127.0.0.1');

$servers->setValue('server','base',array('dc=navine,dc=com'));

$servers->setValue('login','bind_id','cn=admin,dc=navine,dc=com');
```

A continuación se reinicia el servicio web con:

```
/etc/init.d/apache2 restart o service apache2 reload
```

GLOSARIO

BIND Acrónimo de *Berkeley Internet Name Domain,* es el servidor de DNS más comúnmente usado en Internet.

BIOS Acrónimo de *Basic Input Output System* (Sistema Básico de Entrada/Salida). Programa residente normalmente en ROM que controla las interacciones básicas entre el hardware y el software.

DNS *Acrónimo de Domain Name System* (Sistema de Nombres de Dominio). Sistema para traducir los nombres de los ordenadores en direcciones IP numéricas.

FQDN Acrónimo de *Fully Qualified Domain Name*, es el nombre completo de un recurso en el dominio.

FTP *Acrónimo de File Transfer Protocol* (protocolo de transferencia de archivos), un protocolo de Internet que permite que un usuario transfiera archivos hacia y desde otros equipos.

HTML *Hyper Text Markup Language* (Lenguaje de Marcado de Hipertexto). Lenguaje en el que se escriben los documentos que se acceden a través de visualizadores WWW. Admite componentes hipertexto y multimedia.

HTTP Acrónimo de *Hipertext Transfer Protocol* (protocolo de transferencia de hipertexto), es el protocolo en que se basa la tecnología de World Wide Web. Http es el conjunto de reglas que gobiernan el software que transporta los documentos HTML a través de Internet.

LAM Acrónimo *de LDAP Account Manager*.

LAN Acrónimo de *Local Area Network* (red de área local), una red que conecta dos o más equipos que están dentro de un área relativamente pequeña, normalmente en el local de una organización, con el propósito de comunicarlos y compartir archivos.

LDA *Local Delivery Agent,* agente Local empleado para Entregar correo desde el MDA al MUA.

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

MIC *Ministry of Information and Communication* (Ministerio de la Informática y de la Comunicación).

MOM *Acrónimo de Manager of Manager*.

MTA *Mail Transport Agent*, es un programa encargado de recoger mensajes y enviarlos, comunicando con otros MTA según sea preciso.

MUA *Mail User Agent*, es un programa que permite leer y escribir correos.

NIS *Network Information Service* protocolo, nombrado originalmente como Páginas Amarillas.

NSS *Acrónimo de Name Server Switch*.

PAM *Acrónimo de Pluggable Authentication Modules*, es un mecanismo de autenticación flexible que permite abstraer las aplicaciones y otro software del proceso de identificación.

POP *Post Office Transport Protocol*, se utiliza para obtener/descargar los mensajes guardados en el servidor al usuario.

SMTP *Simple Mail Transfer Protocol*, es el protocolo principal del MTA.

SSL *Secure Sockets Layer* (Capa de Conexiones Seguras), es un protocolo que hace uso de certificados digitales para establecer comunicaciones seguras a través de Internet.

TCP/IP Acrónimos de Transport Control Protocol and Internet Protcol (protocolo de control de transmisión y protocolo Internet), los dos protocolos que gobiernan la manera en que los equipos y las redes administran el flujo de información que pasa a través de Internet.

TLS *Transport Layer Security*, protocolo basado en SSL.

UNIX Sistema operativo multitarea, multiusuario. Gran parte de las características de otros sistemas más conocidos como MS-DOS están basadas en este sistema muy

extendido para miniordenadores. Internet no se puede comprender en su totalidad sin conocer el UNIX, ya que las comunicaciones con TCP/IP son una parte fundamental de este sistema operativo.

URL/URI *Universal Resource Locator/Universal Resource Identifier* (Localizador Universal de Recursos/Identificador Universal de Recursos). Sistema unificado de identificación de recursos en la red. Las direcciones se componen de protocolo, FQDN y dirección local del documento dentro del servidor. Este tipo de direcciones permite identificar objetos WWW, Gopher, FTP, News, etc.