UNIVERSIDAD NACIONAL DEL ALTIPLANO - PUNO

FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

"DISEÑO E IMPLEMENTACIÓN DE UN SERVIDOR CON EL PROTOCOLO DE ACCESO A DIRECTORIO LDAP PARA LA SEGURIDAD Y CONTROL DE LOS TRABAJADORES EN EL USO DE DISPOSITIVOS Y COMPUTADORAS DE LA EMPRESA CLARO EN LA REGIÓN PUNO"

TESIS

PRESENTADA POR:

EDWIN MARTIN ZAVALAGA CCOSI

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ELECTRÓNICO

PUNO – PERÚ

2018

UNIVERSIDAD NACIONAL DEL ALTIPLANO-PUNO

FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

DISEÑO E IMPLEMENTACIÓN DE UN SERVIDOR CON EL PROTOCOLO DE ACCESO A DIRECTORIO LDAP PARA LA SEGURIDAD Y CONTROL DE LOS TRABAJADORES EN EL USO DE DISPOSITIVOS Y COMPUTADORAS DE LA EMPRESA CLARO EN LA REGIÓN PUNO

TESIS PRESENTADA POR:

EDWIN MARTIN ZAVALAGA CCOSI

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ELECTRÓNICO

FECHA DE SUSTENTACIÓN: 10/12/2018

APROBADO POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE: _____
Mg. LUIS ENRIQUE BACA WIESSE

PRIMER MIEMBRO: _____
M.Sc. EDWIN WILBER CHAMBI MAMANI

SEGUNDO MIEMBRO: _____
M.Sc. JASMANY RUELAS CHAMBI

DIRECTOR/ASESOR: _____
M.Sc. GAVINO JOSÉ FLORES CHIPANA

Área : Telecomunicaciones

Tema : Telecomunicaciones y Redes de Datos

DEDICATORIA

Dedico este trabajo a las personas que siempre me han apoyado en todo momento y no dudo que lo seguirán haciendo por siempre, a las personas que siempre han estado conmigo.

Con todo mi amor:

A MI PADRE Y MADRE:

Victor Albio Zavalaga Gomez y Graciela Ccosi Arpazi

A MIS HERMANAS:

Eliana Elizabeth Zavalaga Ccosi y Edith Zavalaga Ccosi

A MI PAREJA E HIJO:

Sandy Giuliana Rivas Aguilar y Dan Edwin Zavalaga Rivas

La familia que tanto amo.

Edwin Martin Zavalaga Ccosi

AGRADECIMIENTO

Doy gracias a mi padre, madre, hermanas, pareja y a mi hijo que es el motor de mi vida porque sin ellos no hubiera sido posible este objetivo y a todas las personas que me demostraron su cariño y soporte durante este periodo tan importante para mí.

A mi Madre por darme la vida, el amor y el aliento que necesito para seguir adelante.

Tengo mucho que agradecer a tantas personas que no quisiera omitir a nadie. Por eso, de antemano agradezco a todas las que han contribuido a la formación de mi persona como hijo, hermano, amigo, estudiante, alumno, compañero y por consecuencia, como profesional

Agradezco a DIOS por acompañarme siempre en todo lo que hago y permitirme llegar a este dichoso momento

Edwin Martin Zavalaga Ccosi

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

Se desarrolló un sistema de autenticación de usuarios centralizado, un servidor contiene y mantiene los usuarios, sus respectivas contraseñas y otras características, de esta forma el administrador de la red no necesitó configurar usuarios en cada máquina de la organización cada vez que aparece un nuevo equipo o un nuevo usuario, o incluso cada vez que se requiera una modificación. El administrador de la red se concentró en configurar y mantener los usuarios con sus respectivas características en un servidor centralizado, de esta forma las computadoras y dispositivos de la organización en vez de consultar a sus bases de datos locales por un usuario o usuarios, lo hacen al servidor central. Si una organización, por ejemplo, tiene cien computadoras de escritorio distribuidas en toda la infraestructura, el administrador de red deberá crear todos los usuarios actuales en las cien máquinas y cuando se deba ingresar o eliminar un usuario debe hacer ese mantenimiento en todas las máquinas. El servidor está basado en hardware y software, usa LDAP (Lightweight Directory Access Protocol) como el sistema de autenticación centralizado, sobre un sistema operativo Linux Debian o Ubuntu Server. Se debe aclarar que la autenticación que se hizo no es sobre una aplicación, como la de un programa, una página web, o una aplicación móvil. Este tipo de autenticación se realizó al momento que el usuario inició sesión en el sistema operativo. Si un usuario trata de ingresar a un sistema operativo Linux o Windows, las máquinas clientes consultarán al servidor central LDAP a cerca de la existencia del usuario. El objetivo general fue diseñar e implementar el protocolo de acceso a directorio LDAP para mejorar la seguridad y control de los trabajadores de la empresa Claro en la región Puno.

Palabras Clave: acceso a directorio, autenticación de usuarios, directorio ligero, dominio de servicios, protocolo de autenticación.

ABSTRACT

A centralized user authentication system was developed, a server contains and maintains the users, their respective passwords and other features, so the network administrator did not need to configure users on each machine in the organization each time it appears a new team or a new user, or even each time a modification is required. The network administrator concentrated on configuring and maintaining the users with their respective characteristics in a centralized server, in this way the computers and devices of the organization instead of consulting their local databases for a user or users, do so to the central server. If an organization, for example, has a hundred desktop computers distributed throughout the infrastructure, the network administrator must create all the current users on the hundred machines and when a user must be entered or deleted, he must do this maintenance on all the machines. The server is based on hardware and software, uses LDAP (Lightweight Directory Access Protocol) as the centralized authentication system, on a Linux Debian operating system or Ubuntu Server. It must be clarified that the authentication that was made is not about an application, such as that of a program, a web page, or a mobile application. This type of authentication was performed at the time the user logged in to the operating system. If a user tries to enter a Linux or Windows operating system, the client machines will consult the central LDAP server about the existence of the user. The general objective was to design and implement the LDAP directory access protocol to improve the safety and control of Claro company workers in the Puno region.

Keywords: authentication protocol, directory access, domain services, light directory, user authentication.

CAPITULO I

1.1 INTRODUCCIÓN

En la informática de las organizaciones de hoy en día, incluidas las universidades, las pequeñas y las medianas empresas, necesitan proporcionar una amplia gama de servicios a un gran número de usuarios. Muchos de estos servicios requieren una forma de autenticación y/o autorización para verificar de forma segura la identidad de sus respectivos abonados. Los servicios que pueden requerir tal autenticación incluyen clientes de correo electrónico como Zimbra y clientes de terminales remotos como SSH. Un ataque de denegación de servicio en un Servidor de Protocolo de Acceso a Directorio Ligero (servidor LDAP) que quedaba vulnerable podría efectivamente interrumpir la productividad una organización o institución.

Esta investigación pretende afirmar el argumento de que los sistemas de directorio activo como LDAP en sus estados actuales son opciones recomendables como servicios de autenticación para los trabajadores de la empresa Claro Puno. El servicio es instalado en un Ubuntu de la versión estable 16.04 TLS aún vigente hasta Abril del año 2021; para fines de análisis fue necesario instalar el capturador de paquetes en Ubuntu.

Se explica el planteamiento de la investigación en los cuales muestran las secciones como la formulación del problema, justificación del problema, objetivos de la investigación, hipótesis de la investigación. En los objetivos se muestra el objetivo general donde se basa la investigación, también los objetivos específicos para detallar más a profundo los ámbitos de esta investigación.

El marco teórico muestra los antecedentes de la investigación relacionados con esta tesis, posteriormente explica el Active Directory usado por Windows, donde señala como se relaciona los diferentes sub árboles, a continuación esta Novell Directory Services donde se indica su principal creador y los diferentes protocolos que intervienen en este tipo de servidor. Los anteriores tipos de servicios de autenticación son de creadores no relacionados con GNU/Linux, pero los servicios como OpenLDAP, que se estudia en esta investigación, indica que la creación e implementación es usando GNU/Linux, al igual que Red Hat Directory Server, siendo de una partición diferente a OpenLDAP pero del mismo creador general.

El capítulo de la metodología, indica el tipo de investigación, localidad, población y muestra, al igual también el proceso de implementación del servidor LDAP, haciendo la configuración mediante una interfaz gráfica abierta desde un navegador el cual es phpLDAPadmin, luego se muestra los paso que se hicieron para la creación de usuarios en el servidor. En la parte final muestra la configuración para el cliente en Windows con el programa pGina.

En los resultados se analiza tres pruebas realizadas, las cuales fueron; para la autenticación correcta de usuario y de la contraseña, posteriormente para determinar si el servidor cumple con su función de registro si se ingresa una contraseña invalida; y para terminar se realiza la prueba de un usuario incorrecto esto con fines idénticas a la segunda prueba. En este capítulo se presentan cuadros y figuras tomadas en el software Wireshark, también figuras capturadas en el cliente.

Es posible hacer la autenticación de usuarios de clientes Windows 10 al servidor Ubuntu GNU/Linux, de esta forma el diseño e implementación del sistema de autenticación de usuarios en un dominio de servicios de directorio activo a nivel del sistema operativo

Windows se ha logrado con el objetivo de controlar a los trabajadores de la empresa Claro. Sin embargo, para lograr esto fue necesario diseñar e implementar un dominio de servicios de directorio activo con el software phpLDAPadmin que permitió la administración del directorio activo de forma remota por un entorno web.

También fue necesario mejorar el control y monitoreo de usuarios basándose en la autenticación cuando el usuario accede a la máquina. LDAP trabajó confiablemente para las pruebas, es ligero en términos de uso de recursos tanto en maestro como en réplica, es altamente configurable, en desarrollo activo, razonablemente bien documentado y está bien soportado en la lista de OpenLDAP. Si estuviéramos procesando un gran número de actualizaciones, sin embargo, nuestro maestro LDAP normalmente recibe entre 50 y 100 cambios por hora, lo cual es bastante mínimo y no genera tráfico maestro-réplica significativo para propósitos de sincronización.

De esta forma se ha logrado controlar el acceso de los trabajadores (usuarios) a las máquinas usando su respectivo usuario y contraseña; así se tiene un registro de la dirección de las que acceden los usuarios, la fecha y la hora. Cuando un usuario ingrese a un sitio indebido, el servidor afectado registrará su dirección IP y LDAP registrará qué usuario usó esa dirección IP, sin el uso de LDAP solamente se podría saber la dirección IP de la máquina y no se tendría conocimiento del usuario que actuó en ella.

1.2 EL PROBLEMA, OBJETIVOS E HIPÓTESIS

1.2.1 PROBLEMA DE INVESTIGACIÓN

1.2.1.1 ANÁLISIS DE LA SITUACIÓN PROBLEMÁTICA

Los trabajadores de la empresa claro en la región puno tienen acceso a diferentes dispositivos de red en diferentes tipos de plataformas, sin embargo, no existe el control

centralizado de usuarios y contraseñas, es por ello que cualquier usuario puede acceder a dispositivos de red y no se registra el control. En cuanto al uso de aplicaciones web, programas y otro tipo de aplicaciones móviles, se puede registrar el acceso de usuarios sin ningún problema, pero en cuanto al acceso de usuarios a sistemas operativos el acceso no tiene ningún tipo de control o registro. Por ejemplo, si un usuario ingresa a una aplicación por página web o a de algún programa, existe un servidor que registra el acceso del usuario a esa página web o a ese programa, sin embargo no hubo registro del acceso al sistema operativo, del cual podría anónimamente de hacer uso de muchas aplicaciones y programas no convenientes o maliciosos.

1.2.1.2 DEFINICIÓN DEL PROBLEMA

Según el análisis de la problemática se ha identificado los siguientes problemas:

- No existe un control de usuarios en el acceso a máquinas o dispositivos en el nivel de sistema operativo.
- La baja seguridad existente en la red de área local ante ataques informáticos y malware.

1.3 PLANTEAMIENTO DEL PROBLEMA

1.3.1 PROBLEMA GENERAL

¿El diseño e implementación protocolo de acceso a directorio LDAP podrá mejorar la seguridad y control de los trabajadores de la empresa Claro en la región Puno?

1.3.2 PROBLEMAS ESPECÍFICOS

- ¿Cómo diseñar e implementar protocolo de acceso a directorio LDAP?
- ¿Cómo mejorar la seguridad y control de la red organizacional?

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

Diseñar e implementar el protocolo de acceso a directorio LDAP para mejorar la seguridad y control de los trabajadores de la empresa Claro en la región Puno.

1.4.2 OBJETIVOS ESPECÍFICOS

- Diseñar el protocolo de acceso a directorio LDAP, permitiendo la autenticación de los usuarios a lo largo de la organización.
- Implementar el protocolo de acceso a directorio LDAP, para mejorar la seguridad y control de la red organizacional.

1.5 HIPÓTESIS

1.5.1 HIPÓTESIS GENERAL

El diseño e implementación del protocolo de acceso a directorio LDAP mejorará la seguridad y control de los trabajadores en el uso de dispositivos y computadoras de la empresa Claro en la región Puno.

1.5.2 HIPÓTESIS ESPECÍFICAS

- Se diseñar e implementará protocolo de acceso a directorio LDAP.
- Se mejorará la seguridad y control de la red organizacional.

CAPITULO II

REVISIÓN DE LA LITERATURA

2.1 ACTIVE DIRECTORY

Active Directory es el nombre utilizado por Microsoft (desde Windows 2000) como almacén centralizado de información de uno de sus dominios de administración. Bajo este nombre se encuentra realmente un esquema (definición de los campos que pueden ser consultados) LDAP versión 3 lo que permite integrar otros sistemas que soporten el protocolo. En este LDAP se almacena información de usuarios, recursos de la red, políticas de seguridad, configuración, asignación de permisos, etc. (Arnaert, 2016).

Active Directory es un repositorio común para obtener información sobre los objetos que residen en la red, como usuarios, grupos, computadoras, impresoras, aplicaciones y archivos. El esquema predeterminado de Active Directory admite numerosos atributos para cada clase de objeto que se puede utilizar para almacenar una variedad de información. Las listas de control de acceso (ACL) también se almacenan con cada objeto, lo que le permite mantener permisos para quién puede acceder y administrar el objeto. Tener una sola fuente para esta información la hace más accesible y más fácil de manejar; Sin embargo, lograr esto requiere una cantidad significativa de conocimiento sobre temas tales como el LDAP (Lightweight Directory Access Protocol), los Kerberos, el DNS (Domain Name System), la replicación de múltiples masters, las políticas de grupo y el particionamiento de datos, por nombrar algunos. (Negus, 2015)

2.2 EVOLUCIÓN DE MICROSOFT NOS

El sistema operativo de red, o "NOS", es el término utilizado para describir un entorno en red en el que varios tipos de recursos, como las cuentas de usuario, de grupo y de computadora, se almacenan en un repositorio central controlado por administradores y accesible A los usuarios finales. Normalmente, un entorno NOS está compuesto por uno o más servidores que proporcionan servicios NOS, como autenticación, autorización y manipulación de cuentas, y múltiples usuarios finales que acceden a dichos servicios.

El primer entorno integrado NOS de Microsoft se hizo disponible en 1990 con el lanzamiento de Windows NT 3.0, que combinaba muchas características de los protocolos LAN Manager y del sistema operativo OS/2. La NT NOS evolucionó lentamente durante los próximos ocho años hasta que Active Directory fue lanzado por primera vez en forma beta en 1997.

En Windows NT, se introdujo el concepto de "dominio", que proporciona una forma de agrupar recursos basados en límites administrativos y de seguridad. Los dominios NT eran estructuras planas limitadas a unos 40.000 objetos (usuarios, grupos y computadoras). Para las grandes organizaciones, esta limitación imponía límites superficiales en el diseño de la estructura principal. A menudo, los dominios también estaban geográficamente limitados debido a que la replicación de datos entre los controladores de dominio (es decir, los servidores que proporcionaban los servicios NOS a los usuarios finales) tuvo un rendimiento bajo sobre los enlaces de alta latencia o bajo ancho de banda. Otro problema significativo con el NT NOS fue la delegación de administración, que típicamente tendía a ser una cuestión de todo o nada en el nivel de dominio. Microsoft era muy consciente de estas limitaciones y la necesidad de reestructurar su modelo NOS en algo que sería mucho más escalable y flexible. Parecía

que los servicios de directorio basados en LDAP eran una posible solución. (Nutter, 2014)

2.3 HISTORIA DE LOS DIRECTORIOS

En términos generales, un servicio de directorio es un repositorio de información de red, aplicación o NOS que es útil para múltiples aplicaciones o usuarios. Bajo esta definición, él NOS de Windows NT es un tipo de servicio de directorio. De hecho, hay muchos tipos diferentes de directorios, incluyendo páginas blancas de Internet, sistemas de correo electrónico e incluso el Sistema de nombres de dominio (DNS). Aunque cada uno de estos sistemas tiene características de un servicio de directorio, X.500 y Lightweight Directory Access Protocol (LDAP) definen los estándares de cómo se implementa y se accede a un servicio de directorio verdadero.

En 1988, la Unión Internacional de Telecomunicaciones (UIT) y la Organización Internacional de Normalización (ISO) se unieron para desarrollar una serie de normas sobre los servicios de directorio, que se conoce como X.500. Aunque X.500 resultó ser un buen modelo para estructurar un directorio y proporcionó mucha funcionalidad en torno a operaciones avanzadas y seguridad, fue difícil implementar clientes que pudieran utilizarlo. Una de las razones es que X.500 se basa en la pila de protocolo de interconexión de sistema abierto (OSI) en lugar de TCP/IP, que se había convertido en el estándar para Internet.

El protocolo de acceso a directorios (DAP) era muy complejo e implementaba muchas funciones que la mayoría de los clientes nunca necesitaban. Esto evitó la adopción a gran escala. Fue por esta razón que un grupo encabezado por la Universidad

de Michigan comenzó a trabajar en un protocolo de acceso que haría X.500 más fácil de utilizar.

La primera versión del protocolo ligero del acceso del directorio (LDAP) fue lanzada en 1993 como petición para los comentarios (RFC) 1487, pero debido a la ausencia de muchas características proporcionadas por X.500, nunca despegó realmente. No fue hasta LDAPv2 fue lanzado en 1995 como RFC 1777 que LDAP comenzó a ganar popularidad. Antes de LDAPv2, el uso principal de LDAP era como una pasarela entre servidores X.500. Los clientes simplificados interactuarían con la puerta de enlace LDAP, que traduciría las solicitudes y las enviaría al servidor X. 500. El equipo de la Universidad de Michigan pensó que si LDAP podía proporcionar la mayor parte de la funcionalidad necesaria para la mayoría de los clientes, podría eliminar el intermediario (la puerta de enlace) y desarrollar un servidor de directorio habilitado para LDAP. Este servidor de directorio podría utilizar muchos de los conceptos de X.500, incluyendo el modelo de datos, pero dejaría fuera toda la sobrecarga resultante de las numerosas características que implementó. Así, el primer servidor de directorio LDAP fue lanzado a finales de 1995 por el equipo de la Universidad de Michigan, y se convirtió en la base de muchos futuros servidores de directorio.

En 1997, la última actualización importante de la especificación LDAP, LDAPv3, se describió en RFC 2251. Proporcionó varias nuevas características y LDAP hizo lo suficientemente robusto y extensible suficiente para ser adecuado para la mayoría de los proveedores de implementar. Desde entonces, compañías como Netscape, Sun, Novell, IBM, la Fundación OpenLDAP y Microsoft han desarrollado servidores de directorio basados en LDAP. Más recientemente, se publicó RFC 3377, que enumera todas las principales RFC LDAP. (Desmond, 2013)

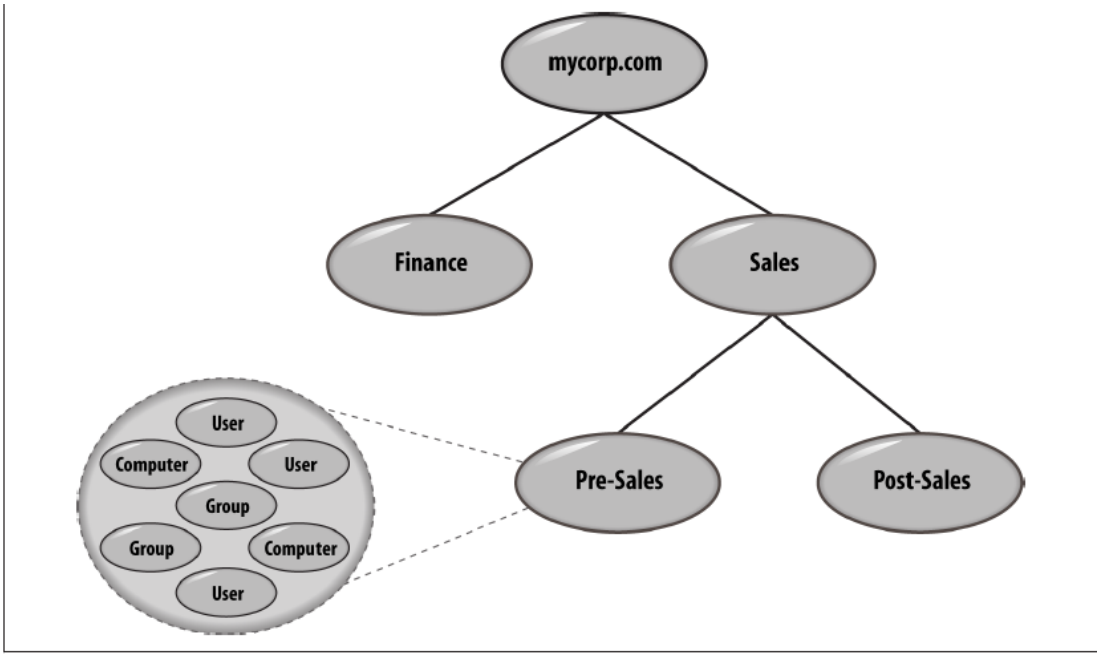
2.4 CÓMO SE ALMACENAN Y SE IDENTIFICAN LOS OBJETOS

Los datos almacenados en Active Directory se presentan al usuario de una manera jerárquica similar a la forma en que se almacenan los datos en un sistema de archivos. Cada entrada se conoce como un objeto. A nivel estructural, existen dos tipos de objetos: contenedores y no contenedores. Los objetos sin contenedor también se conocen como nodos de hoja. Uno o más contenedores se ramifican de una manera jerárquica a partir de un contenedor de raíces. Cada contenedor puede contener nodos de hojas u otros recipientes. Como su nombre indica, sin embargo, un nodo hoja no puede contener ningún otro objeto.

Aunque los datos en Active Directory se presentan de forma jerárquica, en realidad se almacenan en filas y columnas planas de la base de datos. El archivo de árbol de información de directorio (DIT) es un archivo de base de datos de ESE (Extensible Storage Engine). Esto responde a la pregunta "¿Utiliza Active Directory la tecnología de base de datos JET o ESE?" ESE es una tecnología JET.

Considere las relaciones padre-hijo de los contenedores y las hojas en la Figura 2.1. La raíz de este árbol tiene dos hijos, Finance y Sales. Ambos son contenedores de otros objetos. Las Sales tienen dos hijos propios, Pre-sales y Post-sales. Sólo se muestra el contenedor de Pre-sales como conteniendo objetos secundarios adicionales. El contenedor Pre-sales contiene objetos de usuario, grupo y equipo como un ejemplo. Se dice que cada uno de estos nodos hijo tiene el contenedor Pre-sales como su padre. La Figura 1 representa lo que se conoce en Active Directory como un dominio. (Lammle, 2016)

Figura 2.1 - Domino de Active Directory



Fuente: Desmond, B., Richards, J., Allen, R., & Lowe-Norris, A. (2013). *Active Directory*

2.5 IDENTIFICAR OBJETOS DE FORMA EXCLUSIVA

Cuando está almacenando potencialmente millones de objetos en Active Directory, cada objeto debe identificarse de forma única. Para ello, los objetos tienen un identificador único global (GUID) asignado a ellos por el sistema en la creación. Este número de 128 bits es la implementación de Microsoft del concepto de identificador universalmente único (UUID) de Digital Equipment Corporation. UUIDs/GUIDs son comúnmente malentendidos que se garantiza que sean únicos. Este no es el caso; El número es sólo estadísticamente improbable para duplicarse antes del año 3400 dC. En la documentación de la función de API de creación de GUID, Microsoft dice: "En un grado muy alto de certeza, esta función devuelve un valor único". El GUID del objeto permanece con el objeto hasta que se elimina, independientemente de si se cambia el nombre o se mueve Dentro del árbol de información del directorio (DIT). El GUID del

objeto también se conservará si mueve un objeto entre dominios dentro de un bosque multidominio.

Un movimiento entre bosques de un principal de seguridad utilizando una herramienta como la Herramienta de migración de Microsoft Active Directory (ADMT) no conservará GUID del objeto.

Aunque GUID de un objeto es resistente, no es muy fácil de recordar, ni se basa en la jerarquía de directorios. Por este motivo, se utiliza más comúnmente otra forma de referenciar objetos, denominada DN (nombre distinguido). (Desmond, 2013)

2.6 NOMBRES DISTINGUIDOS

Las rutas jerárquicas en Active Directory se conocen como nombres distinguidos y se pueden utilizar para referenciar un objeto de forma exclusiva. Los nombres distinguidos se definen en el estándar LDAP como un medio para referirse a cualquier objeto del directorio.

Los nombres distinguidos para los objetos de Active Directory se representan normalmente utilizando la sintaxis y las reglas definidas en los estándares LDAP. Echemos un vistazo a cómo una ruta a la raíz de la Figura 2.1 Mira:

dc = mycorp , dc = com

En el nombre distintivo anterior, representa la raíz del dominio, *mycorp .com*, separando cada parte con una coma y prefijando cada parte con las letras "*dc*". Si el dominio se llamara *mydomain.mycorp .com*, el nombre distinguido de la raíz se parecería a esto:

dc = mydomain, dc = mycorp , dc = com

Un nombre distinguido relativo (RDN) es el nombre utilizado para hacer referencia única a un objeto dentro de su contenedor principal en el directorio. Por ejemplo, este es el DN de la cuenta de administrador predeterminada en el contenedor usuarios en el dominio mycorp .com

cn=Administrator,cn=Users,dc=mycorp ,dc=com

Este es el RDN del usuario:

cn=Administrator

Los RDN deben ser siempre únicos dentro del contenedor en el que existen. Es permisible tener dos objetos con *cn=Administrator* en el directorio; Sin embargo, deben estar ubicados dentro de diferentes contenedores parentales. No puede haber dos objetos con un RDN de *cn=Administrator* en el contenedor usuarios.

Los DN se componen de nombres y prefijos separados por el signo igual (=). Otro prefijo que le será muy familiar es *ou*, que significa unidad organizativa. Aquí hay un ejemplo:

Cn = Keith Cooper, ou = Northlight IT Ltd, dc = mycorp , dc = com

Todos los RDN utilizan un prefijo para indicar la clase del objeto al que se hace referencia. Cualquier clase de objeto que no tenga un código de letra específico utiliza el valor predeterminado de *cn*, que significa nombre común. Tabla 2.1 Proporciona una lista completa de los tipos de atributos más comunes entre las implementaciones del servidor de directorios. La lista proviene del RFC 2253, "Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names". (Desmond, 2013)

Tabla 2.2 - Tipos de atributo de RFC 2253

Key	Attribute
CN	Common name
L	Locality name
ST	State or province name
O	Organization name
OU	Organizational unit name
C	Country name
STREET	Street address
DC	Domain component
UID	User ID

Fuente: Desmond, B., Richards, J., Allen, R., & Lowe-Norris, A. (2013). *Active Directory*

2.7 BLOQUES DE CONSTRUCCIÓN

Ahora que hemos mostrado cómo se estructuran y se hace referencia a los objetos, veamos los conceptos básicos detrás de Active Directory.

2.8 DOMINIOS Y ÁRBOLES DE DOMINIOS

La estructura lógica de Active Directory se basa en el concepto de dominios. Los dominios se introdujeron en Windows NT 3.x y 4.0. Sin embargo, en Active Directory, los dominios se han actualizado significativamente de la estructura plana e inflexible impuesta por Windows NT. Un dominio de Active Directory se compone de los siguientes componentes:

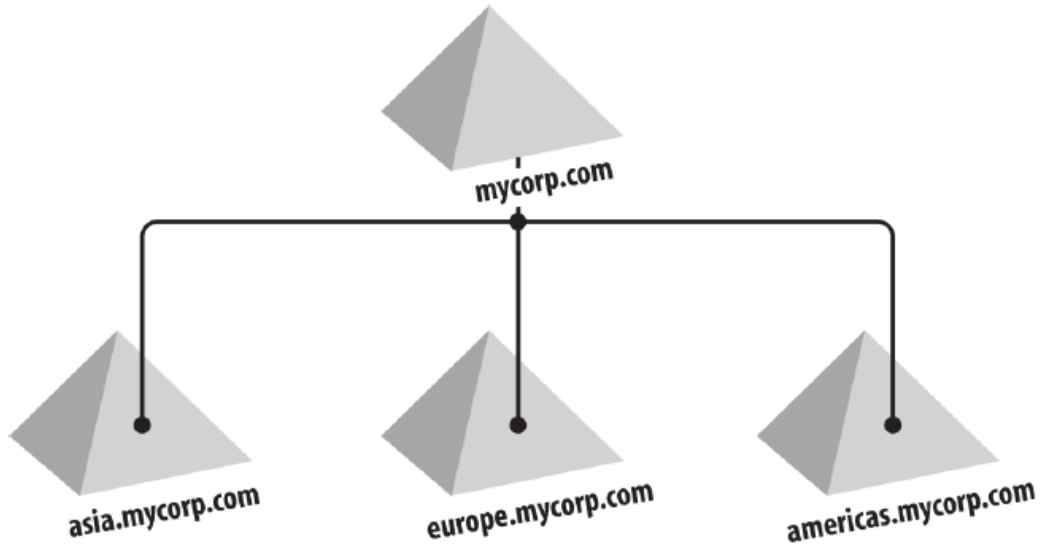
- Una estructura jerárquica basada en X.500 de contenedores y objetos.

- Un nombre de dominio DNS como identificador único.
- Un servicio de seguridad, que autentica y autoriza cualquier acceso a recursos a través de cuentas en el dominio o confía en otros dominios.
- Políticas que determinan cómo la funcionalidad está restringida para usuarios o máquinas dentro de ese dominio.

Un controlador de dominio (DC) puede ser autoritario para uno y sólo un dominio. No es posible alojar múltiples dominios en un solo DC. Por ejemplo, *mycorp* ya ha sido asignado un nombre de dominio DNS para su empresa llamada *mycorp .com*, por lo que decide que el primer dominio de Active Directory que va a construir debe ser llamado *mycorp .com* .Sin embargo, este es sólo el primer dominio de una serie que puede ser necesario crear, Y *mycorp .com* es de hecho la raíz de un árbol de dominio.

El propio dominio *mycorp .com*, ignorando su contenido, se crea automáticamente como el nodo raíz de una estructura jerárquica denominada árbol de dominio. Esto es literalmente una serie de dominios conectados entre sí de una manera jerárquica, todos utilizando un esquema de nombres contiguos. Si *mycorp* agregara dominios llamados Europa, Asia y América, entonces los nombres serían *europe.mycorp*.com, *asia.mycorp.com*, y *americas.mycorp.com* .Cada árbol de dominio se llama por el nombre dado a la raíz del árbol; Por lo tanto, este árbol de dominio se conoce como el árbol *mycorp.com*, como se ilustra en la Figura 2.2 .Puede ver que en la configuración de *mycorp* ahora tenemos un conjunto contiguo de dominios que encajan en un árbol limpio. Incluso si tuviéramos sólo un dominio, seguiría siendo un árbol de dominio, aunque con sólo un dominio. (Desmond, 2013)

Figura 2.2 - El árbol del dominio mycorp .com



Fuente: Desmond, B., Richards, J., Allen, R., & Lowe-Norris, A. (2013). *Active Directory*

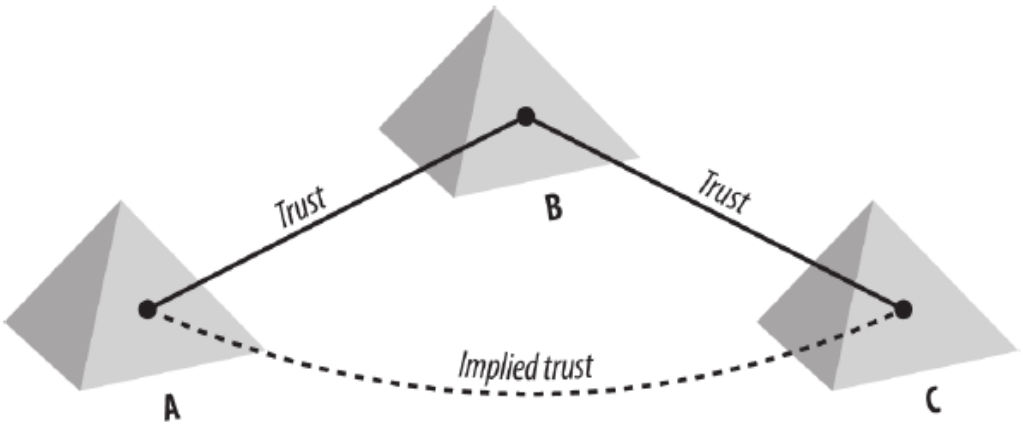
Los árboles facilitan la gestión y el acceso a los recursos, ya que todos los dominios de un árbol de dominio confían mutuamente. Si el dominio A confía en el dominio B y el dominio B confía en el dominio C, esto implica que el dominio A confía en el dominio C también. Ponga mucho más simplemente, el administrador de *asia.mycorp.com* puede permitir que cualquier usuario en el árbol tenga acceso a cualesquiera de los recursos en el dominio de Asia que el administrador desee. El usuario que accede al recurso no tiene que estar en el mismo dominio. (Ferguson, 2016)

2.9 BOSQUES

Cuando un árbol de dominio era una colección de dominios, un bosque es una colección de uno o más árboles de dominio. Estos árboles de dominio comparten un contenedor de *Schema* y *Configuration* común y los árboles en su conjunto están conectados entre sí a través de confianzas transitivas. Si agrega cualquier dominio al árbol de dominio inicial o agrega nuevos árboles de dominio, todavía tiene un bosque.

A medida que continuamos con *mycorp*, encontramos que tiene un negocio subsidiario llamado Othercorp. El nombre de dominio DNS asignado y utilizado por *othercorp* es othercorp.com .En el caso de *Othercorp*, todo lo que tendría que hacer es crear la raíz del árbol de othercorp.com como un miembro del bosque existente; *othercorp.com* y *mycorp.com* pueden entonces existir juntos y compartir recursos, como se muestra en la Figura 2.3. El bosque que contiene los árboles de dominio *mycorp.com* y *othercorp.com* es conocido como el bosque *mycorp.com*, en el que *mycorp.com* es el dominio raíz del bosque.

Figura 2.3 - Trusts transitivos



Fuente: Desmond, B., Richards, J., Allen, R., & Lowe-Norris, A. (2013). *Active Directory*

Un fideicomiso de bosque permite que un administrador cree una única confianza transitiva unidireccional o bidireccional entre dos dominios de raíz de bosque. Esta confianza permite que todos los dominios de un bosque confíen en todos los dominios de otro bosque y viceversa. (Desmond, 2013)

2.10 UNIDADES ORGANIZACIONALES

Habiendo cubierto la vista a gran escala (dominios, árboles y bosques) de Active Directory, hablaremos ahora de la pequeña escala. Cuando mire dentro de un dominio de

Active Directory, verá una estructura jerárquica de objetos. Esta jerarquía está formada por objetos que pueden actuar como contenedores y objetos que no pueden. El tipo primario de contenedor que creará para alojar objetos se denomina unidad organizativa (OU). Otro tipo de contenedor, que en realidad se denomina contenedor, también se puede usar para almacenar una jerarquía de objetos y contenedores.

Aunque ambos pueden contener enormes jerarquías de contenedores y objetos, una unidad organizativa puede tener políticas de grupo aplicadas a ella. Por esta razón, las UO se utilizan a menudo casi exclusivamente para la construcción de objetos Jerarquías dentro de un dominio.

2.11 EL CATÁLOGO GLOBAL

El catálogo global (GC) es una parte muy importante de Active Directory porque se utiliza para realizar búsquedas en todo el bosque. Como su nombre lo indica, el Catálogo Global es un catálogo de todos los objetos de un bosque que contiene un subconjunto de atributos para cada objeto. El GC se puede acceder a través de LDAP a través del puerto 3268 o LDAP/SSL a través del puerto 3269. El catálogo global es de sólo lectura y no se puede actualizar directamente.

En bosques de multidominio, por lo general primero debe realizar una consulta en contra de la GC para localizar los objetos de interés. A continuación, puede realizar una consulta más dirigida contra un controlador de dominio para el dominio en el que se encuentra el objeto si desea tener acceso a todos los atributos disponibles en el objeto.

2.12 FUNCIONES FLEXIBLES DEL OPERADOR MAESTRO ÚNICO (FSMO)

Aunque Active Directory es un directorio multimaster, hay algunas situaciones en las que sólo debe haber un solo controlador de dominio que pueda realizar ciertas funciones .En estos casos, Active Directory nombra un servidor para actuar como el maestro para esas funciones. Hay cinco funciones de este tipo que deben tener lugar en un solo servidor. El servidor que es el maestro para una función o función particular es conocido como el Operador maestro único (FSMO, pronunciado "fizmo").

De los cinco roles, tres existen para cada dominio, y dos se aplican a todo el bosque. Si hay cuatro dominios en su bosque, habrá 14 funciones de FSMO:

- 2 FSMO de todo el bosque
- 4 conjuntos de 3 FSMOs a nivel de dominio

El número de diferentes propietarios de roles puede variar considerablemente dependiendo de si tiene controladores de dominio que cumplen funciones múltiples, como suele suceder. Las diferentes funciones de FSMO son las siguientes:

- Maestro de esquema (forest-wide) El propietario de la función de maestro de esquema es el controlador de dominio que está autorizado a realizar actualizaciones en el esquema. Ningún otro servidor puede procesar cambios en el esquema. Si intenta actualizar el esquema en un controlador de dominio que no contiene el FSMO maestro de esquema, el DC devolverá una referencia al titular de función de maestro de esquema. El propietario de la función maestro de esquema predeterminado es el primer servidor que se promociona a un controlador de dominio en el bosque.

- Maestro de nombres de dominio (en todo el bosque) El propietario de la función maestra de nomenclatura de dominio es el servidor que controla los cambios en el espacio de nombres de todo el bosque. Este servidor agrega y elimina dominios y es necesario para cambiar el nombre o mover dominios dentro de un bosque, así como para autorizar la creación de particiones de aplicaciones y la adición/eliminación de sus réplicas. Al igual que el maestro de esquema, este propietario de rol predeterminado es el primer DC que promueve en un bosque.
- Emulador PDC (todo el dominio) Para fines de compatibilidad con versiones anteriores, un DC de Active Directory debe actuar como el controlador de dominio principal (PDC) de Windows NT .Este servidor actúa como el explorador maestro de Windows NT y también actúa como el PDC para clientes de nivel inferior. Aunque el PDC tiene funciones legadas muy importantes, no se deje engañar pensando que ya no es importante una vez que haya eliminado todos los clientes antiguos.
- El emulador PDC también tiene otras funciones importantes: por ejemplo, intenta mantener la última contraseña para cualquier cuenta. Esto se impone al hacer que los demás DC envíen inmediatamente cualquier cambio de contraseña de cuenta directamente al PDC. El PDC también es el servidor de destino de la mayoría de las herramientas de administración de directiva de grupo. Esto se hace para disminuir la posibilidad de que la misma política sea modificada de diferentes maneras por diferentes administradores en diferentes DCs al mismo tiempo.
- Maestro RID (en todo el dominio) Existe un maestro de identificador relativo (RID) por dominio. Cada principal de seguridad en un dominio tiene un identificador de seguridad (SID) que está compuesto de varios componentes, incluido un RID. El sistema utiliza el SID para identificar de forma exclusiva ese objeto para los permisos de seguridad. En cierto modo, esto es similar al GUID que cada objeto tiene, pero el SID

sólo se da a objetos habilitados para seguridad y se utiliza sólo para propósitos de verificación de seguridad. En un dominio, los SID deben ser únicos en todo el dominio. Como cada DC puede crear objetos habilitados para la seguridad, debe existir algún mecanismo para que nunca se creen dos SID idénticos. Para evitar que se produzcan conflictos, el maestro RID mantiene un grupo grande de valores RID únicos. Cuando se agrega un DC a la red, se le asigna un subconjunto de 500 valores del grupo RID para su propio uso. Siempre que un DC necesita crear un SID, toma el siguiente valor disponible de su propio grupo RID para crear el SID con un valor único.

- Infraestructura maestra (en todo el dominio) El maestro de infraestructura se utiliza para mantener referencias a objetos en otros dominios, conocidos como fantasmas. El maestro de infraestructura también es responsable de arreglar las referencias obsoletas de objetos de su dominio a objetos de otros dominios ("obsoleto" significa referencias a objetos que han sido movidos o cambiados de nombre para que la copia local del nombre del objeto remoto esté obsoleta). El maestro de Infraestructura escribe las actualizaciones que encuentre en sus objetos y, a continuación, replica la información actualizada alrededor de otros DC del dominio.

La colocación del maestro de infraestructura y si puede ser colocado en un catálogo global sin causar problemas es a menudo una fuente de gran confusión. La Tabla 2.2 proporciona Una matriz de permutaciones permitidas para los bosques donde La Papelera de reciclaje del directorio no está habilitada. (Negus, 2015)

Tabla 2.3 - Reglas de ubicación maestra de infraestructura

	Single-domain forest	**Multiple-domain forest**	
		All domain controllers are GCs	All domain controllers are *not* GCs
Infrastructure master relevant	No	No	Yes
Infrastructure master permitted on GC	Yes	Yes	No

Fuente: Desmond, B., Richards, J., Allen, R., & Lowe-Norris, A. (2013). *Active Directory*

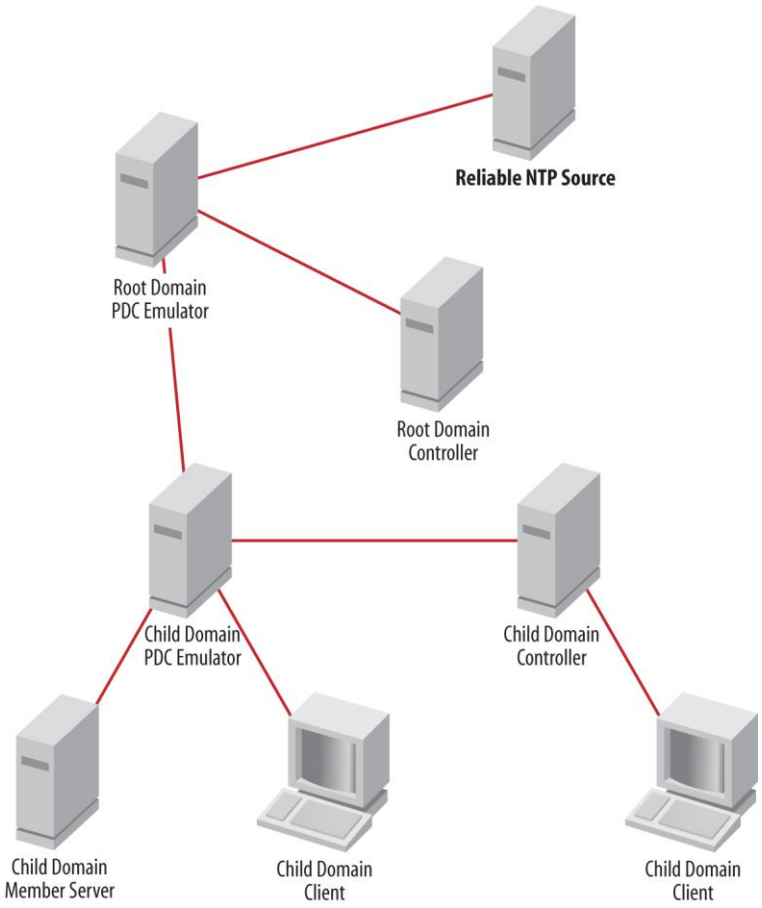
2.13 SINCRONIZACIÓN DE TIEMPO EN ACTIVE DIRECTORY

Active Directory depende en gran medida de todos los controladores de dominio y miembros de dominio que tienen relojes sincronizados. Kerberos (que es el protocolo de autenticación subyacente para los clientes de Active Directory) utiliza relojes del sistema para verificar la autenticidad de los paquetes Kerberos. De forma predeterminada, Active Directory admite una tolerancia de más o menos cinco minutos para los relojes. Si la diferencia de tiempo excede este valor, es posible que los clientes no puedan autenticarse y, en el caso de los controladores de dominio, no se produzca la replicación. (Ferguson, 2016)

Afortunadamente, Active Directory y Windows implementan colectivamente un sistema de sincronización de tiempo basado en el Network Time Protocol (NTP) que asegura que cada máquina en el bosque tenga un reloj sincronizado. El servicio w32time implementa sincronización de tiempo en todas las máquinas con Windows 2000 o más nuevas del bosque. La jerarquía de sincronización de tiempo se describe en la lista siguiente y en la Figura 2.4 se muestra una posible jerarquía efectiva:

- El emulador PDC del dominio raíz del bosque sincroniza su reloj con una fuente de tiempo externa confiable (como un reloj de hardware, una fuente gubernamental u otro servidor NTP confiable).
- El emulador PDC de cada dominio secundario sincroniza su reloj con una fuente de hora confiable en su dominio o en el dominio padre.
- Cada controlador de dominio sincroniza su reloj con el emulador PDC de su dominio o el dominio primario.
- Cada miembro del dominio sincroniza su reloj con el controlador de dominio al que se autentica.

Figura 2.4 - Jerarquía de sincronización de tiempo



Fuente: Desmond, B., Richards, J., Allen, R., & Lowe-Norris, A. (2013). *Active Directory*

2.14 GRUPOS

Active Directory admite tres ámbitos de grupo: dominio local, dominio global y universal. Los grupos en cada uno de estos ámbitos se comportan de forma ligeramente diferente en función del dominio Y los niveles funcionales de los bosques. Para complicar aún más las cosas, cada ámbito de grupo puede tener dos tipos: distribución y seguridad.

Los grupos de distribución se utilizan generalmente como una lista de mensajes (un conjunto de usuarios que puede enviar o enviar mensajes instantáneos a todos a la vez), aunque es posible utilizarlos para grupos de seguridad para aplicaciones basadas en

LDAP o para otras aplicaciones que no utilizan el modelo de seguridad estándar de Windows.Microsoft Exchange representa listas de distribución con grupos de distribución de Active Directory. Los grupos de seguridad, por el contrario, se enumeran durante el inicio de sesión y los SID de cualquier grupo del que el usuario es miembro se agregan al token de seguridad del usuario. Los grupos de seguridad también pueden ser aprovechados por Exchange como listas de distribución.

También conocido como eDirectory es la implementación de Novell utilizada para manejar el acceso a recursos en diferentes servidores y computadoras de una red. Básicamente está compuesto por una base de datos jerárquica y orientada a objetos, que representa cada servidor, computadora, impresora, servicio, personas, etc. entre los cuales se crean permisos para el control de acceso, por medio de herencia. La ventaja de esta implementación es que corre en diversas plataformas, por lo que puede adaptarse fácilmente a entornos que utilicen más de un sistema operativo. (Binnie, 2016)

2.15 NOVELL DIRECTORY SERVICES

2.15.1 ESQUEMA

El esquema NDS consiste en el conjunto de normas que regulan la estructura del árbol de directorios. En él se definen los objetos que pueden existir en el árbol, incluyendo cómo pueden construirse las entradas, que los valores de atributos están permitidos, cómo los Nombres Distinguidos pueden ser construidos, y otras características de utilidad para el propio directorio. Estas reglas de objetos y de atributos se especifican a través de un diccionario de datos que proporciona un conjunto estándar de los tipos de datos a partir de la que los objetos pueden ser creados. Cada objeto en el directorio pertenece a una clase de objeto que especifica los atributos que se pueden

asociar con el objeto. Todos los atributos se basan en un conjunto de tipos de atributos estándar, que a su vez se basan en la sintaxis de atributos estándar.

El conjunto de reglas que controla la creación de un tipo de objeto particular se llama una clase de objeto. Cada clase de objeto se define en términos de atributos o propiedades. Un atributo es una pieza específica de información que puede existir para un objeto. Los atributos, a su vez, se definen en términos de un conjunto base de los tipos de datos denominados *Atributo Sintaxis*. Las sintaxis de atributos definen los tipos de datos primarios para los valores almacenados en el directorio. Por lo tanto el esquema dicta los requisitos, límites y relaciones de objetos y atributos de los objetos que pueden ser creados y utilizados en el árbol de directorios. (Greenblatt, B., 2002)

2.15.2 ESQUEMA DINÁMICO

El esquema es extensible y dinámico en el que los administradores pueden definir nuevas clases y atributos de objetos, además de los proporcionados por el esquema de base. Ampliación del esquema se lleva a cabo a través de la modificación o creación de nuevas definiciones de objeto (atributo o un objeto de clase) y luego la adición de estas nuevas definiciones al esquema de base.

Debido a que el esquema se ha movido en el espacio de objetos jerárquico regular en el árbol de información de directorios (DIT), objetos de esquema general se nombran y se comportan como objetos normales NDS en instanciación, sincronización y modificación. Además, las funciones normales NDS en el DSAPIs se pueden utilizar para manipular definiciones de esquema (crear, ver, modificar, etc.).

Una utilidad de mantenimiento del esquema, Administrador de esquema, permite a los administradores modificar fácilmente el esquema de funcionamiento de

NDS. El Administrador de esquema permite a los administradores ver, ampliar, modificar, imprimir, comparar y diagnosticar su esquema NDS. También proporciona soporte para los desarrolladores y proveedores de software independientes que necesitan para ampliar el esquema para sus aplicaciones. (Greenblatt, B., 2002)

2.15.3 ESQUEMA GLOBAL

La liberación básica NDS se limita a una única partición en el árbol NDS, por tanto, el esquema es global para ese árbol por defecto.

2.15.4 NOMBRAR

El servicio de nombres NDS mapas de nombres de red a direcciones. Se trata de una base de datos global de información orientado a objetos que utiliza un espacio de nombres jerárquico en vez de un espacio de nombres plana. Esta jerarquía permite a la base de datos para ser mapeada como un árbol que puede ser dividido por sus subárboles. Debido a que los nombres de objetos contienen la información de jerarquía, los usuarios pueden tener acceso a recursos de red a nivel mundial, y los administradores pueden administrar todo el árbol y sus objetos desde un único punto. (Colvin, 2015)

2.16 OBJETOS Y ATRIBUTOS NDS

NDS se compone de objetos y atributos basados en un esquema extensible. Los objetos están representados en cada servidor mediante una entrada física. El esquema contiene las reglas para la formación de estos objetos, atributos y la jerarquía. El esquema NDS define dos tipos de objetos:

- Los objetos contenedores son simplemente aquellos que pueden contener otros objetos. El nombre del árbol, país, localidad, organización y objetos de unidad administrativa son objetos contenedores.
- objetos de hoja son aquellos que no pueden contener otros objetos. Estos objetos suelen representar los recursos de red. Usuarios, impresoras y servidores de NCP son ejemplos de objetos hoja. (Greenblatt, 2002)

2.17 LOS NOMBRES DE OBJETOS

Cada objeto debe tener un nombre único. Este nombre y su ubicación en la jerarquía forman contexto el nombre del objeto. Por ejemplo, vamos a utilizar el siguiente nombre:

.CN = RobF.OU = HQ.O = Novell.T = Novell_Inc

En este nombre, CN = RobF es el nombre del objeto más subordinada, mientras que T = Novell_Inc es el nombre del objeto más superior.

2.18 RESOLUCIÓN DE NOMBRES

La resolución de un nombre implica recorrer el árbol NDS para localizar un objeto en particular. Un identificador de entrada para ese objeto se devuelve, que el cliente puede utilizar para averiguar información sobre el objeto.

La resolución de nombres se puede hacer a través de tres operaciones:

- Lista
- Buscar
- Resolver el nombre

La lista de operación se enumera los objetos subordinados inmediatos a un objeto contenedor especificado y cualquier información especificada sobre esos objetos. La lista de operación no "recorrer el árbol" porque la información que devuelve es local al servidor de nombres actual.

El cliente puede especificar tres filtros por los que pueda seleccionar la información acerca de cada objeto:

- El nombre del filtro permite al cliente solicitar sólo los objetos que tienen un cierto nombre completo relativo.
- El filtro de la clase permite al cliente solicitar sólo los objetos que pertenecen a una determinada clase de objeto.
- El tiempo de filtro permite al cliente solicitar sólo los objetos modificados después de un cierto tiempo.

La operación de búsqueda; busca en una región del árbol de directorios para los objetos cuyos atributos satisfacer los criterios especificados. Entradas no deseadas son eliminadas mediante la colocación de las afirmaciones sobre los valores de atributo, o por una combinación booleana de estas afirmaciones. Las afirmaciones deben ajustarse a las reglas de concordancia definidos para la sintaxis de un atributo en el esquema NDS. La búsqueda se aplica no sólo a los contenedores directamente subordinados al objeto especificado, sino a todos los objetos subordinados a ese objeto. Los mismos filtros usados en la lista de funcionamiento se aplican a la búsqueda operación.

Para resolver el nombre de operación toma el nombre distinguido de un objeto y lo utiliza para obtener un identificador de entrada, que luego se pueden utilizar para acceder a un registro de entrada de NDS correspondiente en el archivo *ENTRY.NDS*. El

cliente puede especificar los parámetros de solicitud y banderas para determinar las respuestas aceptables.

Se utilizan tres tipos de solicitudes:

- Peticiones basados en conexión, que se utilizan para encontrar un ID de entrada en un servidor específico.
- Solicitudes de referencia, que están hechas por un cliente a medida que se acerca el árbol distribuido para localizar un objeto específico. Esta operación da las referencias de clientes que le ayudan a recorrer el árbol y buscar el objeto.
- Solicitudes encadenados, que son utilizados por los clientes cuyos limitados recursos requiere que el servidor de manejar el proceso de referencia.

2.19 GESTIÓN DE ENTRADA

Pocas empresas y organizaciones son estáticos, por lo NDS permite a los administradores modificar los objetos de directorio para reflejar los cambios. Además, los administradores pueden obtener información acerca de los objetos de directorio y sus atributos. Las siguientes secciones describen las operaciones que gestionan los objetos de directorio.

2.20 ADICIÓN DE UNA ENTRADA

NDS permite a los administradores añadir una entrada o un alias para el directorio NDS. El esquema dicta donde se crean las entradas y cuáles son sus atributos pueden ser. Para que sea válido en el directorio, la nueva entrada debe mantener el atributo de objeto de clase, que tiene como valor una clase de base NDS válido. Estas clases de objetos son:

- Los atributos obligatorios para una entrada
- Los atributos que pueden aparecer en su nombre completo relativo
- Los atributos opcionales, si los hay

2.21 LA COMPARACIÓN DE LOS VALORES

La comparación operación compara un valor dado a un atributo existente y su valor. Por ejemplo, el cliente puede utilizar de comparación para determinar si el grupo *Everyone. Novell* tiene un atributo de miembro que tiene un valor de *Admin.Novell*. La operación devuelve verdadero si los valores coinciden o falso si no lo hacen. Esta operación se verifica la existencia de un valor sin tener que leer el valor. Las comparaciones se basan en las reglas de concordancia de la sintaxis de atributo asignado al atributo. (Odom, 2016)

2.22 MODIFICACIÓN DE UNA ENTRADA

Modificar una entrada permite a un cliente a modificar, añadir o eliminar atributos de una entrada en la misma operación, excepto atributos de nombres. Por ejemplo, el cliente puede utilizar modificar una entrada para modificar el número de teléfono de un usuario.

Todas las modificaciones deben obedecer las reglas definidas por la clase de la entrada. Por ejemplo, atributos de sólo lectura y los atributos "ocultos" no se pueden cambiar.

2.23 LA MODIFICACIÓN DE NOMBRE DE UNA ENTRADA

Para cambiar nombre completo relativo de una entrada (RDN), el cliente utiliza Modificar RDN. Mediante esta operación, un cliente puede especificar un nuevo

atributo de nombre y el valor que servirá como RDN de la entrada o la operación puede especificar un nuevo valor para el atributo de nombre existente. En cualquier caso, el nuevo nombre deberá cumplir con las normas para ser modificadas las clases de objetos de la entrada. Unas clases de objetos actualmente definen más de un atributo de nombre.

2.24 MOVER UNA ENTRADA

La operación comprende dos rutinas, comenzar a ingresar para mover y finalizar movimiento, hace lugar a la entrada en una ubicación diferente en el árbol de directorios, que cambia de nombre distinguido de la entrada.

2.25 LA LECTURA DE UNA ENTRADA

Devuelve la información sobre un ingreso almacenado en el directorio. Se lee la información de atributos asociada con la entrada. Los atributos de la entrada dependen de la clase de objeto.

El cliente puede optar por devolver información sobre:

- Atributos específicos o una lista completa de los atributos de una entrada.
- Los nombres de atributo o atributos solamente nombres y valores.
- Privilegios de control de acceso efectivos del usuario actual o de otro usuario.

2.26 ELIMINACIÓN DE UNA ENTRADA

La entrada se eliminada debe ser una hoja o un contenedor que no hay subordinados. Si la entrada tiene subordinados, la operación falla. Un cliente o servidor puede utilizar eliminar entrada para eliminar un alias del directorio.

2.27 PERSONALIDADES NDS

NDS proporciona un conjunto de APIs multi-personalidad que permiten a los siguientes servicios de nombres para acceder a una base de datos NDS:

2.28 USO DE NDAP

Novell Directory Access Protocol (NDAP) es el protocolo estándar que permite el acceso a NDS NetWare.

2.29 SERVICIO BINDERY

NDS trata la encuadernación como una base de datos emulado almacenada en un espacio de nombre plano específico de un servidor. El código de los servicios de encuadernación NDS proporciona una vista alternativa de los mismos datos almacenados en el directorio de base de información de NDS (DIB). Sin embargo, son limitadas; que permiten la compatibilidad hacia atrás, pero no pueden expresar toda la información almacenada en el directorio. (Lammle, 2016)

NDS logra la compatibilidad hacia atrás con el enlace de NetWare mediante la asignación de campos en uno de los bloques de memoria compartida de los primitivos.

Servicios Bindery de NDS utiliza los objetos de encuadernación dinámicas y estáticas de la siguiente manera:

- Objetos Bindery dinámicos no se almacenan en contenedores NDS normales. Ellos no tienen un contenedor específico de los padres de enlace, pero se almacenan en un dedicado "partición de enlace." Se puede acceder por un cliente de enlace o el servidor, pero no están disponibles desde las API de NDS.

- Objetos Bindery estáticos se almacenan en contenedores NDS normales especificados en la ruta de contexto de enlace. Son objetos normales NDS y se almacenan de forma persistente en la base de datos NDS. El contexto de enlace puede contener hasta 16 contenedores.

2.30 USO DE LDAP

El agente de servidor LDAP de Novell se basa en la Universidad de Michigan SLAPD aplicación versión 3.3 y proporciona LDAP versión 2 de acceso como se especifica en el RFC 1777 para NDS incluyendo extensiones para las remisiones. Esto significa LDAP de Novell soporta el enlace simple (acceso sin autenticación y acceso donde el cliente LDAP proporciona el nombre y la contraseña), buscar, modificar, añadir, borrar, modificar RDN, comparar abandono y solicitudes de desvinculación. Además, LDAP de Novell no admite clientes si no están ejecutando también LDAP versión 2. Debido a LDAP de Novell es una personalidad NDS, LDAP de Novell debe ejecutarse en un servidor donde NDS también se está ejecutando.

LDAP sin conexión (CLDAP) contará con el apoyo para el acceso autenticado a través de UDP. CLDAP apoya CLDAP v2 como se especifica en el RFC 1798. Inicialmente, LDAP de Novell compatible con al menos 100 clientes.

El esquema LDAP (basado en X.500) y el esquema NDS (X.500 compatibles) se asignan para corresponder entre sí, permitiendo que un cliente LDAP para buscar el directorio NDS. En la versión inicial, LDAP de Novell soporta una asignación estática entre estos dos esquemas. En versiones posteriores, los administradores de LDAP de Novell podrán configurar esta asignación.

Actualmente, la única forma de autenticación LDAP de Novell que apoyará es el nombre del usuario y la contraseña. Suponemos que los clientes LDAP se utilizan SSL para sus sesiones LDAP y serán capaces de cifrar esta información. LDAP de Novell utiliza SSL v3 para crear un canal cifrado para los clientes LDAP para proporcionar su nombre y contraseña de LDAP de Novell.

LDAP de Novell está configurado con la herramienta NWAdmin. Inicialmente, la configuración residirá en un archivo en el sistema de archivos local. En versiones posteriores, la configuración de LDAP de Novell residirá en NDS. LDAP de Novell está configurado de forma dinámica para que los parámetros pueden restablecer sin tener que reiniciarlo. LDAP de Novell se basa en la interfaz de sockets TCP / IP. LDAP de Novell también se ejecutará en la interfaz WinSock de Windows NT. (Marshall, 2015)

2.31 AUTENTICACIÓN

La autenticación es el proceso mediante el cual los usuarios a establecer su identidad cuando se accede a un servicio de aplicaciones de red. NDS es compatible con los siguientes tipos de autenticación:

2.32 NETWARE 3 DE AUTENTICACIÓN

NetWare 3 utiliza una clave de reto y cifrado de la contraseña para autenticar un usuario sin transmitir la contraseña en el cable. Un usuario NetWare 3 utiliza este proceso para iniciar sesión en un servidor NDS donde se ha establecido el contexto de enlace. El intercambio de autenticación comienza después de que se haya establecido una conexión NCP y se han negociado tamaños de búfer.

2.33 NETWARE 4 DE AUTENTICACIÓN

Autenticación en un entorno NetWare 4 consta de dos procesos:

- Inicio de sesión de usuario. Los registros del cliente en un servidor NDS para establecer la identidad del usuario. No importa dónde el usuario inicia sesión, NDS recorre el árbol hasta que encuentra una copia grabable del objeto del usuario. NDS continuación, recupera la clave privada del cliente.
- Autenticación de fondo. Después de iniciar la sesión, el usuario tiene que autenticarse, o establecer su identidad, a un servidor que tiene un servicio de red. Esto se hace a través de la autenticación de fondo.

Desde la perspectiva del usuario, inicio de sesión se lo invita a entrar una contraseña y luego se bloquea el servidor. El servidor devuelve el contexto del usuario, y el usuario está listo para trabajar en la red.

2.34 AUTORIZACIÓN

Novell Directory Services utiliza un proceso llamado control de acceso para autorizar a los usuarios para llevar a cabo operaciones de directorio en otras entradas y sus atributos. El control de acceso restringe muchas operaciones diferentes, incluyendo la creación de objetos, la lectura y la modificación de atributos de entrada, y la comparación de los valores de atributo. Debido a NDS define y hace cumplir de acceso al directorio, una aplicación cliente de directorio no puede acceder al directorio sin necesidad de utilizar el control de acceso de NDS. Si una aplicación cliente de directorios debe utilizar el control de acceso en su reunión con los clientes, NDS puede almacenar y recuperar información centralizada de control de acceso. (Odom, 2016)

2.35 ATRIBUTO DE LA LISTA DE CONTROL DE ACCESO

La lista de control de acceso (ACL) es el componente clave en la determinación de control de acceso al directorio. Este atributo determina qué operaciones de una entrada de administrador puede hacer en otra entrada y sus atributos. En el esquema de directorio, una ACL es un tipo de atributo de varios valores asignados como un atributo opcional a la clase de objeto superior. Debido a que todas las clases de objetos heredan las características de arriba, todas las entradas pueden tener una ACL. (Marshall, 2015)

La lista de control de acceso atributo contiene los siguientes valores:

- Atributo ID protegido. Este campo contiene una referencia al atributo que el conjunto de privilegios de campo se aplica a. También podría contener un identificador como [Derechos de entrada] o [todos los atributos Derechos]. Si este campo contiene [Derechos de entrada], los privilegios de acceso se aplican a la entrada que lleva a cabo esta ACL.
- Identificación fiduciario. Este campo contiene un identificador de entrada para una entrada específica en el Directorio. Sin embargo, también podría contener una referencia de entrada especial, como [Filtro de derechos heredados], [público], [Root], [Creador] o [Auto]. Una ACL con [filtro de derechos heredados] como las máscaras fiduciario o privilegios concedidos a los filtros de entrada.
- Identificador de clase. Esto puede contener una clase de objeto específico o que incluye todas las clases de objetos.
- Conjunto de privilegios. Este campo enumera los privilegios que se han otorgado al tema. Si el nombre del sujeto es [Filtro de derechos heredados], el conjunto de privilegios campo enumera los derechos que se pueden conceder en ese ingreso, a pesar de que no se hayan concedido.

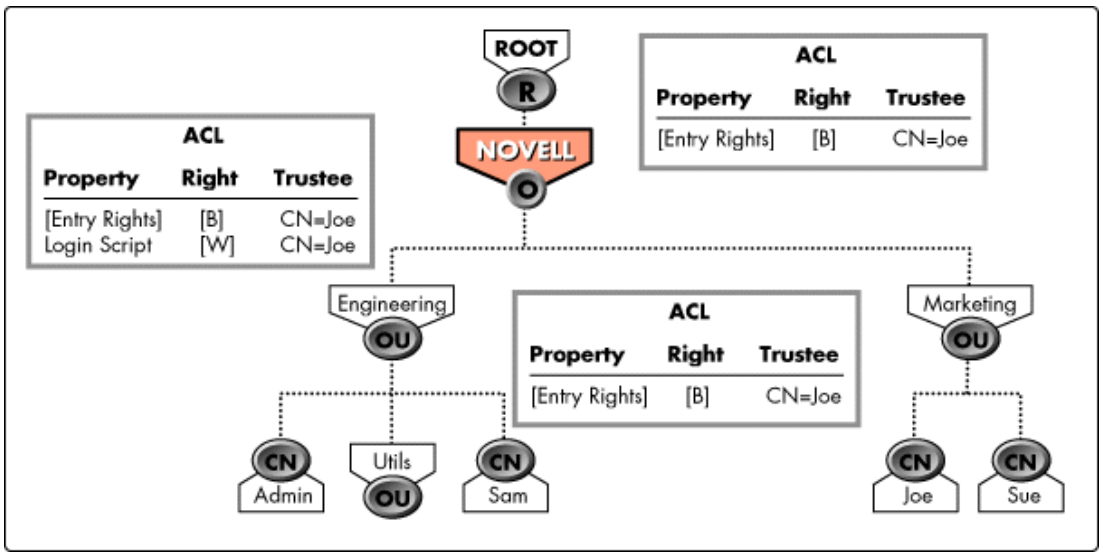
- Controles. Actualmente el único valor posible es heredable. Si este bit está establecido, el conjunto de privilegios que se concede se hereda en objetos subordinados.
- El ACL (Tabla 2.3) es un atributo en el objeto que se está accediendo. El ACL enumera los síndicos y los derechos que tienen al objeto. Por ejemplo, si el objeto Joe y Browse [Derechos de entrada] sobre OU = Engineering, el atributo de ACL en el objeto OU=Engineering se vería así:

Tabla 2.4 - Valores explicados

Campo	Valor
atributo ID	[Cualquier nombre]
fiduciario ID	CN = Joe.OU = Marketing.O = Novell
Atributo Protegida ID	[Derechos de entrada]
conjunto de privilegios	Browse
controles	DS_ENTRY_CTL

Fuente: Greenblatt, B. (2002). Building LDAP-enabled applications with Microsoft's Active Directory and Novell's NDS

Figura 2.5 - Herencia

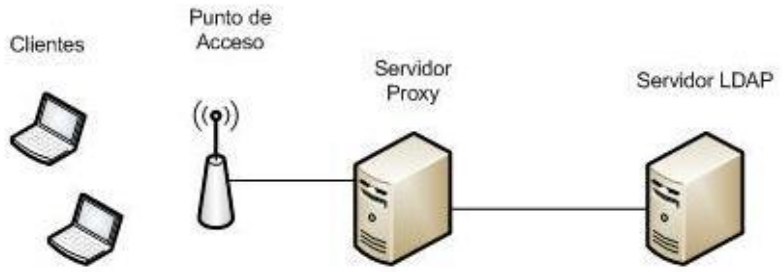


Fuente: Greenblatt, B. (2002). Building LDAP-enabled applications with Microsoft's Active Directory and Novell's NDS

2.36 RED HAT DIRECTORY SERVER

Directory Server es un servidor basado en LDAP que centraliza configuración de aplicaciones, perfiles de usuarios, información de grupos, políticas así como información de control de acceso dentro de un sistema operativo independiente de la plataforma. Forma un repositorio central para la infraestructura de manejo de identidad, Red Hat Directory Server simplifica el manejo de usuarios, eliminando la redundancia de datos y automatizando su mantenimiento. A continuación se detallan los elementos utilizados en el proceso de autentificación de los usuarios en un servidor LDAP. (Brunson, 2015).

Figura 2.6 - Elementos que Intervienen en la Autentificación de LDAP



Fuente: Brunson, R. and Walberg, S. (2015). CompTIA Linux+ / LPIC-1 Cert Guide: (Exams LX0-103 & LX0-104/101-400 & 102-400) (Certification Guide)

Servidor LDAP: protocolo de tipo cliente-servidor para acceder a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Servidor Proxy: intercepta las conexiones de red que un cliente hace a un servidor de destino, por varios motivos como seguridad, rendimiento, etc. Esta función puede ser realizada por un programa o dispositivo. Este servidor es el encargado de permitir o no que determinado usuario tenga acceso a los recursos de la red.

Punto de Acceso: dispositivo que recepta las conexiones de los clientes, y el que asigna una dirección ip a cada usuario.

Clientes: dispositivos que envían peticiones a los servidores para acceder a la red. (Bryant, 2015).

Red Hat Directory Server incluye un servicio de directorio, un servidor de administración para gestionar múltiples instancias de servidor, y una consola basada en Java para administrar las instancias de servidor a través de una interfaz gráfica. El servidor de directorios es un servidor robusto, escalable diseñado para administrar un

directorio de toda la empresa de los usuarios y los recursos. Se basa en un protocolo de servidor de sistemas abiertos llamado el Lightweight Directory Access Protocol (LDAP). Directory Server se ejecuta el ns-slapd en el ordenador central. El servidor gestiona las bases de datos de directorio y responde a las peticiones de los clientes.

Directory Server se compone de varios componentes, que trabajan en conjunto:

- El servidor de directorio es el demonio del servidor LDAP principal. Es compatible con los estándares LDAP v3. Este componente incluye la administración de servidores de línea de comandos y programas de administración y secuencias de comandos para operaciones comunes como la exportación y copia de seguridad de bases de datos.
- La consola de Directory Server es la interfaz de usuario que simplifica la Gestión de usuarios, grupos y otros datos LDAP para su empresa. La consola se utiliza para todos los aspectos de la administración de servidores, incluyendo hacer copias de seguridad; configuración de la seguridad, la replicación, y bases de datos; la adición de entradas; y el seguimiento de los servidores y la visualización de las estadísticas.
- El Servidor de Administración es el agente de administración que administra las instancias del servidor de directorio. Se comunica con la consola de Directory Server y realiza operaciones en las instancias de Directory Server. También proporciona una interfaz HTML simple y páginas de ayuda en línea.
- La mayoría de las tareas administrativas de Directory Server están disponibles a través de la consola de Directory Server, pero también es posible administrar el servidor de directorio modificando manualmente los archivos de configuración o mediante el uso de las utilidades de línea de comandos. (Kouka, A., 2015)

2.37 DESCRIPCIÓN DE DIRECTORY SERVER

Directory Server proporciona las siguientes características principales:

- La replicación multi-master - Proporciona un servicio de directorio altamente disponible para las operaciones de lectura y escritura. replicación Multi-master se puede combinar con escenarios simples y replicación en cascada para proporcionar un entorno de replicación altamente flexible y escalable.
- Encadenamiento y referencias - Aumenta el poder de su directorio mediante el almacenamiento de una vista lógica completa de su directorio en un único servidor, mientras que el mantenimiento de datos sobre un gran número de servidores de directorio de forma transparente para los clientes.
- Roles y clases de servicio - Proporciona un mecanismo flexible para agrupar y compartir los atributos entre las entradas de una manera dinámica.
- Mecanismos de control de acceso mejorado - Proporciona soporte para macros que reducen drásticamente el número de sentencias de control de acceso utilizados en el directorio y aumentan la escalabilidad de evaluación de control de acceso.
- Recursos plazos de DN - concede el poder de controlar la cantidad de recursos de servidor asignados a las operaciones de búsqueda basado en el DN del cliente.
- Múltiples bases de datos - Proporciona una forma sencilla de descomponer los datos de directorio para simplificar la implementación de la replicación y encadenar en su servicio de directorio.
- política de contraseña y bloqueo de cuentas - define un conjunto de reglas que rigen la forma contraseñas y cuentas de usuario se gestionan en el servidor de directorios.

- TLS y SSL - Proporciona una autenticación segura y la comunicación en la red, utilizando la red de Mozilla Servicios de Seguridad (NSS) bibliotecas para la criptografía.
- Los principales componentes del servidor de directorio se incluyen los siguientes:
- Un servidor LDAP - El LDAP v3 compatible demonio de red.
- Directory Server Console - Una consola de administración gráfica que reduce drásticamente el esfuerzo de la creación y el mantenimiento de su servicio de directorio.
- agente SNMP - Se puede controlar el servidor de directorio utilizando el protocolo simple de administración de redes (SNMP).
- Directorio Gateway - Una aplicación web que permite a los usuarios la búsqueda de información en el servidor de directorios, además de proporcionar acceso de autoservicio a su propia información, incluyendo los cambios de contraseña, para reducir los costos de asistencia de los usuarios.
- Gráfico Org - Una aplicación web que muestra una vista gráfica de la estructura de su organización. (Kouka, A., 2015)

2.38 UBICACIONES DE LAS HERRAMIENTAS LDAP

Red Hat Directory Server utiliza herramientas LDAP; como por ejemplo *ldapsearch, ldapmodifyy ldapdelete*- para las operaciones de línea de comandos. Las herramientas se instalan con MozLDAP Directory Server.

Tabla 2.5 - Ubicación del directorio según la plataforma

Plataforma	Ubicación del directorio
Red Hat Enterprise Linux 4 i386	/ Usr / lib / mozldap6
Red Hat Enterprise Linux 4 x86_64	/ Usr / lib64 / mozldap6
Red Hat Enterprise Linux 5 i386	/ Usr / lib / mozldap
Red Hat Enterprise Linux 5 x86_64	/ Usr / lib64 / mozldap
Sun Solaris	/ Usr / lib / sparcv9 / mozldap
HP-UX	/ Opt / DirSrv / bin

Fuente: Juliet Kemp. (2009). Linux system administration recipes: a problem-solution approach

Para todas las guías de Red Hat Directory Server y la documentación, las herramientas LDAP utilizados en los ejemplos, tales como **ldapsearch**e **ldapmodify**, son las herramientas LDAP Mozilla. Para la mayoría de los sistemas Linux, herramientas OpenLDAP ya están instalados en el directorio **/usr/bin/**. Estas herramientas OpenLDAP no son compatibles con las operaciones de Directory Server. Para obtener los mejores resultados con el servidor de directorios, asegúrese de que la ruta de acceso a las herramientas LDAP Mozilla es lo primero en el *path* o utilice la ruta completa y el nombre de archivo para cada operación LDAP.

Sin embargo, estas herramientas OpenLDAP se pueden utilizar para operaciones de directorio del servidor con ciertas precauciones:

- La salida de las otras herramientas puede ser diferente, por lo que puede no parecerse a los ejemplos en la documentación.

- Las herramientas de OpenLDAP requieren un **–x** argumento para desactivar SASL para que pueda ser utilizado por un enlace simple, es decir, el **–D** y **–w** argumentos o un enlace anónimo.

Los argumentos de las herramientas de OpenLDAP para usar TLS/SSL y SASL son muy diferentes de los argumentos LDAP. (Juliet Kemp., 2009)

2.39 OPENLDAP

Se trata de una implementación libre del protocolo que soporta múltiples esquemas por lo que puede utilizarse para conectarse a cualquier otro LDAP. Tiene su propia licencia, la OpenLDAP Public License. Al ser un protocolo independiente de la plataforma, varias distribuciones Linux y BSD lo incluyen, al igual que AIX, HP-UX, Mac OS X, Solaris, Windows (2000/XP) y z/OS. (Bresnahan, 2015)

OpenLDAP tiene cuatro componentes principales:

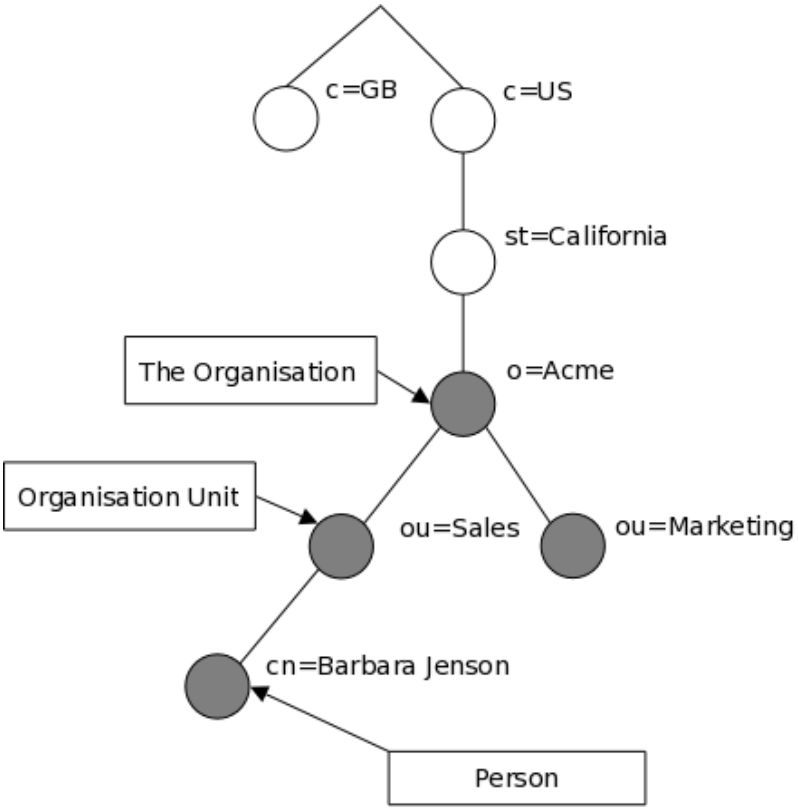
- slapd - demonio LDAP autónomo.
- slurpd - demonio de replicación de actualizaciones LDAP autónomo.
- Rutinas de biblioteca de soporte del protocolo LDAP.
- Utilidades, herramientas y clientes.

El proyecto OpenLDAP se inició en 1998 por Kurt Zeilenga. El proyecto se inició mediante la clonación de la fuente de referencia LDAP de la Universidad de Michigan donde un proyecto de larga duración ha apoyado el desarrollo y la evolución del protocolo LDAP hasta el lanzamiento final de ese proyecto en 1996. A partir de mayo el año 2015, el proyecto OpenLDAP tiene cuatro miembros del equipo central: Howard Chu (arquitecto jefe), Quanah Gibson-Mount, Hallvard Furuseth, y Kurt Zeilenga. Hay

numerosos otros colaboradores importantes y activos incluidos Luke Howard, Ryan Tandy, y Gavin Henry. Miembros del equipo central últimos incluyen Pierangelo Masarati.

¿Cómo está dispuesta la información? En LDAP, entradas de directorio están organizadas en una estructura jerárquica en forma de árbol. Tradicionalmente, esta estructura refleja los límites geográficos y/o de organización. Las entradas que representan los países aparecen en la parte superior del árbol. Debajo de ellos son entradas que representan estados y organizaciones nacionales. Debajo de ellos podrían ser entradas que representan unidades organizativas, la gente, impresoras, documentos o cualquier otro que se pueda imaginar. La figura 2.7 muestra un ejemplo de árbol de directorios LDAP utilizando nomenclatura tradicional. (Butcher, 2012)

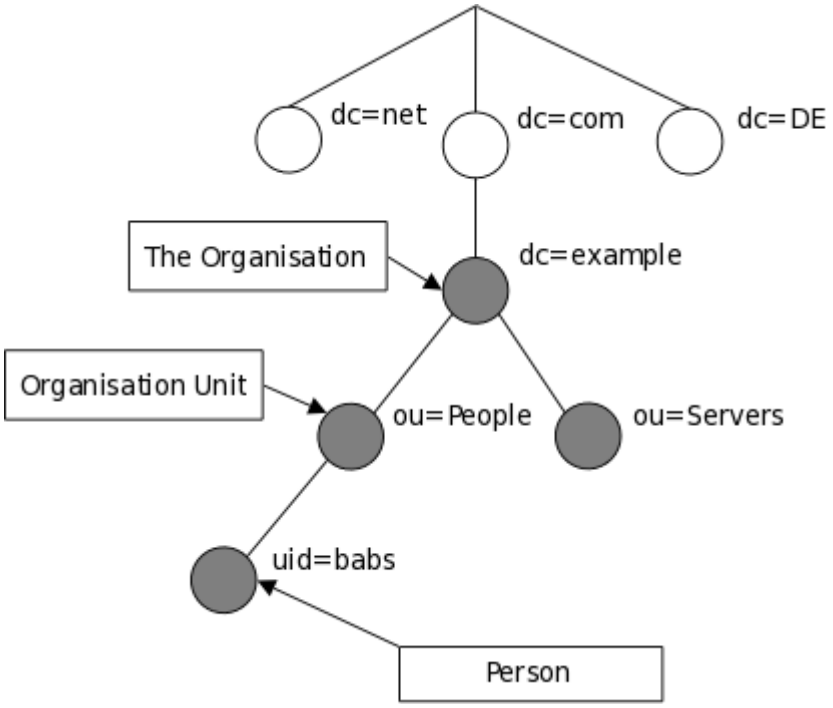
Figura 2.7 - Árbol de directorios LDAP (nomenclatura tradicional)



Fuente: Butcher, M. (2012). Mastering OpenLDAP.

El árbol también se puede disponer sobre la base de nombres de dominio de Internet. Este enfoque se está convirtiendo en nombres vez más popular, ya que permite a los servicios de directorio que se encuentran utilizando el DNS. Figura 2.8 muestra un ejemplo LDAP árbol del directorio utilizando nombres basado en dominio.

Figura 2.8 - Árbol de directorios LDAP (nombres de Internet)



Fuente: Butcher, M. (2012). Mastering OpenLDAP.

En general, se debe utilizar un servidor de directorio en caso de requerir datos a ser gestionado de forma centralizada, almacenada y accesible a través de métodos basados en estándares.

Algunos ejemplos comunes que se encuentran en toda la industria son, pero no limitado a:

- Autenticación de la máquina
- Autenticación de usuario
- Grupos de usuario / Sistema

- Directorio
- Representación organización
- Seguimiento de activos
- Almacén de información de la telefonía
- La gestión de recursos de usuario
- E-mail búsquedas de direcciones
- Almacén de configuración de aplicaciones
- Almacén de configuración de PBX

Siempre hay nuevas formas de utilizar un directorio LDAP y aplicar los principios para hacer frente a ciertos problemas, por lo tanto, no hay una respuesta sencilla a esta pregunta.

LDAPv3 fue desarrollado en la década de 1990 para sustituir a LDAPv2. LDAPv3 añade las siguientes características a LDAP:

- Servicios de autenticación y seguridad de datos a través de fuertes SASL
- Servicios de autenticación de certificado y de seguridad de datos a través de TLS (SSL)
- Internacionalización a través del uso de Unicode
- Referencias y continuaciones
- Esquema Descubrimiento
- Extensibilidad (controles, operaciones extendidas, y más)

LDAPv2 es histórico (RFC3494). Como la mayoría de los llamados implementaciones de LDAPv2 (incluyendo slapd (8)) no se ajustan a la especificación técnica LDAPv2, la interoperabilidad entre las implementaciones que reclaman el apoyo LDAPv2 es

limitado. Como LDAPv2 difiere significativamente de LDAPv3, desplegando tanto LDAPv2 y LDAPv3 al mismo tiempo es bastante problemático. LDAPv2 debe ser evitado. LDAPv2 está desactivado por defecto.

2.40 PROTOCOLO LDAP VS PROTOCOLO RDBMS

Esta cuestión se plantea muchas veces, en diferentes formas. El más común, sin embargo, es: ¿Por qué no usar OpenLDAP un sistema relacional de gestión de base de datos (RDBMS) en lugar de un almacén de claves/valor intrínseco como LMDB? En general, esperando que los sofisticados algoritmos implementados por RDBMS de calidad comercial se hacen OpenLDAP ser más rápido o de alguna manera mejor y, al mismo tiempo, permitir que el intercambio de datos con otras aplicaciones.

La respuesta corta es que el uso de un sistema de indexación de base de datos y personalizado incrustado permite OpenLDAP para proporcionar un mayor rendimiento y escalabilidad sin pérdida de fiabilidad. Ahora para la respuesta larga. Todos nos enfrentamos todo el tiempo con los RDBMSs elección frente a directorios. Es una decisión difícil y no existe una respuesta sencilla. Es tentador pensar que tener un backend RDBMS al directorio resuelve todos los problemas. Sin embargo, es malo. Esto se debe a que los modelos de datos son muy diferentes. Que representan los datos del directorio con una base de datos relacional va a requerir dividir los datos en varias tablas.

Piense por un momento acerca de la persona de objeto. Su definición requiere tipos atributo objectclass, SN y CN y permite que los tipos de atributos userPassword, telephoneNumber, seeAlso y descripción. Todos estos atributos son varios valores, por lo que una normalización requiere poner cada tipo de atributo en una tabla separada. Ahora usted tiene que decidir sobre las teclas apropiadas para esas tablas. La clave

principal podría ser una combinación de la DN, pero esto se hace bastante ineficiente en la mayoría de las implementaciones de bases de datos. El gran problema ahora es que el acceso a los datos de una entrada requiere que buscan en diferentes áreas del disco. En algunas aplicaciones esto puede ser aceptable, pero en muchas aplicaciones de rendimiento se resiente.

Los únicos tipos de atributos que se pueden poner en la entrada principal de la tabla son los que son obligatorios y de un solo valor. Puede añadir también los atributos de valor único opcionales y los puso a NULL o algo si no está presente. Pero espera, la entrada puede tener varias clases de objetos y están organizados en una jerarquía de herencia. Una entrada de organizationalPerson de objeto ahora tiene los atributos de la persona además de algunos otros y algunos tipos de atributos anteriormente opcionales son ahora obligatorio.

Una vez alcanzado este punto, tres enfoques vienen a la mente. Una de ellas es hacer plena normalización de manera que cada tipo de atributo, no importa qué, tiene su propia tabla separada. El enfoque simplista, donde el DN es parte de la clave primaria es extremadamente derrochadora, y pide que se enfoque en el que la entrada tiene un identificador numérico único que se utiliza en lugar de las teclas y una tabla principal que se asigna a los identificadores de DN. El enfoque, de todos modos, es muy ineficiente cuando se solicitan varios tipos de atributos de una o más entradas. Esta base de datos, aunque torpemente, se puede gestionar desde aplicaciones de SQL. (Nutter, 2014)

El segundo enfoque es poner toda la entrada como una mancha en una mesa compartida por todas las entradas independientemente de la clase de objeto y tienen tablas adicionales que actúan como índices de la primera tabla. Tablas de índice no son índices de bases de datos, pero están totalmente gestionados por la aplicación del lado

del servidor LDAP. Sin embargo, la base de datos se vuelve inutilizable desde SQL. Y, por lo tanto, un sistema de base de datos en toda regla proporciona poca o ninguna ventaja. La generalidad completa de la base de datos es que no sean necesarios. Mucho mejor usar algo ligero y rápido, como LMDB.

Una forma completamente diferente de ver esto es que renunciar a las esperanzas de la implementación del modelo de datos de directorio. En este caso, LDAP se usa como un protocolo de acceso a los datos que proporciona sólo superficialmente el modelo de datos de directorio. Por ejemplo, puede ser de sólo lectura o, cuando se permiten cambios, se aplican restricciones, tales como hacer de un solo valor de tipos de atributos que permitan varios valores. O la imposibilidad de añadir nuevas clases de objetos a una entrada existente o eliminar uno de los presentes. Las restricciones abarcan toda la gama de las restricciones permitidas (que puede estar en otra parte el resultado de control de acceso) para violaciones directas del modelo de datos. Puede ser, sin embargo, un método para proporcionar acceso a los datos LDAP preexistente que es utilizado por otras aplicaciones. Sin embargo, en el entendimiento de que en realidad no tenemos un "directorio".

Existentes implementaciones del servidor LDAP comerciales que utilizan una base de datos relacional son ya sea desde la primera clase o el tercero. No sé de cualquier aplicación que utiliza una base de datos relacional para hacer lo que hace ineficiente BDB eficiente. Para aquellos que estén interesados en la "tercera vía" (la exposición de los datos existentes de RDBMS como árbol LDAP, que tiene algunas limitaciones en comparación con el modelo clásico de LDAP, pero lo que es posible interoperar entre las aplicaciones LDAP y SQL):

OpenLDAP incluye back-SQL - el servidor que lo hace posible. Se utiliza ODBC + metainformación adicional sobre la traducción de las consultas LDAP a las consultas SQL en el esquema RDBMS, proporcionando diferentes niveles de acceso - a partir de sólo lectura para acceso completo dependiendo de RDBMS que utiliza, y su esquema. (Butcher, M., 2012)

2.41 SERVIDOR SLAPD

Slapd es un servidor de directorio LDAP que se ejecuta en muchas plataformas diferentes. Se puede utilizar para proporcionar un servicio de directorio de su propio. Su directorio puede contener casi cualquiera que usted quiere poner en él. Se puede conectar al servicio de directorio LDAP global, o ejecutar un servicio por sí mismo. Algunas de las características y capacidades más interesantes de slapd incluyen:

2.41.1 USO LDAPV3

Slapd implementa la versión 3 de Lightweight Directory Access Protocol. Slapd soporta LDAP sobre ambosIPv4 y IPv6 y Unix IPC.

2.41.2 AUTENTICACIÓN SALS

Slapd soporta autenticación fuerte y de datos de seguridad (integridad y confidencialidad) servicios a través del uso de SASL. slapd aplicación SASL's utiliza Cyrus SASL software que es compatible con una serie de mecanismos que incluye DIGEST-MD5, EXTERNO y GSSAPI.

2.41.3 TRANSPORT LAYER SECURITY

Slapd soporta la autenticación y seguridad de datos basada en certificados (integridad y confidencialidad) servicios a través del uso de TLS (o SSL). slapd aplicación TLS's puede utilizar OpenSSL , GnuTLS o MozNSS software.

2.41.4 TOPOLOGÍA DE CONTROL

Slapd puede ser configurado para restringir el acceso a la capa de conexión en base a información de la topología de red. Esta característica utiliza envoltorios TCP.

2.41.5 CONTROL DE ACCESO

Slapd proporciona una facilidad de control de acceso rico y poderoso, que le permite controlar el acceso a la información en su base de datos (s). Puede controlar el acceso a las entradas en base a información de autorización LDAP, direcciones IP, nombre de dominio y otros criterios. Slapd soporta tanto estática y dinámica de información de control de acceso.

2.41.6 ELECCIÓN DE BACKENDS DE BASES DE DATOS

Slapd viene con una variedad de diferentes bases de datos que puede elegir. Incluyen MDB, un backend de base de datos transaccional jerárquico de alto rendimiento; BDB, un backend transaccional de base de datos de alto rendimiento (obsoleto); HDB, un backend transaccional jerárquico (obsoleto) de alto rendimiento; SHELL, una interfaz de back-end para scripts de shell arbitrarios; Y PASSWD, una simple interfaz de backend al archivo passwd. El backend de MDB utiliza LMDB, un reemplazo de alto rendimiento para la base de datos Berkeley de Oracle. Los backends BDB y HDB utilizan Oracle Corporation Berkeley DB. Estos backends han sido

obsoletos ya que LMDB proporciona un rendimiento de lectura / escritura significativamente mayor y confiabilidad de datos.

2.41.7 MÚLTIPLES INSTANCIAS DE BASE

Slapd se pueden configurar para servir a múltiples bases de datos al mismo tiempo. Esto significa que una sola slapd servidor puede responder a las peticiones de muchas porciones lógicamente diferentes del árbol LDAP, utilizando los mismos o diferentes backends de bases de datos.

2.41.8 MÓDULOS GENÉRICOS API

Si necesita aún más personalización, slapd le permite escribir sus propios módulos fácilmente. slapd se compone de dos partes distintas: un extremo frontal que se encarga de la comunicación de protocolo con los clientes LDAP; y los módulos que manejan tareas específicas, tales como las operaciones de bases de datos. Debido a que estas dos piezas se comunican a través de un bien definido API, Puede escribir sus propios módulos personalizados que se extienden slapd de muchas maneras. Además, un número de base de datos programables se proporciona módulos. Estos permiten exponer a fuentes de datos externas slapd utilizando lenguajes de programación (Perl , shell, y SQL).

2.41.9 HILOS

Slapd se rosca para un alto rendimiento. Un multi-hilo sola slapd proceso maneja todas las peticiones entrantes utilizando un grupo de subprocesos. Esto reduce la cantidad de sobrecarga de sistema necesario mientras que proporciona un alto rendimiento.

2.41.10 REPLICACIÓN

Slapd se puede configurar para mantener las instantáneas de la información del directorio. Este maestro-único/múltiple-esclavo esquema de replicación es vital en entornos de gran volumen en un solo slapd instalación simplemente no facilita la disponibilidad o fiabilidad necesaria. Para entornos extremadamente exigentes en las que un único punto de fallo no es aceptable, multi-maestro de replicación también está disponible. Slapd incluye soporte para *LDAP Sync* basado en replicación.

2.41.11 CONFIGURACIÓN

Slapd es altamente configurable a través de un único archivo de configuración que le permite cambiar casi todo lo que nunca quiere cambiar. Las opciones de configuración tienen valores por defecto razonables, haciendo su trabajo mucho más fácil. La configuración también puede realizarse dinámicamente usando LDAP en sí, que mejora en gran medida la manejabilidad. (Butcher, 2012)

2.42 PHPLDAPADMIN

phpLDAPadmin (también conocido como PLA) es un cliente LDAP basado en la web. Proporciona administración multilingüe fácil de acceder desde cualquier lugar para su servidor LDAP. Su navegador de árbol jerárquico y la funcionalidad de búsqueda avanzada hacen que sea intuitivo navegar y administrar su directorio LDAP.

phpLDAPadmin es el navegador LDAP perfecto para el profesional de LDAP y para principiantes. Su base de usuarios se compone principalmente de profesionales de administración de LDAP. (Main Page - phpLDAPadmin. 2017)

Tiene las siguientes características:

- Navegador de árbol LDAP
- Edición de entrada basada en plantillas
- Copie las entradas LDAP (incluso copie entre diferentes servidores)
- Copiar recursivamente árboles enteros
- Eliminar entradas LDAP
- Recursivamente eliminar árboles enteros
- Ver y editar los atributos de la imagen
- Navegador de esquema LDAP avanzado
- Creación de entradas basadas en plantillas
- Búsquedas LDAP (tanto simples como avanzadas)
- Renombrar entradas LDAP
- Determine automáticamente el DN raíz de su servidor LDAP
- Incrementar automáticamente los números de UID
- Disponible en 10 idiomas
- Extensible. (Main Page - phpLDAPadmin. 2017).

2.43 ANTECEDENTES DE LA INVESTIGACIÓN

El autor Dennis Stephen Cohn Muroy, con la tesis titulada: **"**ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN PARA LA ADMINISTRACIÓN DE LAS HERRAMIENTAS DE SEGURIDAD EN UNA RED LOCAL" del año 2008, para optar el título de Ingeniero Informático en la Pontificia Universidad Católica del Perú, indica que: Desde el año en que se estableció la primera red de computadoras (ARPANET), hasta nuestros días, Internet ha pasado a través de un largo proceso evolutivo. Siendo utilizado actualmente como fuente de conocimiento, medio de comunicación y una amplia plataforma para hacer negocios (ebusiness).

Lastimosamente, también es un canal a través del cual se perpetran ataques que han ocasionado pérdidas de información no sólo a las empresas de diversos tamaños, sino también a las personas naturales. Como mecanismo de prevención, es necesario hacer uso de una serie de herramientas de tipo software y/o hardware, así como políticas de seguridad a fin de proteger la confidencialidad, integridad y disponibilidad de la información. Sin embargo, las soluciones presentes en el mercado, a pesar de poseer un adecuado desempeño en cuanto a la prevención y la detección de los ataques, carecen de un entorno intuitivo y de fácil uso, lo cual influye en el registro de reglas débiles o erróneas; provocando agujeros en el perímetro de la seguridad de la red local. Es por ello, que en el presente documento se plantea como solución analizar, diseñar e implementar una aplicación para facilitar la administración de las herramientas Iptables, Squid y Snort; utilizadas para proteger la información dentro de una red local. De esta forma concluye que luego de llevar a cabo la implantación y pruebas de la solución propuesta, se concluye lo siguiente: Mientras más sencilla y fácil de utilizar sea una aplicación para los usuarios, los riesgos de llevar a cabo una inadecuada configuración son menores; asimismo, el tiempo invertido en llevar a cabo las configuraciones es menor, lo cual permite asignar dicho personal a tareas críticas. El riesgo que muchos usuarios carezcan de conocimientos en cuanto a reglas de seguridad, puede ser aminorado haciendo uso de una solución que le ayude a establecer reglas y configuraciones iniciales claras y fáciles de comprender. A pesar de las soluciones que permiten configurar y asegurar las redes y sistemas; siempre existirá un nivel de riesgo a ataques, mientras los usuarios no conciencien sobre los riesgos a los que se hallan expuestos.

El autor Rafael Villanueva Castrejón, con la tesis titulada **“**ANALISIS, DISEÑO E IMPLEMENTACIÓN DE TECNOLOGÍA FIREWALL PARA MEJORAR LA GESTIÓN Y ADMINISTRACIÓN DE LA RED DE DATOS DE LA EMPRESA S&B

SERVICIOS GENERALES" del año 2013, para optar el título profesional de Ingeniero de Sistemas en la Universidad Privada del Norte, indica que: el presente trabajo tuvo como objetivo general el "Análisis, Diseño e Implementación de Tecnología Firewall para Mejorar la Gestión y Administración de la Red de la Empresa S&B Servicios Generales". Actualmente se observa las preocupaciones que tienen las empresas hoy día es como llevar a cabo sus transacciones electrónicas manteniendo altos niveles de seguridad y confidencialidad. La conectividad total se ha convertido en una necesidad para poder sobrevivir en el ambiente competitivo del nuevo milenio. Esto ha traído, al mismo tiempo, serios problemas de seguridad al facilitar el acceso desde el mundo exterior a través de internet y así exponer los recursos internos de la red. Para impedir que personas no autorizadas penetren en la red o que accedan a más información de la permitida, se utiliza un sistema de defensa peri mental llamado firewall (cortafuegos), el cual se coloca como una barrera de protección entre internet y la red local de la empresa. A veces se utilizan firewall adicionales internamente para separar distintos departamentos. Un sistema basado en firewall no es la panacea para la seguridad. Esta tesis pretende completar una mejor seguridad en la red de datos, en la cual se implementó un firewall el cual brinda una máxima seguridad, donde se aplicaron políticas de seguridad en las áreas de trabajo de la empresa, además se definen las reglas que se aplican en el firewall. Finalmente concluye que: las reglas especifican el origen, destino, servicio y acción a realizar para cualquier transacción. También definen que eventos deben guardarse (logs), puesto que las posibles brechas de seguridad son más fácilmente identificables de esta manera. Es imprescindible que las políticas de seguridad se diseñaron de acuerdo a los activos y condiciones específicas de información que quiso proteger la empresa. Creando la zona verde que está dedicada a la red LAN de la empresa, la zona naranja donde se encuentran los servidores web y la zona roja que incluye la red

WAN. En las distintas zonas de aplicaron las políticas de seguridad, restringir el acceso a servicios específicos.

El autor Jorge Alonso López Mori, con la tesis titulada: "DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE GESTIÓN DE ACCESOS A UNA RED WI-FI UTILIZANDO SOFTWARE LIBRE" del año 2008, para optar el título de Ingeniero de Telecomunicaciones en la Pontificia Universidad Católica del Perú, resume que: el reciente aumento en la implementación de redes inalámbricas nos obliga a contemplar con más cuidado el aspecto de la seguridad en este tipo de redes. Así como en el caso de las típicas redes de datos con cables (siendo la tecnología Ethernet la más utilizada para estos casos), tiene que asegurarse que los usuarios de una red inalámbrica se encuentren conectados a ésta de una manera segura, teniendo en cuenta que ahora el medio de transmisión ya no se restringe a un cable, sino que se encuentra en todo el ambiente que lo rodea. Debe de comprobarse que el usuario sea quien dice ser (autenticación), que solo tenga acceso a los recursos que le corresponda (autorización) y también llevar a cabo un registro de las actividades que haga dentro de la red (contabilidad); realizando todo esto de una manera segura y sin que sujetos ajenos a la red puedan estar leyendo información confidencial ni mucho menos tratar de modificarla. En esta tesis se tiene pensado explicar el diseño e implementación que se debería de llevar a cabo dentro de un escenario dado para la instalación de una red inalámbrica segura que contemple la administración de sus usuarios por medio de una plataforma de gestión Web basada en PHP, integrada a un servidor de directorios LDAP con compatibilidad hacia implementaciones libres y cerradas de dicho protocolo, un servidor de autenticación RADIUS y un servidor de base de datos MySQL. Se estudiarán los principales aspectos aplicados en redes inalámbricas Wi-Fi, poniendo especial énfasis en la seguridad de la red y de sus usuarios con mecanismos tales como: WPA2 (IEEE

802.11i), 802.1X, EAP, RADIUS, entre otros. Finalmente concluye que en OpenLDAP se puede encontrar requerimientos de un elemento adicional para que pueda 'conversar' éste con los clientes móviles (notebooks con sistema operativo MS Windows XP o Windows Vista). Fue así que, tras luego de haber investigado, se ubicó al servicio de SAMBA como una solución ideal para este problema. Integrando el servidor OpenLDAP con el servicio SAMBA pudo implementarse un dominio de usuarios para los cuales aparecería transparente y sin mayores inconvenientes para el momento de registrarse y contar con acceso a la red.

El autor Joselito Cieza Pérez de la tesis titulada "IMPLEMENTACION DE SISTEMA DE APOYO AL SOPORTE TECNICO EN PLATAFORMA MOVIL" del año 2014, en resumen: este documento contiene una visión general de alto nivel del diseño y la documentación de siete de los procesos de ITIL en la Gerencia de Servicios Informáticos en CARO PERU S.A. ITIL (Biblioteca de Infraestructura de Tecnología de la Información) es un marco público que describe las mejores prácticas en la gestión de los servicios de tecnología de la información. ITIL representa las experiencias de aprendizaje y liderazgo de los mejores proveedores de servicios del mundo. Proporciona un marco para la gobernanza de la tecnología de la información y se centra en la medición y la mejora continua de la calidad de los servicios de las tecnologías de la información ofrecidos, tanto desde la perspectiva empresarial y la perspectiva del cliente. Prácticas de ITIL se basan en el ciclo de vida de servicio y contienen cinco elementos: la estrategia de servicio, el diseño del servicio, la transición del servicio, la operación del servicio y la mejora continua del servicio. Cada uno de ellos se basa en los principios del servicio, procesos, funciones y medidas de la ejecución. Algunos de los beneficios de aplicar prácticas de ITIL son: la satisfacción del cliente con los servicios de tecnología de la información, la mejora de la disponibilidad del servicio, ahorros financieros por la

reducción de los reprocesos y el tiempo perdido, la mejora de la gestión de los recursos, mejora en la toma de decisiones y optimización del riesgo. Este documento es para cualquier persona que tenga interés en conocer los objetivos, el enfoque, las normas, el contenido y los indicadores clave de rendimiento de los siguientes procesos de ITIL: Administración de Niveles de Servicio, Service Desk, gestión de incidentes, el Servicio de Gestión de Requisitos, Problema Gestión, Gestión del Cambio y Configuración y Servicio de Gestión de Activos. Conluye que: dentro de los procesos para la gestión de los servicios informáticos establecidos por las mejores prácticas ITIL, Gestión de Niveles de Servicio, función Mesa de Servicios, Gestión de Incidentes, Gestión de Requerimientos, Gestión de Problemas, Gestión de Cambios y Gestión de Configuración y Activos de los Servicios Informáticos son requeridos con mayor prioridad dentro de la GSI y con un nivel de madurez 3, es decir, procesos normalizados y definidos. Algunos de los usuarios de la Organización desconocen los servicios, los términos y las condiciones en las que estos son prestados.

El autor Juan Jacob Bueno Rosales en el año 2013 con la tesis titulada "SISTEMA DE CONTROL Y SEGURIDAD ENDIAN FIREWALL PARA LA EMPRESA FRADA SPORT" para optar el título de Ingeniero de Sistemas Informáticos en la Universidad de Israel (Quito, Ecuador); en resumen indica que: los niveles de vulnerabilidad e importancia de toda la información de la empresa, a través de la red global de datos, orienta procesos o mecanismos de seguridad, únicos centralizados, y segmentados en la empresa Frada Sport. Constituye una parte fundamental para la empresa, ya que no solo corresponde a la herramienta de seguridad como tal, sino a representar gráficamente, sistemáticamente, procesos y modelos de desarrollo actuales, mediante la implementación de un sistema único de seguridad, que a más de brindar soluciones de seguridad, estructura y establece medios de estabilidad, vigilancia, modelos de

desarrollo, altamente disponible e inteligentes, control organizativo y fundamental, entre otros. Teniendo en cuenta toda la infraestructura de la red, se realiza un diseño de la red y se levanta la información de los procesos o modelos actuales, para posteriormente brindar soporte a soluciones específicas orientadas al control mayor y seguridad. La herramienta de seguridad y control Endian Firewall, es un sistema único open source, específicamente estructurado como bitácora, capaz de brindar soluciones óptimas a la red de datos, captando y estabilizando mejores formas gráficas de control organizativo en la red de datos.Se concluye que: el sistema de seguridad Endian firewall, representa una manera estratégica de control, seguridad, disponibilidad, rendimiento y administración de la red global de datos, lo que se describe inicialmente, es un análisis de le empresa en técnicas de la investigación, basado en observación directa, encuestas descriptivas y referencia cruzada, para determinar el eje principal de la problemática de la empresa y abarcar procesos centrales de desarrollo. Establecido el proceso de desarrollo en base a las encuestas, se analiza los diferentes medios de información vulnerables, puntos críticos, manejo de información de cada departamento de la empresa, entre otros, para representar gráficamente la situación actual referente a los diferentes problemas de la empresa como proceso de diseño. Después, se procedió a estructurar la propia metodología de desarrollo, basada en 6 etapas de mejoramiento, cada etapa sigue diferente proceso de administración, control, seguridad, centralización y alta disponibilidad de datos. Además, es importante trabajar con herramientas de entorno gráfico para tareas complejas, como crear reglas de filtrado, políticas, servicios, registros, entre otros. Teniendo en cuenta todo el proceso a seguir, se puede manifestar que en la empresa Frada Sport, es aplicable el sistema de seguridad open source, basado en costos representativos mínimos para la empresa, toda la estructura física y lógica que gestiona el sistema de seguridad EFW, es de vital importancia, ya que cuenta con diferentes

medios que ayuda a la empresa a tener principalmente seguridad centralizada de alta disponibilidad, y además, la correcta administración y control de todos los componentes de la red global de datos, destacando principalmente su alto rendimiento o performance

El autor José Antonio Senso Ruiz de la tesis titulada "SISTEMAS DE METADATOS EN RECUPERACIÓN DE INFORMACIÓN: PROPUESTA DE MODELO DE TRABAJO EN EL USO DE RDF SOBRE DIRECTORIOS LDAP" del año 2010, en resumen indica que: teniendo en cuenta la complejidad y amplitud del tema a tratar, así como la casi nula existencia de documentación previa en castellano que aborde el uso de sistemas de metadatos en la recuperación de información utilizando una metodología científica que nos sirviera de guía, creímos conveniente acometer este estudio desde dos perspectivas. La primera (que coincide con el primer bloque del trabajo) es esencialmente descriptiva, y trata de analizar la situación actual de los sistemas de metadatos con el objetivo claro de analizar si éstos son capaces de ofrecer las soluciones necesarias para realizar una descripción homogénea de los recursos electrónicos sin limitar las opciones de localización y recuperación. Para ello se hace un recorrido, primero por el concepto de metadato y, más adelante, por las principales tendencias en los sistemas de representación de contenido en recursos electrónicos. La segunda perspectiva, más valiosa científicamente hablando por lo que de novedoso tiene, consiste en estructurar y aplicar un modelo de trabajo válido para el desarrollo de sistemas de información basados en metadatos. En concreto se propone el uso de RDF (Resource Description Framework) como mecanismo de representación de la información y de LDAP (Lightweight Directory Access Protocol) como herramienta de gestión y búsqueda de la misma. Para ello se describe, paso por paso, una implementación básica y real y se evalúan los resultados en comparación con otros mecanismos de consulta. En conclusión: a modo de conclusión, podemos afirmar que los

URI engloban a los URL y a los URN. Los URL describen una ubicación física, los URN un nombre único y los URI lo describen todo, también podemos observar cómo la sintaxis serializada muestra un modelo de transposición del recurso a RDF muy claro, con pocas complicaciones y bien estructurado. El problema lo encontraremos cuando se desee profundizar más en determinados detalles del recurso. Para eso se tendrá que utilizar la sintaxis abreviada.

Los autores Gladis Sofía Asadovay Lema y Liliana Mercedes Caiza Ortiz en la tesis titulada "ANÁLISIS COMPARATIVO DE SERVIDORES DE AUTENTIFICACIÓN RADIUS Y LDAP CON EL USO DE CERTIFICADOS DIGITALES PARA MEJORAR LA SEGURIDAD EN EL CONTROL DE ACCESO A REDES WIFI" del año 2013, para optar el título de Ingeniería Electrónica, Telecomunicaciones y Redes, en la Escuela Superior Politécnica de Chimborazo (Ecuador); en resumen se indica que: el análisis comparativo de servidores de autentificación RADIUS y LDAP complementado con el uso de certificados digitales para mejorar la seguridad en el acceso a redes Wifi y su aplicación como prototipo de un sistema de autentificación para una red inalámbrica en general. Utilizando el método científico experimental, se desarrolló ambientes de prueba de los servidores en estudio para establecer las diferencias que existen entre estos, mediante las herramientas de software daloRADIUS, MySQL, OpenLDAP 2.2.5, ZeroShell 2.0 y Squid-2.5. Además de la observación que permitió determinar cuál es el servidor más eficiente y para qué tipo de red son más adaptables, utilizando técnicas de revisión bibliográfica para la instalación, configuración, administración y monitoreo de los servidores. Se compararon los siguientes indicadores: Gestión de usuarios, Autentificación, Gestión de Datos, Rendimiento, Funcionalidad y seguridad, dando como resultado que LDAP tiene el 70,52% de eficiencia y RADIUS un 74,67%, estableciéndose una diferencia de

aproximadamente un 6% entre los dos servidores de autentificación. Se concluye que el servidor de autentificación más eficiente para ser implementado en una red inalámbrica es RADIUS, debido a la facilidad de administración que presenta y la rapidez con la que se realizan las actualizaciones de datos. Se recomienda la utilización de certificados digitales como credenciales para poder conectarse a la red, en lugar del usuario y contraseña tradicional que se utiliza en la autentificación convencional, para añadirle mayor seguridad en el proceso de autentificación. LDAP es un protocolo de acceso unificado a un conjunto de información sobre una red. Existen diversas implementaciones y aplicaciones reales del protocolo LDAP: Active Directory Active Directory es el nombre utilizado por Microsoft (desde Windows 2000) como almacén centralizado de información de uno de sus dominios de administración. Bajo este nombre se encuentra realmente un esquema (definición de los campos que pueden ser consultados). Novell Directory Services También conocido como eDirectory es la implementación de Novell utilizada para manejar el acceso a recursos en diferentes servidores y computadoras de una red. Básicamente está compuesto por una base de datos jerárquica y orientada a objetos, que representa cada servidor, computadora, impresora, servicio, personas, etc. entre los cuales se crean permisos para el control de acceso, por medio de herencia.

El autor Juan Luis Angel Cano Moreno, en la tesis titulada: "IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA" del año 2014, para optar el título profesional de Ingeniero en Ciencias y Sistemas en la Universidad San Carlos de Guatemala; en resumen: OpenLDAP es un sistema OpenSource que implementa un protocolo ligero de

acceso a directorios, por las siglas en inglés. Permite el almacenamiento de los usuarios en un sistema centralizado y especializado para el control de usuarios en una estructura de directorios. Asímismo, permite una alta gama de mecanismos de encriptación para las contraseñas de los usuarios, lo que permitió hacer correctamente la migración de usuarios de la base de datos relacional a este sistema. A medida que los sistemas informáticos proliferan para soportar los procesos del negocio, tanto los usuarios, como administradores de sistemas se enfrentan a una tarea complicada para completar las funciones laborales. Los usuarios típicamente se tienen que autenticar en múltiples sistemas, necesitando una pantalla de autenticación por cada uno de los sistemas, esto podría involucrar usuarios y contraseñas distintas, mientras que los administradores de sistemas se enfrentan a la tarea de estar administrando las cuentas de los usuarios en cada uno de estos sistemas, y de estarlos coordinando para que la información sea consistente e integra de acuerdo a las políticas de seguridad de la organización. El Centro de Cálculo e Investigación Educativa se estaba enfrentando a esta problemática, por lo que se detectó la oportunidad de mejora en la implementación de un sistema que permita la unificación de usuarios de los distintos aplicativos informáticos que son administrados por la institución y que al mismo tiempo les permitiera escalar en algún futuro a tecnologías que son manejadas a través de internet, como lo es, el sistema de inicio de sesión de Google. En conclusión: OpenAM es una plataforma poderosa de SSO que permite la integración con varias plataformas de programación web. En el caso de Java, permite la integración fácil a través del archivo de configuraciones web.xml y permite disminuir el tiempo de desarrollo, porque los desarrolladores se pueden enfocar en la lógica del negocio y no se deben de preocupar por el módulo de seguridad y autenticación, ya que SSO administra la seguridad de las aplicaciones.

CAPITULO III

MATERIALES Y MÉTODOS

3.1 MATERIALES

3.1.1 HARDWARE

Cliente (laptop)

- Procesador: Intel(R) Core(TM) i5-2430 2.40GHz
- Memoria instalada (RAM): 4.00GB de RAM.
- Tipo de sistema: Sistema Operativo de 64 bits Windows 10

Servidor (estación de trabajo)

- Procesador: intel(R) Core(TM)2 Duo E8500 3.16GHz
- Memoria instalada (RAM): 4.00GB
- Tipo de sistema: sistema operativo de 64bits Ubuntu 16.04-STABLE

3.1.2 SOFTWARE

- Sistema Operativo de 64 bits Windows 10.
- Sistema Operativo de 64 bits Ubuntu 16.04-STABLE.
- Capturador de paquetes Wireshark v2.2.1
- Microsoft Excel Profesional Plus 2013 v15.0.4569.1506

3.2 DISEÑO, NIVEL Y TIPO DE INVESTIGACIÓN

3.2.1 DISEÑO DE LA INVESTIGACIÓN

El diseño de la investigación es el plan o estrategia que el investigador deberá seguir para responder las preguntas planteadas por la investigación (Sabino, 1992). Bajo este enfoque la presente investigación es experimental y descriptiva, experimental porque para responder la pregunta principal es necesario someter el prototipo a un experimento bajo condiciones reales en campo, y descriptiva porque para lograr los objetivos específicos es necesario conocer los aspectos más importantes de las tecnologías de las que podemos disponer para la conclusión del prototipo.

3.2.2 NIVEL DE LA INVESTIGACIÓN

El nivel de investigación se refiere a la profundidad del conocimiento que se busca lograr con la investigación, por tanto el nivel de la presente investigación es perceptual, porque no se pretende tener un conocimiento profundo de las distintas partes que componen la presente investigación, y en cierto modo es exploratoria pues (Baptista Lucio, Hernadez Sampieri, & Fernandez Collado, 2006) señalan que las investigaciones exploratorias buscan abrir nuevos caminos en el desarrollo del conocimiento humano. Y la presente investigación siendo un prototipo busca abrir un camino para un nuevo método de monitoreo volcánico.

3.3 POBLACIÓN Y MUESTRA DE LA INVESTIGACIÓN

3.3.1 POBLACIÓN

Siendo la población el conjunto de las entidades, o cosas respecto a las cuales se basa las conclusiones de una investigación, para nuestro caso la población está definida

por los trabajadores de la empresa Claro en la región Puno. Se considera 570 trabajadores.

3.3.2 MUESTRA

Se define la muestra como parte que se estudia y es representativa de la población, es decir un segmento que tiene las características y propiedades de la población, por tanto la muestra está conformada por los trabajadores que hacen uso de una computadora de la oficina de control interno.

3.4 UBICACIÓN Y DESCRIPCION DE LA INVESTIGACIÓN

3.4.1 UBICACIÓN

El ámbito donde se realiza la investigación es en las instalaciones de los trabajadores de la empresa Claro de la ciudad de Puno entre el mes de setiembre del año 2016 hasta el mes de enero del año 2017, período 2017-I.

3.4.2 DESCRIPCIÓN DE LA INVESTIGACIÓN

La presente investigación pretende demostrar la utilidad e implementar un sistema de autenticación de usuarios en un dominio de servicios de directorio activo a nivel de sistema operativo, esta investigación consta de tres fases que se describen a continuación.

- **Fase 1:** Esta fase trata esencialmente búsqueda de información, consulta bibliográfica y sitios web, sobre plataformas de desarrollo, elección de componentes y técnicas o métodos implementación del sistema de autenticación.
- **Fase2:** Esta fase comprende el diseño e implementación del sistema y pruebas preliminares.

- **Fase 3:** Comprende la obtención y el proceso de verificación, que consiste en correlacionar los resultados, para luego analizar los resultados, ya que es necesario para ver el buen funcionamiento.

3.5 TECNICAS E INSTRUMENTOS DE RECOLECCION DE DATOS

La recolección de datos se refiere a cómo y qué medios se usan para la obtención de la información que será de utilidad para la corroboración de nuestras hipótesis. La técnica para la recolección de datos es por la observación, por el motivo de que se actúa sobre los resultados con un instrumento. El instrumento son las formas de observación de los resultados, pasos para la recolección para su posterior análisis.

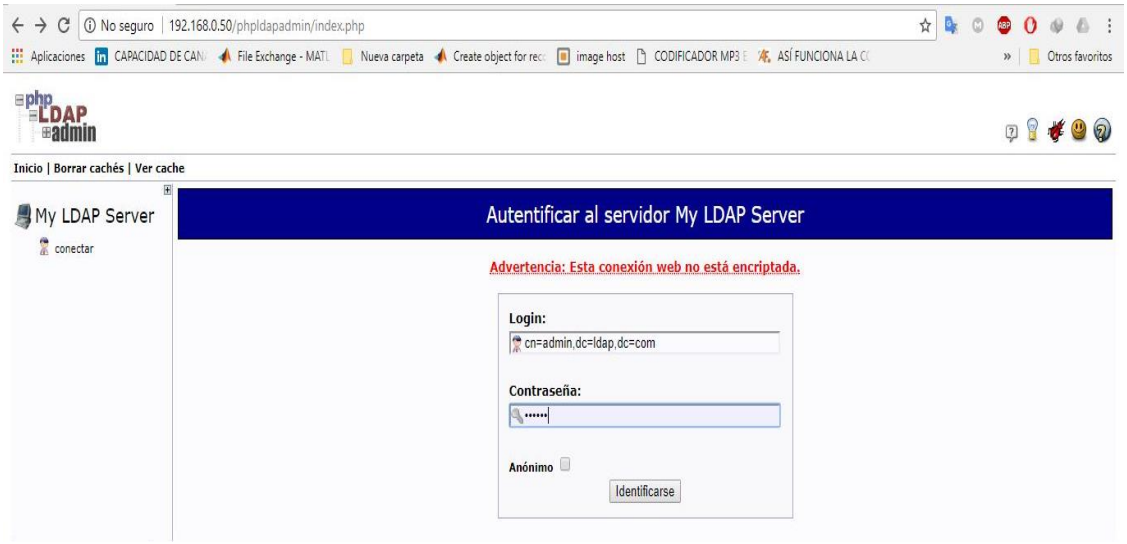
3.6 PROCEDIMIENTO DE IMPLEMENTACION

3.6.1 INSTALACIÓN DEL SERVIDOR LDAP

Lightweight Directory Access Protocol (LDAP) es un protocolo estándar diseñado para gestionar y acceder a la información del directorio jerárquico través de una red. Puede ser utilizado para almacenar cualquier tipo de información, aunque se utiliza más a menudo como un sistema de autenticación centralizada o para directorios de correo electrónico y números de teléfono de empresa.

Mostraremos la instalación y configuración el servidor OpenLDAP en Ubuntu 16.04. en el ANEXO A. Despues se ingresó con phpLDAPadmin desde el navegador de un equipo de la red. https://192.168.0.50/phpldapadmin.

Figura 3.1 - Ingreso al servidor phpLDAPadmin desde el navegador



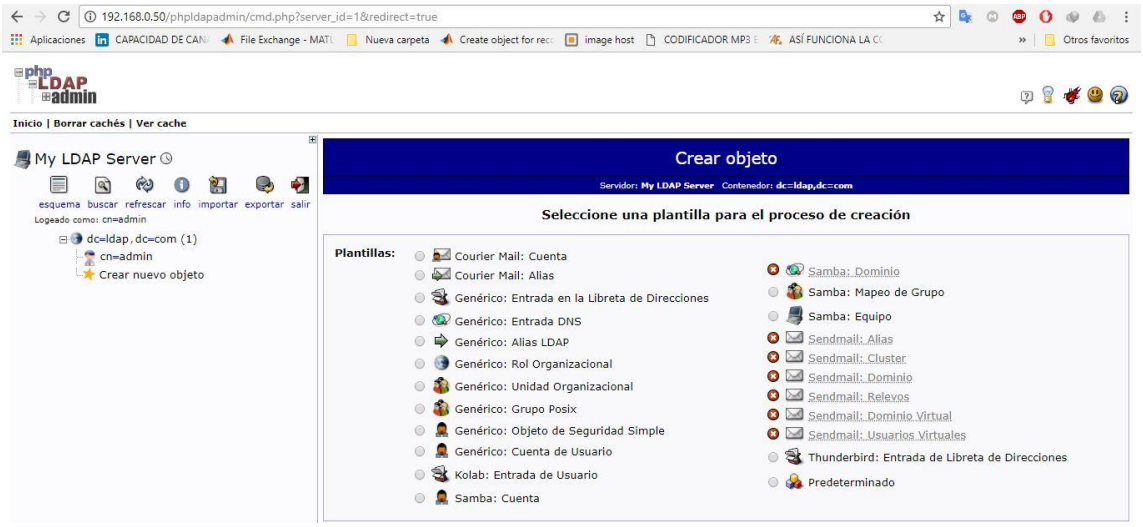
Elaboración propia

La página phpLDAPadmin se carga. Luego se hizo clic en el enlace conectar en el menú de la izquierda de la página. Se presentará un formulario de entrada, el **DN de entrada** es el nombre de usuario que va a utilizar. Contiene el nombre de cuenta como una cn=sección y el nombre de dominio que ha seleccionado para el servidor dividido en dc=secciones como se describe anteriormente. La cuenta de administrador por defecto que hemos creado durante la instalación se llama **admin**, por lo que para nuestro ejemplo podríamos escribir el siguiente: *cn=admin,dc=ldap,dc=com*

3.6.2 CONFIGURAR USUARIOS LDAP

Después de introducir la cadena apropiada para el dominio, se escribió la contraseña de administrador que ha creado durante la configuración. Después se observa el interfaz principal:

Figura 3.2 - Inicio de phpLDAPadmin para realizar las configuraciones



Elaboración propia

En este punto, se inicia la sesión en la interfaz phpLDAPadmin, Para añadir usuarios, unidades organizativas, grupos y las relaciones.

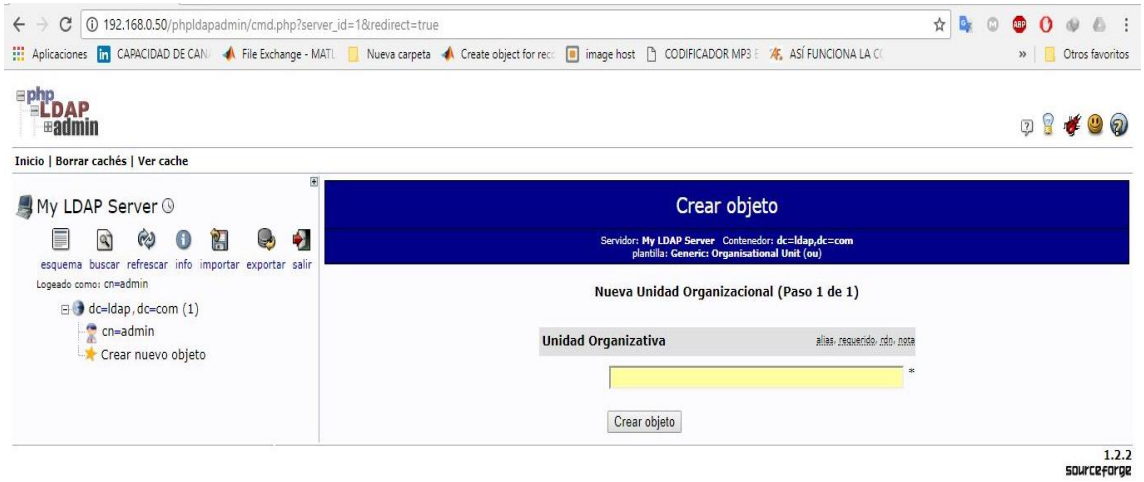
LDAP es muy flexible. Puede crear jerarquías y relaciones de muchas maneras diferentes, dependiendo de qué tipo de información necesita accesible y qué tipo de caso de uso tiene. Crearemos una estructura básica para nuestra información y luego la rellenaremos con información.

3.6.3 CREAR UNIDADES ORGANIZATIVAS

En primer lugar, Debido a que se trata de una configuración básica, sólo necesitaremos dos categorías: grupos y usuarios. En el enlace "Crear nueva objeto" en el lado izquierdo. (Figura 3.3) Después se puede ver los diferentes tipos de entradas que podemos crear.

Debido a que sólo estamos utilizando esto como una estructura organizativa, en lugar de una entrada de información pesada, se utilizara el "Genérico: Unidad Organizacional". Al ingresar al enlace se muestra la figura 3.3.

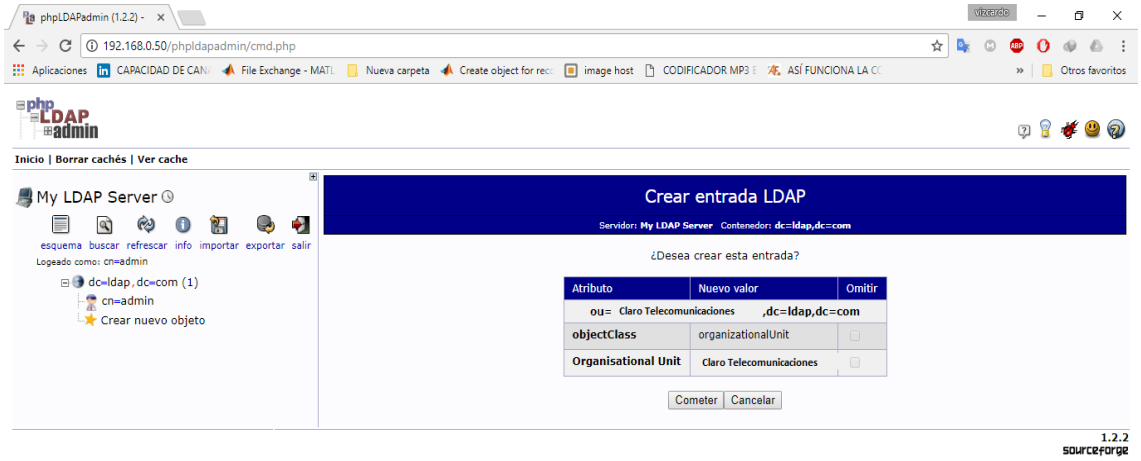
Figura 3.3 - Ventana para ingresar la Unidad Organizacional



Elaboración propia

Se solicita crear un nombre para nuestra unidad organizativa. Se creara con el nombre de “Claro Telecomunicaciones” .

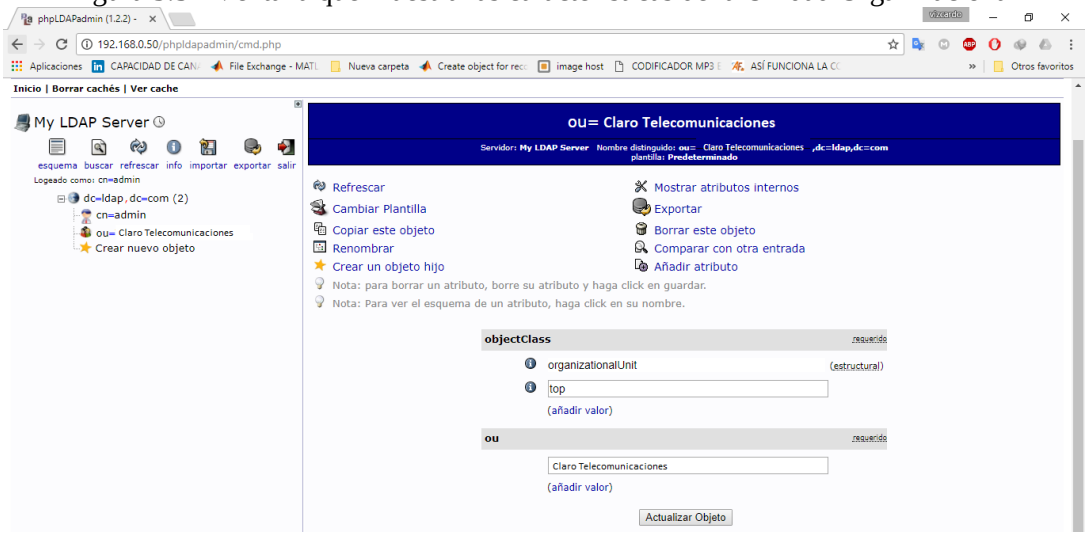
Figura 3.4 - Ventana para la confirmación de cambios



Elaboración propia

A continuación, tendrá que confirmar los cambios, haciendo clic a “cometer”. Cuando esto se complete, podemos ver una nueva entrada en el lado izquierdo. Ya se puede crear los grupos.

Figura 3.5 - Ventana que muestra las características de la Unidad Organizacional

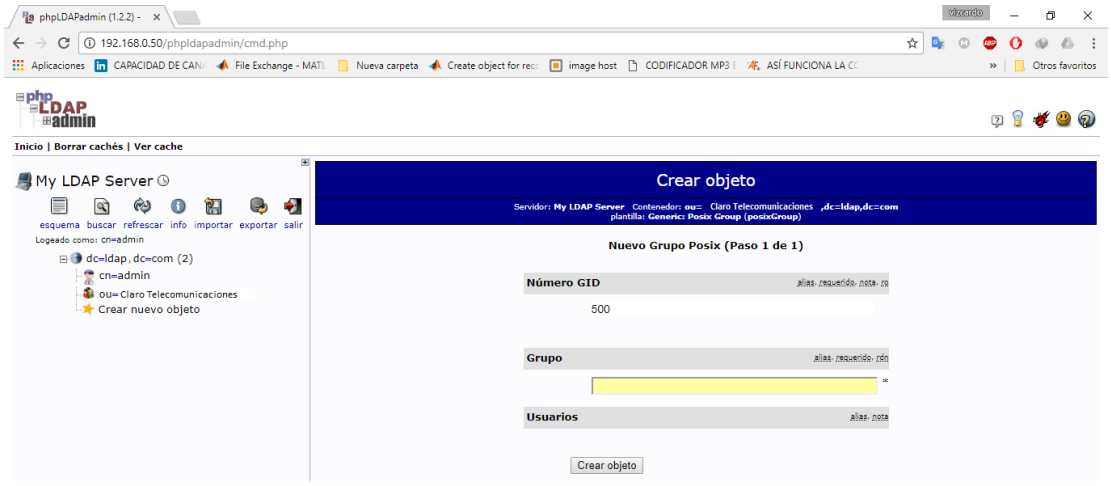


Elaboración propia

3.6.4 CREAR GRUPOS

A continuación, podríamos permitir que los miembros de diferentes grupos se autentican si configuramos la autenticación de cliente LDAP. Se creó los grupos dentro de la unidad organizativa " Claro Telecomunicaciones". Haga clic en la categoría " Claro Telecomunicaciones " que creamos. En el panel principal (figura 3.5), se hace clic en "crear un objeto hijo" dentro de la categoría de grupos. Esta vez, se eligió la categoría "Genérico: Posix Group".

Figura 3.6 - Ventana para ingresar el grupo



Elaboración propia

Se rellenó " Oficina de Soporte Técnico " como el nombre del grupo. Luego clic en "Crear objeto" y, a continuación, confirmar en la página siguiente.

Figura 3.7 - Ventana para la confirmación de cambios de grupos

Elaboración propia

Se repite el proceso, pero simplemente reemplace el nombre " Oficina de Soporte Técnico " por el nombre de otras oficinas del Claro Telecomunicaciones.

3.6.5 CREAR USUARIOS

A continuación, vamos a crear usuarios para poner en estos grupos. Haciendo clic en la categoría "cn = Oficina de Soporte Técnico". Y luego clic en "Crear un objeto hijo". Después elegimos "Genérico: cuenta de usuario " para estas entradas.

Figura 3.8 - Ventana para ingresar datos de usuario

Elaboración propia

Se rellena todas las entradas con información que tenga sentido para el usuario. Algo a tener en cuenta es que el "nombre común" debe ser único para cada entrada en una categoría. Por lo tanto, puede que desee utilizar un formato de nombre de usuario en lugar del predeterminado que se rellena automáticamente. Es necesario escribir la contraseña para la autenticación del usuario y después “crear objeto”.

Figura 3.9 - Características del usuario creado

Elaboración propia

La figura 3.9 muestra las características del usuario creado, tómese en cuenta el nombre de usuario ya que esto se usara para la autenticación.

3.6.6 CONFIGURAR CLIENTE LDAP EN WINDOWS

Para los clientes en Windows se usara el software pGina. PGina es un reemplazo para el proveedor de credenciales de Windows predeterminado, o GINA (para XP y anteriores). A través de los complementos, pGina le permite configurar muchos aspectos del proceso de inicio de sesión desde la autenticación y la autorización hasta los eventos de registro y terminales. Tenga en cuenta que el marco Proveedor de credenciales es compatible con Windows 7/Vista y superior. Las versiones anteriores de Windows utilizaban un marco llamado GINA. Para gran parte de esta guía, simplemente lo referiremos como Proveedor de Credenciales. Para la configuración se ingresa a la pestaña de “plugin selection” donde se selecciona las tres opciones de LDAP. Donde

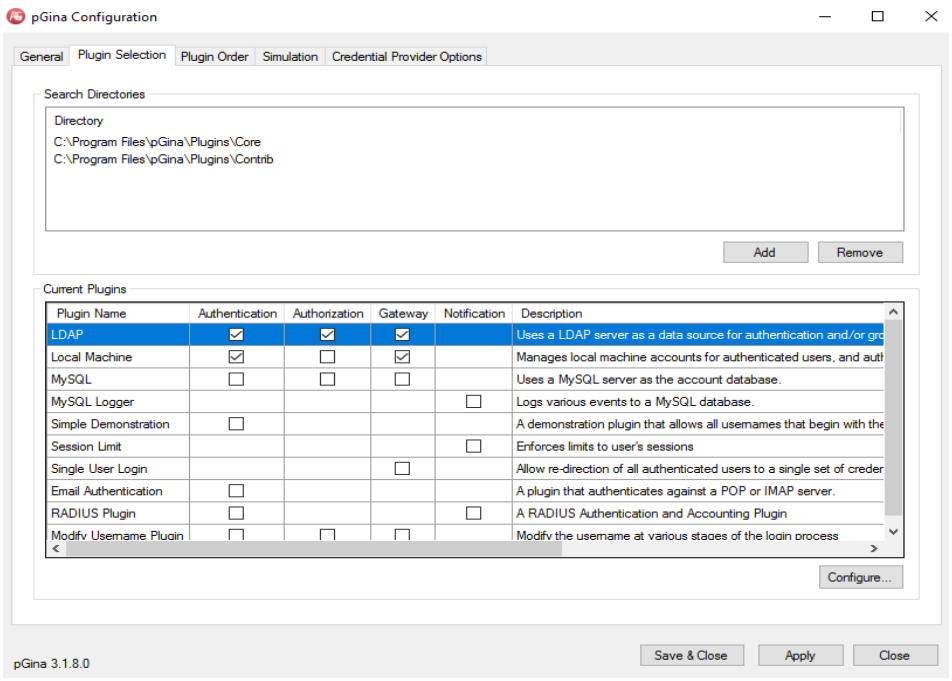
pGina gestiona el proceso de inicio de sesión delegando el trabajo a un conjunto de cero o más complementos. Los complementos tienen el trabajo de decidir si el usuario es quien dice ser (autenticación), si el usuario debe tener acceso (autorización) y tomar otras acciones de tiempo de inicio de sesión. Cada etapa incluye cero o más complementos, y un plugin dado puede estar involucrado en múltiples etapas. Después de que el usuario proporciona sus credenciales, pGina pasa a través de las tres etapas de una en una, en el orden mostrado arriba. Cada etapa puede tener éxito o fallar dependiendo de los resultados de los plugins implicados. El propósito de cada etapa se resume a continuación.

Autenticación - Plugins involucrados en esta etapa validan que el usuario es quien dice ser. Esto puede hacerse validando las credenciales contra alguna base de datos externa u otra fuente.

Autorización - Esta etapa tiene como objetivo determinar si el usuario (que ya está autenticado) puede acceder a los recursos que se solicitan. Por ejemplo, a un usuario solo se le puede permitir iniciar sesión si es miembro de ciertos grupos.

Gateway - Esto es similar a la etapa de Autorización, ya que puede fallar, sin embargo, la intención no es autorizar a los usuarios, sino proporcionar gestión de cuentas post-autorización que puede fallar. Si por alguna razón, esto falla, el usuario no puede iniciar sesión y este complemento debe detener el proceso de inicio de sesión y Proporcione un mensaje de error apropiado.

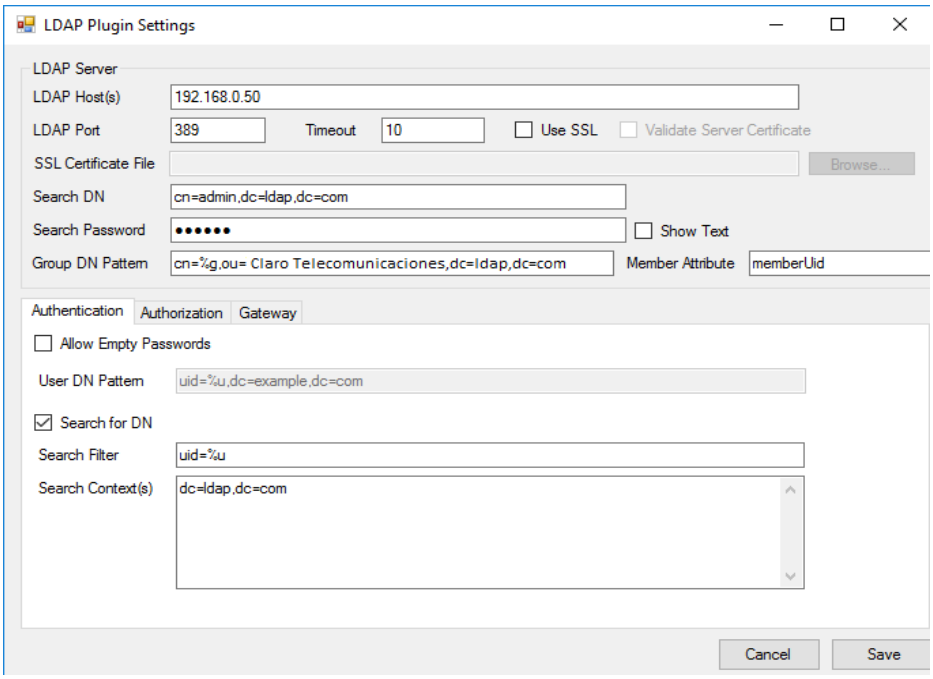
Figura 3.10 - Selección de plugin para el protocolo LDAP



Elaboración propia

Seleccionado los plugins se da clic en la opción configuración para la posterior autenticación, mostrando la figura 3.11.

Figura 3.11 - Configuración general y autenticación para conectar con el servidor LDAP



Elaboración propia

- **Host LDAP (s):** - Una lista separada por espacio de uno o más servidores LDAP. Este campo es compatible con direcciones IP o nombres de dominio completos.
- **Puerto LDAP** - El puerto que se utiliza cuando se conecta al servidor (s) de LDAP. Normalmente, esto es 389 para las conexiones no SSL (o conexiones utilizando StartTLS), y 636 cuando se usa SSL.
- **Tiempo de espera** - Este es el número de segundos para esperar una respuesta de un servidor antes de renunciar (y posiblemente de pasar al siguiente servidor de la lista).
- **Utilizar SSL** - Sea o no utilizar el cifrado SSL cuando se conecta al servidor (s).
- **Verificar certificado del servidor** - Si o no para verificar el certificado público del servidor con un certificado local o almacén de certificados. Cuando se selecciona esta opción, la conexión fallará si el certificado del servidor no valida.
- **Archivo de certificado SSL** (opcional) - Si ha seleccionado "verificar el certificado del servidor", puede proporcionar una copia del certificado SSL pública del servidor aquí. El certificado debe proporcionarse en el formato "PEM", que es el valor predeterminado para OpenSSL. Si este campo se deja en blanco, el plugin intentará utilizar el almacén de certificados de Windows para validar el certificado.
- **DN de búsqueda** - El DN utilizar al vincular al servidor con el fin de realizar las búsquedas. Si esto se deja vacío, el plugin intentará unir de forma anónima. Esto se utiliza para la búsqueda de pertenencia a un grupo o para la búsqueda de DN de un usuario.
- **Buscar Contraseña** - La contraseña a usar cuando se enlaza con el DN anterior. Esto es ignorado si el campo "Buscar DN" está vacía (enlace anónimo).
- **Grupo Pattern DN** - El patrón que se utiliza cuando la conversión de un nombre de grupo a un LDAP DN. Utilizar %g como un marcador de posición para el nombre del grupo. utilizando el plugin LDAP en la autorización y / o etapas de puerta de enlace.

- **Atributo Miembro** - El atributo LDAP que se utiliza para almacenar los miembros del grupo.

Las opciones de autenticación son las siguientes:

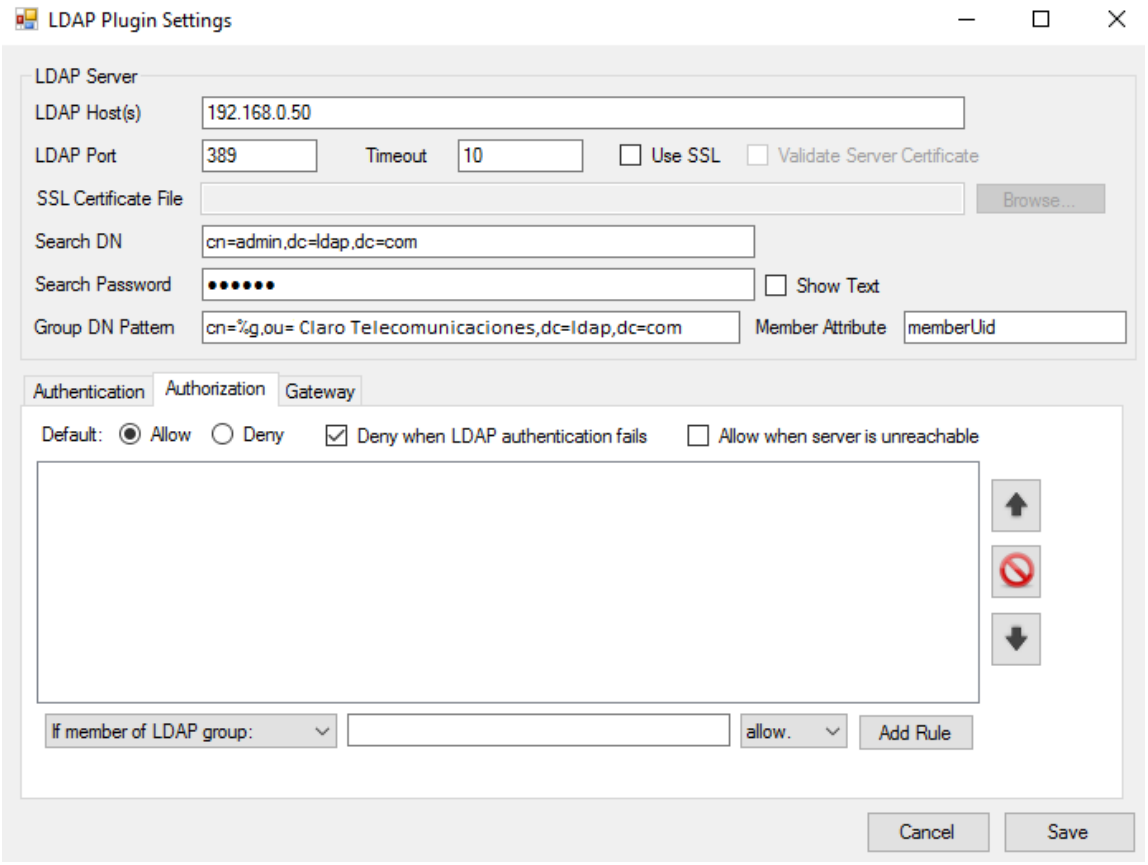
- **Permitir contraseñas vacías** - Si se marca esta opción, el plugin intentará autenticar al usuario, incluso si la contraseña está vacía. Cuando no está marcada, no será capaz de autenticar al usuario sin intentar enlazar con el servidor LDAP. Tenga en cuenta que algunos servidores pueden tratar una contraseña vacía como un enlace anónimo, por lo que es una buena idea dejar sin marcar (por defecto).
- **Patrón DN** - Si "Buscar DN" no está seleccionado, el nombre de usuario se asigna a un DN usando este patrón. La subcadena %u será reemplazado con el nombre de usuario.

Buscar DN - Cuando se selecciona esta opción, el "modelo de DN" (mencionado anteriormente) se ignora. En su lugar, el plugin intentará conectarse al servidor LDAP (mediante credenciales Search DN y Search Password), y la búsqueda de una entrada que utiliza el filtro de búsqueda y la búsqueda de contexto (s) proporcionado.

- **Filtro de búsqueda** - Se trata de un filtro de búsqueda LDAP para ser usado cuando se busca el DN. Para obtener más información sobre los filtros de búsqueda LDAP, consulte este RFC , o cualquier libro de LDAP. Si la cadena %u aparece en el filtro, será reemplazado por el nombre de usuario.

Búsqueda en contexto (s) - Esta es una lista de números internos (uno por línea) que se van a utilizar como contextos de búsqueda. Esto significa que la búsqueda se realizará en el subárbol LDAP arraigada en cada uno de estos números internos.

Figura 3.12 - Configuración de autorización para conectar con el servidor LDAP



Elaboración propia

La ficha de autorización proporciona una interfaz para crear, eliminar y eliminar las reglas de autorización. Las reglas son probadas por el plugin en orden y se aplica la primera regla de concordancia. Si ninguna de las reglas coincide, se aplica la regla por defecto. El valor por defecto se configura mediante los botones de radio en la parte superior de la interfaz de pestañas.

Las otras opciones de configuración se describen a continuación:

- **Denegar cuando falla la autenticación LDAP** - Si está marcada, la autorización falla cuando el plugin LDAP no puede autenticar al usuario en la etapa de autenticación o el plugin LDAP no se ejecuta en esa etapa.

- **Permitir al servidor es inalcanzable** - Cuando se activa esta opción, el plugin permite (éxito autorización) cuando el servidor LDAP no está disponible o se produce algún otro error que provoca una falta de contacto con el servidor LDAP. Si no está marcada, el usuario se le deniega la autorización en las mismas circunstancies.

CAPITULO IV

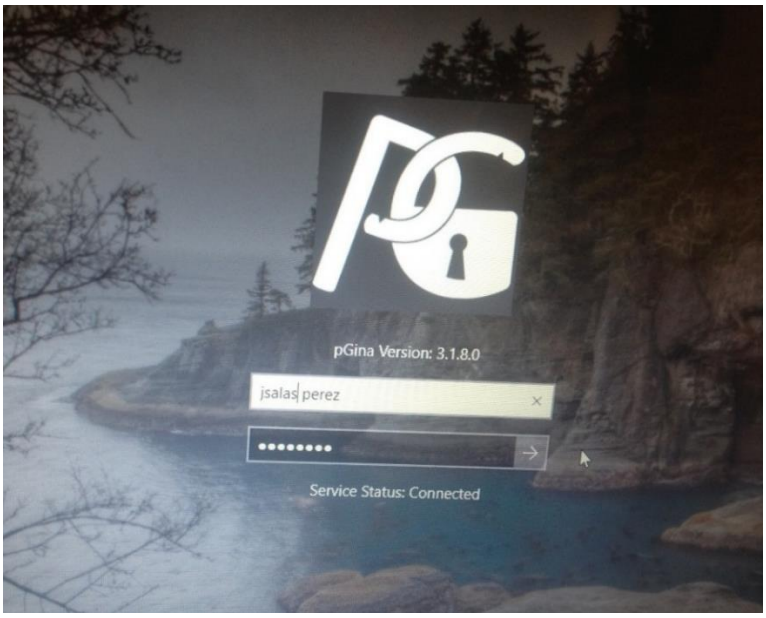
RESULTADOS Y DISCUSIÓN

Los resultados mostraran según cada resultado, es decir, uno en la cual la autenticacion del usuario y la contraseña es correcta, otra cuando la contraseña de autenticacion es incorrecta y al final si el usuario es incorrecto. Estos resultados son obtenidos tomando una foto del cliente en Windows y capturas de paquetes en el servidor, instalando Wireshark en ubuntu.

4.1 RESULTADOS CON USUARIO Y CONTRASEÑA CORRECTA

Explicada anteriormente presentaremos los resultados en el cliente, esto se muestra en la figura 4.1, donde se ingresó el nombre de usuario y contraseña configurados en la interfaz gráfica mostrada en el navegador. También se observa que el estado del servidor (service status) es “conectado” (connected).

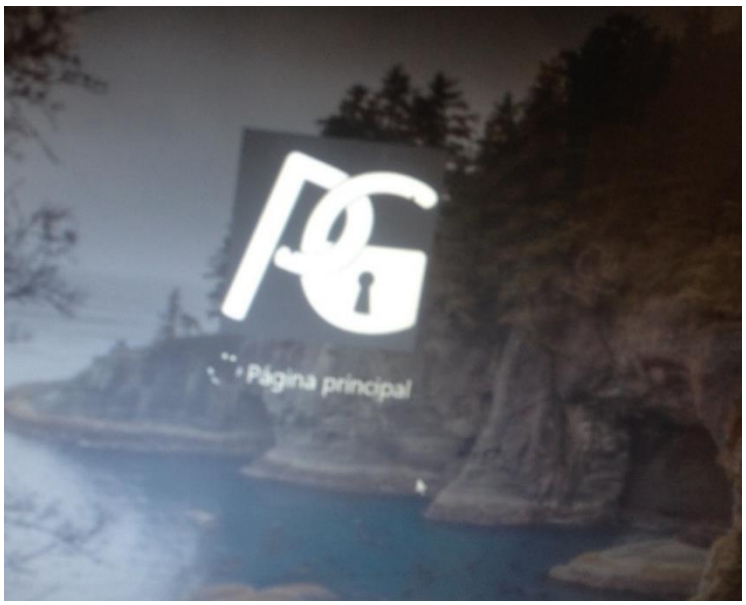
Figura 4.1 - Autenticación de usuario y contraseña



Elaboración propia

Después de ingresar correctamente el usuario y contraseña. El clientes consulta con el servidor y este le da acceso a la pc, tal como indica la figura 4.2.

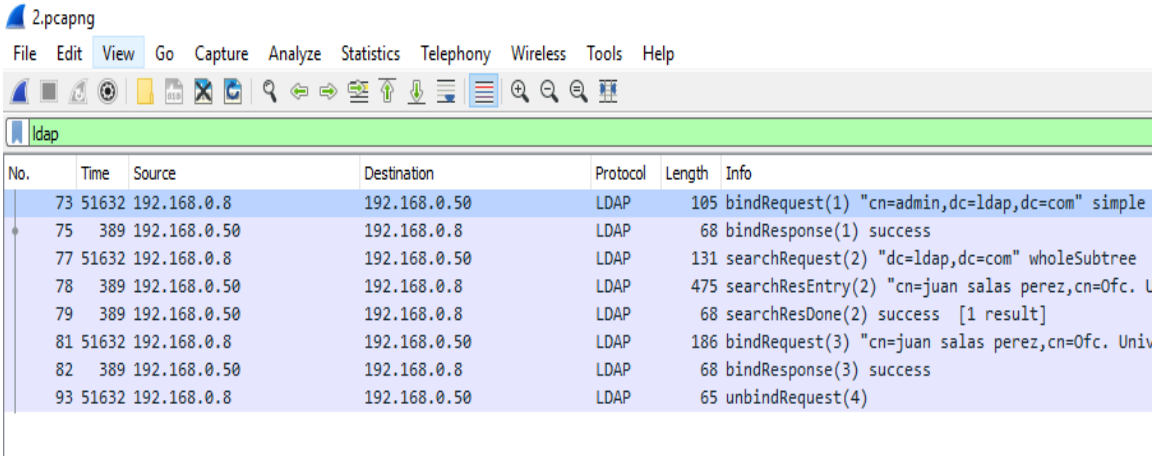
Figura 4.2 - Acceso autorizado en el cliente



Elaboración propia

Para ver lo sucedido en el servidor, analizaremos los paquetes LDAP de entrada (paquetes recibidas del cliente) y salida (paquetes enviados al cliente) del servidor, estos paquetes son capturados con el software Wireshark, tal como indica en la figura 4.3. La dirección IP del servidor es 192.168.0.50 y del cliente es 192.168.0.8.

Figura 4.3 - Paquetes LDAP en Wireshark



Elaboración propia

La cantidad de paquetes en total para la correcta autenticación son 8 observando la figura 4.3. Para iniciar el análisis se tomaran los dos primeros paquetes (nº 73 y 75) y posteriormente los paquetes siguientes.

Tabla 4.1 - Despliegue de los dos paquetes iniciales

No.	Time	Source	Destination	Protocol	Length	Info
73	51632	192.168.0.8	192.168.0.50	LDAP	105	bindRequest(1) "cn=admin,dc=ldap,dc=com" simple
Lightweight Directory Access Protocol LDAPMessage bindRequest(1) "cn=admin,dc=ldap,dc=com" simple messageID: 1 protocolOp: bindRequest (0) bindRequest version: 3 name: cn=admin,dc=ldap,dc=com authentication: simple (0) simple: 123456 [Response In: 75]						
75	389	192.168.0.50	192.168.0.8	LDAP	68	bindResponse(1) success
Lightweight Directory Access Protocol LDAPMessage bindResponse(1) success messageID: 1 protocolOp: bindResponse (1) bindResponse resultCode: success (0) matchedDN: errorMessage: [Response To: 73] [Time: 0.000326877 seconds]						

Elaboración propia

Ambos paquetes son relacionados ya que en la sección *messageID* son idénticas, esto ya que Todos los mensajes LDAP que encapsulan, las respuestas contienen el valor *messageID* idéntica a la solicitud del mensaje LDAP.

- BindRequest: es la secuenciación de la solicitud de enlace enviado por el cliente, tal como se muestra en el Tabla 4.1 este solicita acceso como administrador al servidor

ldap.com, en el campo de versión indica la versión del protocolo en este caso es la versión 3, en espacio de autenticación indica simple ya que no se configuro el tipo de encriptación en el cliente, por lo que se muestra la contraseña enviada.

- BindResponse: consiste simplemente en una indicación desde el servidor el estado de la solicitud del cliente para la autenticación. Si el enlace tuvo éxito, el resultCode muestra *success* esto indica que la conexión fue correcta.

Tabla 4.2 - Despliegue de los paquetes de autenticación de usuario

No	Time	Source	Destination	Protocol	Length	Info
77	51632	192.168.0.8	192.168.0.50	LDAP	131	searchRequest(2) "dc=ldap,dc=com" wholeSubtree
Lightweight Directory Access Protocol LDAPMessage searchRequest(2) "dc=ldap,dc=com" wholeSubtree messageID: 2 protocolOp: searchRequest (3) searchRequest baseObject: dc=ldap,dc=com scope: wholeSubtree (2) derefAliases: neverDerefAliases (0) sizeLimit: 0 timeLimit: 0 typesOnly: False Filter: (uid=jsalas perez) filter: equalityMatch (3) equalityMatch attributeDesc: uid assertionValue: jsalas perez attributes: 0 items [Response In: 78]						
78	389	192.168.0.50	192.168.0.8	LDAP	475	searchResEntry(2) "cn=juan salas perez,cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com"
Lightweight Directory Access Protocol LDAPMessage searchResEntry(2) "cn=juan salas perez,cn=Oficina de Soporte Técnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com" [1 result] messageID: 2 protocolOp: searchResEntry (4) searchResEntry objectName: cn=juan salas perez,cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com						

attributes: 10 items
PartialAttributeList item cn
type: cn
vals: 1 item
AttributeValue: juan salas perez
PartialAttributeList item givenName
type: givenNa me
vals: 1 item
AttributeValue: juan
PartialAttributeList item gidNumber
type: gidNumber
vals: 1 item
AttributeValue: 500
PartialAttributeList item homeDirectory
type: homeDirectory
vals: 1 item
AttributeValue: /home/users/jsalas perez
PartialAttributeList item sn
type: sn
vals: 1 item
AttributeValue: salas perez
PartialAttributeList item loginShell
type: loginShell
vals: 1 item
AttributeValue: /bin/sh
PartialAttributeList item objectClass
type: objectClass
vals: 3 items
AttributeValue: inetOrgPerson
AttributeValue: posixAccount
AttributeValue: top
PartialAttributeList item userPassword
type: userPassword
vals: 1 item
AttributeValue: {MD5}fWOgDEoSl91mv6qRPWqxRQ==
PartialAttributeList item uidNumber
type: uidNumber
vals: 1 item
AttributeValue: 1000
PartialAttributeList item uid
type: uid
vals: 1 item
AttributeValue: jsalas perez
[Response To: 77]
[Time: 0.000456537 seconds]

79	389	192.168.0.50	192.168.0.8	LDAP	68	searchResDone(2) success [1 result]

Lightweight Directory Access Protocol
LDAPMessage searchResDone(2) success [1 result]
messageID: 2
protocolOp: searchResDone (5)
searchResDone
resultCode: success (0)
matchedDN:
errorMessage:
[Response To: 77]
[Time: 0.000485283 seconds]

Elaboración propia

- searchRequest: en este campo indica la petición de búsqueda que tiene los siguientes valores, iniciando con *baseObject*, este es la entrada de objeto de base con relación a la que la búsqueda se va a realizar la cual es: *dc=ldap,dc=com*. la parte de *scope* es un indicador del alcance lo que muestra a *wholeSubtree, lo cual indica el alcance a sub árboles completo. D*erefAliases es un indicador son los alias de objetos (tal como se definen como deben ser gestionados las búsquedas), neverDerefAliases es par no eliminar la referencia del alias. SizeLimit y timeLimit muestran la cantidad máxima de entradas y el tiempo límite respectivamente, typesOnly se refiere si los resultados de búsqueda tendrán tipos de atributos y valores en este caso se encuentra desactivado. Filter muestra las condiciones de búsqueda, para resumir los resultados, en la petición enviada muestra que se desea buscar a al usuario jsalas perez.
- searchResEntry: muestra los resultados de la búsqueda enviados por el servidor al recibir una solicitud de búsqueda, se devuelven en las respuestas de búsqueda LDAP. Se observa en el cuadro 6 que en el campo *objectName* responde con las características del usuario que son resumidas, debajo de este se indica que hay 10 items, en la que se describe las diferentes tipos de campos configurados en el navegador tales como cn (nombres y apellidos), homeDirectory (ubicación de archivos), sn (apellidos), userPassword (encriptado con md5), (uid nombre del usuario)

- searchResDone: indica la cantidad de resultados y autentica con la petición, este
- campo indica *success,* es decir que la autenticación es correcta viendo esto se procederá a la autenticación de la contraseña.

Tabla 4.3 - Despliegue de los paquetes de autenticación de contraseña

No	Time	Source	Destination	Protocol	Length	Info
81	51632	192.168.0.8	192.168.0.50	LDAP	186	bindRequest(3) "cn=juan salas perez,cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com" simple
Lightweight Directory Access Protocol LDAPMessage bindRequest(3) "cn=juan salas perez,cn=Oficina de Soporte Técnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com" simple messageID: 3 protocolOp: bindRequest (0) bindRequest version: 3 name: cn=juan salas perez,cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com authentication: simple (0) simple: b40f0324 [Response In: 82]						
82	389	192.168.0.50	192.168.0.8	LDAP	68	bindResponse(3) success
Lightweight Directory Access Protocol LDAPMessage bindResponse(3) success messageID: 3 protocolOp: bindResponse (1) bindResponse resultCode: success (0) errorMessage: [Response To: 81] [Time: 0.000411957 seconds]						

Elaboración propia

- BindRequest: es la secuenciación de la solicitud de enlace enviado por el cliente, tal como se muestra en las tablas 4.1 y 4.3 este solicita acceso para el cliente consultado en la tabla 4.2 el cual es: *cn=juan salas perez, cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com* eso se realiza al servidor *ldap.com,* en el campo de

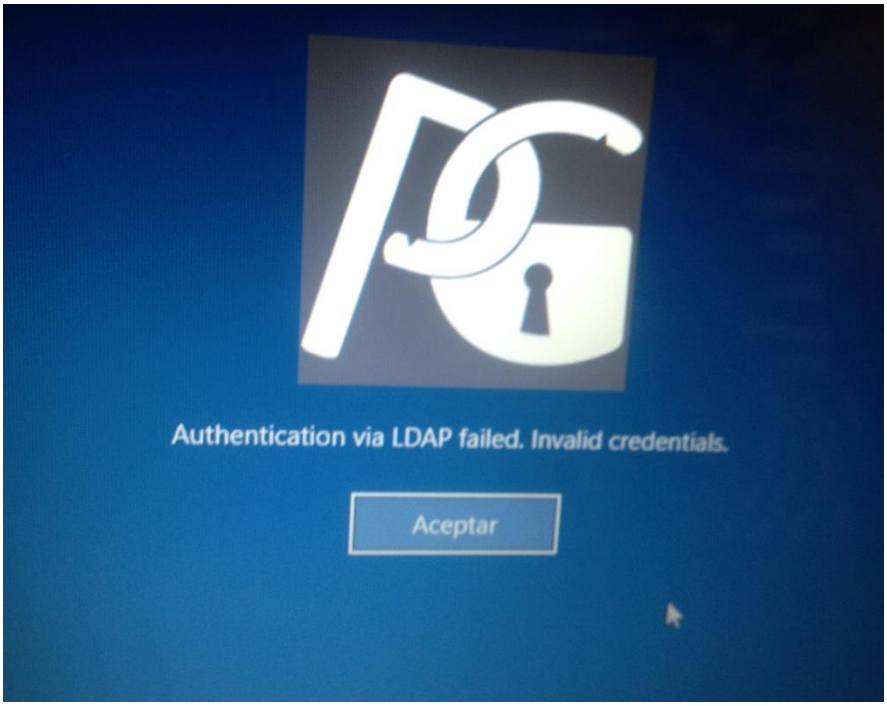
versión indica la versión del protocolo en este caso es la versión 3, en espacio de autenticación indica simple ya que no se configuro el tipo de encriptación en el cliente, por lo que se muestra la contraseña enviada.

- BindResponse: consiste simplemente en una indicación desde el servidor el estado de la solicitud del cliente para la autenticación. Si el enlace tuvo éxito, el resultCode muestra *success* esto indica que la conexión fue correcta.

4.2 RESULTADOS CON CONTRASEÑA INCORRECTA

Se ingresó una contraseña aleatoria diferente a lo registrado en el servidor LDAP. El resultado se muestra en la figura 4.4, en la que indica que la autenticación por LDAP ha fallado. Es inválido la contraseña.

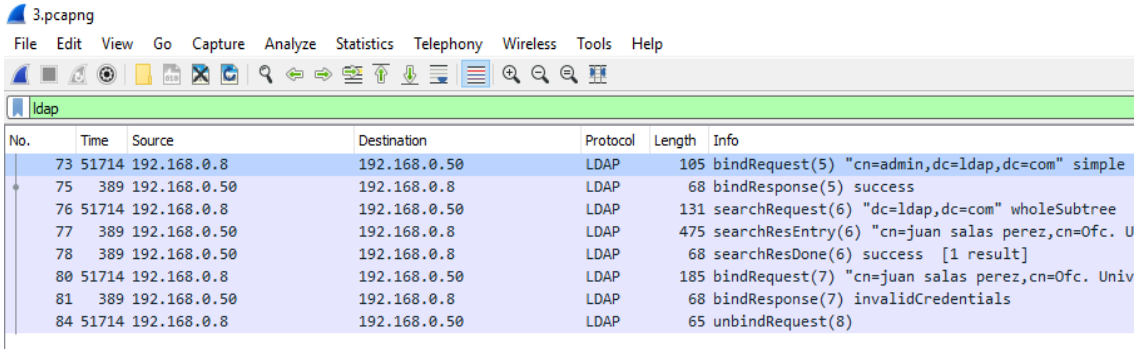
Figura 4.4 - Autenticación de contraseña fallida



Elaboración propia

El resultado en el cliente es lo esperado, pero se constatará por los datos en el servidor lo que se muestra en la figura 4.5.

Figura 4.5 - Paquetes de LDAP para la prueba de contraseña errónea



Elaboración propia

Alguno de estos paquetes son idéntico a los explicados en la sección anterior, tales como:

- BindRequest, para el acceso al servidor LDAP.
- BindRespose, donde otorga el acceso el servidor al cliente.
- SearchRequest, en este realiza la consulta del usuario, es idéntica a la prueba anterior ya que es el mismo usuario.
- searchResEntry, donde muestra los resultados de la búsqueda enviados por el servidor en este se devuelven en las respuestas de búsqueda LDAP idénticas a la prueba anterior.
- searchResDone, en esta prueba indica la misma cantidad de resultados y también este campo indica *success,* obtenidas anteriormente.

Ya mencionado toda los paquetes idénticos se proceden a analizar los paquetes de autenticación de contraseña.

Tabla 4.4 - Despliegue de paquetes LDAP de autenticación de contraseña

No	Time	Source	Destination	Protocol	Length	Info
80	51714	192.168.0.8	192.168.0.50	LDAP	185	bindRequest(7) "cn=juan salas perez,cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com" simple
Lightweight Directory Access Protocol LDAPMessage bindRequest(7) "cn=juan salas perez,cn=Oficina de Soporte Técnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com" simple messageID: 7 protocolOp: bindRequest (0) bindRequest version: 3 name: cn=juan salas perez,cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com authentication: simple (0) simple: 1234567 [Response In: 81]						
81	389	192.168.0.50	192.168.0.8	LDAP	68	bindResponse(7) invalidCredentials
Lightweight Directory Access Protocol LDAPMessage bindResponse(7) invalidCredentials messageID: 7 protocolOp: bindResponse (1) bindResponse resultCode: invalidCredentials (49) matchedDN: errorMessage: [Response To: 80] [Time: 0.000256496 seconds]						

Elaboración propia

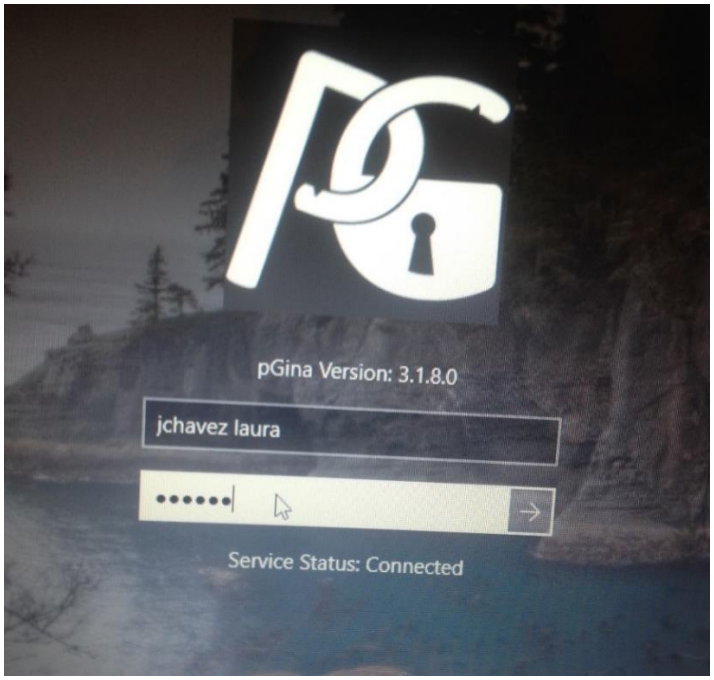
- BindRequest: este solicita acceso para el cliente *cn=juan salas perez,cn=Oficina de Soporte Tecnico,ou=Claro Telecomunicaciones,dc=ldap,dc=com* consultado en la tabla. En el espacio de autenticación indica simple ya que no se configuro el tipo de encriptación en el cliente, por lo que se muestra la contraseña enviada, esta es diferente a la prueba anterior.

- BindResponse: el resultCode no muestra *success* esto indica que la conexión fue incorrecta, por lo que se muestra *invalidCredencial*.

4.3 RESULTADOS CON USUARIO INCORRECTO

Para esta prueba se ingresa el nombre de usuario aleatorio pero idéntico al formato usado para la primera prueba, el usuario ingresado se muestra en la figura 4.6.

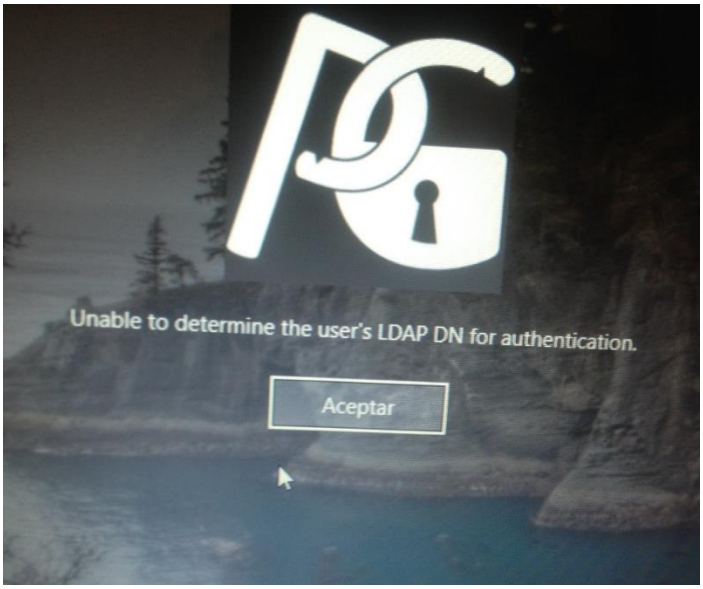
Figura 4.6 - Prueba con usuario no registrado en el servidor LDAP



Elaboración propia

La contraseña ingresada es aleatoria pero no es importante ya que no realizara la autenticación de contraseña. El resultado en el cliente se observa en la figura 4.7, que indica que no puede determinar el DN LDAP del usuario para la autenticación.

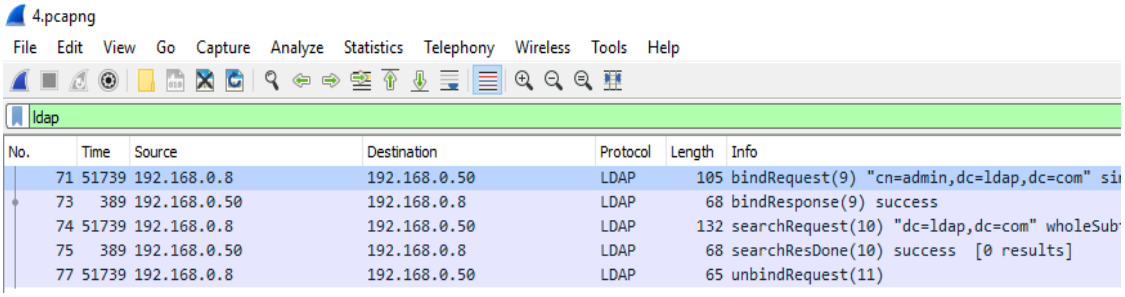
Figura 4.7 - Resultado en el cliente de usuario erróneo.



Elaboración propia

Para ver los resultados en el servidor también se analizaran los paquetes LDAP mostrados en la figura 4.8. Se observa que la cantidad de paquetes se ha disminuido. Los paquetes para la autenticación de la contraseña no se encuentran, ya que el cliente determina que no existe el usuario y por consecuencia tampoco la contraseña.

Figura 4.8 - Paquetes de LDAP para la prueba de usuario erróneo



Elaboración propia

Para el análisis más detallado, desplegaremos los paquetes que realizan la autenticación del usuario. Mencionamos que los dos primeros paquetes son idénticos a la prueba anterior.

Tabla 4.5 - Despliegue de los paquetes LDAP para la autenticación de usuario

No	Time	Source	Destination	Protocol	Length	Info
74	51739	192.168.0.8	192.168.0.50	LDAP	132	searchRequest(10) "dc=ldap,dc=com" wholeSubtree
Lightweight Directory Access Protocol LDAPMessage searchRequest(10) "dc=ldap,dc=com" wholeSubtree messageID: 10 protocolOp: searchRequest (3) searchRequest baseObject: dc=ldap,dc=com scope: wholeSubtree (2) derefAliases: neverDerefAliases (0) sizeLimit: 0 timeLimit: 0 typesOnly: False Filter: (uid=jchavez laura) filter: equalityMatch (3) attributeDesc: uid assertionValue: jchavez laura attributes: 0 items [Response In: 75]						
75	389	192.168.0.50	192.168.0.8	LDAP	68	searchResDone(10) success [0 results]
Lightweight Directory Access Protocol LDAPMessage searchResDone(10) success [0 results] messageID: 10 protocolOp: searchResDone (5) resultCode: success (0) errorMessage: [Response To: 74] [Time: 0.000431883 seconds]						

Elaboración propia

- searchRequest: en este campo indica la petición de búsqueda que tiene los siguientes valores. Todos los campos son idénticos a la prueba inicial pero la diferencia se presenta el Filter. El Filter muestra las condiciones de búsqueda, para resumir los resultados, en la petición enviada muestra que se desea buscar a al usuario *jchavez laura.*

- searchResDone: indica la cantidad de resultados y autentica con la petición, este campo indica ningún resultado pero si indica *success*, ya que la autenticación es no se realizó con ninguna opción idéntica.

CONCLUSIONES

Primero: Se ha logrado diseñar e implementar un sistema de autenticación de usuarios en un dominio de servicios de directorio activo a nivel de sistema operativo para el control de los trabajadores de la empresa Claro en el período 2017-I. Este sistema se ha implementado como prototipo, es decir un sistema funcional en laboratorio, y aunque no se ha implementado en el entorno real, la funcionalidad probada en laboratorio puede servir para sacar conclusiones pero no hacer generalidades.

Segundo: Es posible hacer la autenticación de usuarios de clientes Windows 10 al servidor Ubuntu GNU/Linux, de esta forma el diseño e implementación del sistema de autenticación de usuarios en un dominio de servicios de directorio activo a nivel del sistema operativo Windows se ha logrado con el objetivo de controlar a los trabajadores de la empresa Claro en el primer período del año 2017. Sin embargo, para lograr esto fue necesario diseñar e implementar un dominio de servicios de directorio activo con el software phpLDAPadmin que permitió la administración del directorio activo de forma remota por un entorno web.

Tercero: También fue necesario mejorar el control y monitoreo de usuarios basándose en la autenticación cuando el usuario accede a la máquina. LDAP trabajó confiablemente para las pruebas, es ligero en términos de uso de recursos tanto en maestro como en réplica, es altamente configurable, en desarrollo activo, razonablemente bien documentado y está bien soportado en la lista de OpenLDAP. Si estuviéramos procesando un gran número de actualizaciones, sin embargo, nuestro maestro LDAP normalmente recibe entre 50 y 100 cambios por hora, lo cual es bastante mínimo y no genera tráfico maestro-réplica significativo para propósitos de sincronización. De esta forma se ha logrado controlar el acceso de los trabajadores (usuarios) a las máquinas

usando su respectivo usuario y contraseña; así se tiene un registro de la dirección de las que acceden los usuarios, la fecha y la hora. Cuando un usuario ingrese a un sitio indebido, el servidor afectado registrará su dirección IP y LDAP registrará qué usuario usó esa dirección IP, sin el uso de LDAP solamente se podría saber la dirección IP de la máquina y no se tendría conocimiento del usuario que actuó en ella.

RECOMENDACIONES

Se ha logrado establecer el servidor LDAP en GNU/Linux, agregar usuarios en este servidor y finalmente autenticar los usuarios con el uso de PGina en clientes Windows 10. Para esto se puede agregar clientes GNU/Linux, Android e iOS, sin embargo son necesarias las aplicaciones como PGina en Windows que permitan la autenticación.

Active Directory es un proveedor de servicios de directorio, donde puede agregar un nuevo usuario a un directorio, eliminar o modificar, especificar privilegios, asignar la política, etc. Es justo como un directorio telefónico donde cada persona tiene un número de contacto único. En AD (Active Directory) se considera como objetos y cada objeto se le da un identificador único (similar a un número de contacto único en un directorio telefónico.

Lightweight Directory Access Protocol o LDAP, es una especificación basada en estándares para interactuar con los datos del directorio. Los servicios de directorio pueden implementar la compatibilidad de LDAP para proporcionar interoperabilidad entre aplicaciones de terceros. Active Directory es la implementación de Microsoft de un servicio de directorio que, entre otros protocolos, soporta LDAP para consultar sus datos. Aunque admite LDAP, Active Directory proporciona una gran cantidad de extensiones y conveniencias, como la caducidad de contraseñas y el bloqueo de cuentas.

El servicio de directorio es un sistema de software que almacena, organiza y proporciona acceso a información en el directorio del sistema operativo de un ordenador. En la ingeniería de software, un directorio es un mapa entre nombres y valores. Permite la búsqueda de valores nombrados, similar a un diccionario.

REFERENCIAS BIBLIOGRÁFICAS

Arnaert, M. (2016). Complete Debian Server 2016 Kindle Edition. United States: Lulu.com.

Binnie, C. (2016). Linux Server Security: Hack and Defend 1st Edition. United States: Wiley.

Bresnahan, C. and Blum, R. (2015). LPIC-1 Linux Professional Institute Certification Study Guide: Exam 101-400 and Exam 102-400 4th Edition. United States: Sybex.

Bresnahan, C. and Blum, R. (2015). CompTIA Linux+ Powered by Linux Professional Institute Study Guide: Exam LX0-103 and Exam LX0-104 (Comptia Linux + Study Guide) 3rd Edition. United States: Sybex.

Bresnahan, C. and Blum, R. (2016). LPIC-2: Linux Professional Institute Certification Study Guide: Exam 201 and Exam 202 2nd Edition. United States: Sybex.

Brunson, R. and Walberg, S. (2015). CompTIA Linux+ / LPIC-1 Cert Guide: (Exams LX0-103 & LX0-104/101-400 & 102-400) (Certification Guide) 1st Edition. United States: Pearson IT Certification.

Bryant, C. (2015). Chris Bryant's CCNP SWITCH 300-115 Study Guide (Ccnp Success) Paperback. United States: CreateSpace Independent Publishing Platform.

Butcher, M. (2012). Mastering OpenLDAP. Birmingham: Pactk Publishing.

Bryant, C. (2016). Chris Bryant's CCNP ROUTE 300-101 Study Guide Paperback. United States: CreateSpace Independent Publishing Platform.

Colvin, H. (2015). VirtualBox: An Ultimate Guide Book on Virtualization with VirtualBox Paperback. United States: CreateSpace Independent Publishing Platform.

Crawley, D. R. (2010). The Accidental Administrator: Linux Server Step-by-Step Configuration Guide Paperback. United States: CreateSpace Independent Publishing Platform.

Davis, J. A. Baca, S. and Thomas, O. (2016). VCP6-DCV Official Cert Guide (Exam #2V0-621) (3rd Edition) (VMware Press Certification) 3rd Edition. United States: VMware Press.

Desmond, B., Richards, J., Allen, R. and Lowe-Norris, A. (2013). Active Directory. Sebastopol, CA: O'Reilly & Associates.

Ferguson, B. (2016). vSphere 6 Foundations Exam Official Cert Guide (Exam #2V0-620): VMware Certified Professional 6 (VMware Press) 1st Edition. United States: VMware Press.

Ghori, A. (2015). RHCSA & RHCE Red Hat Enterprise Linux 7: Training and Exam Preparation Guide (EX200 and EX300), Third Edition 3rd Edition. United States: Endeavor Technologies Inc.

Greene, J. (2014). LPIC-1 Primer Kindle Edition. United States: John Greene.

Greenblatt, B. (2002). Building LDAP-enabled applications with Microsoft's Active Directory and Novell's NDS. Upper Saddle River, NJ: Prentice Hall PTR.

Juliet Kemp. (2009). Linux system administration recipes : a problem-solution approach. Apress.

Kouka, A. (2015). Ubuntu Server Essentials Paperback. United States: Packt Publishing - ebooks Account.

LaCroix, J. (2016). Mastering Ubuntu Server Paperback. United States: Packt Publishing - ebooks Account.

Lammle, T. (2013). CCNA Routing and Switching Study Guide: Exams 100-101, 200-101, and 200-120 1st Edition. United States: Sybex.

Lammle, T. (2016). CCNA Routing and Switching Complete Deluxe Study Guide: Exam 100-105, Exam 200-105, Exam 200-125 2nd Edition. . United States: Sybex.

Lammle, T. (2016). CCNA Routing and Switching Complete Study Guide: Exam 100-105, Exam 200-105, Exam 200-1252nd Edition. United States: Sybex.

Main Page - phpLDAPadmin. (2017). Phpldapadmin.sourceforge.net. Retrieved 22 October 2017, from http://phpldapadmin.sourceforge.net/wiki/index.php/Main_Page

Marshall, N. Lowe, S. Orchard, G. and Atwell, J. (2015). Mastering VMware vSphere 6 1st Edition. United States: Sybex.

Nathan, S. (2015). VirtualBox at Warp Speed: Virtualization with VirtualBox Kindle Edition. United States: Senthil Nathan.

Negus, C. (2015). Linux Bible 9th Edition. United States: Wiley.

Nutter, R. (2014). VMware - A Guide for New Admins Kindle Edition. United States: TechBytes Press.

Odom, W. (2016). CCENT/CCNA ICND1 100-105 Official Cert Guide 1st Edition. United States: Cisco Press.

Odom, W. (2016). CCNA Routing and Switching 200-125 Official Cert Guide Library 1st Edition. United States: Cisco Press.

Odom, W. (2016). CCNA Routing and Switching ICND2 200-105 Official Cert Guide 1st Edition. United States: Cisco Press.

Odom, W. (2013). CCENT/CCNA ICND1 100-101 Official Cert Guide 1st Edition. United States: Cisco Press.

Hynes, Byron (November 2006). "The Future Of Windows: Directory Services in Windows Server "Longhorn"". TechNet Magazine. Microsoft. https://technet.microsoft.com/en-us/magazine/2006.11.futureofwindows.aspx

Thomas, Guy. "Windows Server 2008 - New Features". ComputerPerformance.co.uk. Computer Performance Ltd. http://www.computerperformance.co.uk/Longhorn/longhorn_new_features.htm

"The LDAP Application Program Interface". Retrieved 2013-11-26. http://www.ietf.org/rfc/rfc1823.txt

"Active Directory on a Windows Server 2003 Network". Active Directory Collection. Microsoft. 13 March 2003. Retrieved 25 December 2010. http://technet.microsoft.com/en-us/library/cc780036(WS.10).aspx#w2k3tr_ad_over_qbjd

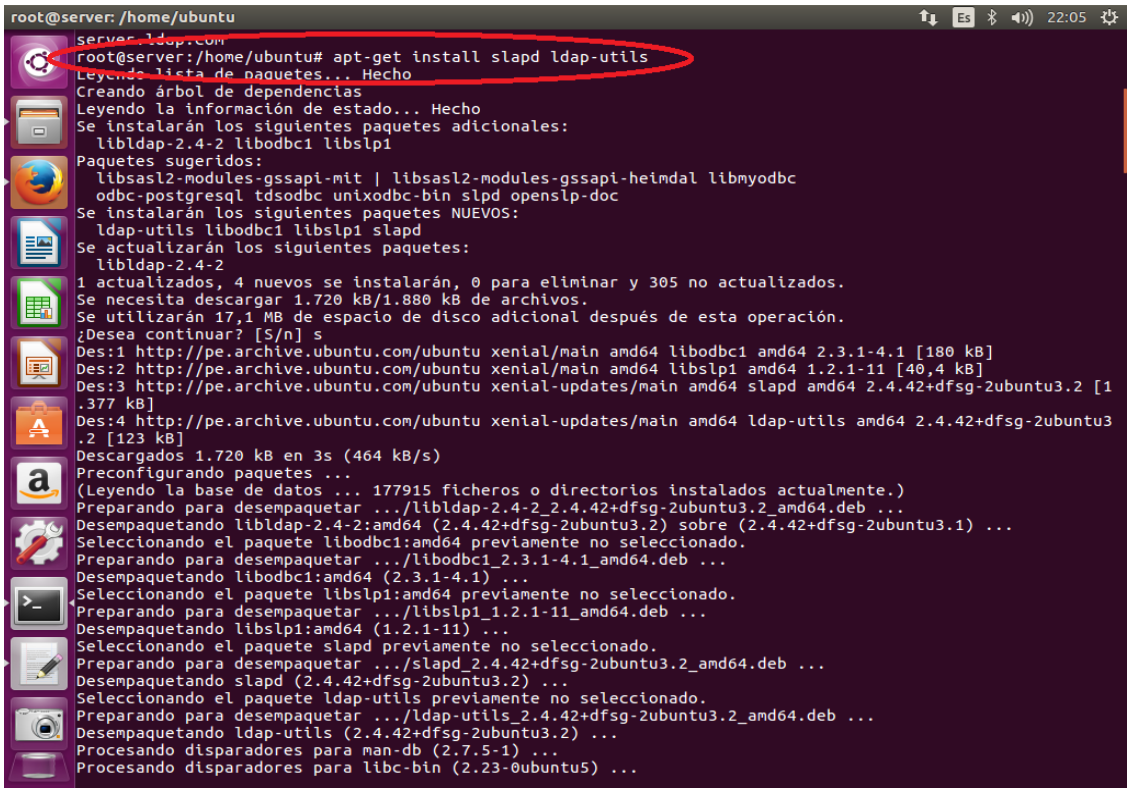
Microsoft Server 2008 Reference, discussing shadow groups used for fine-grained password policies. http://technet.microsoft.com/en-us/library/cc770394%28WS.10%29.aspx

ANEXO: CONFIGURACION DEL SERVIDOR LDAP EN UBUNTU

Para comenzar, debe tener un de Ubuntu desktop 16.04 configurar con Apache y PHP. Nuestro primer paso es instalar el servidor LDAP y algunas utilidades asociadas. Por suerte, los paquetes que necesitamos están todos disponibles en los repositorios por defecto de Ubuntu.

Iniciar sesión en el servidor. Dado que esta es la primera vez que el uso apt-get de esta sesión, vamos a refrescar nuestro índice local de paquetes, a continuación, instalar los paquetes que queremos; con el comando sudo *apt-get install slapd ldap-util*, como se muestra en la figura 1:

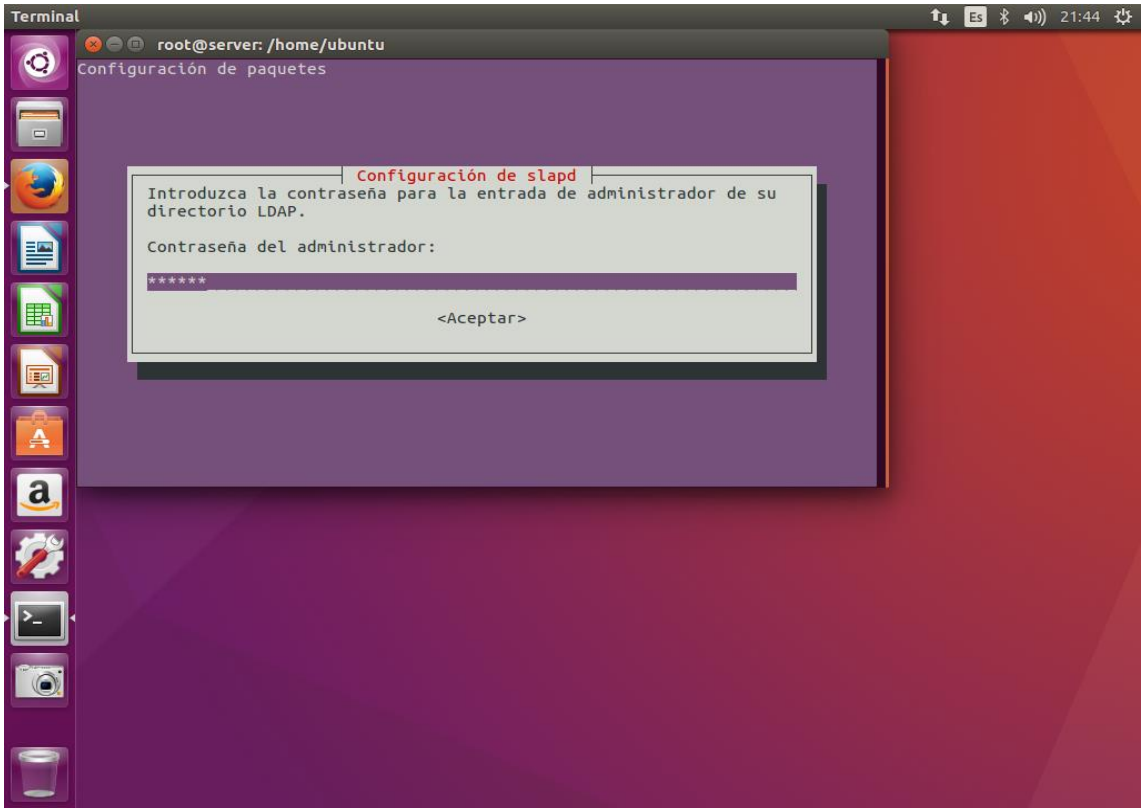
Figura 1 - Ejecución del comando para descarga e instalación paquetes necesarios para LDAP



Elaboración propia

Durante la instalación, se pedirá que seleccione y confirme una contraseña de administrador para LDAP. Esta contraseña será necesaria posteriormente, si se comete un error se podrá cambiar ya que se tendrá la oportunidad de actualizar en otro momento.

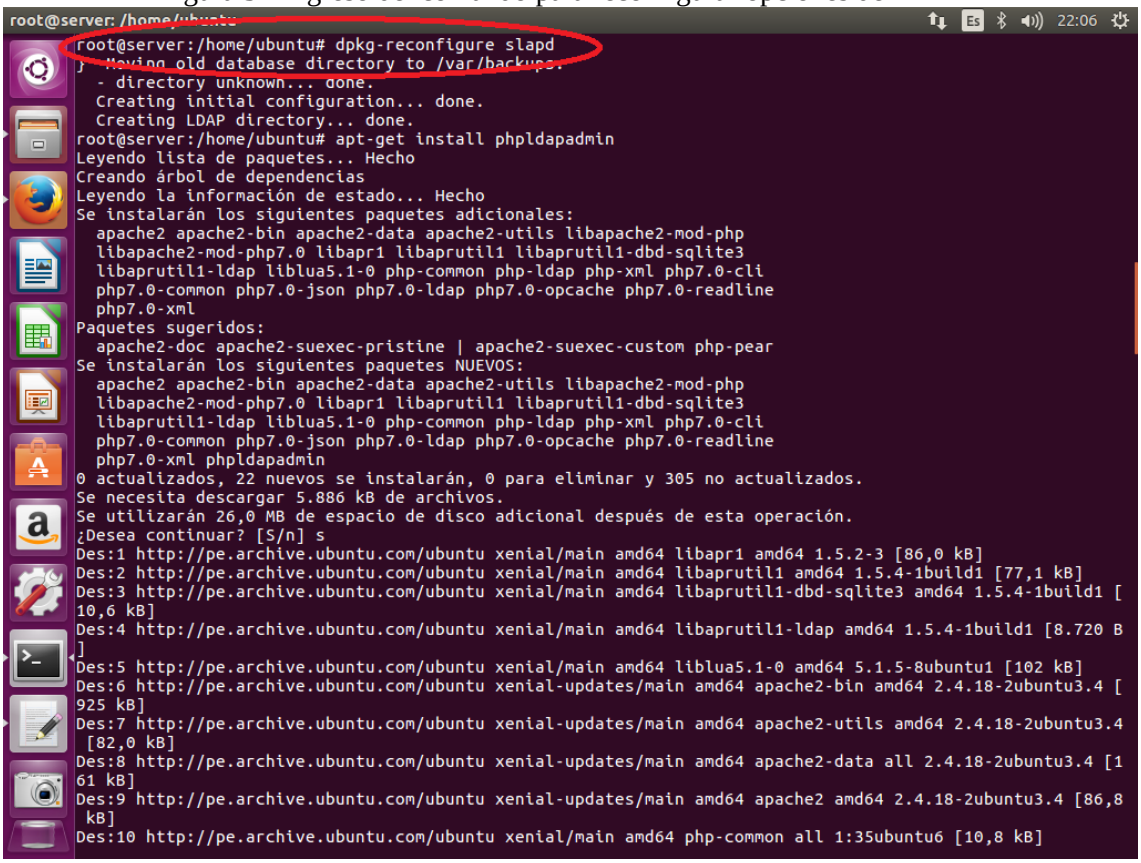
Figura 2 - Ingreso de contraseña para la administración del directorio LDAP



Elaboración propia

A pesar de que acabamos de instalar el paquete, vamos a seguir adelante y reconfigurarlo. El slapd paquete tiene la capacidad de hacer un montón de preguntas de configuración importantes, pero por defecto, que se pasó por alto en el proceso de instalación. Tenemos acceso a todas las preguntas diciendo a nuestro sistema para reconfigurar el paquete, con el comando sudo *dpkg-reconfigure slapd*:

Figura 3 - Ingreso del comando para reconfigurar opciones de LDAP

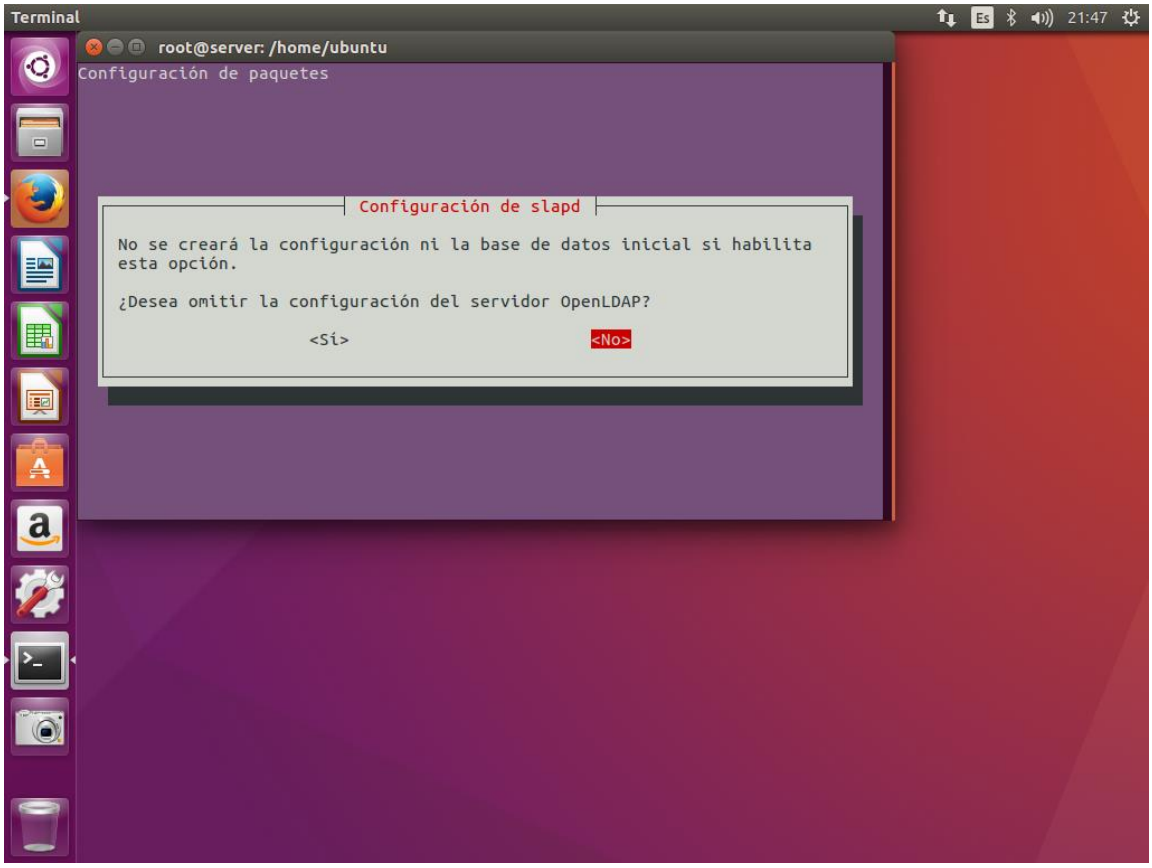


Elaboración propia

Hay un buen número de nuevas preguntas para responder en este proceso. Estaremos aceptando la mayor parte de los valores por defecto. Vamos a través de las preguntas:

La primera pregunta que realiza es: ¿Desea omitir la configuración del servidor OpenLDAP?

Figura 4 - Inicio de dialogo para iniciar a configurar opciones de OpenLDAP



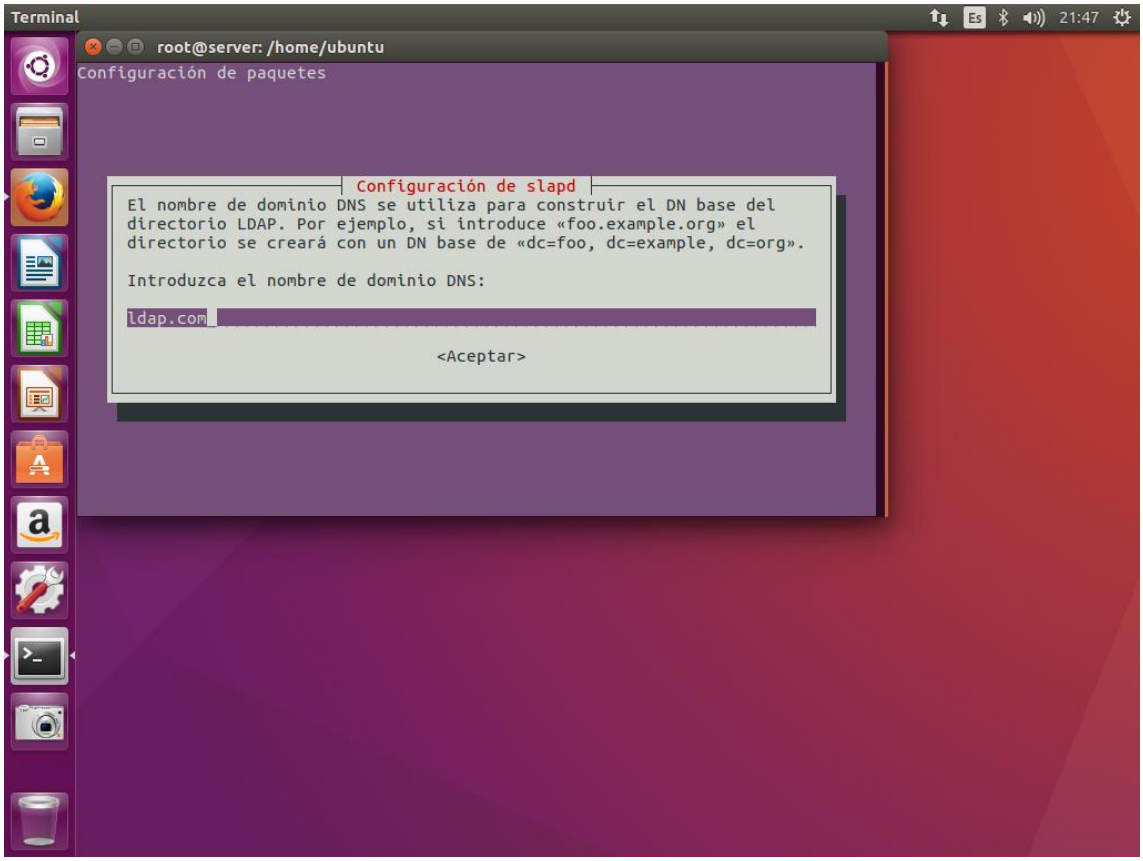
Elaboración propia

Como indica la figura 4 la respuesta es “no”, ya que debemos configurar el servidor OpenLDAP

Nombre de dominio DNS:

Esta opción determinará la estructura base de la ruta del directorio. Lea el mensaje a entender exactamente cómo se va a implementar. En realidad se puede seleccionar cualquier valor que desea, incluso si usted no es dueño del dominio real. El nombre de dominio se utilizará para crear el DN base del directorio LDAP, si se introduce “foo.example.org” el directorio se creara con un DN de “dc:foo,dc=example,dc=org”, Vamos a utilizar ldap.com en todo la instalación, Como se ve en la figura 5.

Figura 5 - Ingreso del domino DNS

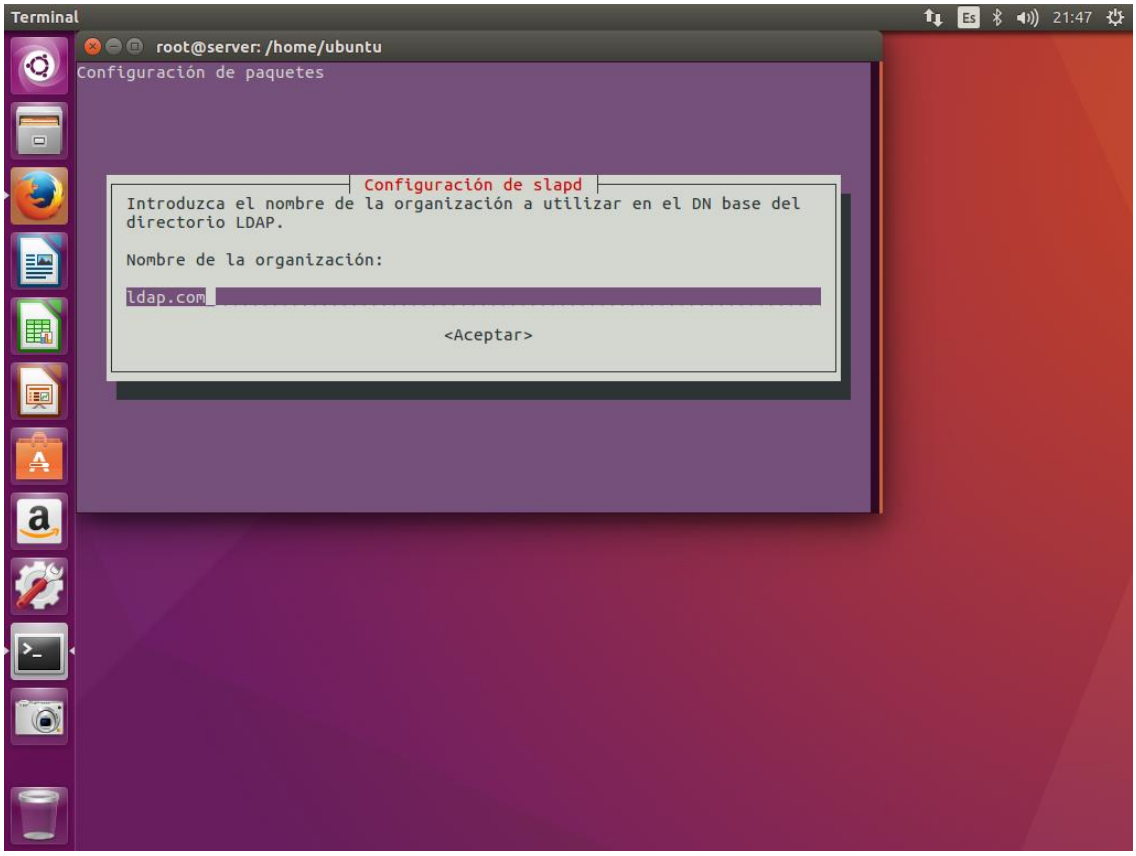


Elaboración propia

Nombre de la Organización:

Para esto, se escribe ldap.com como el nombre de nuestra organización. Esto será el DN base del directorio LDAP.

Figura 6 - Ingreso del nombre de la organización

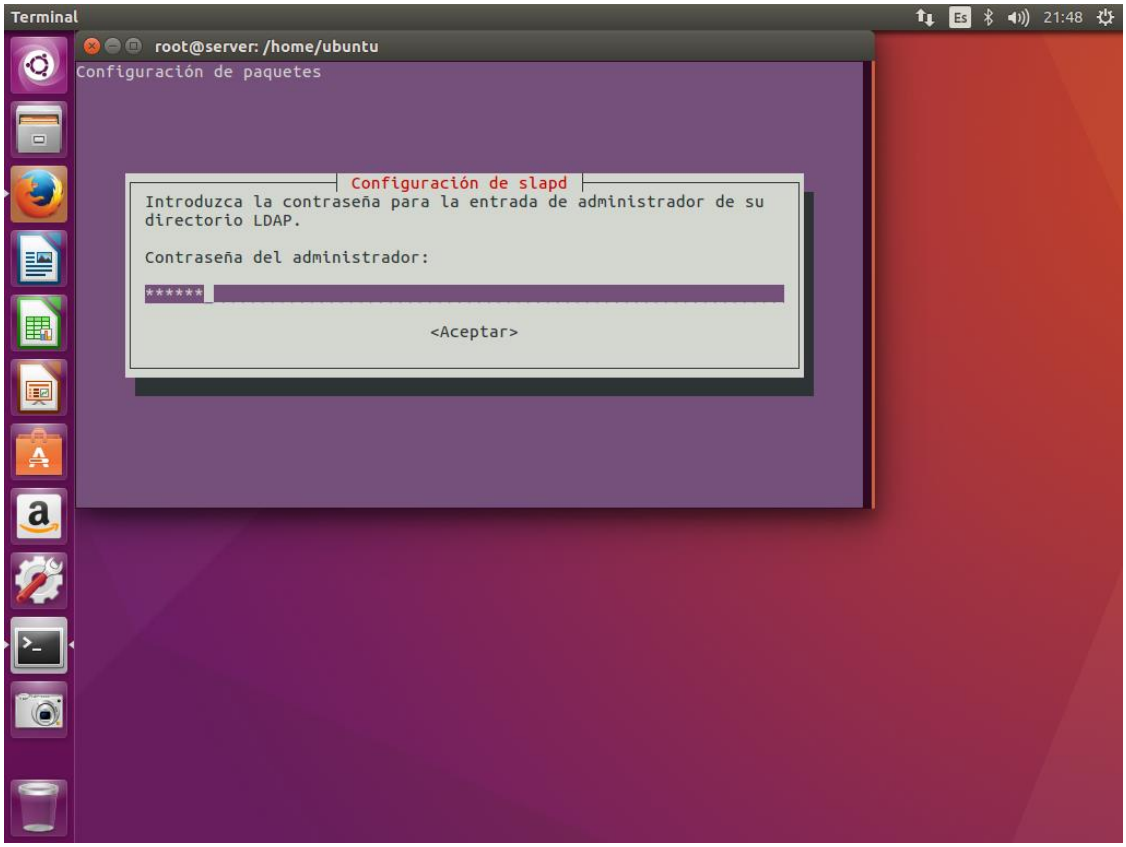


Elaboración propia

Contraseña de administrador:

Introduzca la contraseña para la entrada del administrador esto para ingresar a sus directorios, esto se utilizará en la configuración por medio del navegador. El pedido de la contraseña será dos veces, la segunda para la autenticación.

Figura 7 - Ingreso de la contraseña para administrar los directorios LDAP



Elaboración propia

Motor(backend) de base de datos a utilizar:

Hdb es una variante del *bdb* original que fue escrito por primera vez para su uso con BDB. *Hdb* utiliza un diseño de base de datos jerárquico que admite renombres de subárbol. Por lo demás, es idéntico al comportamiento de *bdb*, y se aplican todas las mismas opciones de configuración.

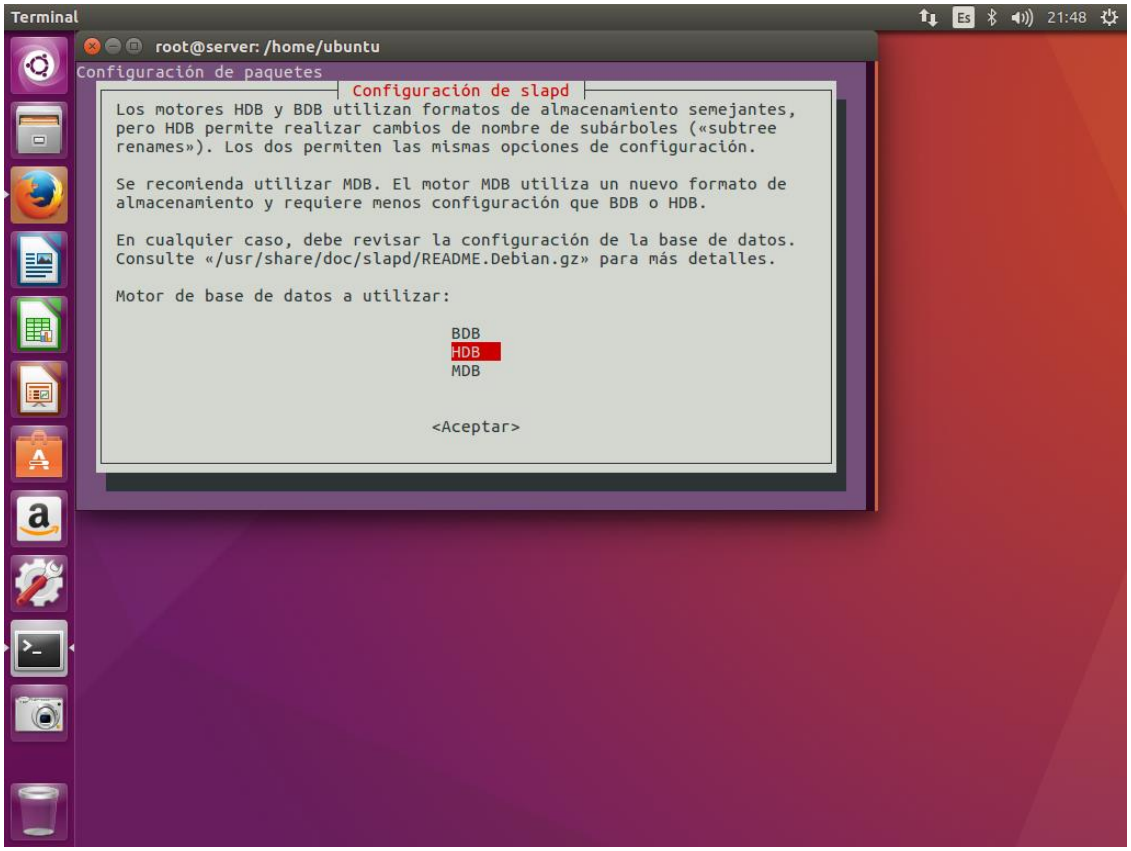
El backend de *mdb* es el primario recomendado para una base de datos *slapd* normal. Utiliza la propia base de datos. Para almacenar datos y está destinado a reemplazar los backends de Berkeley DB.

Es compatible con la indexación como los backends BDB, pero no utiliza almacenamiento en caché y no requiere ajuste para ofrecer el máximo rendimiento de

búsqueda. Al igual que *hdb*, también es completamente jerárquico y admite renombres de subárbol en tiempo constante.

Como indica la figura 8, se eligió el backend hdb.

Figura 8 - Selección del motor de base de datos a utilizar

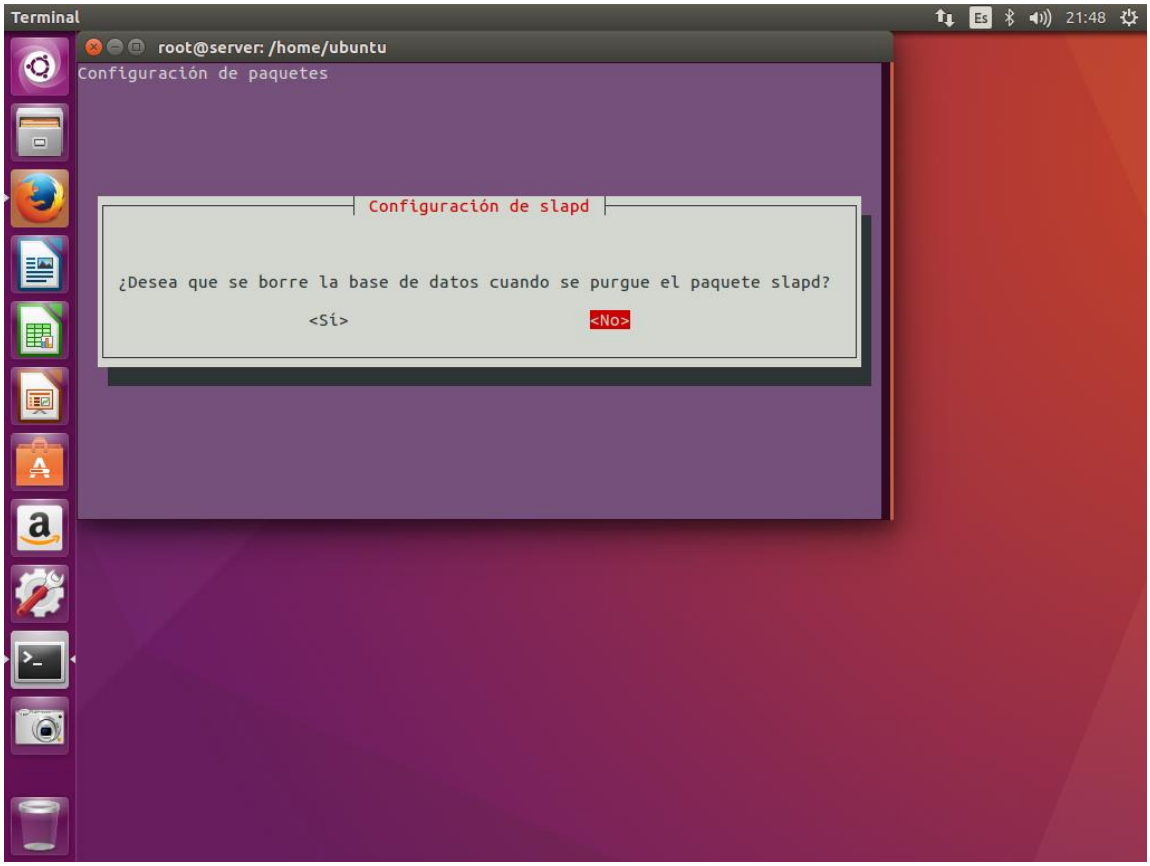


Elaboración propia

¿Desea que se borre la base de datos cuando se purga slapd?

En esta opción la respuesta fue no.

Figura 9 - Negación a borrar datos cuando se purgue paquetes sladp.

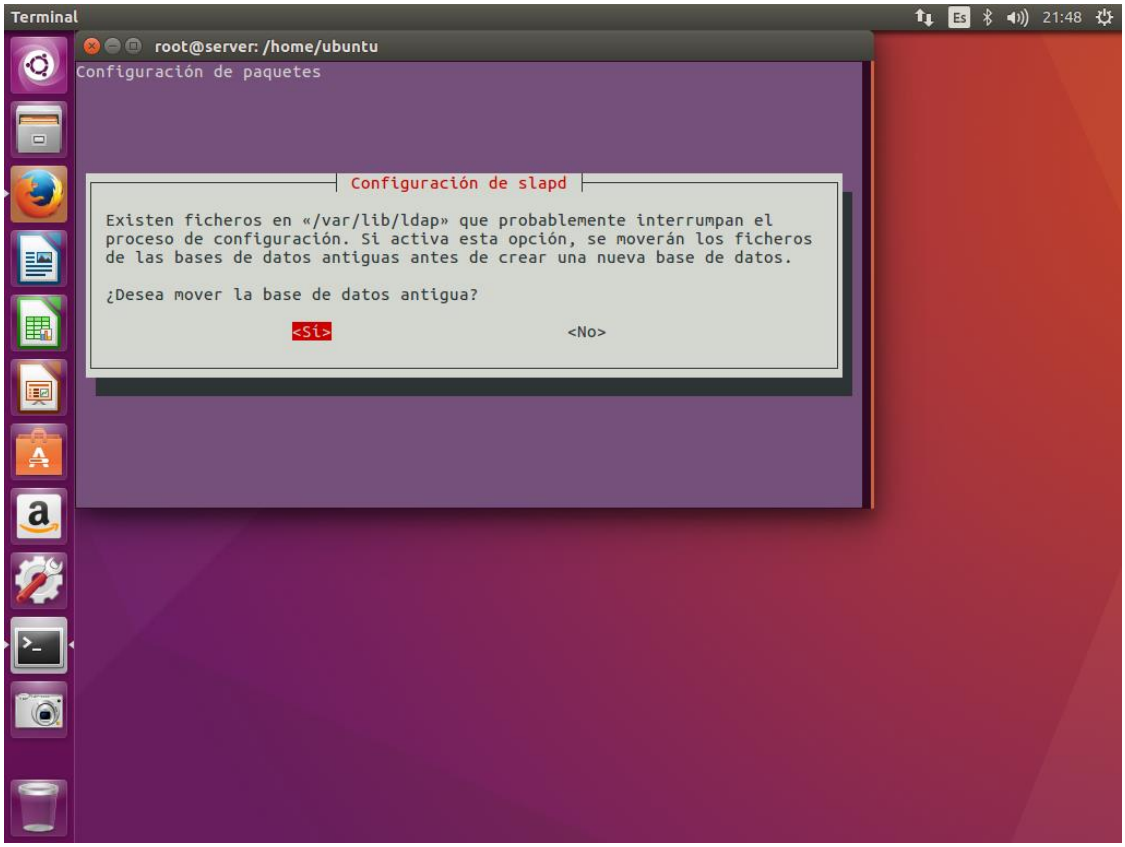


Elaboración propia

¿Desea mover la base de datos antigua?

Esta opción indica si se mueve los ficheros de las bases de datos antigua antes de crear una nueva base de datos.

Figura 10 - Confirmar para mover baso de datos antigua

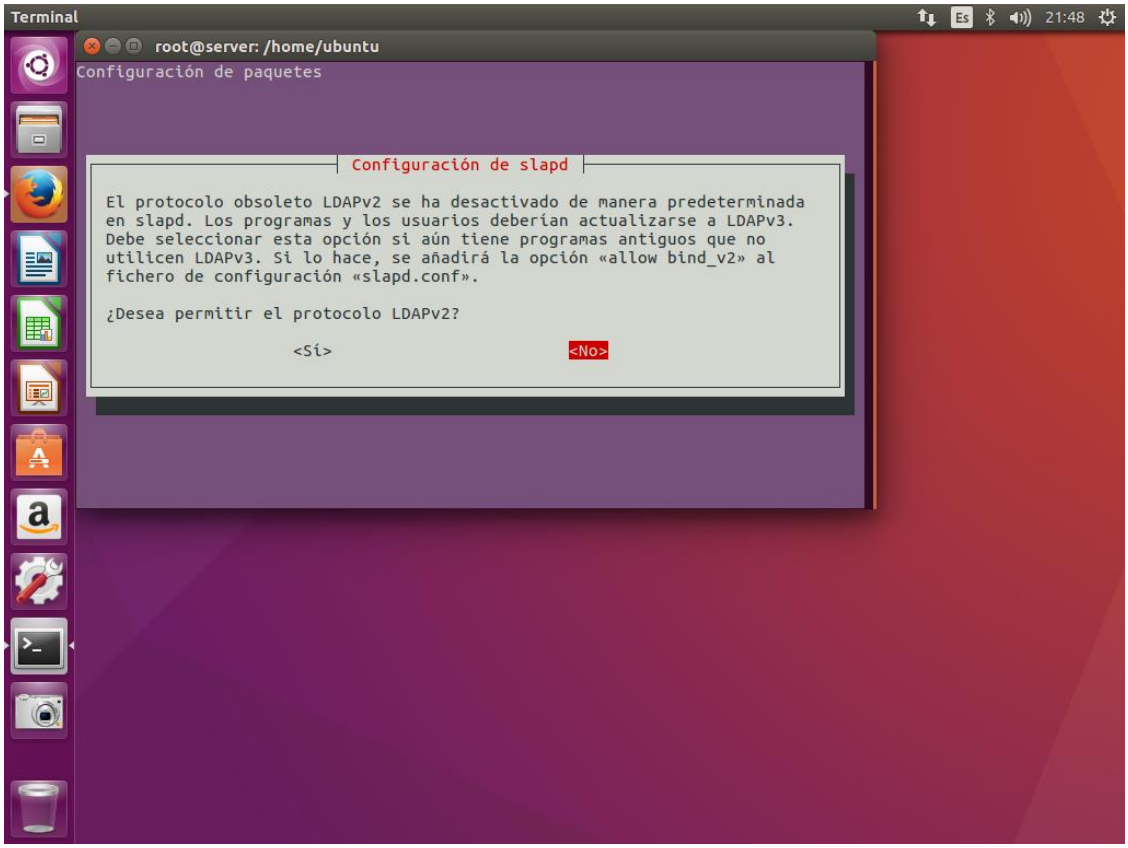


Elaboración propia

¿Desea permitir el protocolo LDAPv2?

Indica que el protocolo LDAPv2 esta desactivado por defecto pero si aún los programas usados no cuentan la versión 3 de LDAP, entonces debe responder afirmativamente la respuesta, en caso de esta investigación se usara programas relacionados con LDAPv3, por lo que la respuesta es negativa.

Figura 11 - Deshabilitar el protocolo LDAPv2

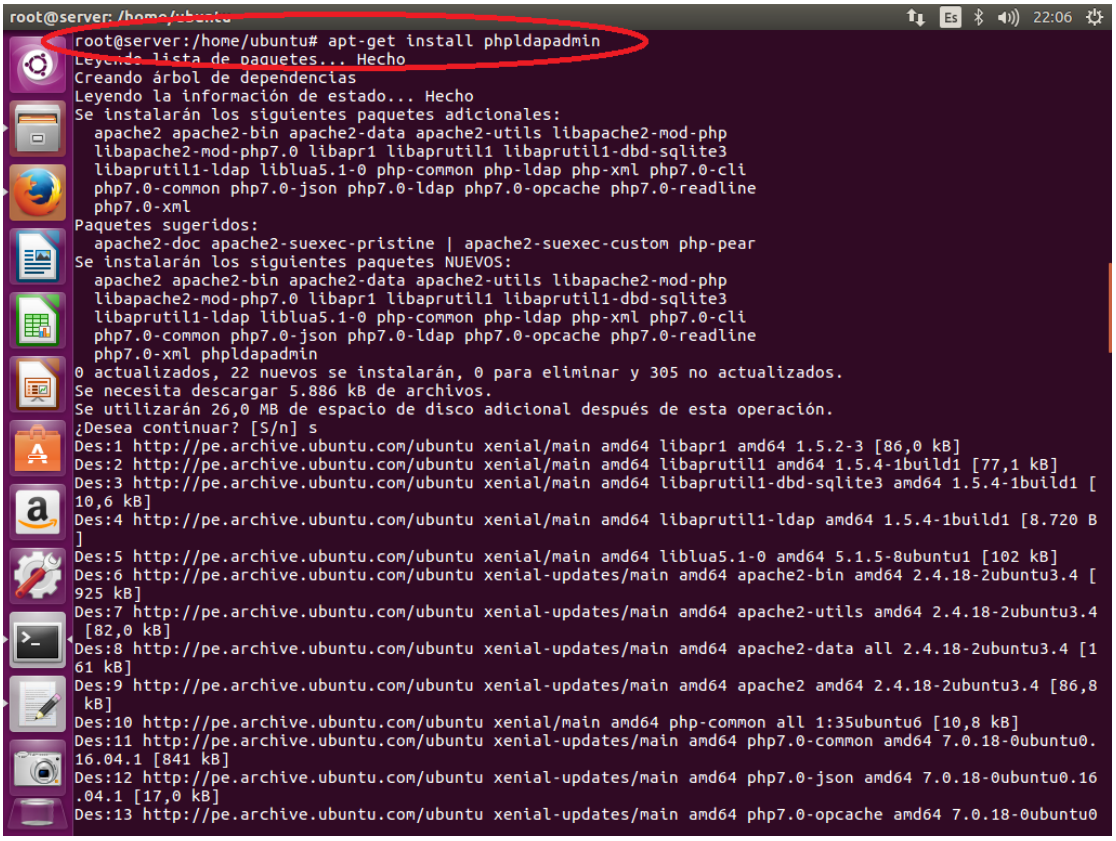


Elaboración propia

Aunque es muy posible administrar LDAP a través de la línea de comandos, la mayoría de los usuarios les resulta más fácil de usar una interfaz web. Vamos a instalar phpLDAPadmin, una aplicación PHP que proporciona esta funcionalidad.

Los repositorios de Ubuntu contienen un paquete phpLDAPadmin. Se puede instalar con *sudo apt-get install phpldapadmin*, tal como se muestra en la figura 12:

Figura 12 - Ingreso del comando para descargar e instalar phpldapadmin

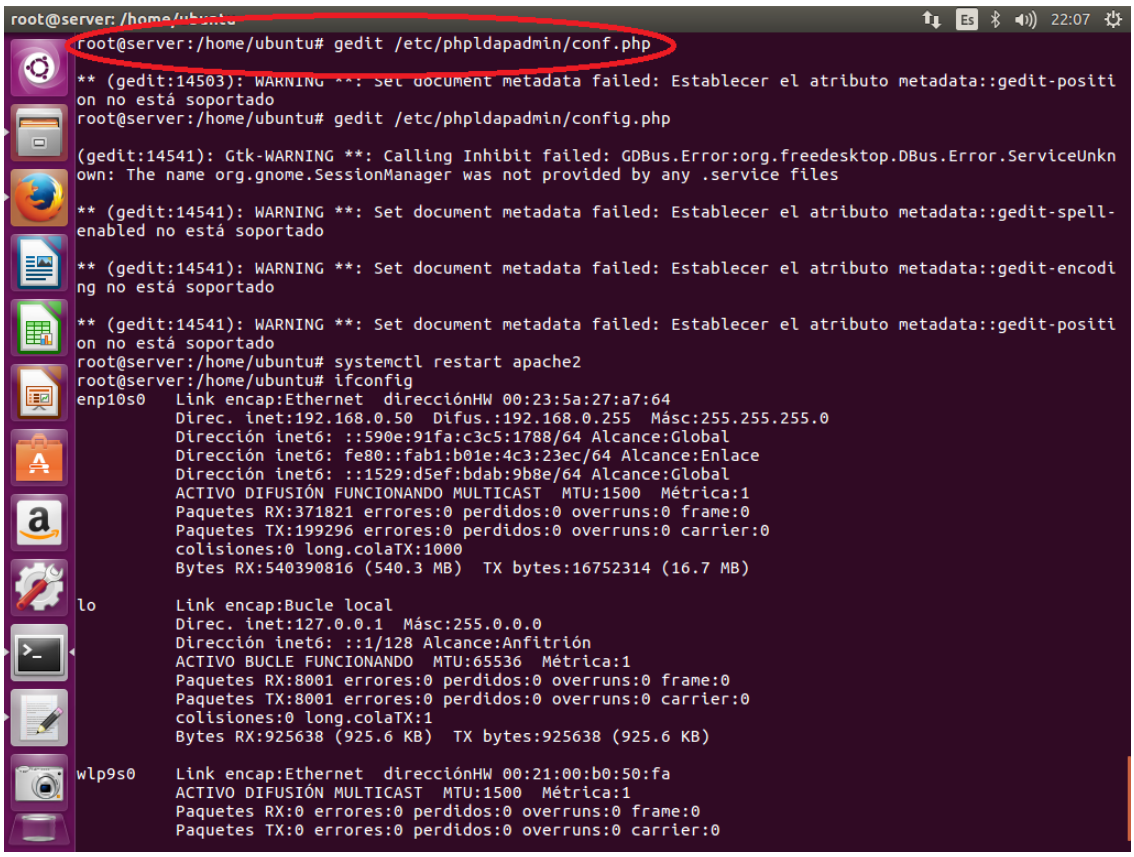


Elaboración propia

Esto instalará la aplicación, activar las configuraciones de Apache necesarias, y volver a cargar Apache. El servidor web está configurado para servir a la aplicación, pero tenemos que hacer algunos cambios adicionales. Tenemos que configurar phpLDAPadmin utilizar nuestro dominio, y para no autocompletar por la información de inicio de sesión LDAP.

Comience abriendo el archivo de configuración principal con privilegios de root en el editor de texto con *gedit /etc/phpldapadmin/config.php:*

Figura 13 - Ingreso del comando para agregar los detalles de configuración para el servidor LDAP

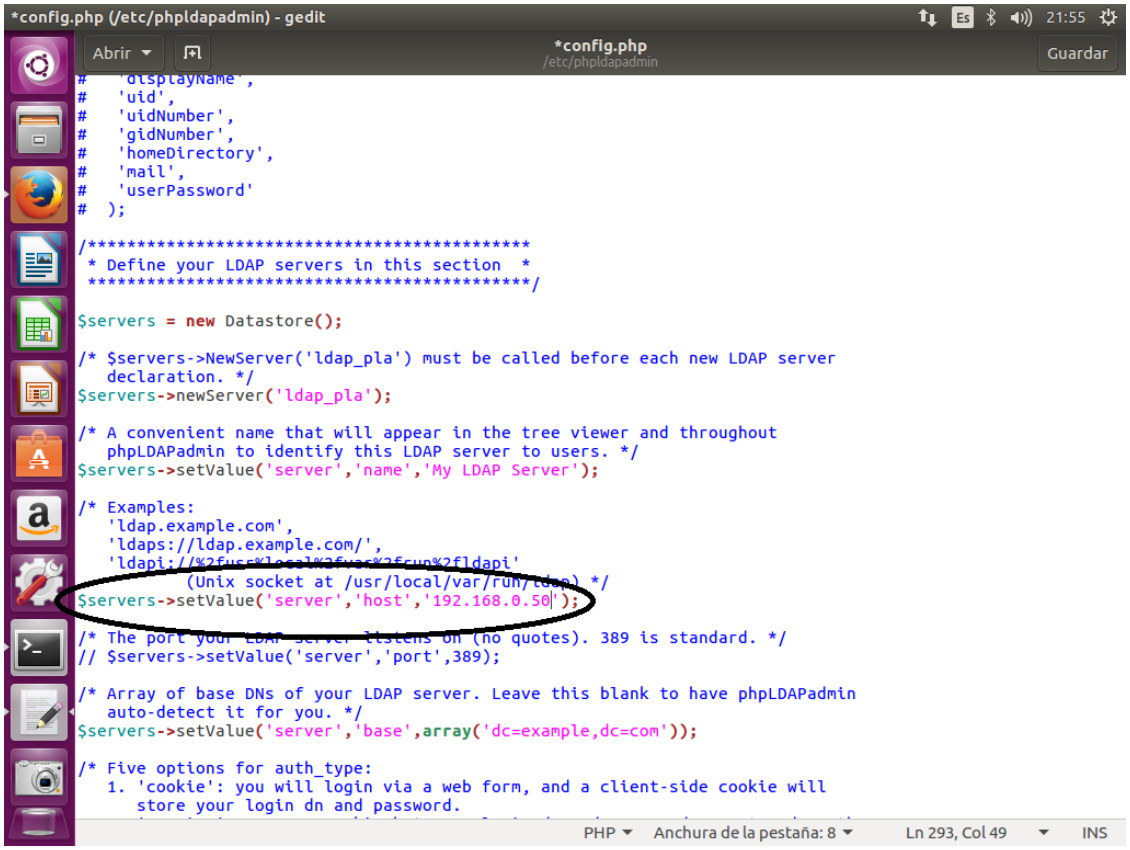


Elaboración propia

En este archivo, necesitamos agregar los detalles de configuración el servidor LDAP. Comience por buscar el parámetro host y establecerlo en el nombre de dominio de su servidor o en la dirección IP pública. Este parámetro debe reflejar la forma en que planea acceder a la interfaz web:

Modificamos la línea que comienza con $servers->setValue('server','name', server_domain_name_or_IP');

Figura 14 - Ingreso de la dirección IP del servidor



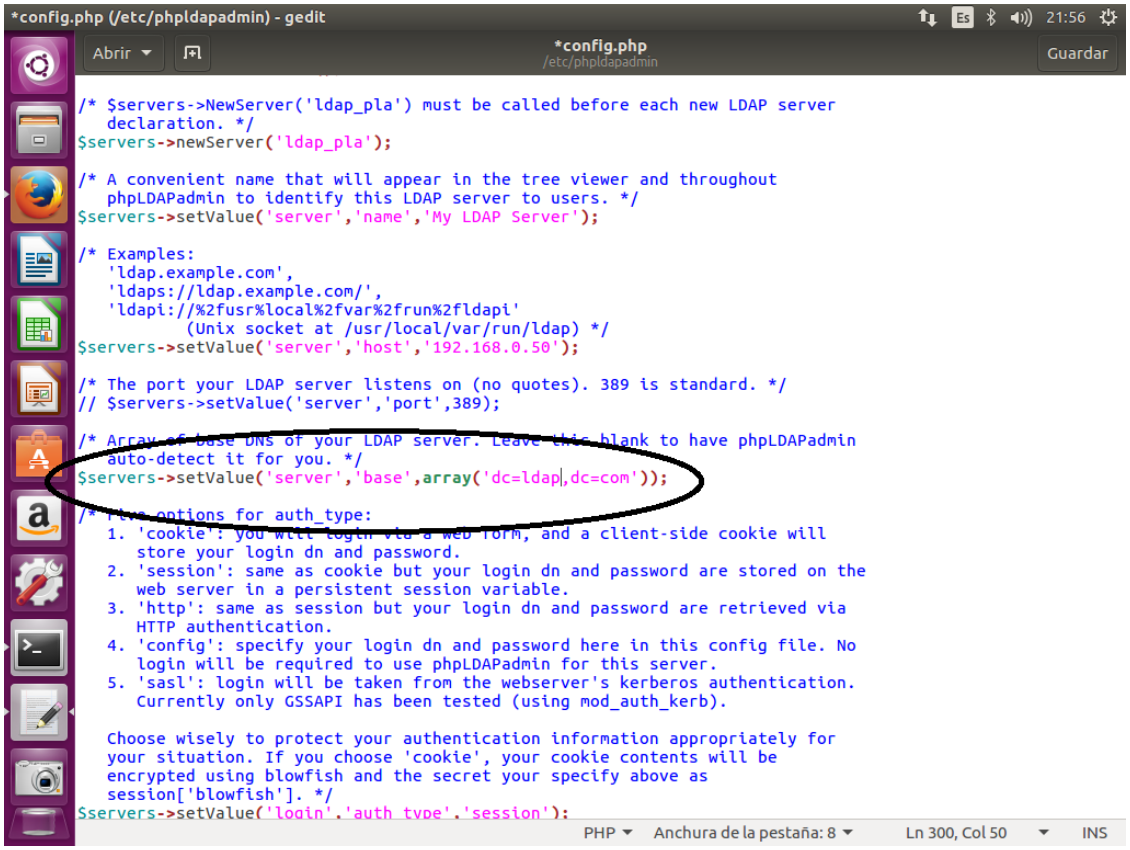
Elaboración propia

En la figura 14 se muestra que se cambió por la dirección IP del servidor que es 192.168.0.50, esto para ingresar desde el navegador de algún dispositivo en la red.

A continuación, se configuro el nombre de dominio que seleccionó para el servidor LDAP. Recuerde, en esta investigación se seleccionó ldap.com. Necesitamos traducir esto en sintaxis de LDAP reemplazando cada componente de dominio (todo no un punto) en el valor de una dc especificado. Todo esto significa que en lugar de escribir ldap.com, vamos a escribir algo como dc=ldap,dc=com. Se debe encontrar el parámetro que establece el parámetro base del servidor y utilizar el formato que acabamos de comentar para referenciar el dominio que decidimos. Esta configuración le dice a phpLDAPadmin cuál es la raíz de la jerarquía LDAP es. Esto se basa en el valor que escribió en al

reconfigurar el paquetes slapd, buscamos la línea con $servers->setValue('server','base', array('dc=example,dc=com')) y modificamos como se muestra en la figura 15:

Figura 15 - Edición del dominio del servidor

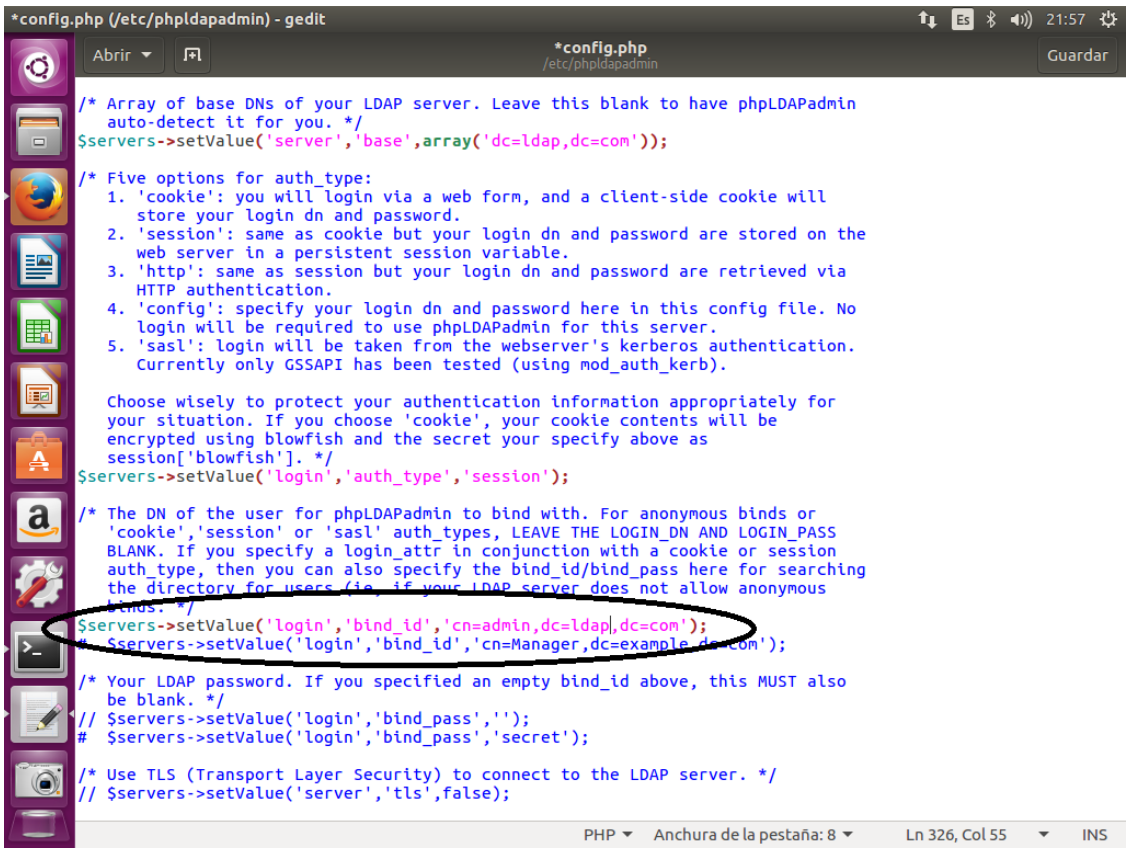


Elaboración propia

Se ajustó esta misma en nuestro parámetro *login bind_id*. El parámetro *cn* ya está configurado como "admin". Esto es correcto. Sólo se ajustó las porciones *dc*, al igual que lo hicimos arriba

La entrada bind_id en la línea de configuración y comentarlo con un principio de la línea:

Figura 16 - Cambio de detalles de administración al iniciar la sesión

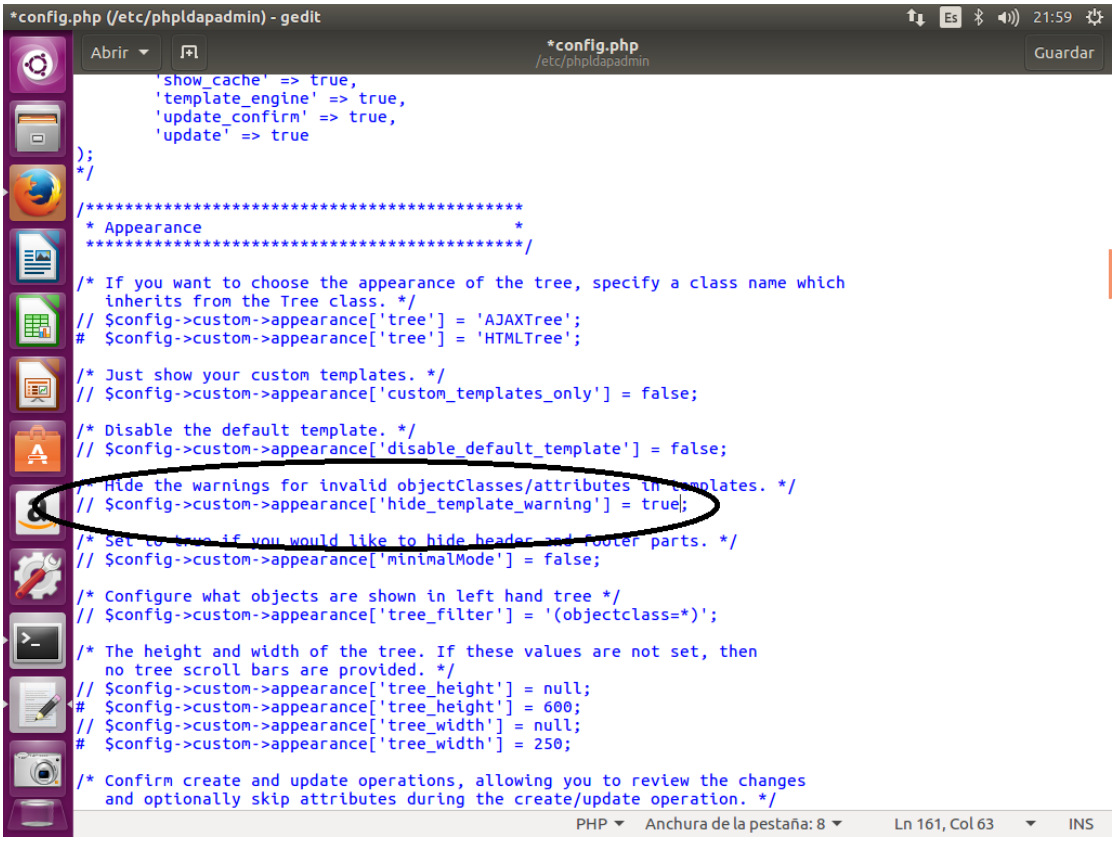


Elaboración propia

Esta opción pre-rellena los detalles de administración de inicio de sesión en la interfaz web. Esta es información que no deben compartir si nuestra página phpLDAPadmin es accesible al público.

Lo último que necesitamos para ajustar la visibilidad de algunos mensajes de advertencia phpLDAPadmin. Por defecto, la aplicación mostrará un buen número de mensajes de advertencia acerca de los archivos de plantilla. Estos no tienen impacto en nuestro uso actual del software. Podemos ocultarlos mediante la búsqueda el parámetro hide_template_warning, eliminando el comentario de la línea que lo contiene, y se establece a $config->custom->appearance['hide_template_warning'] = true.

Figura 17 - Modificación para mostrar mensajes de advertencia



Elaboración propia