UNIVERSIDAD DE OVIEDO

ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN

MÁSTER EN INGENIERÍA INFORMÁTICA

TRABAJO FIN DE MÁSTER

SISTEMAS DE CONTROL DE ACCESO PARA INFRAESTRUCTURAS DE COMUNICACIONES CABLEADAS.

Giovanni Domenico Mazzei

Junio, 2014

UNIVERSIDAD DE OVIEDO

ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN

MÁSTER EN INGENIERÍA INFORMÁTICA

TRABAJO FIN DE MÁSTER

SISTEMAS DE CONTROL DE ACCESO PARA INFRAESTRUCTURAS DE COMUNICACIONES CABLEADAS.

DOCUMENTO Nº 1

MEMORIA, SISTEMA y PRUEBAS.

Giovanni Domenico Mazzei

Junio, 2014

ÁREA DE INGENIERÍA TELEMÁTICA

TUTOR: DAVID MELENDI PALACIO

Índice.

RESUMEN

El presente Trabajo Final de Máster permite comparar una amplia variedad de sistemas de acceso a infraestructuras de redes cableadas, su definición, funcionamiento, arquitectura, comportamiento y evolución en el mercado, así como también la seguridad que estas representan en un entorno corporativo, al mismo tiempo, se realizó una evaluación de su interacción con el protocolo "Lightweight Directory Access Protocol", en sus versiones open source y bajo licencia. La iniciativa de escoger este tema surgió debido a la necesidad entre usuarios del campus a conectarse a una red vía alámbrica, sin que esta solicite una autenticación, como la red inalámbrica, donde se nos pide nuestro usuario y clave para acceder, ya sea con nuestro móvil, laptop, etc. Para este cometido se trazó un principal objetivo; realizar un análisis de los diferentes mecanismos de control de acceso existentes en el mercado para infraestructuras de comunicaciones cableadas y evaluar los principios y aplicaciones de los sistemas de control de acceso para un óptimo funcionamiento. Además, por medio de la elaboración de un piloto y de realizar pruebas en el entorno de la red de la universidad, pudimos obtener conclusiones positivas en la implementación de un sistema de autenticación de usuario, en un principio hacia la nube de internet, posteriormente validando el usuario creado en el software de openldap, la seguridad del sistema propuesto, la integridad de los datos, así como el rendimiento o throughput de la red, entre otros.

ABSTRACT

This Final Master allows you to compare a wide variety of systems for access to wired network infrastructure, its definition, performance, architecture, behavior and evolution in the market, as well as the security that these represent in a corporate environment, at the same time, an evaluation of the interaction was carried out with the protocol; Lightweight Directory Access Protocol, in versions open source and licensed. The initiative of choosing this topic arose because of the need between campus users to connect to a wired via network, without this request authentication, and wireless network, where we are asked your username and password to access, either with our mobile, laptop, etc. For this purpose a principal object was drawn; an analysis of different control mechanisms existing in the market access for wired communications infrastructure and evaluate the principles and application of access control systems for optimum performance. In addition, through the development of a pilot and testing in the vicinity of the university network, we could obtain positive conclusions on the implementation of a system of user authentication, initially to the internet cloud, then validating the user created in openldap software, security of the proposed system, the integrity of the data and the performance or throughput of the network, among others.

Lista de Figuras.

Lista de Tablas

1.- Introducción

Este proyecto consiste en realizar un trabajo científico-técnico, que tiene como objetivo principal definir y evaluar de la forma más exacta los diferentes mecanismos que permiten controlar el acceso a los usuarios a las redes de comunicaciones, principalmente sobre infraestructuras cableadas.

La necesidad de la definición de este estudio surge motivado a observar que los principales sistemas de control de acceso a las redes, se presentan sobre infraestructuras inalámbricas, un ejemplo de ello podemos observarlo en los laboratorios de la Universidad de Oviedo, donde cualquier usuario que coloque un laptop y se conecte vía cable de red a un jack libre, este permite conectarse a internet, sin pedir validación de credenciales, por ejemplo un usuario y contraseña, caso contrario ocurre, cuando se realiza validación de usuario y contraseña (UO) para acceder a internet por ejemplo a través de la red inalámbrica de la universidad. Al no tener en consideración estas variables, podría tener varias consecuencias, una de ellas sería la de comprometer la seguridad de la Universidad y de la gestión de los recursos de red.

Por tal razón se requiere un buen sistema de control de acceso para impedir el ingreso de personas extrañas y además para controlar y medir el uso de los recursos por parte de los usuarios autorizados.

Un sistema de esta naturaleza suele ser bastante complejo y por lo general cumple las siguientes funciones básicas: autenticación, autorización, contabililidad y auditoría. Estas 4 funciones son conocidas en inglés como AAAA (Authentication, Authorization, Accounting and Auditing), aunque en la práctica se utiliza la sigla AAA, ya que Accounting y Auditing se consideran a menudo como parte del mismo proceso.

En el proceso de desarrollo del presente trabajo, además de la evaluación de los diferentes mecanismos de control de acceso a los usuarios a redes ya sea privada o pública, sobre infraestructuras cableadas, es la realización de un piloto o prototipo que implemente unos de los mecanismos que serán estudiados en el desarrollo de este trabajo.

1.1Objetivos.

El principal objetivo que se persigue con la realización de este trabajo de investigación en el entorno a los distintitos mecanismos de control de acceso a las redes es el siguiente:

- ✓ Realizar un estudio de los diferentes mecanismos de control de accesos existentes en el mercado para infraestructuras de comunicaciones cableadas.
- ✓ Conocer los principios y aplicaciones de los sistemas de control de acceso que realizan las funciones de autenticación de los usuarios.

Dentro de estos objetivos principales subyacen los siguientes sub-objetivos:

- ✓ Comprender el funcionamiento de cada uno de los sistemas de control de acceso existes en el mercado actual.
- ✓ Aprender y gestionar el sistema de control de acceso mediante las herramientas de hardware disponibles en el laboratorio.
- ✓ Realizar un prototipo totalmente funcional que permita gestionar el acceso o no de los usuarios a una red.
- ✓ Aprender a configurar y gestionar el equipo de red disponible en el laboratorio de la Universidad de Oviedo, para llevar a cabo el piloto de manera satisfactoria.
- ✓ Integrar los diferentes componentes de software y hardware para la realización del piloto.
- ✓ Comprender el uso y aplicación de un servicio de directorio.

1.2 Alcance.

El alcance del presente proyecto es realizar una evaluación y estudio de los diferentes mecanismos de control de acceso de usuarios a redes basadas en una infraestructura cableadas, además de la incorporación de un sistema de directorios para almacenar cuentas de usuarios. Para este trabajo, se contempla la realización de un pequeño demo, en conjunto de una batería de pruebas para dar acceso a un usuario o no a la red.

2. -Sistemas de Control de Acceso.

2.1 Sistema AAA.

Como lo describimos en el apartado introductorio del presente trabajo de investigación, cuando se hace mención de un sistema de control de acceso a un sistema de redes, debemos hacer mención del sistema AAA. Cuando se maneja un gran número de usuarios, el control de acceso puede hacerse más eficazmente centralizando las funciones AAA en uno o más servidores AAA, también llamados servidores de autenticación. Un servidor de esta naturaleza debe ser capaz de recibir peticiones, examinar el contenido de dichas peticiones, determinar qué autorización se está pidiendo, bajar las políticas o reglas que necesite de un repositorio, evaluar la petición y obtener la respuesta a la petición, o bien reenviar la petición a otro servidor AAA. A continuación se establecerá la definición breve de cada uno de ellos.

Autenticación es el proceso mediante el cual se verifica la identidad del usuario. Puede basarse en lo que esa persona sabe (ej. contraseña, clave, PIN); lo que la persona posee (ej. token, tarjeta, certificado digital); lo que los persona es (ej. huella dactilar, iris, cara, mano, voz).

Autorización es el proceso de aceptar o negar el acceso de un usuario a los recursos, una vez que haya sido autenticado con éxito. El tipo de datos y servicios a los que el usuario podrá acceder dependen del nivel de autorización que tenga establecido. Por ejemplo, sólo los usuarios del departamento de nómina podrían tener acceso a los datos de nómina de la empresa. La autorización también controla el tipo de recursos que se asignan (ej., ancho de banda, dirección IP, etc.)

Contabilidad es el proceso de supervisar la actividad del usuario en lo que se refiere a uso de los recursos, la cantidad de tiempo que permanece conectado, los servicios a los que accede, así como la cantidad de datos transferidos durante la sesión. Esta información se utiliza principalmente para facturación, pero también se utiliza con fines estadísticos, de planeamiento de capacidad y de asignación de costos. Una función cada vez más importante es para la auditoría, con el fin de investigar incidentes de seguridad, accesos no autorizados, abusos de los recursos, etc.

2.2 TACACS. (Terminal Access Controller Access Control System).

Durante los años de la gran expansión de las redes ARPANET y MILNET, se crearon los protocolos TAC (Terminal Access Controller). Estos protocolos eran utilizados en la red de Milnet para permitir el acceso de los equipos remotos mediante MÓDEM, gestionando su autenticación y posterior acceso a la red. Este primer protocolo TAC, utilizado por Milnet en sus nodos de acceso a la red, como YUMA-TAC se pasó a llamar TACACS (Access Control System). TACACS permitió que las credenciales de usuario no se almacenaran en el equipo TAC y se pudiera descentralizar las bases de datos de credenciales hacia equipos que la almacenaban. El TACACS original fue desarrollado por el Ministerio de Defensa de EE.UU., y la empresa privada BBN Planet Corp. El diseño original de este protocolo consistía en un esquema simple de usuario y contraseña. TACACS se basó en UNIX y al servidor se le llama TACACSSD o TACACS daemon (demonio) y utiliza como puertos de autenticación el 49 tanto en TCP como en UDP indistintamente. TACACS no incorpora seguridad ni encriptación en sus transmisiones, con lo que simplemente interceptando el tráfico se puede recopilar todas las credenciales. Es un protocolo modelo cliente-servidor, donde el NAS envía una solicitud al servidor y éste responde afirmativamente o negativamente.

En 1990, CISCO adoptó este protocolo pasándose a llamar XTACACS e incorporando el arqueo o contabilidad y la auditoría. Posteriormente desarrolló TACACS+, realizando importantes aportaciones y mejoras en el mismo pero separándolo de sus antecesores, de tal manera que fuera incompatible. CISCO lo convirtió en un protocolo extensible aceptando plugins de autenticación: como tarjeta inteligente y otros sistemas basados en challege o desafío, además de añadir encriptación en las comunicaciones entre cliente y servidor. Se modularizó, basándose en AAA. TACACS+ es un protocolo de segunda generación basado en AAA, y que habita principalmente en equipos de CISCO, ya que es un protocolo propietario de CISCO y no está abierto a otros fabricantes. Algunas características de TACACS+ son:

- ✓ TACACS+ usa TCP como protocolo de transporte.
- ✓ TACACS+ encripta el cuerpo entero del mensaje.

- ✓ TACACS+ separa el proceso de autenticación, permitiendo utilizar sólo la autorización y el accounting de forma independiente. De esa manera puede utilizar otros protocolos de autenticación como Kerberos.
- ✓ TACACS+ da soporte a los siguientes protocolos: Netbios, x25, Appletalk, Novell NASI.
- ✓ TACACS+ controla la configuración de seguridad del acceso a routers.

Formato del Paquete

El paquete *header* de TACACS+, siempre comienza con doce octetos. En la figura 1, se aprecia la composición del *header* en TACACS+.

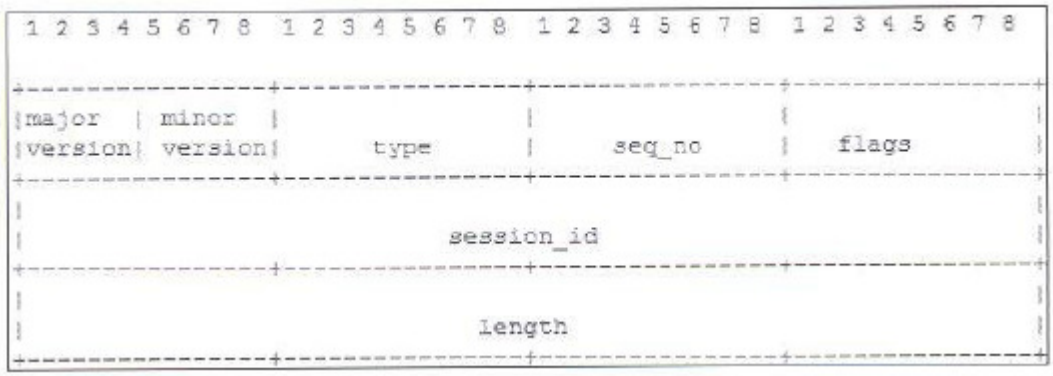


Figura 1. Header de TACACS+.

Fuente: CISCO System

Los contenidos del paquete se describen a continuación:

Major Version: Es el número de la versión de TACACS+.
Minor Version: Se entiende como las revisiones del protocolo con las cuales guarda compatibilidad hacia atrás, cuando se recibe un paquete con un número menor de versión del protocolo, que no es soportada, se genera de inmediato un error con un código de versión cerrada.
Type: Se refiere al tipo de paquete. Entre los valores posibles se encuentran: *authorization, authentication y accounting.*

Seq_no: Numero de secuencia del paquete en la sesión actual. El primer paquete en la sesión debe tener el número 1 y los siguientes paquetes deben incrementar este número en uno. Los clientes solamente envían paquetes con número impar y el servidor con número par.
Flags: Es una bandera que indica si el cuerpo del mensaje que sigue al *header* está encriptado o no.
Session_Id: Identificador de sesión que no cambia durante la duración de la misma.
Length: Longitud total del cuerpo del paquete, no se incluye el header. El paquete nunca es enviado sin ésta información.

El paquete del cuerpo (*body packet)* se define en el paquete *header*, las siguientes son las reglas generales que aplican al tipo de paquete *body* de TACACS+:

1. La totalidad del paquete body está protegido por el mecanismo de encriptación indicado en el *header.*
2. Cualquier variable de longitud de datos de un campo vacío debe ser 0.
3. Campos de longitud fija que no sean utilizados pueden tener longitud 0.
4. Ninguno de los datos y campos de mensaje en el paquete TACACS+, deben terminar en nulo.
5. Todos los valores de longitud están sin firma y en orden al octeto de la red.
6. No debe estar relleno ningún campo 0 al final del paquete.

Autenticación y Autorización.

La autenticación de TACACS+, tiene tres paquetes: Start, Continue y Replay. Los dos primeros son enviados por el cliente y el último por el demonio de TACACS+.

La authentication inicia con el envío del mensaje *Start* por parte del cliente al demonio, allí se especifica el tipo de autorización a realizar, y además, puede contener un nombre de usuario y algunos datos de autenticación. El paquete *Start* es enviado únicamente en el primer mensaje en la sesión o al reiniciar el servicio.

En respuesta al paquete *Start* el demonio envía Replay, lo mismo sucede si el mensaje es un *Continue*. De esta manera la comunicación se mantiene porque la única forma de terminarla es con una señal de interrupción *(abort),* en cuyo caso, la sesión es inmediatamente interrumpida.

El proceso de Authorization es una forma extendida de dar el servicio de autenticación remota, la sesión está definida como un simple par de mensajes: la petición (Request) seguida de una respuesta (Response).

El mensaje de petición contiene un número fijo de campos que describen la autenticidad del usuario o del proceso, y un número variable de argumentos que describen el servicio y las opciones para las cuales se pide la autorización. La respuesta contiene un grupo variable de argumentos de respuesta (AVP), los cuales pueden restringir o modificar las acciones de los clientes. Los argumentos para ambos mensajes tanto la petición como la respuesta pueden ser especificados como mandatario u opcional. Un argumento opcional es aquel que puede ser o no usado y uno mandatorio es aquel que debe ser usado.

En la siguiente figura, podemos detallar un poco más la secuencia que se realiza, en la autenticación y autorización por parte de TACACS+: En primer lugar un usuario realiza una solicitud a un servidor de acceso (NAS), que requiere autenticación al TACACS+, esta a su vez va a solicitar a través de la (NAS) un usuario y una contraseña, que previamente deben estar creados en el servidor AAA. Una vez que dichas credenciales sean las correctas, el servidor responde aceptando a dicho usuario.

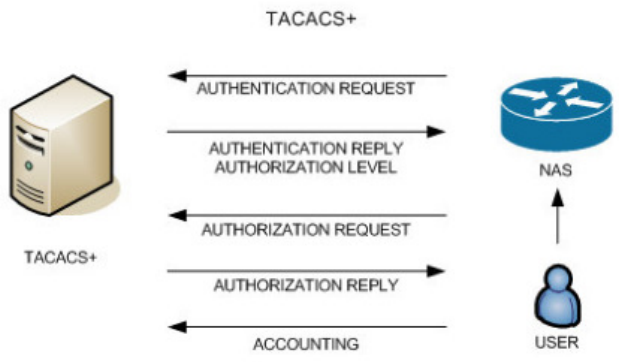


Figura 2. Secuencia de mensajes TACACS+.

Fuente: tacacs.net

Los RFC que tratan sobre TACACS son el RFC 1492 de 1993 ("Un protocolo de control de acceso, TACACS"), RFC 927 ("Opción de Telnet para TACACS") y RFC 2975 de 2000 ("Introducción a la gestión de contabilidad o arqueo").

2.3 RADIUS.

RADIUS, fue diseñado originalmente para controlar el acceso de usuarios remotos que se conectaban vía módems fónicos mediante un RAS (Remote Access Server), también conocido como NAS (Network Access Server), pero hoy día se utiliza para el control de acceso a cualquier tipo de sistema o servicio (VPN, firewall, LAN inalámbrica, DSL). RADIUS es un protocolo cliente/servidor que opera bajo TCP/IP. El cliente RADIUS es típicamente un NAS y el servidor RADIUS es generalmente un proceso tipo demonio que corre en una máquina Unix o Windows. RADIUS almacena información de identificación sobre todos los usuarios de la red con perfiles individuales, que pueden incluir restricciones de acceso, enrutamiento específico para el destino, filtrado de paquetes e información de cuentas. Con RADIUS se puede controlar el acceso a una máquina específica, a un servicio específico como telnet o a un protocolo específico como PPP (Point-to-Point Protocol). Por ejemplo, cuando un usuario se conecta vía modem a un ISP (Internet Service Provider), RADIUS lo podría autorizar para correr PPP usando la dirección IP 217.213.21.5 y a partir de eso momento empieza a contabilizar el uso. Podemos observar la secuencia de la petición en la siguiente figura número 3, por ejemplo.

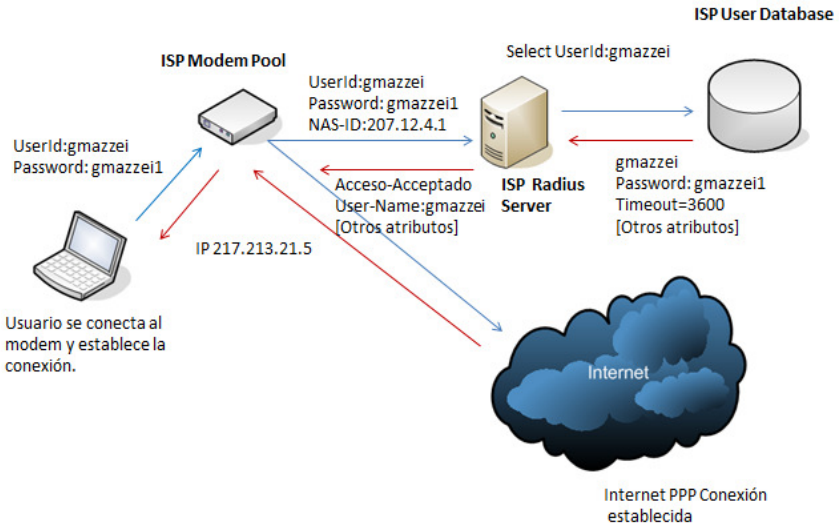


Figura 3. Secuencia de acceso a la red.

Fuente: Propia.

RADIUS también se utiliza para controlar el acceso remoto del administrador de la red a la configuración de equipos tales como routers y switches, eliminando así el riesgo de usar contraseñas guardadas localmente en el propio equipo.

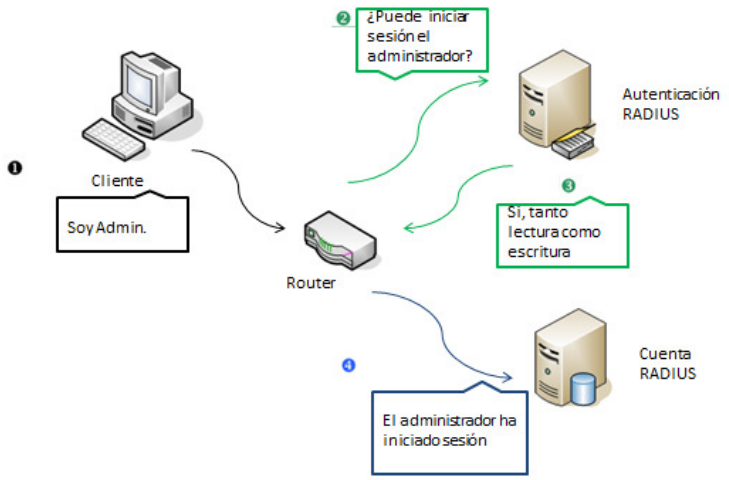


Figura 4. Secuencia de Acceso del usuario Admin.

Fuente: Propia

Una infraestructura completa para autenticación, autorización y contabilidad basada en RADIUS está formada por los siguientes componentes: Clientes de acceso, servidores de acceso (clientes

RADIUS), proxy RADIUS, servidores RADIUS, bases de datos de cuentas de usuario. Estos componentes se muestran en la siguiente figura:

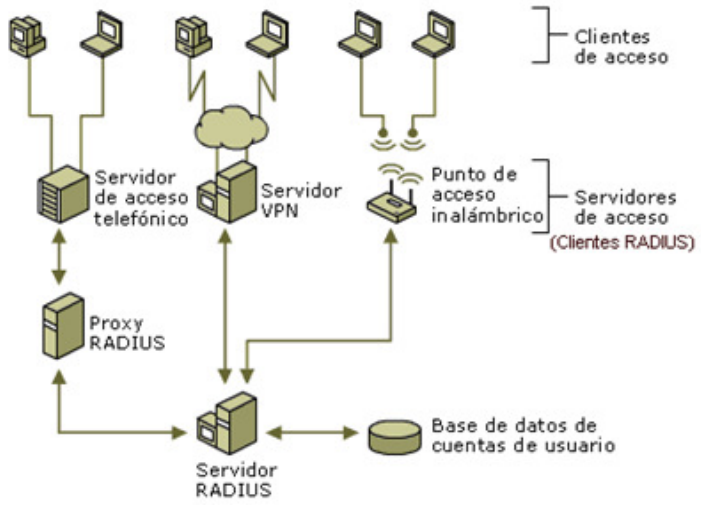


Figura 5. Componentes que actúan en el acceso a la red por RADIUS.

Fuente: Vincenzo Mendillo. UCV

Un cliente de acceso es un equipo que requiere un cierto nivel de acceso a la red. Los clientes de acceso telefónico o de red privada virtual (VPN), los clientes inalámbricos o los clientes LAN conectados a un switch Ethernet son ejemplos de clientes de acceso.

Un servidor de acceso es un dispositivo que proporciona cierto nivel de acceso a la red. Un servidor de acceso que utiliza una infraestructura RADIUS es un cliente RADIUS y envía peticiones de conexión y mensajes de contabilidad a un servidor RADIUS. Un ejemplo es un equipo Windows Server que ejecuta el servicio de Enrutamiento y acceso remoto y que proporciona los servicios tradicionales de acceso telefónico remoto o de acceso remoto de red privada virtual (VPN) a la intranet de una organización. Otro ejemplo es un punto de acceso inalámbrico (AP), que proporciona acceso de nivel físico a la red de una organización, por medio de tecnologías inalámbricas de transmisión y recepción.

Un proxy RADIUS es un dispositivo que reenvía o enruta peticiones de conexión y mensajes de contabilidad entre clientes RADIUS y servidores RADIUS. Un proxy RADIUS se puede

utilizar como punto de reenvío de los mensajes RADIUS cuando los servicios AAA deben tener lugar en varios servidores RADIUS de diferentes organizaciones.

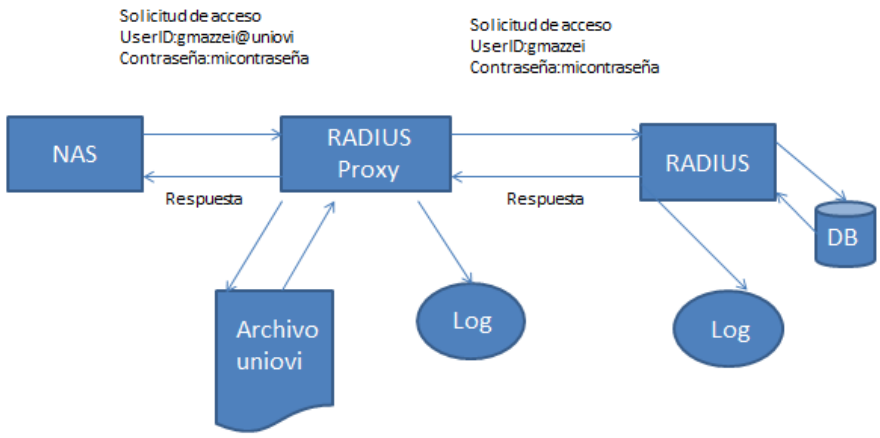


Figura 6. Procedimiento de acceso a la red a través de RADIUS Proxy.

Fuente: Propia

Un servidor RADIUS es un dispositivo que recibe y procesa peticiones de conexión o mensajes de contabilidad enviados por clientes RADIUS o proxy RADIUS. En el caso de las peticiones de conexión, el servidor RADIUS procesa la lista de atributos RADIUS de la petición de conexión. El servidor RADIUS autentica y autoriza la conexión, y devuelve un mensaje de aceptación o rechazo de acceso, basándose en un conjunto de reglas y la información de la base de datos de cuentas de usuario. El mensaje de aceptación de acceso puede contener restricciones de conexión que implementa el servidor de acceso durante el transcurso de la conexión.

La base de datos de cuentas de usuario es la lista de cuentas de usuario, junto con sus propiedades, que un servidor RADIUS puede comprobar para verificar las credenciales de autenticación y las propiedades de cuentas de usuario que contienen información de parámetros de autorización y conexión.

Podemos observar la operación del protocolo RADIUS que se ilustra en la siguiente figura.

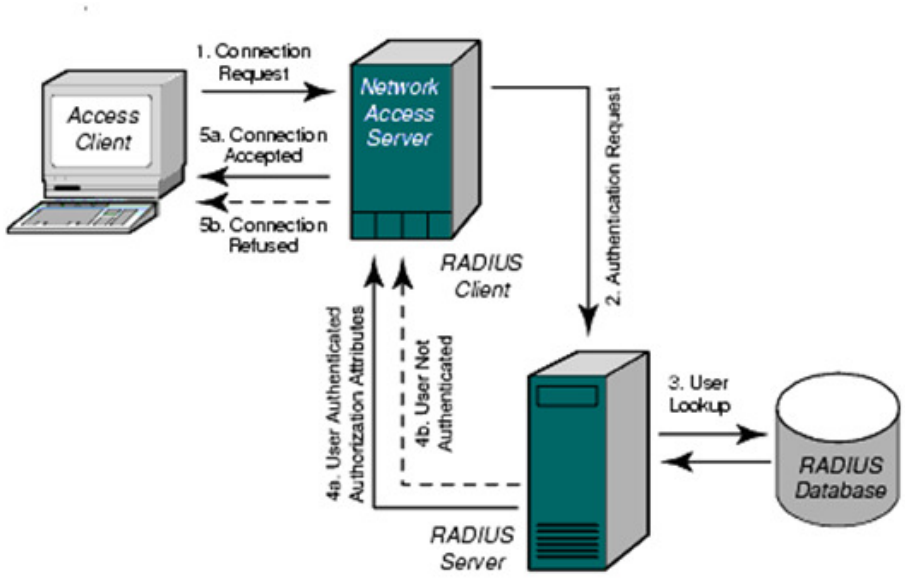


Figura 7. Secuencia de acción del protocolo RADIUS.

Fuente: Vincenzo Mendillo. UCV

1. Un cliente del acceso remoto (que podría ser por ejemplo un usuario con modem o un usuario con PC inalámbrica), envía al NAS una petición de conexión.
2. Cuando el NAS recibe la petición de conexión, realiza con el usuario una negociación inicial del acceso con información sobre la conexión solicitada (nombre del usuario, contraseña, tipo de conexión, puerto de acceso, etc.). El NAS entonces remite esa información como una petición de autenticación al servidor RADIUS. Este servidor y el NAS comparten una clave secreta previamente configurada y el servidor sólo acepta peticiones del NAS si la clave es correcta.
3. El servidor RADIUS autentica la información del usuario en una base de datos local o remota. El servidor RADIUS podría simplemente verificar que el nombre y la contraseña del usuario sean correctos, o podría comparar esa información con una lista de comprobación del acceso para verificar otros parámetros.
4. Si la autenticación es exitosa, el servidor RADIUS devuelve un mensaje de aceptación (4a). Este mensaje podría también incluir los parámetros de la conexión. Adicionalmente el servidor RADIUS podrá enviar un desafío (*challenge*) solicitando más información antes de aceptar o de rechazar al usuario. Si la autenticación falla, se devuelve un mensaje del rechazo (4b).

5. De acuerdo con la información que recibe del servidor RADIUS, el NAS acepta o rechaza la petición de conexión.

Después de autenticar al usuario y establecer la conexión, el NAS puede remitir los datos de contabilidad y auditoría al servidor RADIUS, el cual a su vez los puede almacenar o remitir a otra máquina para otros fines.

RADIUS soporta una amplia variedad de protocolos para autenticar a los usuarios y proteger la información que se transmite. Los dos más comunes son PAP (Password Authentication Protocol) y CHAP (Challenge-Handshaking Authentication Protocol). RADIUS también soporta el sistema de autenticación de Unix y de Windows, así como el reciente EAP (Extensible Autenticaction Protocol), muy utilizado en LANs inalámbricas (WLANs). Los detalles del protocolo RADIUS están claramente explicados en RFC 2865 y RFC 2866 (accounting).

En esos RFC se definen los siguientes tipos de mensajes RADIUS, con su respectivo código:

1. *Access-Request* (solicitud de acceso)

Enviado por un cliente RADIUS para solicitar autenticación y autorización de un intento de conexión.

2. *Access-Accept* (aceptación de acceso)

Enviado por un servidor RADIUS como respuesta a un mensaje Access-Request. En él se informa al cliente RADIUS de que se ha autenticado y autorizado el intento de conexión.

3. *Access-Reject* (rechazo de acceso)

Enviado por un servidor RADIUS como respuesta a un mensaje Access-Request. En él se informa al cliente RADIUS de que se ha rechazado el intento de conexión. Un servidor RADIUS envía este mensaje si las credenciales no son auténticas o si no se ha autorizado el intento de conexión.

4. *Accounting-Request* (solicitud de contabilidad)

Enviado por un cliente RADIUS para especificar información de contabilidad de una conexión.

5. *Accounting-Response* (respuesta de contabilidad)

Enviado por el servidor RADIUS como respuesta a un mensaje Accounting-Request. En este mensaje se confirman la recepción y el procesamiento correctos del mensaje.

11. *Access-Challenge* (desafío de acceso)

Enviado opcionalmente por un servidor RADIUS como respuesta a un mensaje Access-Request. Este mensaje es un desafío al cliente RADIUS que exige una respuesta.

En la figura siguiente se ilustra el flujo de los distintos mensajes.

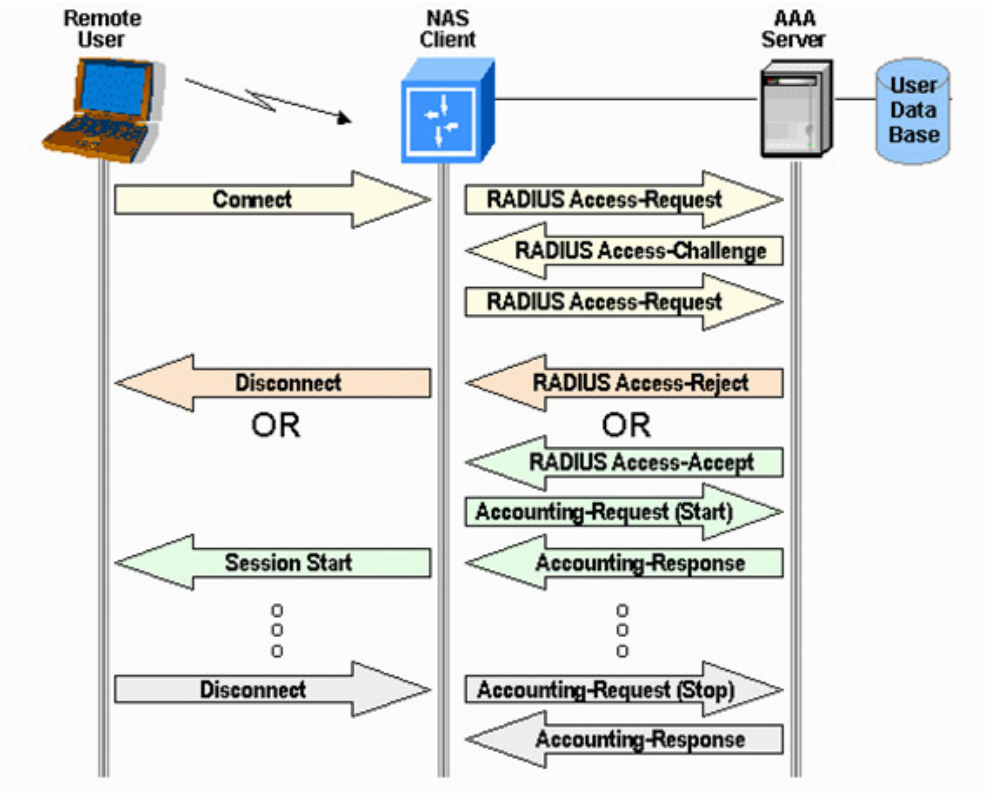


Figura 8. Secuencia de flujo de mensajes.

Fuente: Vincenzo Mendillo. UCV

Los mensajes RADIUS se envían sobre UDP (User Datagram Protocol). El puerto UDP 1812 (o 1645) se utiliza para los mensajes de autenticación y el puerto UDP 1813 (o 1646) para los mensajes de contabilidad. Se utiliza UDP en vez de TCP para acelerar el proceso de autenticación, ya normalmente que no hace falta la retransmisión de datos ni tampoco la

confirmación que brinda TCP. En caso de no respuesta por parte del servidor RADIUS primario, las solicitudes suelen ser redirigidas a un servidor alterno.

Un mensaje RADIUS está formado por un encabezado RADIUS con 4 campos, seguido de los atributos y sus valores. Cada atributo especifica una información determinada acerca del intento de conexión. Por ejemplo, existen atributos para el nombre de usuario, la contraseña de usuario, el tipo de servicio solicitado por el usuario y la dirección IP del servidor de acceso.

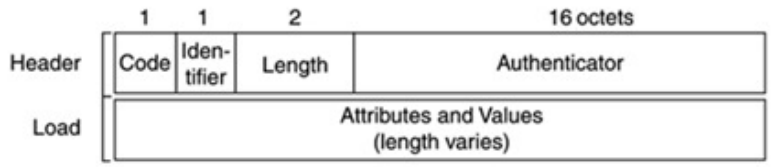


Figura 9. Mensaje Radius.

Fuente: Vincenzo Mendillo. UCV

El código (*Code*) es 1 octeto e identifica el tipo de paquete. Por ejemplo, un mensaje Access-Request tiene Code = 1.

El identificador (*Identifier*) es 1 octeto y permite asociar una petición con la respectiva respuesta. Generalmente es un contador que se incrementa cada vez que se envía una petición.

La longitud (*Length)* son 2 octetos y representa la longitud total del mensaje RADIUS, incluyendo el encabezado.

El autenticador (*Authenticator*) son 16 octetos y se utiliza para autenticar la respuesta del servidor RADIUS y también se utiliza como un mecanismo para encriptar la contraseña del usuario.

La carga útil que sigue al encabezado está formada por una secuencia de parejas de atributos y valores (*AVP*). Para cada atributo se especifica el tipo, la longitud y el valor, de acuerdo a un diccionario previamente establecido. Los atributos son opcionales y se utilizan para proporcionar información entre clientes RADIUS, proxy RADIUS y servidores RADIUS. Por ejemplo, la lista de atributos del mensaje Access-Request incluye información acerca de las

credenciales de usuario y los parámetros del intento de conexión. Por el contrario, el mensaje Access-Accept incluye información acerca del tipo de conexión que se puede establecer, las restricciones de conexión y los atributos específicos del proveedor. Como se indicó en un principio en el RFC 2865 se define una serie de atributos bastante comunes, entre los que tenemos:

- ✓ Attribute type 1— User-Name (defines usernames, such as numeric, simple ASCII characters, or a Simple Mail Transfer Protocol [SMTP] address).
- ✓ Attribute type 2— User-Password (defines the password, which is encrypted using Message Digest 5 [MD5]).
- ✓ Attribute type 3— CHAP-Password (used only in access-request packets).
- ✓ Attribute type 4— NAS-IP-Address (defines the NAS's IP address; used only in access-request packets).
- ✓ Attribute type 5— NAS-Port (this is not the User Datagram Protocol [UDP] port number; it indicates the NAS's physical port number, ranging from 0 to 65,535).
- ✓ Attribute type 6— Service-Type of service requested or type of service to be provided.
- ✓ Attribute type 7— Framed-Protocol defines required framing; for example, PPP is defined when this attribute is set to 1 and SLIP is set to 2.
- ✓ Attribute type 8— Framed-IP-Address defines the IP address to be used by the remote user.
- ✓ Attribute type 9— Framed-IP-Netmask defines the subnet mask to be used by the remote user.
- ✓ Attribute type 10— Framed-Routing.
- ✓ Attribute type 13 —Framed-Compression.
- ✓ Attribute type 19— Callback-Number.
- ✓ Attribute type 26— Vendor-Specific Attributes (VSA).
- ✓ Attribute type 61— NAS-Port-Type.

2.4 Diameter.

Tras la creación del grupo de trabajo en la IETF en 1995 dedicado a crear el RFC correspondiente a RADIUS, se pensó en crear un nuevo código limpio y mejorado de RADIUS que fue

denominado RADIUS v.2. Pero la IETF no permitió este cambio, debido a que RADIUS todavía no había sido ratificado en una RFC funcional y corregida, y no se debía crear otro estándar hasta que el primero hubiera sido publicado. Por ello, el nombre que recibió este nuevo estándar no pudo ser RADIUS v2 y se optó por Diameter (dos veces el radio o "twice as good as RADIUS" o "dos veces tan bueno como RADIUS"). Diameter fue diseñado en 1996 por Pat Calhoun de la compañía Black Storm Networks. [5]

El RFC que regula Diameter pasó a ser el RFC-2588 ("Diameter Base Protocol"), y posteriormente se han ido creando diferentes RFC que regulan su aplicación en MobileIP, EAP, etc. [5]

Según [Hansen, Fernández Yago, 2008]. Diameter es un protocolo de segunda generación, basado en AAA, que como posible sucesor de RADIUS pretendía mejorar todas sus carencias y puntos débiles. Unas de las premisas más importantes en su diseño fue que tenía que ser compatible con RADIUS ("legacy compatible") para que pudiera asumir todas las instalaciones en forma de migración. Por eso cada vez que se modifica un RFC relacionado con RADIUS es necesario modificar esas nuevas características en Diameter para seguir permitiendo mantener ese principio de compatibilidad entre ambos sistemas. Algunas de las mejoras que incorpora son: la sustitución de UDP por TCP y SCTP mejorando el control de errores en la transmisión, el uso de túneles mediante IPSec o TLS, y su cambio de modelo hacia peer to peer en vez de cliente-servidor, con lo que un servidor puede realizar consultas hacia un cliente, permitiendo sesiones dinámicas.

Diameter mejora RADIUS en muchos aspectos como la gestión de las comunicaciones mediante SCTP, previendo de una forma muy adecuada el timeout en los envíos de mensaje y en la búsqueda de rutas alternativas hacia el servidor o servidores. Diameter firma los mensajes mediante un código de tiempo, que impide duplicidades en la recepción de respuestas simultáneas, además de usar cifrado basado en certificados y firma digital. Diameter se apoya en un módulo criptográfico llamado CMS (Criptographic Message Syntax) integrado en su plataforma, que se encarga del cifrado de todos los mensajes. Diameter da soporte al nuevo estándar de gestión de NAS llamado NASREQ.

Otro aspecto que mejora Diameter es el de la comunicación entre servidores de autenticación (Diameter), permitiendo definir cadenas de proxy para los envíos de mensajes e implantando mejores modelos de confianza que RADIUS. Todos estos componentes de Diameter es la de roaming que permite crear grandes redes distribuidas.

Con los años RADIUS fue creciendo y mejorando, con lo que muchas de las motivaciones que instaron a la creación de Diameter se sumieron en lo innecesario, por eso RADIUS ha seguido manteniéndose como un gran estándar en la actualidad.

Sin embargo Diameter continúa su proceso de expansión y comienza a implantarse en sectores como la telefonía móvil o la VoIP debido a todos esos componentes que incorpora, que lo convierten en un servicio ideal para este tipo de instalaciones de gran nivel.

Arquitectura.

Diamenter está definido en términos de un protocolo base y un grupo de aplicaciones. El protocolo base debe utilizarse de la mano con una aplicación Diamenter, de tal forma, que cada aplicación depende del protocolo base sobre el servidor para soportar el tipo específico de acceso a la red. Las aplicaciones NASREQ, soportan *dia-lin* PPP/IP mientras que el protocolo base define el formato del mensaje, los datos son llevados como una colección de AVP (*Atribute Value Pairs*) equivalente a los atributos de RADIUS, de forma tal que los AVP, están formados por múltiples campos: AVP code, Length, Flags y Data. Algunos de los campos son utilizados por el protocolo y otros por la aplicación. Otras aplicaciones de Diameter las constituyen Mobile-IP, Mobil IPv4 y Mobile IPv6. Como se ha comentado anteriormente Diameter se encuentra estrechamente conectado con la aplicación CMS (Cryptographic Message Syntax) para proveer seguridad en todas las aplicaciones, como si tuvieran diferentes funciones, todas ellas soportadas por el protocolo base. En la siguiente figura puede observarse la arquitectura de Diameter representada por el protocolo base, las aplicaciones Diameter y la aplicación CMS.

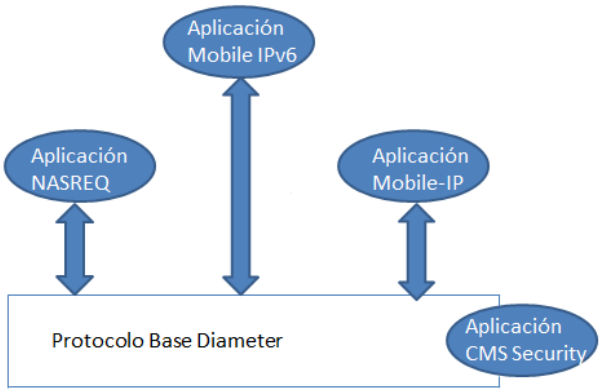


Figura 10. Arquitectura de Diameter.

Fuente: Propia.

Formato del Paquete.

El paquete Diamter tiene una longitud fija de 20 octetos seguidos de una cantidad variable de AVP. El *header* del mensaje de Diameter contiene dos identificadores de 32 bits, un identificador *Hop-by-Hop* y un identificador *End-to-End*. Esto puede verse en la imagen 11.

El identificador *Hop-by-Hop*, ayuda a hacer juego entre la petición y su respuesta. El identificador *End-to-End*, en conjunto con la identidad del host de origen es utilizado para identificar los mensajes de petición de duplicados.

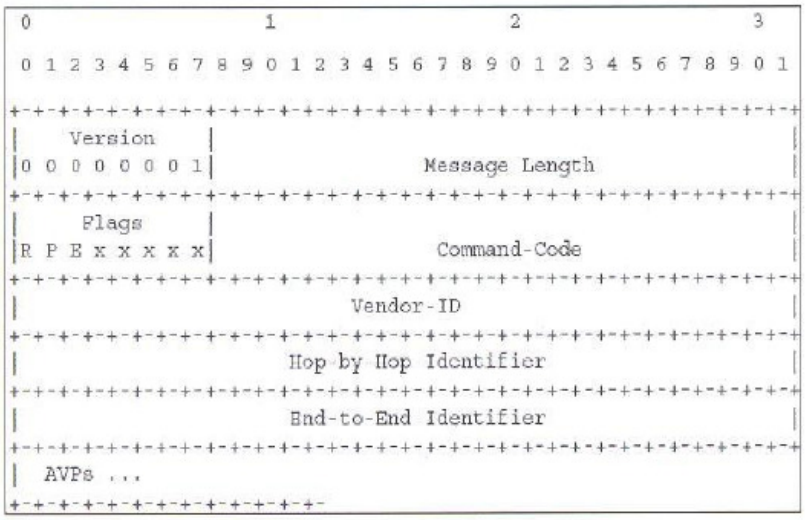


Figura 11. Head de Diameter.

Fuente: IEEE

El formato del header AVP de Diameter se ilustra en la siguiente imagen:

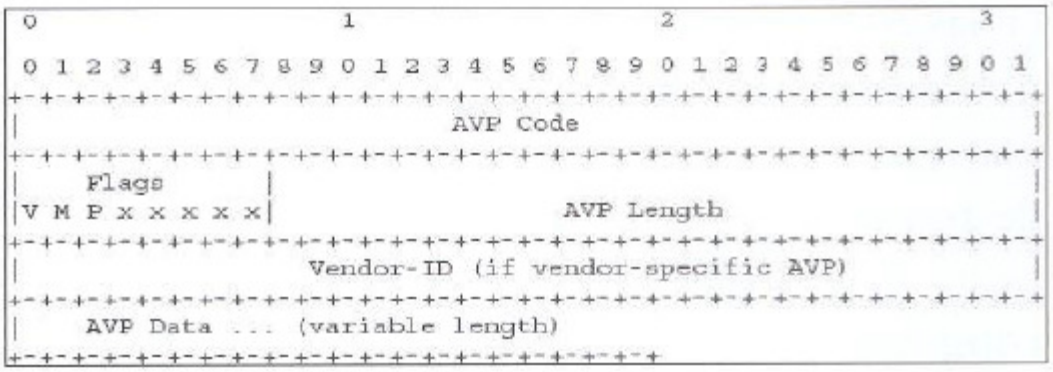


Figura 12. Head AVP de Diameter.

Fuente: IEEE

El *AVP Code*, más el *Vendor-ID*, únicamente identifican el atributo. Los primeros 256 números AVP con el *Vendor-ID* en cero, son reservados para compatibilidad con Radius.

El *AVP Data* contiene información específica del atributo. Diameter define los siguientes tipos de datos: Integer32, Unsigned32, Integer64, Unsigned64, Float32, Float64, Float128, OctetString y Grouped. Todos los tipos de datos se explican así mismo a diferencia del tipo Grouped que es un tipo de dato que contiene una secuencia de AVP.

2.5 Kerberos

Kerberos es un protocolo de seguridad creado por el MIT (Massachusetts Institute of Technology) que usa una criptografía de claves simétricas, es decir, tanto el cliente y el servidor comparten una llave común que es usada para encriptar y descifrar la comunicación de la red, para validar usuarios con los servicios de red, evitando así tener que enviar contraseñas a través de ésta. Al validar los usuarios para los servicios de la red por medio de Kerberos, se pretende evitar los intentos de usuarios no autorizados que intentan interceptar contraseñas en la red.

Funcionamiento de Kerberos.

A continuación se describe de manera general el funcionamiento del protocolo Kerberos, para su seguimiento y compresión se coloca el significado de los siguientes acrónimos:

- ✓ AS = Authentication Server
- ✓ TGS = Ticket Granting Server
- ✓ SS = Service Server.

El funcionamiento es el siguiente: como primer paso, el cliente se autentica a sí mismo contra el AS, así demuestra al TGS que está autorizado para recibir un ticket de servicio (y lo recibe) y ya puede demostrar al SS que ha sido aprobado para hacer uso del servicio kerberizado [16].

A continuación, entraremos en detalle, en su funcionamiento:

1. Un usuario ingresa su nombre de usuario y password en el cliente
2. El cliente genera una clave hash a partir del password y la usará como la clave secreta del cliente.
3. El cliente envía un mensaje en texto plano al AS solicitando servicio en nombre del usuario.
4. El AS comprueba si el cliente está en su base de datos. Si es así, el AS envía dos mensajes al cliente:
    1. Mensaje A: Client/TGS session key cifrada usando la clave secreta del usuario
    2. Mensaje B: Ticket-Granting Ticket (que incluye el ID de cliente, la dirección de red del cliente, el período de validez y el Client/TGS session key) cifrado usando la clave secreta del TGS.
5. Una vez que el cliente ha recibido los mensajes, descifra el mensaje (A) para obtener el client/TGS session key. Esta session key se usa para las posteriores comunicaciones con el TGS. (El cliente no puede descifrar el mensaje B pues para cifrar éste se ha usado la clave del TGS). En este punto el cliente ya se puede autenticar contra el TGS.
6. Una vez autenticado, el cliente envía los siguientes mensajes al TGS:
    1. Mensaje C: Compuesto del Ticket-Granting Ticket del mensaje B y el ID del servicio solicitado.

    2. Mensaje D: Autenticador (compuesto por el ID de cliente y una marca de tiempo), cifrado usando el client/TGS session key.
7. Cuando recibe los mensajes anteriores, el TGS descifra el mensaje D (autenticador) usando el client/TGS session key y envía los siguientes mensajes al cliente:
    1. Mensaje E: Client-to-server ticket (que incluye el ID de cliente, la dirección de red del cliente, el período de validez y una Client/Server session key) cifrado usando la clave secreta del servicio.
    2. Mensaje F: Client/server session key cifrada usando el client/TGS session key.
8. Cuando el cliente recibe los mensajes E y F, ya tiene suficiente información para autenticarse contra el SS. El cliente se conecta al SS y envía los siguientes mensajes:
    1. Mensaje E del paso anterior.
    2. Mensaje G: un nuevo Autenticador que incluye el ID de cliente, una marca de tiempo y que está cifrado usando el client/server session key.
9. El SS descifra el ticket usando su propia clave secreta y envía el siguiente mensaje al cliente para confirmar su identidad:
    1. Mensaje H: la marca de tiempo encontrada en el último Autenticador recibido del cliente más uno, cifrado el client/server session key.
10. El cliente descifra la confirmación usando el client/server session key y chequea si la marca de tiempo está correctamente actualizada. Si esto es así, el cliente confiará en el servidor y podrá comenzar a usar el servicio que este ofrece.
11. El servidor provee del servicio al cliente.

En la siguiente ilustración puede detallarse gráficamente el proceso de funcionamiento de Kerberos:

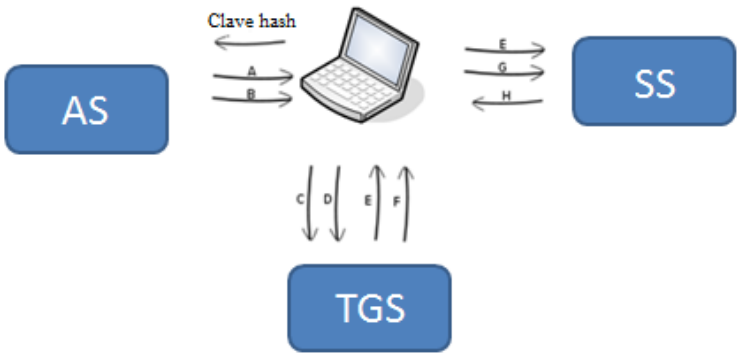


Figura 13. Secuencia Kerberos.

Fuente: Propia

Ventajas.

Las ventajas que establece el uso del protocolo Kerberos es el siguiente:

Los sistemas de redes actuales utilizan el esquema tradicional de autenticación basados en las conocidas contraseñas, es decir, cuando un usuario requiere autenticarse para acceder a un servidor o a un sistema de red, debe proporcionar un usuario y una contraseña. Ahora bien, esta información de autenticación en muchos servicios se transmite sin cifrar o estar encriptada. En este caso, para que la red sea segura, no tiene que ser accesible para usuarios externos, en caso contrario, por ejemplo, cuando la red se conecta a internet, esta deja de ser segura. En este caso, cualquier persona que puede tener acceso a la red y que utilice un sniffer o analizador de tráfico, puede interceptar cualquier contraseña que viaje en texto plano o sin encriptar, comprometiendo así la integridad de los datos. Se puede establecer entonces, como principal ventaja del protocolo de Kerberos, la de evitar transmitir contraseñas que no estén encriptadas a través de la red.

Desventajas.

Entre las principales desventajas que puede presentar el uso del sistema de Kerberos, se puede establecer los siguientes:

- ✓ Dificultad en la migración de contraseñas de usuarios desde una base de datos de contraseñas estándar UNIX, tal como /etc/passwd o /etc/shadow, a una base de datos de contraseñas Kerberos.

- ✓ Para que una aplicación use Kerberos, el código debe ser modificado para hacer las llamadas apropiadas a las librerías de Kerberos. Las aplicaciones que son modificadas de esta forma son consideradas kerberizadas. Para algunas aplicaciones, esto supone un esfuerzo excesivo de programación, debido al tamaño de la aplicación o su diseño. Para otras aplicaciones incompatibles, los cambios se deben realizar en el modo en que el servidor de red y sus clientes se comunican; nuevamente, esto puede suponer realizar programación para su adaptación. En general, las aplicaciones de código cerrado que no tienen soporte de Kerberos son usualmente las más problemáticas.
- ✓ Una desventaja notable de usar Kerberos en una red, es que se debe tener en cuenta que si se transmite cualquier contraseña a un servicio que no usa Kerberos para autenticar, se corre el riesgo de que el paquete pueda ser interceptado. Así, nuestra red no obtendrá ningún beneficio de usar Kerberos. Para asegurar la red con Kerberos, solo se debe utilizar las versiones kerberizadas, es decir, que funcionen con Kerberos de todas las aplicaciones cliente/servidor que envíen contraseñas sin encriptar o no utilizar ninguna de estas aplicaciones en la red.

2.6 Estándar 802.1X

808.1X no es un protocolo de comunicaciones, sino una extensión del sistema de autenticación RADIUS, a las capas más bajas de una red. 802.1X es una gran novedad en la implementación de redes. Cuando nos referimos a seguridad en las capas más bajas de la red, hablamos de la capa 2 o capa de enlace. El campo de actuación de 802.1X es la capa de enlace en redes cableadas o inalámbricas, para asegurar la conexión de dispositivos a la infraestructura de red de la organización. El procedimiento de conexión utilizado por este estándar se basa en la provisión de un sistema de autenticación por puertos en los que se establece un canal o puerto blindado de comunicación entre el suplicante y el NAS a fin de permitir únicamente su autenticación.

La seguridad de las redes se basa en los siguientes conceptos:

- ✓ **Control de Acceso o Control de Admisión.** La seguridad en el acceso a la red es un punto de gran importancia, debiéndose denegar totalmente el acceso a la red, a cualquiera

que no esté autorizado a acceder. El control de acceso a la red debe igualmente ser muy restrictivo con los servicios a los que esté autorizado a utilizar el usuario o equipo solicitante.

- ✓ **Privacidad**: La privacidad es otro punto vital para evitar la interceptación de los datos trasmitidos por un usuario o equipo, que esté utilizando como medio de transporte una red de datos. Gran parte de la información que viaja por la red se transmite en texto en claro, a pesar de ser información privada y sensible. La privacidad siempre se debe basar en la encriptación o cifrado de las comunicaciones en las capas 2 y 3 del modelo OSI.

- ✓ **Autenticación y Autorización**: La autenticación y autorización son el motor que va a permitir llevar a cabo, de forma íntegra, los dos puntos anteriores. La autenticación debe asegurar los medios para su propia integridad, impidiendo la interceptación de la credenciales o lo intentos de penetración no autorizados. La autorización se ocupará de limitar el uso del canal y de los recursos por parte del equipo o usuario.

La seguridad de una red Ethernet, siempre ha comenzado en las capas superiores, desde la capa 7 y descendiendo. Cuando entramos en el correo o en una página http nos autenticamos en la capa de aplicación, al igual que cuando establecemos una sesión SSL. Las redes VPN, que nos dan un gran soporte de seguridad en las comunicaciones punto a punto, es permitir que trabajen en la capa tres (red). Los protocoles más comunes que trabajan sobre esta capa son IPSec, PPTP, etc. Su implantación es también bastante compleja. El problema de trabajar sobre sistemas VPN como IPSec es que sólo garantizan la seguridad sobre el protocolo IP en la capa tres y superiores, dejando abierto el tráfico de otros protocolos de la capa tres y de los protocolos de autenticación, que actúan sobre la capa dos.

El protocolo 802.1x, es necesario para garantizar la seguridad a partir de la capa dos del modelo OSI. La seguridad en la capa de enlace provee de los mecanismos necesarios para viabilizar los procesos de autenticación y envíos de trama de control. Todo esto es lo que se conoce como la

seguridad en la capa de enlace (Link Layer Security) que además expande esa seguridad a todas las capas superiores.

A la 802.1x se le llama comúnmente ESPOL (Extensible Authentication Protocolo over Lan) o EAP sobre Ethernet. Si lo aplicamos a tecnologías inalámbricas se suele llamar EAPoW (EAP over Wireless). EAP es el protocolo de transporte de la autenticación nativo de 802.1x, ya que trabaja contra servidores de autenticación sobre tramas Ethernet, lo que permite trabajar en la capa dos del modelo OSI (Capa de enlace de datos o capa MAC). También se define como un sistema de autenticación basado en puertos puesto que el control de admisión se realiza a través de puertos virtuales LAN.

Un equipo NAS que trabaje sobre 802.1x emplea un sistema virtual de dividir cada puerto físico LAN (PAE Port Access Entity) en dos puertos virtuales, un puerto controlado y un puerto incontrolado (controlled port & uncontrolled port). Desde el momento de la conexión de un dispositivo al puerto físico LAN hasta el momento en el que se produzca una autenticación exitosa, sólo el puerto virtual incontrolado está abierto. Y este puerto incontrolado solamente permite el paso de paquetes del tipo ESPOL a fin de permitir el proceso de autenticación del suplicante.

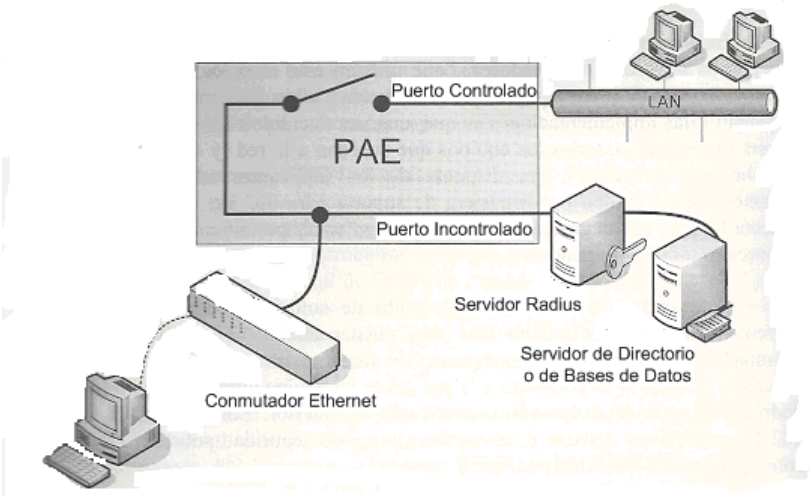


Figura 14. Estado de Puerto 802.1X

Fuente: Hansen, Fernandez Yago, 2008

Una vez que el suplicante se haya autenticado exitosamente, el puerto pasa a un estado autorizado o controlado y se abre para ese dispositivo. Cuando está abierto, el equipo autorizado puede pasar simplemente a hacer uso de la red, de forma normal. A través del proceso de autorización, que gestiona el servidor RADIUS, se le pueden asignar al suplicante características como una dirección IP estática/dinámica de un ámbito definido, o simplemente dejar a ese equipo en una red virtual (VLAN), segmentando la red en diferentes partes para diferentes usos. Por ejemplo: un equipo de recepción de una empresa que se conecta automáticamente mediante su suplicante al switch de la red (compatible 802.1x) y tras el proceso de autenticación contra RADIUS, se le asigna una dirección IP reservada para él y se le abre el puerto, aislándola en una VLAN del departamento de recepción, en la que solo coexisten los equipos de recepción y la puerta de enlace para conectarse a Internet. Las redes VLAN trabajan en la capa 2 y 3 de OSI y necesitan enrutamiento IP. La Fig. 15 muestra la configuración para un usuario, al que se le asigna una VLAN en el momento de la conexión:

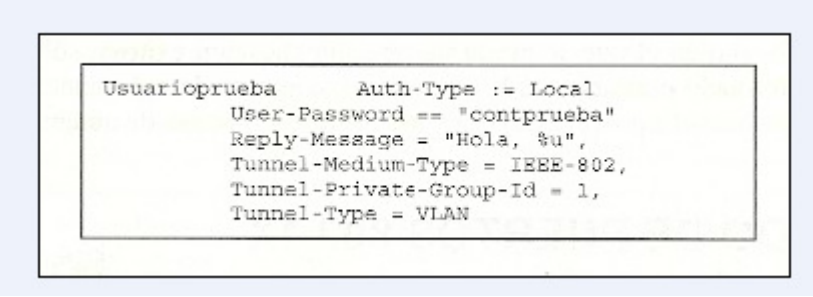


Figura 15. Configuración de usuario.

Fuente: Hansen, Fernández Yago, 2008

Existe una gran variedad de modelos de equipos de NAS como switches de red, puntos de acceso, enrutadores, que utilizan esta tecnología para su propio acceso a la red o el de los equipos que se conecten a ellos. Si bien, un problema que presentan estas implementaciones es que, una vez decidido el uso de este estándar en una organización, todos los equipos que accedan a la red deben cumplir este estándar, o sea disponer de los suplicantes adecuados para su conexión. Si un equipo no dispusiera de soporte para 802.1x, y se tendría que conectar a la red degradaríamos la seguridad total, permitiendo su acceso.

Este estándar puede ir acompañado de soluciones de encriptación de tráfico, en caso de redes inalámbricas como puede ser WPA2 o similares. Si no va acompañado de encriptación en la capa de Red, no ofrece el cien por cien de seguridad, ya que si se accediera a la red desde un equipo autorizado para hacerlo, podríamos seguir interceptando el tráfico en su interior. Por lo tanto el uso de 802.1x no excluye del uso de otras soluciones de seguridad sobre la capa tres, como puede ser VPN de tipo IPsec o similares.

Lo que hay detrás de las tecnologías 802.1x no es sino un servidor de autenticación, en la mayor parte de los casos RADIUS, que gestiona todo el sistema AAA contra los suplicantes, a través del equipo NAS. Esta autenticación se realiza de forma muy segura, utilizando sistemas vía túneles basados en certificados PKI, como EAP-TLS, EAP-PEAP u otros. Se puede, aunque no se recomienda, utilizar versiones más débiles de EAP como EAP-MD5, ya que esto descendería el nivel de seguridad general. De esa manera el incremento de seguridad es máximo, creando un sistema de autenticación bidireccional (mutua) donde el servidor identifica al cliente y el cliente identifica al servidor. Así se evitan graves riesgos de seguridad, como la posible existencia de *rogue-AP* actuando como MiTM (Man in The Middle). Este tipo de autenticación además de autenticar al usuario puede hacer lo mismo con el equipo, por lo que se puede disponer de un certificado de equipo para poder entrar en la red. RADIUS utiliza en este caso una tecnología que proporciona autenticación, integridad de datos y privacidad, a través del protocolo 802.1x.

Esto ha ido cambiado en los últimos años el concepto que se tenía de la seguridad, ya que ahora se busca la seguridad en el control de acceso al medio y los cortafuegos suben la seguridad hacia las capas más altas (capa 7), cuando antes se basaban en el filtrado de puertos en las capas más bajas.

Una infraestructura 802.1x basada en PKI puede disponer de x equipos que se conectan a la red, que disponen de x certificados de equipos. Además cada usuario dispone de las credenciales necesarias para autenticarse contra la red. La administración de estas infraestructuras es un poco más tediosa, pero el resultado incrementa la seguridad general de forma notable. Quizás el hecho de que su implantación sea un poco más compleja es lo que haya causado que, desde su publicación, su implantación en el mercado sea muy lenta.

Otro motivo que ha dificultado la extensión de 802.1x ha sido la falta de disponibilidad de programas suplicantes adecuados y robustos. Microsoft dispone de un suplicante 802.1x en sus versiones, pero tiene algunos problemas de diseño. Uno de esos problemas es la necesidad de activar el servicio WXCS (servicio de configuración inalámbrica rápida) para disponer de la autenticación 802.1x, aunque sea para una tarjeta Ethernet LAN y no WLAN. Este servicio suele gustar a pocos administradores, y muchos suplicantes para tarjetas inalámbricas tienden a deshabilitarlo, con lo que se deshabilita también el suplicante para redes cableadas. Otro de esos problemas es el propio diseño de las fases de autenticación entre el suplicante 802.1x y el propio sistema de login de Windows hacia un servidor de dominio Kerberos o de directorio activo. Windows pretende autenticarse contra su servidor de dominio, antes de hacerlo contra la red 802.1x, pero su servidor de dominio no estaría disponible hasta que la red estuviera disponible. Para evitar este caos de autenticación, las soluciones pasan por hacer disponible al servidor de dominio en una red virtual VLAN abierta, con los graves problemas de seguridad que eso conlleva; también se puede cambiar un suplicante de otro fabricante, libre o de pago.

El estándar 802.1x se recoge en el RFC 3580 (IEEE 802.1x RADIUS Usage Guidelines) el estándar IEEE 802.11i ratificado en 2004 está incluido en el RFC 4017.

2.7 Sistemas de Soporte.

Cuando hablamos de sistemas de soporte, debemos de hacer referencia a los servicios de directorio, que a su vez son bases de datos optimizadas para la lectura, navegación y búsqueda de datos. En la mayoría de los directorios se tiende a contener información descriptiva basada en atributos, esto facilita enormemente la capacidad de búsqueda cuando se realiza mediante la aplicación de un filtro en particular, esto es muy útil y aplicado en entornos empresariales de gran tamaño o mediano, al igual que universidades, que facilitan la búsqueda de datos de manera óptima y gestionar el acceso de usuarios a los recursos sobre dicha red. Una de las características que debemos resaltar es que no debemos compararlos con los sistemas de bases de datos tradicionales, es decir, no permite realizar transacciones complejas, pero si permiten realizar actualizaciones simples. Es por eso que los directorios están ajustados para dar una rápida respuesta a grandes volúmenes de datos para su búsqueda. Además tienen la capacidad de

replicar la información para incrementar la disponibilidad y la fiabilidad, al tiempo que reducen los tiempos de respuesta. La utilización de este servicio en el desarrollo de nuestro piloto, nos permitirá ver sus ventajas en su utilización, la facilidad de creación de una pequeña base de datos de usuarios, permitirá ejemplificar la gestión del acceso a dichos usuarios a la utilización de los recursos de red.

2.7.1 Servicio de Directorio.

Un servicio de directorio (SD) es una aplicación o un conjunto de aplicaciones que almacena y organiza la información de los usuarios de una red de computadores, permitiendo a los administradores gestionar el acceso de usuarios a los recursos sobre dicha red. Además, los servicios de directorio actúan como una capa de abstracción entre los usuarios y los recursos compartidos.

Los directorios tienden a contener información descriptiva basada en atributos y tienen capacidades de filtrado muy sofisticada. Los directorios generalmente no soportan transacciones complicadas ni esquemas de vuelta atrás (Roll Back) como los que se encuentran en los sistemas de bases de datos diseñados para manejar grandes y complejos volúmenes de actualizaciones. Las actualizaciones de los directorios son normalmente cambios simples.

Un servicio de directorio no debería confundirse con el repositorio de directorio, que es la base de datos, esta es la que contiene la información sobre los objetos nombrados, gestionado por el servicio de directorio. El servicio de directorio proporciona la interfaz de acceso a los datos que se contienen en unos o más espacios de nombre de directorio. La interfaz del servicio de directorio es la encargada de gestionar la autenticación de los accesos al servicio de forma segura, actuando como autoridad central para el acceso a los recursos de sistema que manejan los datos del directorio.

Como base de datos, un servicio de directorio está altamente optimizado para lecturas y proporciona alternativas avanzadas de búsqueda en los diferentes atributos que se puedan asociar a los objetos de un directorio. Los datos que se almacenan en el directorio son definidos por un

esquema extensible y modificable. Los servicios de directorio utilizan un modelo distribuido para almacenar su información y esa información generalmente está replicada entre los servidores que forman el directorio.

Los directorios están afinados para dar una rápida respuesta a grandes volúmenes de búsquedas. Estos tienen la capacidad de replicar la información para incrementar la disponibilidad y la fiabilidad, al tiempo que reducen los tiempos de respuesta. Cuando la información de un directorio se replica, se pueden producir inconsistencias temporales entre las réplicas mientras esta se está sincronizando.

Hay muchas formas diferentes de proveer un servicio de directorio. Diferentes métodos permiten almacenar distintos tipos de información en el directorio, tener distintos requisitos sobre cómo la información ha de ser referenciada, consultada y actualizada, cómo es protegida de los accesos no autorizados, etc. Algunos servicios de directorio son locales, es decir, proveen el servicio a un contexto restringido (como por ejemplo, el servicio "finger" en una única máquina). Otros servicios son globales y proveen servicio a un contexto mucho más amplio (como por ejemplo, Internet). Los servicios globales normalmente son distribuidos, esto significa que los datos están repartidos a lo largo de distintos equipos, los cuales cooperan para dar el servicio de directorio. Típicamente, un servicio global define un espacio de nombres uniforme que da la misma visión de los datos, independientemente de donde se esté, en relación a los propios datos. El servicio DNS (Domain Name System) es un ejemplo de un sistema de directorio globalmente distribuido.

En principio en un servicio de directorio se puede almacenar cualquier tipo de información. Como por ejemplo, nombre, dirección de habitación, email, etc. Sin embargo, la información que se almacena es aquella que permita organizar de manera jerárquica todos los usuarios de la red. Estructurar la información de los usuarios de la red es de utilidad a la hora de restringir el acceso a los servicios y recursos de la red; permitiendo gestionar con mayor facilidad la red.

2.7.2 Lightweight Directory Access Protocol (LDAP).

El acrónimo LDAP en inglés significa Lightweight Directory Access Protocol (LDAP); traducido al español su significado es: Protocolo Ligero para Acceder al Servicio de Directorio, ésta implementación se basa en el estándar X.500, el cual es un conjunto de estándares de redes de computadoras de la ITU-T sobre el servicio de directorios. LDAP se ejecuta sobre TCP/IP o sobre otros servicios de transferencia orientada a conexión; que permite el acceso a la data de un directorio ordenado y distribuido para buscar información.

Habitualmente se almacena información de los usuarios que conforman una red de computadores, como por ejemplo el nombre de usuario, contraseña, etc. Es posible almacenar otro tipo de información tal como, correo electrónico, número de teléfono móvil, etc. En conclusión, LDAP es un protocolo de acceso unificado a un conjunto de información sobre los usuarios de una red de computadores.

Al utilizar LDAP se puede consolidar información para toda una organización dentro de un repositorio central. Por ejemplo, en vez de administrar listas de usuarios para cada grupo dentro de una organización, se puede usar LDAP como directorio central, accesible desde cualquier parte de la red. Puesto que LDAP soporta la Capa de conexión segura (SSL) y la Seguridad de la capa de transporte (TLS), los datos confidenciales se pueden proteger. LDAP también soporta un número de bases de datos “back-end” en las que se almacena la información. Esto permite que los administradores tengan la flexibilidad para desplegar la base de datos más indicada, para el tipo de información. LDAP tiene una interfaz de programación de aplicaciones (API) bien definida, existe un número de aplicaciones acreditadas para LDAP, estás están en distintos lenguajes de programación, tales como C, C++, Java, Perl, PHP, entre otros.

2.7.2.1 Almacenamiento de la información en LDAP.

La información en LDAP es almacenada de la siguiente manera, en principio la información es ordenada en el modelo de LDAP en entradas. Una entrada es una colección de atributos que tienen un único Nombre Global Distinguido (DN). El DN se utiliza para referirse a una entrada

sin ambigüedades. Cada atributo de una entrada posee un tipo y uno o más valores. Los tipos son normalmente palabras nemotécnicas, como "cn" para common name, o "mail" para una dirección de correo. La sintaxis de los atributos depende del tipo de atributo. Por ejemplo, un atributo cn puede contener el valor "Giovannni Mazzei". Un atributo email puede contener un valor "gmazzei@uniovi.edu.es".

Estas entradas están organizadas en una estructura jerárquica en forma de árbol invertido, de la misma manera como se estructura el sistema de archivos de UNIX. Tradicionalmente esta estructura reflejaba los límites geográficos y/u organizacionales. Las entradas que representan países aparecen en la parte superior del árbol. Debajo de ellos, están las entradas que representan los estados y las organizaciones nacionales. Bajo estas, pueden estar las entradas que representan las unidades organizacionales, empleados, impresoras, documentos o todo aquello que pueda imaginarse. La siguiente figura muestra un árbol de directorio LDAP haciendo uso del nombramiento tradicional.

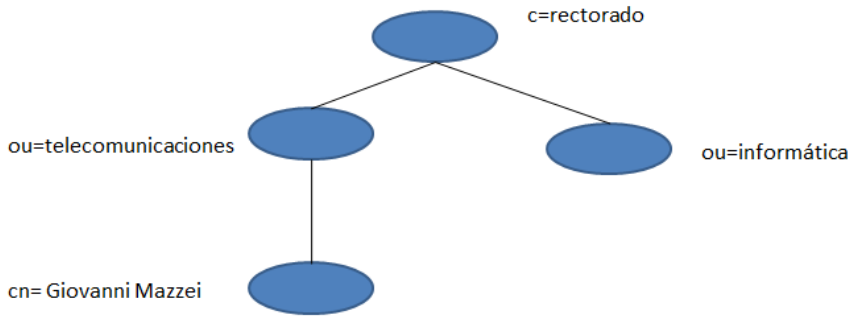


Figura 16 .Árbol de directorio LDAP (nombramiento tradicional).

Fuente: Propia

El árbol también se puede organizar basándose en los nombres de dominio de Internet. Este tipo de nombramiento se está volviendo muy común y en los actuales momentos es el más utilizado, ya que permite localizar un servicio de directorio haciendo uso de los DNS. La siguiente figura muestra un árbol de directorio que hace uso de los nombres basados en dominios.

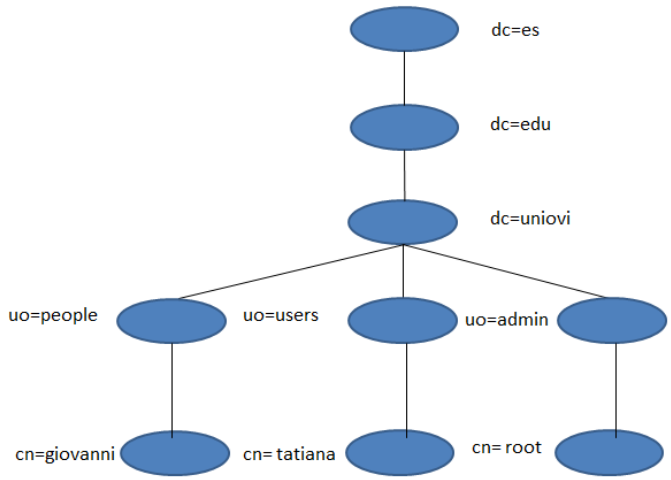


Figura 17: Árbol de directorio LDAP (nombramiento de Internet).

Fuente: Propia.

Un ejemplo del DN sería:

dn: cn=Giovanni Mazzei, ou=people, dc=uniovi, dc=edu, dc=es

Podemos ver que el dn se construye de abajo hacia arriba. Al igual que se construyen los nombres en DNS. Además, LDAP permite controlar qué atributos son requeridos y permitidos en una entrada gracias al uso del atributo denominado objectClass. El valor del atributo objectClass determina qué reglas de diseño (schema rules) ha de seguir la entrada.

2.7.2.2 Referencia de información en LDAP.

Una entrada es referenciada por su nombre distinguido, que es construido por el nombre de la propia entrada llamado Nombre Relativo Distinguido (RDN) y la concatenación de los nombres de las entradas que le anteceden. Por ejemplo, la entrada para giovanni en el ejemplo del nombramiento de Internet anterior tiene el siguiente RDN: uid=giovanni y su DN sería: uid=giovanni,ou=people,dc=uniovi,dc=edu,dc=es. De esta manera se puede acceder a toda la información que se almacenada en el directorio LDAP.

2.7.2.3 Acceso a la información en LDAP.

LDAP provee operaciones para añadir, modificar y eliminar entradas del mismo. La mayor parte del tiempo, LDAP se utiliza para buscar información almacenada en el directorio. Las operaciones de búsqueda de LDAP permiten encontrar entradas que concuerdan con algún criterio especificado dado por un filtro de búsqueda. La información puede ser solicitada desde cada entrada que concuerda con dicho criterio.

Un ejemplo de esto, es que se quiere buscar en el subárbol del directorio que está por debajo de dc=uniovi,dc=edu,dc=es a personas con el nombre Giovanni Mazzei, obteniendo la dirección de correo electrónico de cada entrada que concuerde, LDAP permite realizar esta tarea. O tal vez buscar las organizaciones que posean la cadena uinovi en su nombre o posean un número de fax.

2.7.2.4 Arquitectura de LDAP.

El servicio de directorio de LDAP está basado en el modelo cliente/servidor. Uno o más servidores LDAP contienen los datos que conforman la información del árbol del directorio (DIT). El cliente se conecta a los servidores y les formula preguntas. Los servidores responden con una respuesta o con un puntero donde el cliente puede obtener información adicional (normalmente otro servidor LDAP). No importa a que servidor LDAP se conecte un cliente, este siempre obtendrá la misma visión del directorio; un nombre presentado por un servidor LDAP referencia la misma entrada que cualquier otro servidor LDAP.

2.7.2.5 Estándar X.500.

X.500 es un conjunto de estándares de redes de computadores de la ITU (Unión Internacional de Telecomunicaciones) sobre servicios de directorio, entendidos estos como bases de datos de direcciones electrónicas (o de otros tipos). El estándar se desarrolló conjuntamente con la ISO como parte del modelo de interconexión de sistemas abiertos, para usarlo como soporte del correo electrónico X.400.

Los protocolos definidos por X.500 incluyen:

- ✓ Protocolo de acceso al directorio (DAP)
- ✓ Protocolo de sistema de directorio
- ✓ Protocolo de ocultación de información de directorio
- ✓ Protocolo de gestión de enlaces operativos de directorio.

Dentro de la serie X.500, la especificación que ha resultado ser la más difundida no trata de protocolos de directorio, sino de certificados de clave pública X.509. El protocolo LDAP fue creado como una versión liviana de X.500 y terminó por reemplazarlo. Por esta razón algunos de los conceptos y estándares que utiliza LDAP provienen de la serie de protocolos X.500.

Técnicamente, LDAP es un protocolo de acceso a directorio para el servicio de directorio X.500, del servicio de directorio de OSI. Inicialmente, los clientes LDAP accedían a través de puertas de enlace al servicio de directorio X.500. Esta puerta de enlace ejecutaba LDAP entre el cliente y la puerta de enlace, y el Protocolo X.500 de Acceso al Directorio (DAP) entre la puerta de enlace y el servidor X.500. DAP es un protocolo extremadamente pesado que opera sobre una pila protocolar OSI completa y requiere una cantidad significativa de recursos computacionales. LDAP está diseñado para operar sobre TCP/IP proporcionando una funcionalidad similar a la de DAP, pero con un costo muchísimo menor.

Aunque LDAP se utiliza todavía para acceder al servicio de directorio X.500 a través de puertas de enlace, hoy en día es más común implementar LDAP directamente en los servidores X.500. El demonio autónomo de LDAP, o SLAPD, puede ser visto como un servidor de directorio X.500 ligero. Es decir, no implementa el DAP X.500, sino un subconjunto de modelos de X.500. Es posible replicar datos desde un servidor de directorio LDAP hacia un servidor DAP X.500. Esta operación requiere una puerta de enlace LDAP/DAP. OpenLDAP no suministra dicha puerta de enlace, pero el demonio de replicación que posee puede ser usado para la replicación, como si de una puerta de enlace se tratase.

2.7.2.6 Versión 3 de LDAP.

LDAPv3 incorpora las siguientes características a LDAP:

- ✓ Soporta LDAP sobre IPv4, IPv6 y Unix IPC.
- ✓ Tiene soporte de autentificación fuerte gracias al uso de SASL. La implementación SASL de SLAPD hace uso del software Cyrus SASL, el cual soporta un gran número de mecanismos de autentificación, como: DIGEST-MD5, EXTERNAL, y GSSAPI.
- ✓ Provee protecciones de privacidad e integridad gracias al uso de TLS o SSL. La implementación TLS de SLAPD hace uso del software OpenSSL
- ✓ Puede ser configurado para restringir el acceso a la capa de socket basándose en la información topológica de la red. Esta característica hace uso de los TCP wrappers (Herramienta simple que sirve para monitorear y controlar el tráfico que llega por la red)
- ✓ Provee facilidades de control de acceso muy potentes, permitiéndole controlar el acceso a la información de su(s) base(s) de datos. Puede controlar el acceso a las entradas basándose en la información de autorización de LDAP, en la dirección IP, en los nombres de dominio y otros criterios. SLAPD soporta tanto el control de acceso a la información dinámico como estático.
- ✓ Puede ser configurado como un servicio proxy de caché LDAP.
- ✓ "Simple Authentication and Security Layer" (capa de seguridad y autenticación simple). Es un framework para manejar la autenticación y autorización en protocolos de Internet. Este separa los mecanismos de autenticación de los protocolos de la aplicación.

Como LDAPv2 difiere significativamente de LDAPv3, la interacción entre ambas versiones puede ser problemática.

2.7.2.7 Diferencia entre LDAPv2 y v3.

El LDAP v3 (RFC 2251) está diseñado para corregir algunas de las limitaciones del LDAP v2 en las áreas como la internalización de los caracteres, la autentificación, la remisión, entre otros. A continuación procedemos a explicar alguno de ellos.

- ✓ La versión 3 de LDAP se diferencia de la v2, ya que esta usa UTF-8 para codificar los strings.

- ✓ LDAP v2 soportaba tres tipos de autentificaciones: anónima, simple (password de texto claro), y Kerberos v4. El LDAP v3 usa el marco de trabajo de autentificación "Simple Authentication and Security Layer (SASL)" (RFC 2222) para permitir utilizar diferentes mecanismos de autentificación con el LDAP. SASL especifica un protocolo de desafío-respuesta en el que se intercambian datos entre el cliente y el servidor para propósitos de autentificación.

- ✓ Una remisión es información que un servidor envía de vuelta al cliente indicando que la información solicitada puede encontrarse en otra localización, posiblemente en otro servidor. En el caso de LDAP v2, los servidores manejaban la remisiones y no las devolvían al cliente. Esto es porque manejar las remisiones puede ser muy complicado. Como los servidores eran construidos y desplegados, las remisiones se encontraron útiles, pero no muchos servidores soportan el manejo de remisiones en el lado del servidor. Por eso una forma de reparar el protocolo es permitir que se devuelvan las remisiones. Esto se hace situando la remisión dentro de un mensaje de error de una respuesta de error de "resultado parcial". El LDAP v3 tiene soporte explícito para remisiones y permite a los servidores devolver remisiones directamente al cliente

2.7.3 Directorio Activo de Microsoft.

En el ámbito de las redes de ordenadores, el concepto de directorio (o almacén de datos) es una estructura jerárquica que almacena información sobre objetos en la red, normalmente implementada como una BD optimizada para operaciones de lectura y que soporta búsquedas de

grandes cantidades de información y con capacidades de exploración. Este servicio de directorio (Active Directory) es un servicio de red que almacena información acerca de los recursos de la red y permite el acceso de los usuarios y las aplicaciones a dichos recursos, de forma que se convierte en un medio de organizar, controlar y administrar centralizadamente el acceso a los recursos de la red. Se establece también que el servicio Active Directory proporciona la capacidad de establecer un único inicio de sesión y un repositorio central de información para toda su infraestructura.

Una de las ventajas primordiales que ofrece el Active Directory es que permite separar la infraestructura lógica de una organización (dominios) de la infraestructura física (topología de la red). Esto permite independizar la estructuración de dominios de la organización, de la topología de la red que interconecta los sistemas.

Active Directory ejerce una gran cantidad de funciones, entre las cuales podemos nombrar las siguientes:

- ✓ ***Centralizar el control de los recursos de red***. Al centralizar el control de recursos como servidores, archivos compartidos e impresoras, sólo los usuarios autorizados pueden obtener acceso a los recursos de Active Directory.

- ✓ ***Centralizar y descentralizar la administración de recursos***. Los administradores pueden administrar equipos clientes distribuidos, servicios de red y aplicaciones desde una ubicación central mediante una interfaz de administración coherente o pueden distribuir tareas administrativas mediante la delegación del control de los recursos a otros administradores.

- ✓ ***Almacenar objetos de forma segura en una estructura lógica.*** Active Directory almacena todos los recursos como objetos en una estructura lógica, jerárquica y segura.

- ✓ ***Optimizar el tráfico de red.*** La estructura física de Active Directory permite utilizar el ancho de banda de red de forma más efectiva. Por ejemplo, garantiza que, cuando un

usuario inicie una sesión en la red, la autoridad de autenticación más cercana a él lo autentique, reduciendo así la cantidad de tráfico de red.

Siguiendo con la clasificación de las infraestructuras tanto lógica como física, procedemos a explicar un poco en detalle los mismos:

La estructura lógica de Active Directory comprende los siguientes componentes:
Objetos. Son los componentes más básicos de la estructura lógica. Las clases de objetos son plantillas o planos técnicos para los tipos de objetos que se pueden crear en Active Directory. Cada clase de objetos se define mediante un grupo de atributos, que definen los posibles valores que se pueden asociar a un objeto. Cada objeto posee una única combinación de valores de atributos.

Unidades organizativas. Se pueden utilizar estos objetos contenedores para estructurar otros objetos de modo que admitan los propósitos administrativos. Mediante la estructuración de los objetos por unidades organizativas, se facilita su localización y administración. También se puede delegar la autoridad para administrar una unidad organizativa. Las unidades organizativas pueden estar anidadas en otras unidades organizativas, lo que simplifica la administración de objetos.

Dominios. Se trata de las unidades funcionales centrales en la estructura lógica de Active Directory que son un conjunto de objetos definidos de forma administrativa y que comparten una base de datos, directivas de seguridad y relaciones de confianza comunes con otros dominios.

Árboles de dominios. Los dominios que están agrupados en estructuras jerárquicas se denominan árboles de dominios. Al agregar un segundo dominio a un árbol, se convierte en secundario del dominio raíz del árbol. El dominio al que está adjunto un dominio secundario se denomina dominio primario. Un dominio secundario puede tener a su vez su propio dominio secundario.

El nombre de un dominio secundario se combina con el nombre de su dominio primario para formar su propio nombre único de Sistema de nombres de dominio (DNS, *Domain Name System*).

Bosques. Un bosque es una instancia completa de Active Directory. Consta de uno o varios árboles. En un árbol de sólo dos niveles, que se recomienda para la mayoría de las organizaciones, todos los dominios secundarios se convierten en secundarios del dominio raíz de bosque para formar un árbol contiguo. El primer dominio del bosque se denomina dominio raíz de bosque, que es el primer dominio creado en el mismo. Podemos también agregar que los nombres de dominios en un bosque pueden ser continuos o discontinuos en la jerarquía del DNS. Por ejemplo los continuos, están en el mismo árbol de dominio, en cambio los discontinuos forman varios árboles de dominio.

En referencia a la infraestructura lógica, que se basa primordialmente en requisitos administrativos, la infraestructura física del Active Directory optimiza el tráfico de red, en el proceso de realizar la replicación y el tráfico de la conexión. En cuanto a la infraestructura física de un Active Directory, esta se conforma de los siguientes elementos:

Controladores de Dominio: Un controlador de dominio realiza funciones de almacenamiento y replicación. Un controlador de dominio sólo puede admitir un dominio. Para asegurar la disponibilidad continua de Active Directory, cada dominio debe disponer de más de un controlador de dominio.

Sitios de Active Directory. Estos sitios son grupos de equipos conectados entre sí de manera correcta. Al establecer sitios, los controladores de dominio de un único sitio se comunican con frecuencia entre sí. Esta comunicación minimiza la latencia dentro del sitio, es decir, el tiempo necesario para que un cambio realizado en un controlador de dominio pueda replicarse en otros controladores de dominio. Se pueden crear sitios para optimizar el uso del ancho de banda entre los controladores de dominio que están en ubicaciones diferentes.

***Particiones de Active Directory*:** Cada controlador de dominio contiene las siguientes particiones de Active Directory:

Partición del dominio: Esta contiene replicados todos los objetos de ese dominio. La partición del dominio sólo puede replicarse en otro controlador de dominio del mismo dominio.

Partición de configuración: Contiene la topología del bosque. La topología es un registro de todos los controladores de dominio y las conexiones entre ellos en un bosque.

Partición del esquema: Contiene el esquema de todo el bosque. Cada bosque tiene un esquema para que la definición de las clases de objetos sea coherente. Las particiones de configuración y del esquema pueden replicarse en los controladores de dominio del bosque.

Particiones de aplicaciones: Contienen objetos no relacionados con la seguridad y utilizados por una o varias aplicaciones. Las particiones de aplicaciones pueden replicarse en controladores de dominio especificados del bosque.

2.8. Cut-through proxy.

En el planteamiento de nuestro trabajo de investigación, se estableció la importancia que tiene la implementación de un sistema de control de acceso a los recursos de red, específicamente en la universidad. Un usuario, en este caso un estudiante puede conectarse y tener acceso a la red de la universidad, con solo conectar un cable de red a un puerto o jack, sin que este valide al alumno, caso contrario si lo hace cuando queremos acceder a la red de la universidad vía inalámbricamente, específicamente WiFi. Por tanto, debe existir un sistema que permita autenticar a un usuario de manera eficiente y lo más importante que dicha autenticación ocurra en la capa de aplicación. La arquitectura del sistema que plantearemos a continuación permitirá orientar nuestro piloto a dicho sistema de autenticación de usuarios a los recursos de red, de manera segura.

El mecanismo cut-through proxy, permite que el usuario se identifique mediante Telnet, FTP o HTTP, antes de poder utilizar un aplicativo. Esta autenticación consiste en demostrar la identidad mediante un nombre de usuario y una contraseña, que no tienen por qué ser los mismos datos que los necesarios para conectarse al servidor al que se desea hacer la conexión.

La base de datos de usuarios contra la que se contrasta la información de autenticación puede ser local, configurada en un ASA por ejemplo o remota. En este último caso por ejemplo, mediante protocolo RADIUS o protocolo TACACS/+, el ASA solicita al servidor AAA la validación del usuario. Este procedimiento puede observarse en la siguiente imagen a continuación.

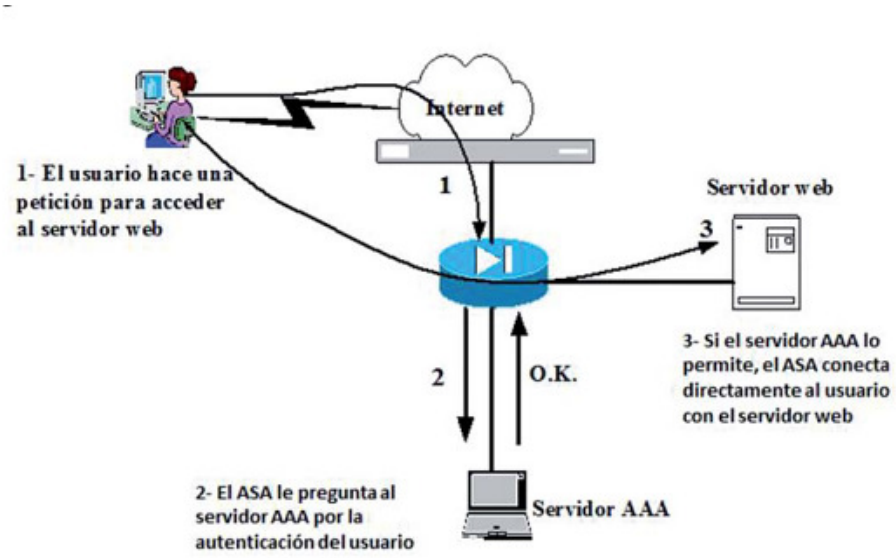


Figura 18.Mecanismo cut-through proxy.

Fuente: G.Diaz, F.Mur, E. Sancristóbal, M. Castro, J. Peire (2012)

En fin, el método cut-through proxy, que permite verificar si los usuarios tienen permisos para ejecutar una aplicación TCP o UDP antes de llegar a la aplicación, es decir, verifica a los usuarios en el mismo firewall. Este tipo de autenticación permite realizarla de manera más granula y basada en el usuario.

Vulnerabilidad Caso Cisco.

A continuación haré mención de una serie de vulnerabilidades que afectan al módulo Firewall Service Module (FWSM) de los switches Catalyst serie 6500 y los routers de la serie 7600 en las versiones 3.1.x, 3.2.x, 4.0.x, y 4.1.x., principalmente del sistema de Authentication Proxy o cut-through. Esto nos permitirá en primer lugar, tener el conocimiento de la existencia de dicha vulnerabilidad para así evitar inconveniente a la hora de su implantación en relación a los ataques de denegación de servicio. Esto siempre y cuando se implemente dicha arquitectura en equipos

del fabricante Cisco, ya que para el fabricante de Juniper una arquitectura similar se le conoce como Web-based authentication, y no se ha encontrado vulnerabilidad alguna.

Denegación de servicio a través de Authentication Proxy.

Revisando la documentación del fabricante Cisco, se ha documentado que existe un error en la autenticación para permitir el acceso a los usuarios de la red conocida como 'Authentication Proxy' o 'cut-through'. Cuando dicha autenticación se configura a través de los comandos 'aaa authentication match' o 'aaa authentication include', puede producirse una denegación de servicio si hay un elevado número de peticiones de autenticación. El identificador asignado a esta vulnerabilidad es el CVE-2011-3297, la cual puede ser consultada en la base de datos de Cisco. [12]

La vulnerabilidad en Cisco Firewall Services Module Cut-Through afecta a switches Cisco Catalyst 6500 Series y routers Cisco 7600 Series con la función de proxy cut-through activada.

La vulnerabilidad en el módulo Cut-Through Proxy de Cisco Firewall Services se produce debido a una condición de carrera que se produce al liberar memoria reservada por la función de proxy cut-through. Un atacante podría explotar esta vulnerabilidad mediante el envío de tráfico, para provocar la condición que invoca la autenticación en el proxy cut-through. Esto podría provocar la recarga del sistema afectado, que podría causar denegación de servicio (DoS) en caso de explotarse repetidamente esta vulnerabilidad. El identificador de esta vulnerabilidad es CVE-2014-0710 y CSCuj16824, al igual que la primera vulnerabilidad comentada, esta puede consultarse en la bases de datos que proporciona el fabricante Cisco. [12].

3.- Descripción del Sistema Propuesto.

Actualmente en las redes de computadoras aparece a menudo un dispositivo llamado proxy que actúa como un agente o intermediario para las comunicaciones.

Quizás el uso más común del proxy es como parte de una barrera de protección (firewall), que controla el tráfico que entra y sale de una red. Hoy día las empresas y universidades empiezan a limitar el uso de Internet para su personal por razones de seguridad y eficiencia. El ancho de banda se consume en tráfico improductivo de aplicaciones P2P, ICQ, IRC, etc. Los proxies permiten controlar a los usuarios, el contenido y los sitios, por ejemplo prohibiendo el acceso a determinadas páginas Web o aplicaciones por ser improductivas o por cualquier otro motivo. El cliente interno se comunica con el servidor proxy en lugar de hacerlo directamente con el servidor externo. El proxy es el encargado de evaluar las solicitudes del cliente y decide cuáles deja pasar y cuáles no. Si una petición es aceptada, el proxy se comunica con el servidor real en nombre del cliente (el término proxy significa representante) y lleva cabo las peticiones de servicio del cliente al servidor externo y transmite las respuestas de éste de nuevo al cliente.

Por tal razón al analizar nuestro planteamiento inicial sobre las vulnerabilidades que pueden producirse al no contar con un sistema de control de usuarios a los recursos de red de la universidad, se diseñó una arquitectura basada en el método denominado cut-through proxy que permite verificar si los usuarios tienen permisos para ejecutar una aplicación determinada, ya sea TCP o UDP antes de llegar a la aplicación, es decir, verifica a los usuarios en el mismo firewall, esto permite realizar la verificación más expedita para cada persona que solicita conectarse a un determinado servicio, identificándose ya sea por HTTP, FTP o Telnet.

3.1 Arquitectura del Sistema.

Para la elaboración del piloto que nos permita visualizar claramente el funcionamiento del sistema de autenticación de usuarios contra un sistema LDAP, mediante la utilización de un Firewall Juniper SG 550, para tener acceso a una determinada red, en este caso, tener acceso a internet, se elaboró una topología de red, donde se muestra los principales elementos de la arquitectura. A continuación se detalla gráficamente la topología de red desarrollada con los

elementos del sistema que intervienen: De manera general, se puede establecer como primer punto, una conexión del equipo Juniper a una de las interfaces de red conectada al servidor LDAP, donde se encuentra configurada las aplicaciones OpenLDAP y JXplore. La segunda interfaz conectada al acceso a Internet. A través de configuraciones y reglas, se redirigió el tráfico de la eth 0/0 a la eth0/2, para así establecer que todo el tráfico pasara por una sola vía, es decir, una única puerta de enlace y aplicar la autenticación.

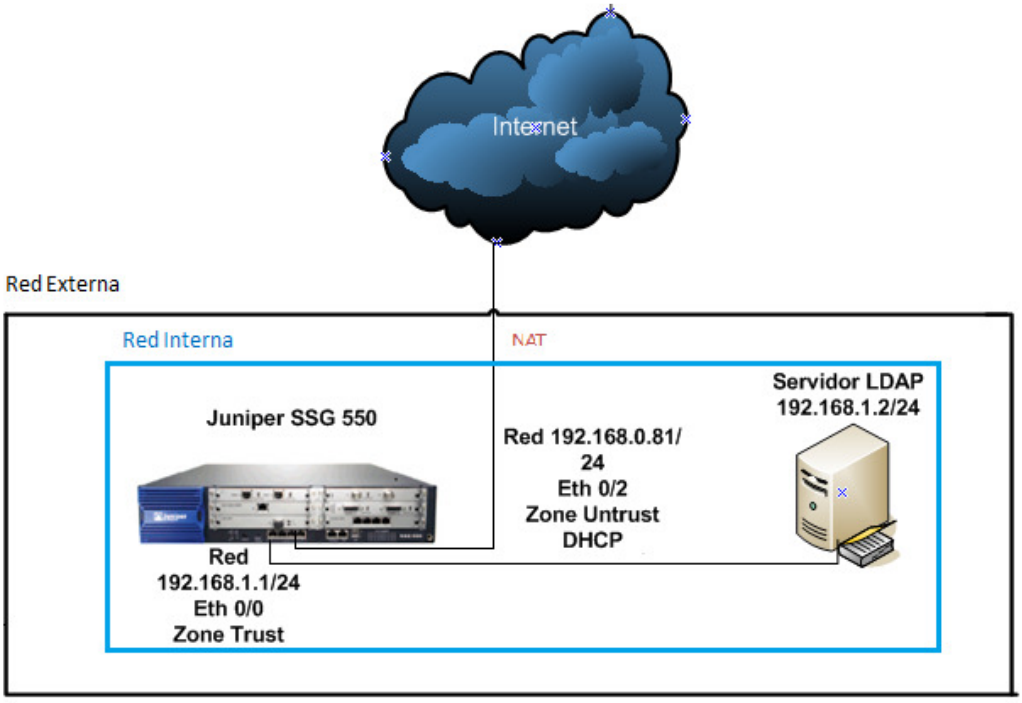


Figura 19.Arquitectura del Sistema.

Fuente: Propia.

Como lo indica la ilustración, se ha divido la topología en dos redes, una red externa, que da acceso al exterior y una red interna, que es la que contiene la interconexión entre el firewall y el servidor LDAP. El firewall posee un módulo con cuatro interfaces de red, una interface será para la red interna y la otra para la externa, la configuración de la red interna, se establece mediante la configuración de la interface de red eth 0/0, configurada en una zona segura denomina Zona Trust, con un direccionamiento IP del segmento 192.168.1.1/24. Para la segunda intefarce de red eth 0/2, configurada en una zona no segura denominada Zona UnTrust, cuyo direccionamiento IP

es proporcionado por el servidor DHCP, cuyo rango es el 192.168.0.0/24, para el tiempo de la puesta en marcha de las configuraciones y pruebas, el servidor proporcionó la dirección IP 192.168.0.81/24, la cual puede detallarse en la figura anterior. La denominación de las zonas se estableció por defecto, pero estas pueden ser cambiadas manualmente si se desea realizar algún cambio de zona y su respectivo segmento de red.

3.2 Definición de Subsistemas.

La definición de la arquitectura del sistema de control de acceso, nos ha dado una visión general del funcionamiento de dicho sistema, es por tal razón, que definiré en subsistemas los componentes que lo conforman, para una mayor compresión de los mismos. El primer susbsitema a definir, es el sistema firewall que permitirá administrar el tráfico entrante y saliente, habilitando políticas o reglas para tal fin. Este subsistema que contiene al firewall, será nuestra puerta de enlace para poder salir a internet. Debemos recordar en este punto, la implementación del método o arquitectura cut-through proxy, es decir, las peticiones que realiza un usuario a un sitio Web son interceptadas por el proxy, y esta solicitará las credenciales, ya sea un usuario y una contraseña, para poder tener acceso a la red o no. Además para que podamos tener acceso a la red externa, previamente nuestro dispositivo de red, realizará una operación de traducción de direcciones IP, conocido como NAT (Network Address Translation). Esta capacidad de traducción de direcciones permite aplicar una técnica llamada IP Masquerading (enmascaramiento de IP), usada muy a menudo para dar acceso a Internet a los equipos de una red de área local compartiendo una única conexión a Internet, y por tanto, una única dirección IP externa. El firewall a través de su servidor DHCP, proporciona un direccionamiento cuyo segmento es el siguiente 192.168.0.0/24. Como se comentó en un principio, para el desarrollo de nuestro piloto, el servidor DHCP, proporcionó la IP 192.168.0.81/24.

En el inicio del desarrollo del piloto se valoraron varias alternativas referentes a equipos de seguridad, desde varios puntos de vista. Dentro de dichas alternativas para tal desarrollo se valoró un PIX de Cisco, un firewall por software y por último el Juniper. Se descartó el PIX de Cisco, por ser un equipo obsoleto en cuanto a las prestaciones que se requerían, en relación al firewall por software, es que esta, no ofrece el mismo nivel de protección que un hardware dedicado para

tal fin. Además un firewall por software debe instalarse en una maquina con ciertas prestaciones de rendimiento muy altos, ya que un firewall maneja gran cantidad de tráfico, saliente o entrante sea el caso, y esto puedo causar latencia en la red, sobrecarga del CPU, retransmisión de paquetes, etc. Después de la valoración de las alternativas, se optó por un equipo hardware relativamente más nuevo que el PIX de Cisco, y más eficiente que el firewall por software. Esta opción ofrece altas prestaciones, rendimiento, CPU, throughput, routing, seguridad, virtualización de zonas, QoS, etc. A continuación se estudiará en detalle las características del firewall de Juniper SG 550 como opción escogida para nuestro piloto:

3.2.1 Firewall Juniper SG 550.

El equipo Juniper Networks Secure Services Gateway 500 Series (SSG) es un dispositivo de seguridad especialmente diseñada que ofrece una perfecta combinación de rendimiento, seguridad y conectividad LAN/WAN para implementaciones que se adaptan para grandes y medianas empresas, universidades, etc. En cuanto al resguardo de la seguridad, incluye cortafuegos, IPS, antivirus (incluye Anti-Spyware, Anti-Adware, Anti-Phishing), Anti-spam, filtrado Web, para detener gusanos, spyware, troyanos, malware y otros ataques emergentes.

A nivel de comunicación la serie SSG 500 soporta cuatro interfaces 10/100/1000 más seis ranuras de interfaz que aceptan una amplia gama de redes LAN y WAN I/O, tarjetas incluyendo T1/E1, DS3, E3, Serial, 10/100/1000, SFP y FE. Además es importante mencionar que el Juniper SSG 550 trabaja en la modalidad tanto en la capa 2 como en la capa 3, la virtualización de zonas y routers, la gestión del tráfico (QoS), VPN, Routing, entre otros.

La serie SSG 500 ofrece un avanzada segmentación de las redes, como las zonas de seguridad, enrutadores virtuales y redes VLAN que permiten a los administradores de redes implementar diferentes niveles de seguridad para un usuario, diferentes grupos, dividir la red en dominios seguros, etc., cada uno de estos con sus propias políticas de seguridad.

También es importante a tener en consideración los diferentes tipos o mecanismos de autenticación de usuarios que soporta este dispositivo de red, entre los que se encuentra RADIUS, RSA SecurID, 802.1X y LDAP, siendo este último utilizado para desarrollar nuestro sistema de autenticación a la red cableada.

El segundo subsistema que definimos en nuestra arquitectura, es el servidor de autenticación. Como ya hemos comentado anteriormente, una de las utilidades más importantes de un servidor LDAP es como servidor de autentificación. Autentificarse es necesario para entrar en un sistema en particular, acceder a algunos servicios como un servidor FTP o a páginas privadas en un servidor web. Para la puesta en marcha de nuestra demo de un sistema de control de acceso a infraestructura de redes cableadas, se evaluaron varias herramientas, tanto open source como bajo licencia, entre ellas podemos nombrar al Active Directory de Microsoft. Para este estudio en particular se consideraron herramientas open source, ya que de forma general, existe gran documentación en la red para su correcta configuración. Entre las herramientas libres, se valoraron Apache Directory Server, este un servidor de directorio escrito completamente en Java y disponible bajo la licencia de Apache Software, es compatible con LDAPv3 certificado por el Open Group, soporta otros protocolos de red tal como Kerberos y NTP (Network Time Protocol), además provee Procedimientos Almacenados, triggers y vistas; características que están presente en las Base de Datos Relacionales pero que no estaban presentes en el mundo LDAP.

Otra herramienta que ofrece la configuración de un servidor LDAP, es Zeroshell. Esta una distribución Linux para servidores y dispositivos embebidos, que provee servicios de red. Dispone de un interfaz Web para su configuración, pero también puede ser administrado desde un terminal remoto (ssh) Está basado en Debian. Se pueden descargar los diferentes paquetes que lo forman, para adaptarlo a nuestro hardware. Entre los servicios que ofrece podemos encontrar, servidor DHCP, RADIUS, configuración de un firewall, creación de VLAN, configuración de Portal Cautivo, configuración de un servidor LDAP, entre otros. Por último, estudiamos la opción de OpenLDAP, también una herramienta open source, que puede ser instalado en varios sistemas operativos. Existe mucha información en internet como en libros para su instalación y configuración. Además puede ser descargado de manera gratuita, al sistema operativo de preferencia, en este caso en particular para Windows. A continuación detallamos las características principales de dicha herramienta:

3.2.2 OpenLDAP.

OpenLDAP es una implementación libre y de código abierto del protocolo Lightweight Directory Access Protocol (LDAP) desarrollada por el proyecto OpenLDAP. Varias distribuciones de

GNU/Linux, incluyen el software OpenLDAP, además corre en diferentes sistemas operativos, como BSD, AIX, HP-UX, Mac OS X, Solaris, Microsoft Windows, etc. La estructura en que se basa OpenLDAP, se compone de tres elementos básicos, entre ellos, tenemos: el archivo slapd, que es el archivo principal de configuración del servicio LDAP, bibliotecas que implementan el protocolo y los diferentes programas clientes, como ldapsearch, ldapadd, ldapdelete

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

En el esquema que a continuación se detalla, expresa el proceso de validación de un usuario y la arquitectura cliente-servidor, ya que es la que este sistema se basa.

Figura 20.Arquitectura Cliente-Servidor.

Fuente: Propia.

3.3 Modelo de Datos del Sistema.

Las entradas de un directorio LDAP están estructuradas en forma de árbol jerárquico. Como sucede en muchas estructuras jerárquicas, tanto mayor sea la profundidad del árbol, mayor será la precisión del contenido almacenado en ella. A la estructura de árbol jerárquico de LDAP se le conoce de manera formal como directory information tree (DIT) o árbol de información del directorio. A la parte superior de esta estructura jerárquica se le conoce como el elemento raíz. La

ruta completa hacia cualquier nodo en la estructura del árbol, misma que define a este nodo en forma única, se le conoce como distinguished name (DN) o nombre diferenciado del nodo u objeto.

Es usual que la estructura de un directorio LDAP refleje límites geográficos u organizacionales. Los límites geográficos pueden ocurrir en las líneas divisorias de un país, de un estado, de una ciudad, un municipio, etc. Los límites organizacionales pueden, por ejemplo, referirse a líneas divisorias entre funciones, departamentos o unidades organizacionales, para nuestro caso en particular de puesta en marcha de nuestro piloto, se adapta a ambos tipos de jerarquía. En una universidad por ejemplo puede conseguirse, departamentos, secciones, edificios departamentales, laboratorios, bibliotecas, salas de lecturas, institutos, diferentes campus ubicados en áreas geográficas diferentes, etc.

Esta referencia nos da pie como introducción para dar a conocer la estructura de datos que compone nuestro servidor LDAP. Como se mencionó en un principio, la disposición de los datos se realiza o se establece de manera jerárquica, esto nos permite entre varias cosas, como está compuesto nuestro directorio, una mejor compresión de la disposición de estos, realizar una búsqueda ordena, agruparlos, etc., todo esto de manera eficiente y clara.

Para el proyecto se creó una pequeña estructura de datos, con el fin de ejemplificar la disposición de los elementos en una base de datos LDAP. En la siguiente ilustración podemos observar su estructura. La parte más alta del directorio, representa la raíz del árbol del directorio, también es conocida como la base. El nombre de esa base es el Nombre Distinguido de la Base, o base DN, en nuestro caso en particular es *dc=maxcrc, dc=com.* Pasando al siguiente nivel, nos encontramos con la Unidad Organizacional (por sus siglas en inglés Organizational Unit), donde solo creamos una, llamada *People.* A modo de ejemplo en las empresas, las Unidades Organizativas pueden ser Departamentos, Secciones, Clientes, Empleados, Dispositivos, Localizaciones, etc. Para el último nivel en nuestras bases de datos, finalizamos con el *cn* (common name), que es un identificador único, en nuestro caso, son *Admin* y *Pepe.*

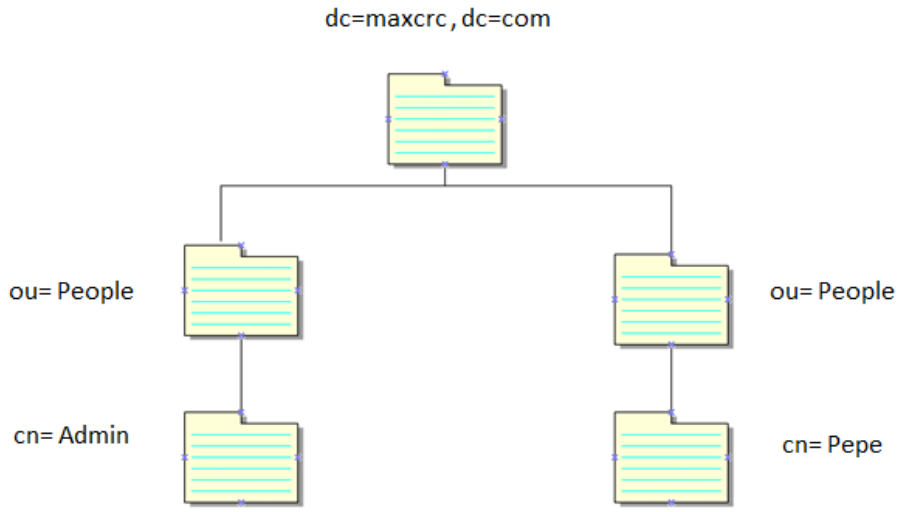


Figura 21. Árbol de directorio LDAP.

Fuente: Propia

La estructura de datos que a continuación detallamos también cuenta con una gran cantidad de elementos, como lo son los atributos y sus respectivos valores, como son una gran cantidad, solo se cumplimentaron los más importantes para la demo. Realizamos una tabla para su mejor comprensión. Los atributos que se encuentran coloreadas en negritas, son atributos que deben estar de manera obligatoria, ya que si no son cargados en la base de datos, no se creará la plantilla a crear.

	Atributo	*Valor*
Admin	cn	Admin
	objectClass	InetOrgPerson
	objectClass	organizationalPerson
	objectClass	top
	sn	gmazzei
	description	
	mail	
	postalAddress	
	postalCode	
	telephoneNumber	
	title	
	audio	
	businessCategory	
	carLicense	
	departamentNumber	

	Atributo	*Valor*
Pepe	cn	Pepe
	objectClass	InetOrgPerson
	objectClass	organizationalPerson
	objectClass	top
	sn	Pepe
	description	
	mail	
	postalAddress	
	postalCode	
	telephoneNumber	
	title	
	audio	
	businessCategory	
	carLicense	
	departamentNumber	

Tabla 1. Atributos y valores de estructura de datos de LDAP.

Fuente: Propia.

3.4 Políticas de Seguridad.

Cada ordenador que se conecta a internet puede ser víctima de alguna vulnerabilidad. La metodología que generalmente usan los intrusos consiste en analizar la red mediante el envío aleatorio de paquetes de datos en busca de un ordenador conectado. Una vez que encuentra un ordenador, el intruso busca un punto débil en el sistema de seguridad para explotarlo y tener acceso a los datos de la máquina. Estas amenazas pueden ser contenidas, pero no en su totalidad, ya que un sistema no va hacer ciento por ciento segura, a menos que se desconecte de internet y este apagado. Por tal razón, vemos la necesidad de introducir un firewall, como se comentó en un principio de la descripción, se tomó la elección para esta demo la incorporación del modelo Juniper SG-550, cuyo principal objetivo o función es la de filtrar los paquetes tanto entrante como salientes, autorizar conexiones, bloquear paquetes, rechazar el pedido de conexión sin informar al que lo envió, es decir, negarlos, etc. Pero además de la puesta en funcionamiento del firewall, debemos de crear ciertas reglas o políticas más específicas para filtrar el tráfico, este tipo de reglas, se les conoces como Políticas de Seguridad. Más adelante podrá detallarse su configuración e implementación en el equipo con sus respectivas capturas de pantalla. En el

análisis y ejecución del proyecto de red, se configuraron 2 reglas muy específicas, cuyo objetivo es solo filtrar el protocolo HTTP de todo del tráfico que permitamos. Las reglas configuras son las siguientes:

- ✓ Permitir desde la red origen 192.168.1.0/24 hacia cualquier red destino, todo tipo de servicios.
- ✓ Permitir desde la red origen 192.168.1.0/24 hacia cualquier red destino, solo los servicios HTTP, HTTPS y HTTP-EXT (Extension Framework).

3.5 Configuración OpenLDAP.

Para la configuración del servidor LDAP, se escogió la herramienta de software libre OpenLDAP, para Windows, una vez que descargamos el archivo *openldapforWindows.exe*, se ejecuta y se inicia la configuración por interfaz gráfica, como se observa en la siguiente captura:

Figura 22. Instalación de OpenLDAP.

Fuente: Propia

La siguiente captura de pantalla, se muestra una ventana de bienvenida para el asistente de la aplicación:

Figura 23. Instalación de OpenLDAP por el Wizard.

Fuente: Propia

La siguiente pantalla es el Acuerdo de Licencia de Usuario Final de OpenLDAP para Windows, que permite utilizar el software de forma gratuita bajo la Licencia Pública Común (CPL).

Figura 24. Instalación de OpenLDAP y licencia.

Fuente: Propia

En el proceso de la instalación, la aplicación detecta unos requisitos previos, la cual las expone en pantalla, para su instalación:

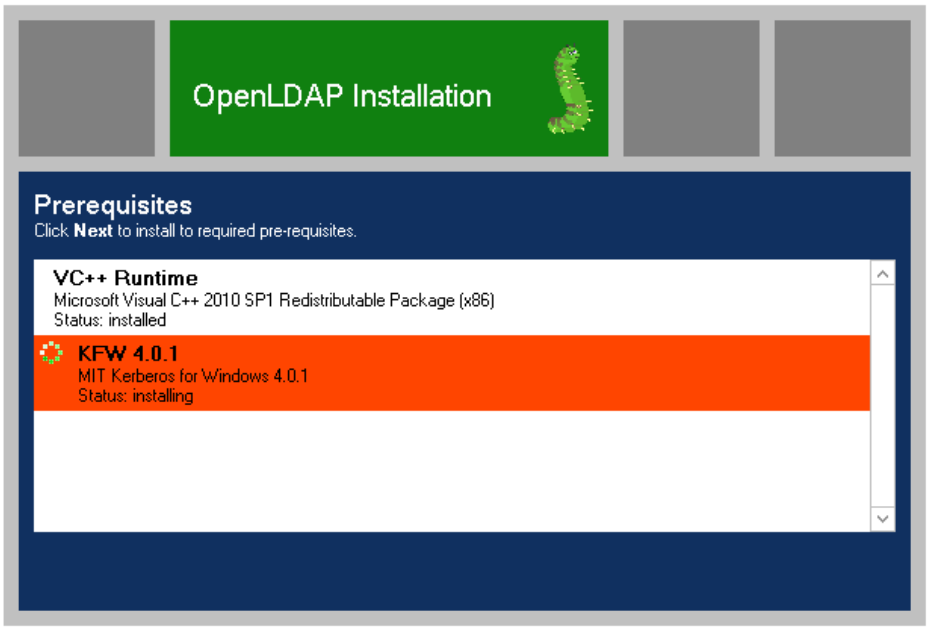


Figura 25. Instalación de OpenLDAP y requisitos propios.

Fuente: Propia

En la siguiente imagen se selecciona la carpeta de destino, donde se va a instalar la aplicación:

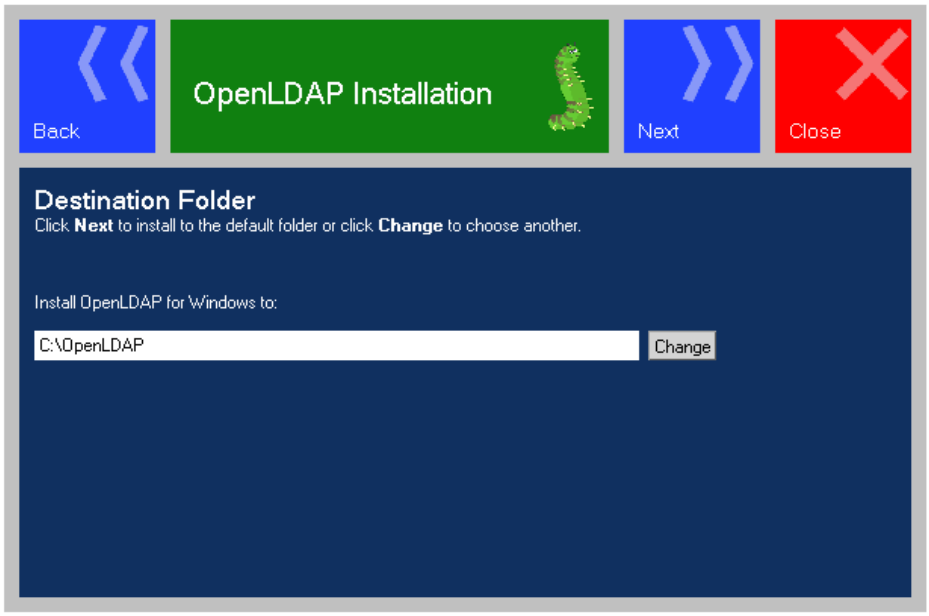


Figura 26. Directorio de instalación OpenLDAP.

Fuente: Propia

A continuación, el asistente permite realizar una configuración personalizada, que permite excluir ciertas características de OpenLDAP para Windows, si no se requieren, en este caso, se dejaron por defecto:

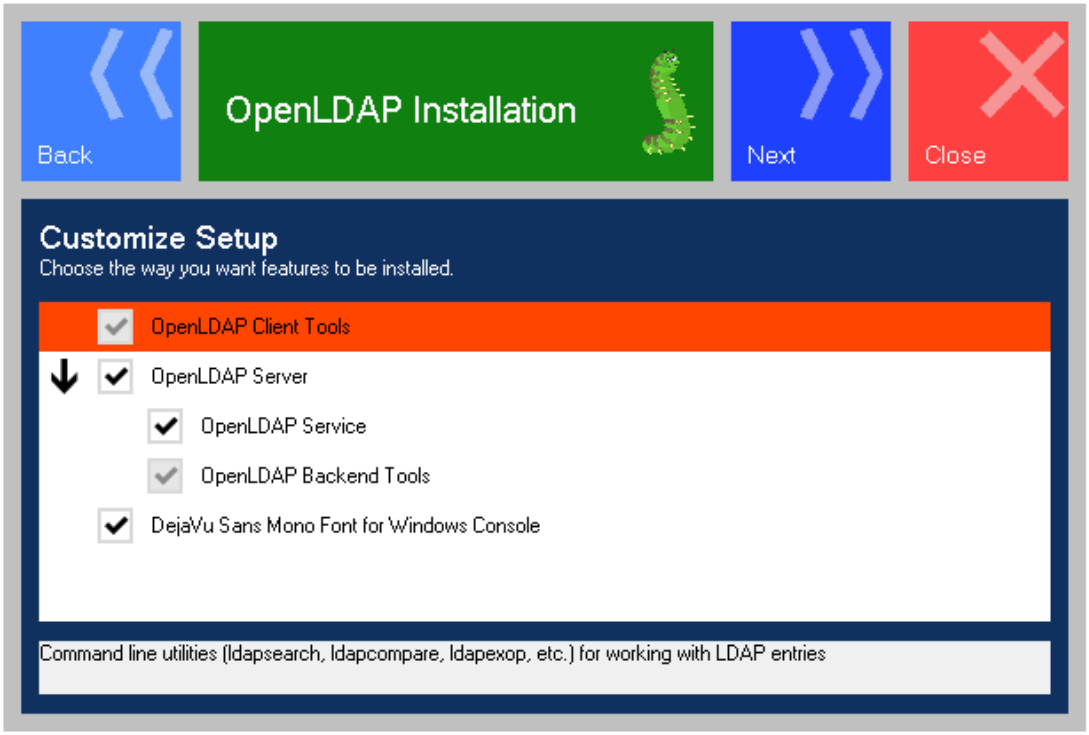


Figura 27. Instalación personalizada de OpenLDAP.

Fuente: Propia

La pantalla Additional Settings o Configuración adicional puede ser útil si necesita cambiar los ajustes por defecto: por ejemplo el server name o dirección IP y los puertos abiertos y SSL del servidor OpenLDAP. También se puede activar o desactivar la opción para activar las estadísticas de la instancia de OpenLDAP. También se permite administrar las propiedades del servidor de forma remota sin necesidad de reiniciar el servicio marcando la opción dinámica backend configuración. En este apartado, como se describió, colocamos la dirección IP que será nuestro servidor, para este caso en la 192.168.1.2 /24, además se especifica claramente los puerto utilizados para la comunicación, en este caso el 389 y para el puerto SSL 636.

Figura 28.Configuracion de la IP.

Fuente: Propia

Se validó, la apertura del puerto TCP 389, que es el puerto para LDAP, la cual se encontraba escuchando, esto se realizó ejecutando el comando *netstat –a*, la cual muestra todas las conexiones activas y los puertos TCP y UDP en el que el equipo está a la escucha.

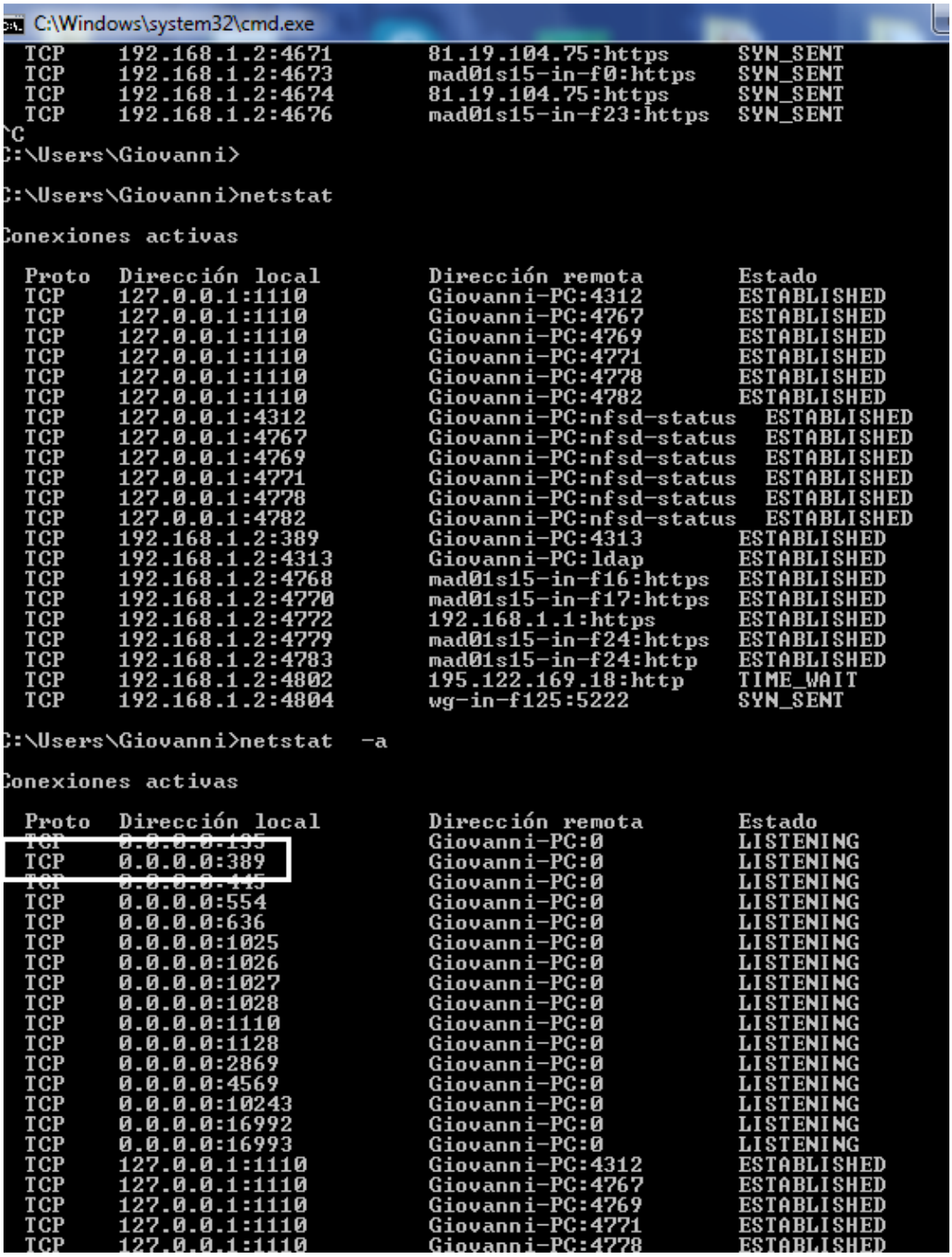


Figura 29. Ejecución del comando netstat -a.

Fuente: Propia.

Para la siguiente imagen, solo se indica que se va a iniciar el proceso de instalación o que podemos volver hacia atrás y realizar cualquier cambio en la configuración de las pantallas anteriores.

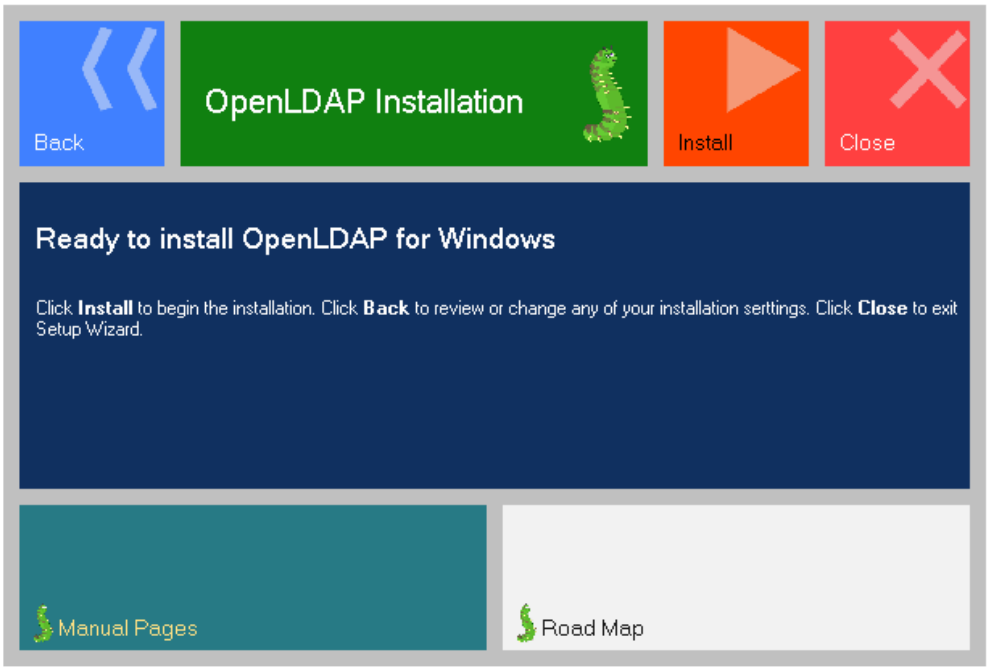


Figura 30. Proceso de instalación OpenLDAP.

Fuente: Propia

El resultado de la instalación se muestra en la ventana final del asistente de instalación. Con este último paso, se finaliza la instalación del servidor LDAP, para nuestro piloto. Para su administración introducimos la siguiente información:

User: cn=Manager,dc=maxcrc,dc=com

Password: secret

Figura 31. Instalación completada de OpenLDAP.

Fuente: Propia

Si nos dirigimos a la ruta de instalación podemos observar la serie de archivos creados, después de su instalación. Dentro de todos estos archivos, es de hacer mención uno importante, que es el slapd. El archivo *slapd.conf* es el archivo principal de OpenLDAP, y es donde se realiza la configuración de todos los parámetros. Slapd permite recibir las peticiones LDAP, servirlas, gestionarlas, además de implementar las funciones básicas de LDAP. Por ejemplo en lo que refiere a control de acceso slapd provee facilidades muy potentes, permitiendo controlar el acceso a la información de sus bases de datos. Puede controlar el acceso a las entradas basándose en la información de autorización de LDAP, en la dirección IP, en los nombres de dominio y otros criterios. Además slapd soporta tanto el control de acceso a la información dinámico como estático.

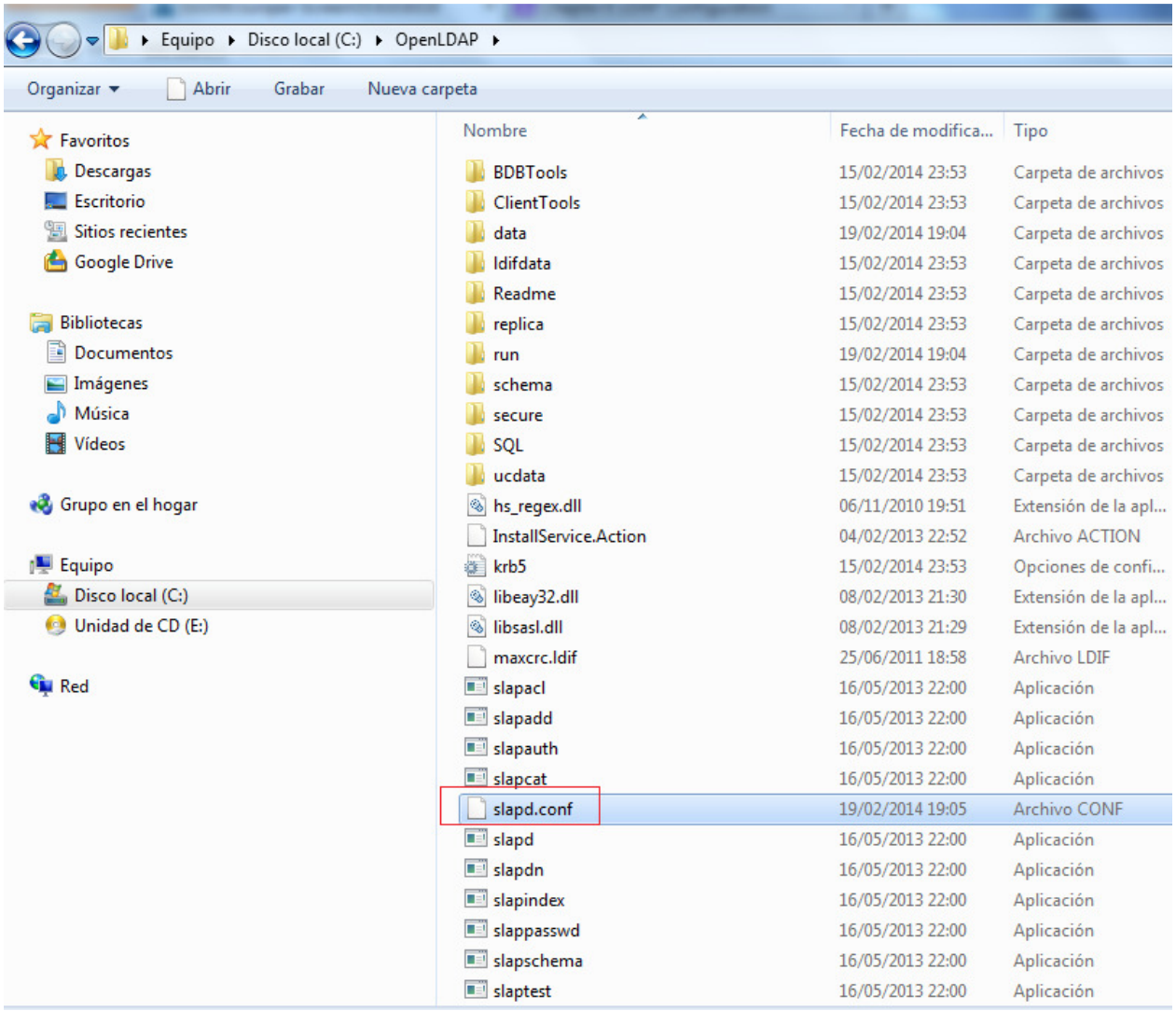


Figura 32. Directorio de instalación OpenLDAP.

Fuente: Propia

En la configuración inicial del archivo slapd.conf, existe unos parámetro que son importante resaltar, que a continuación puede verse en la siguiente ilustración. El parámetro database, simplemente nos indica el tipo de base de datos que hayamos escogido para el almacenamiento de los datos, en este caso, Berkeley DataBase (bdb), la cual es una librería de manejo de base de datos con API para varios lenguajes, entre ellos, C, C++, Java, Perl, Python. La entrada suffix nombra el dominio para el cual el servidor LDAP proveerá información. El parámetro o entrada *rootdn* es el Nombre distinguido (DN) para un usuario que no está restringido por el control de acceso o los parámetros de límites administrativos fijados para operaciones en el directorio LDAP, es decir, es el usuario administrador con privilegios para la gestión del directorio LDAP.

```
database        bdb
suffix          "dc=maxcrc,dc=com"
rootdn          "cn=Manager,dc=maxcrc,dc=com"
```

Figura 33. Configuración parcial del archivo. *slapd.conf.*

Fuente: Propia

3.5.1 Configuración de JXplore.

Al iniciar el presente trabajo, se realizó una recopilación de información de las herramientas de software que fueron utilizados, para llevar a cabo el presente piloto de control de acceso, entre las herramientas que fueron utilizados tenemos a la aplicación, JXplorer, que no es más que una aplicación Java de código abierto que permite navegar y buscar cualquier directorio LDAP. Esta herramienta, como lo veremos a continuación nos permite, conectarnos a un directorio LDAP, y administrarlo, ya sea agregar, modificar, eliminar usuarios, al igual que podemos crear grupos de usuarios para que tengan acceso, por ejemplo en nuestro caso, a una red determinada, es decir, es una herramienta que nos permite administrar LDAP. A continuación se presentará unas capturas de pantallas, para mostrar su utilización en esta demo.

Como lo indicamos anteriormente, para iniciar sesión en la herramienta para entrar a administrar LDAP, entramos con las siguientes credenciales:

User: cn=Manager,dc=maxcrc,dc=com

Password: secret

IP 192.168.1.2.

Protocolol LDAPv2

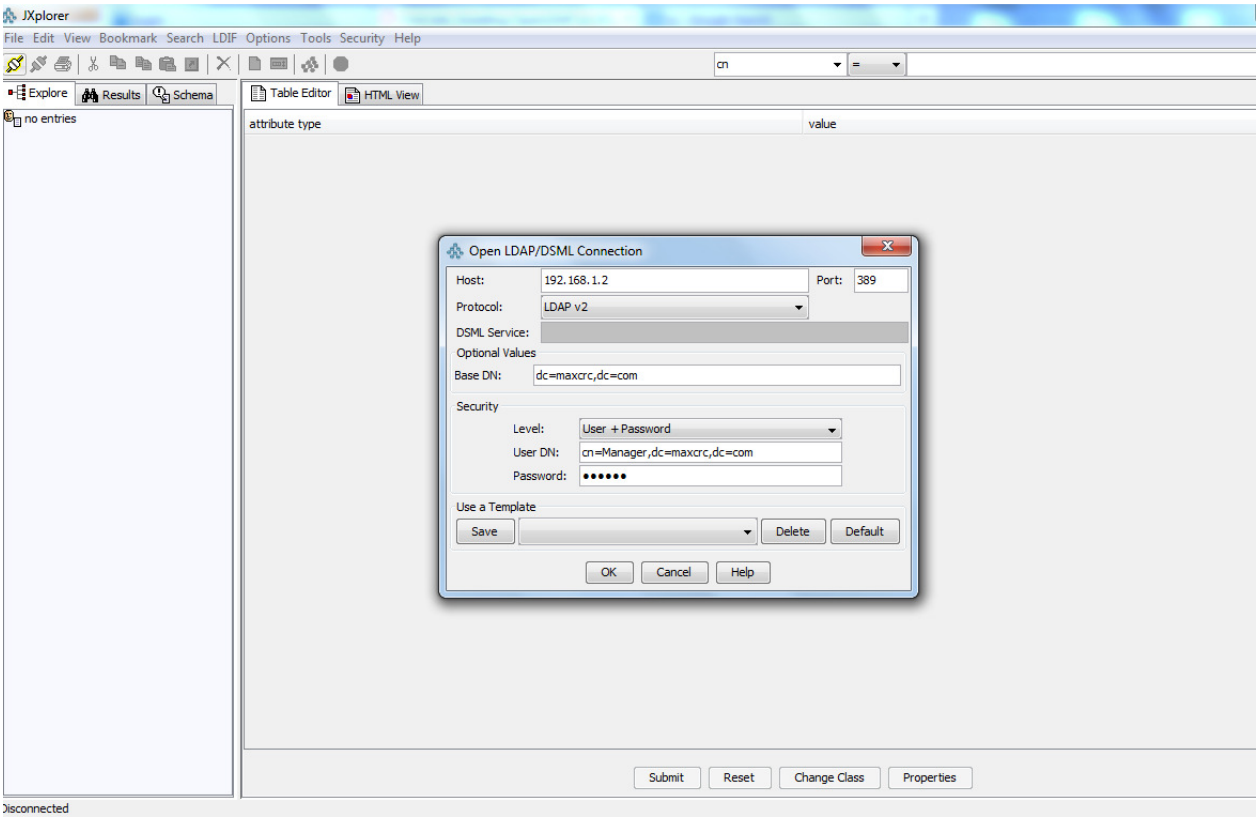


Figura 34. Vista principal de conexión de la herramienta Jxplore.

Fuente: Propia

En la siguiente captura podemos ver la creación de un nuevo usuario, en este caso colocamos en el Common Name *admin,* y en el surname *gmazzei,* para darle una segunda descripción y un password. En esta primera pestaña principal (Main) del HTML, permite agregar una gran cantidad de información de un usuario, por ejemplo, colocar la foto de un usuario, su número telefónico, local, de casa o el de móvil, su correo electrónico, las iniciales, una pequeña descripción, si posee URL, etc.

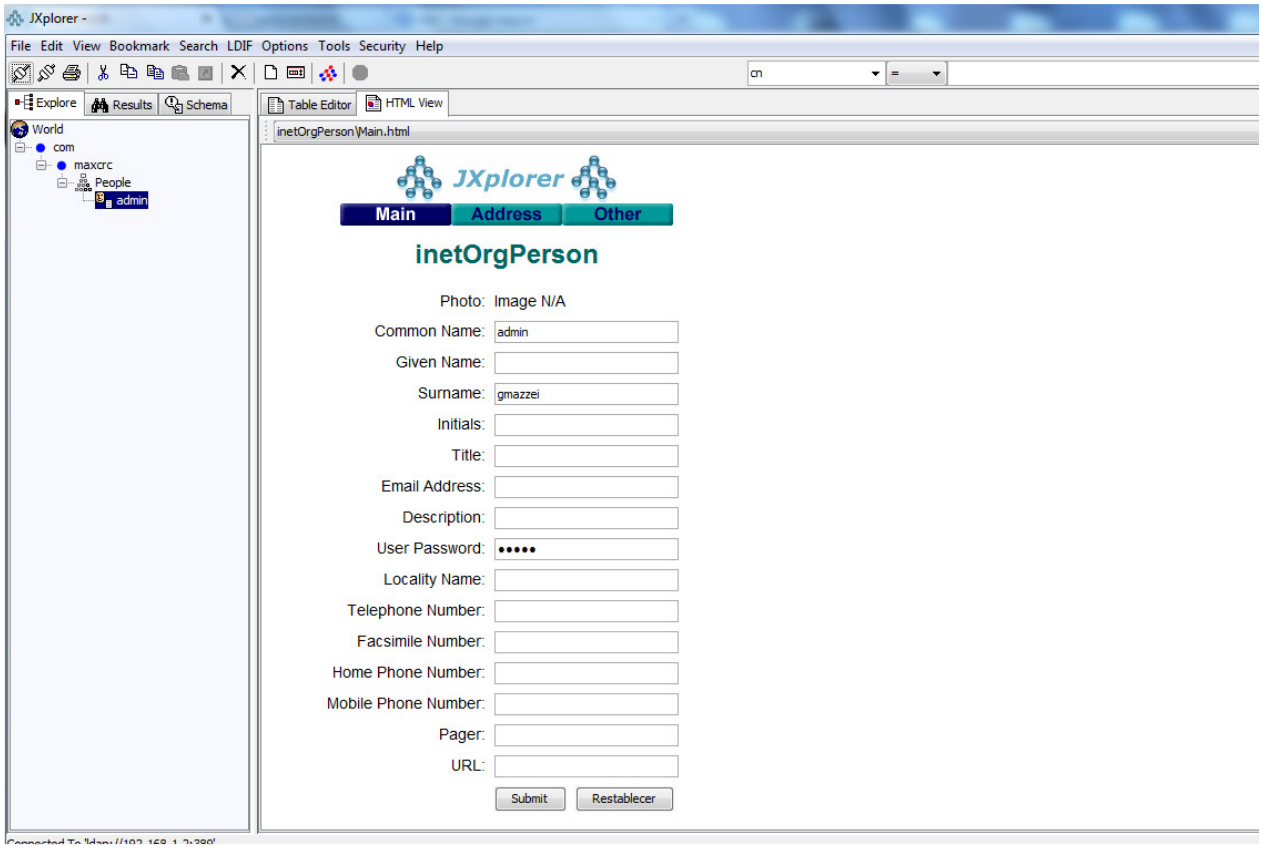


Figura 35. Vista de información del usuario *Admin*.

Fuente: Propia

Se creó un segundo usuario, para probar la autenticación con el LDAP, a través del Juniper. Este usuario es *Pepe*, y su Common Name es *cn=Pepe,ou=People,dc=maxcrc,dc=com* y la contraseña *pepe* igualmente.

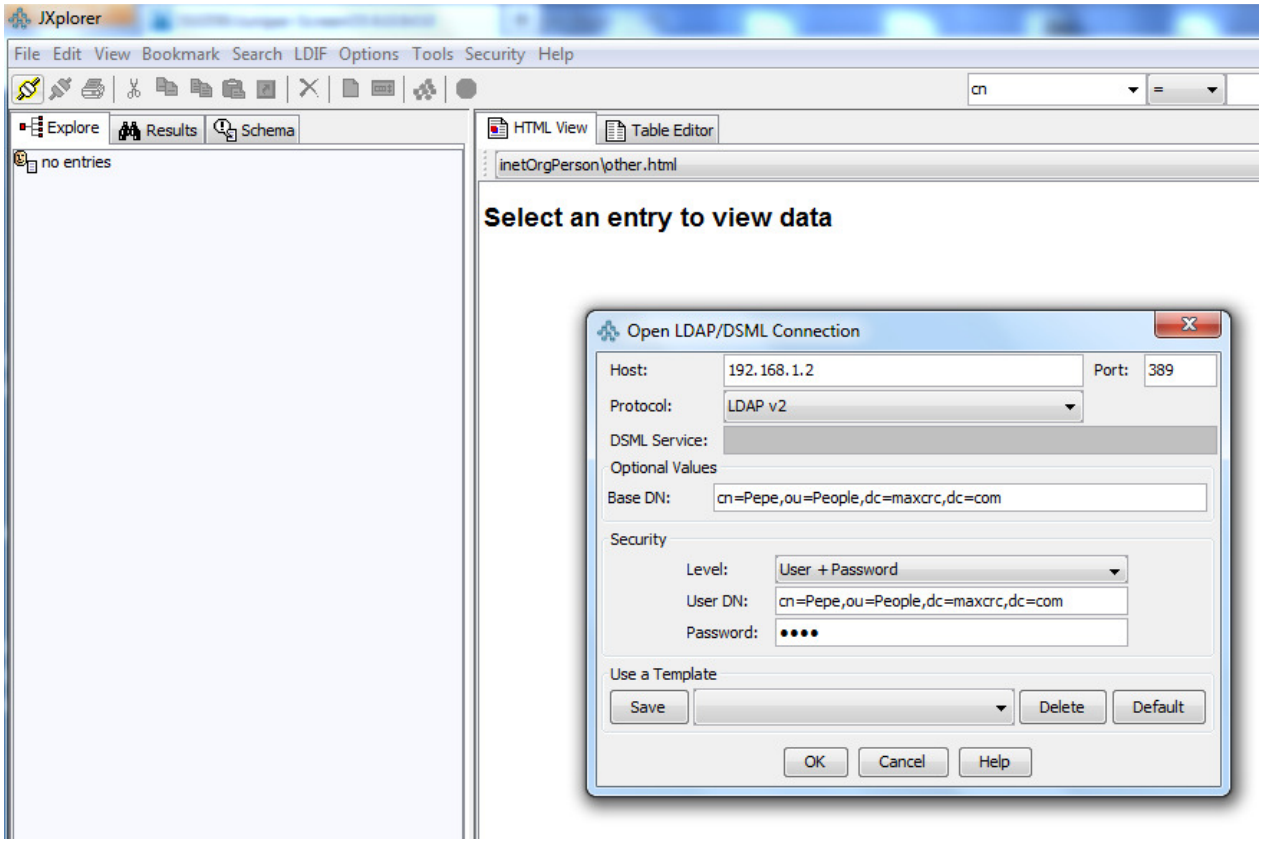


Figura 36. Conexión a LDAP a través del usuario *pepe*.

Fuente: Propia

En la siguiente imagen podemos observar otro atributo para recabar información, como lo es la pestaña (Address), como su propio nombre lo indica, una dirección postal, ya sea de casa u oficina, información de la provincia, etc.

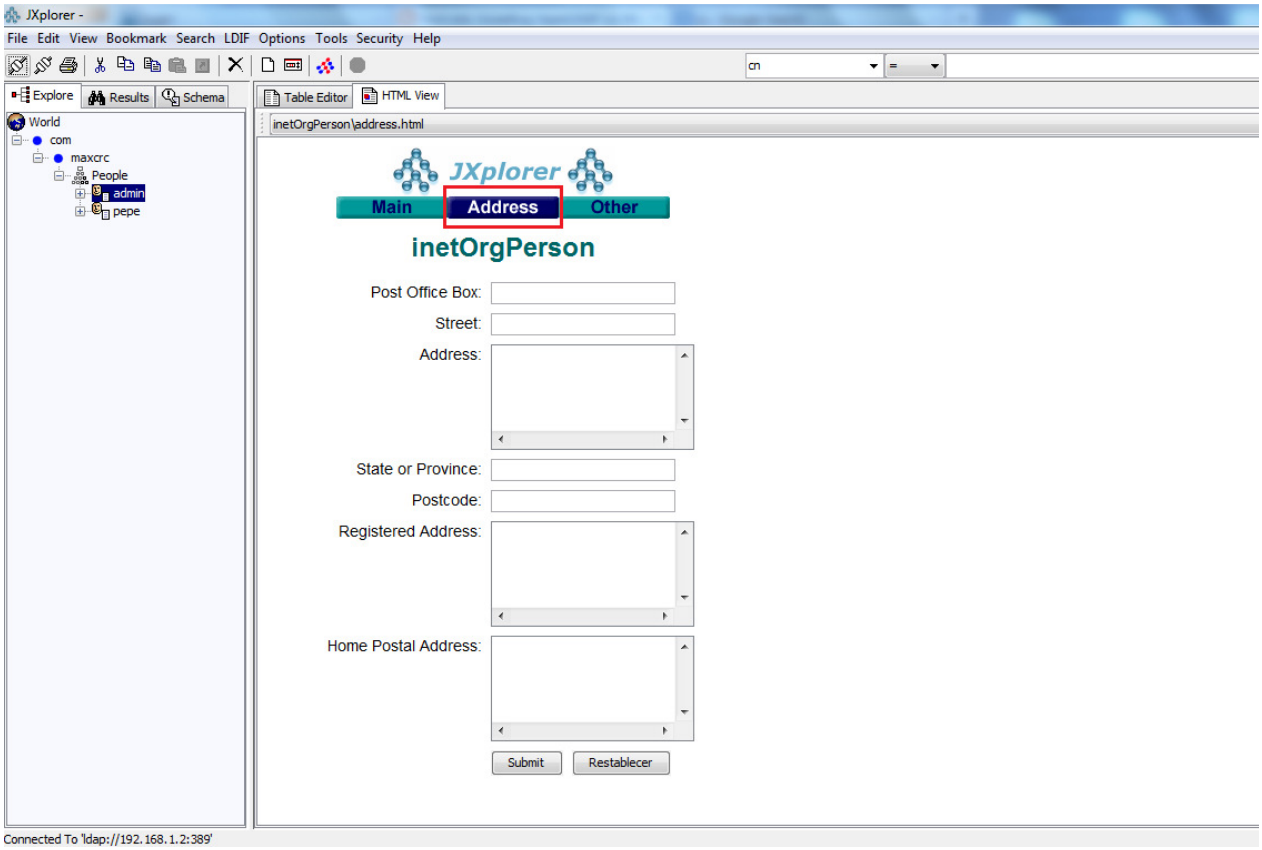


Figura 37. Vista de atributos del usuario *Admin* (Dirección).

Fuente: Propia

En la última pestaña (Others) podemos ver un poco de información variada, por ejemplo, la grabación de un audio, el tipo de empleado, el número de empleado, el departamento, el número de sala, el tipo de negocio, etc., es decir, información de perfil de una persona empleada en una empresa.

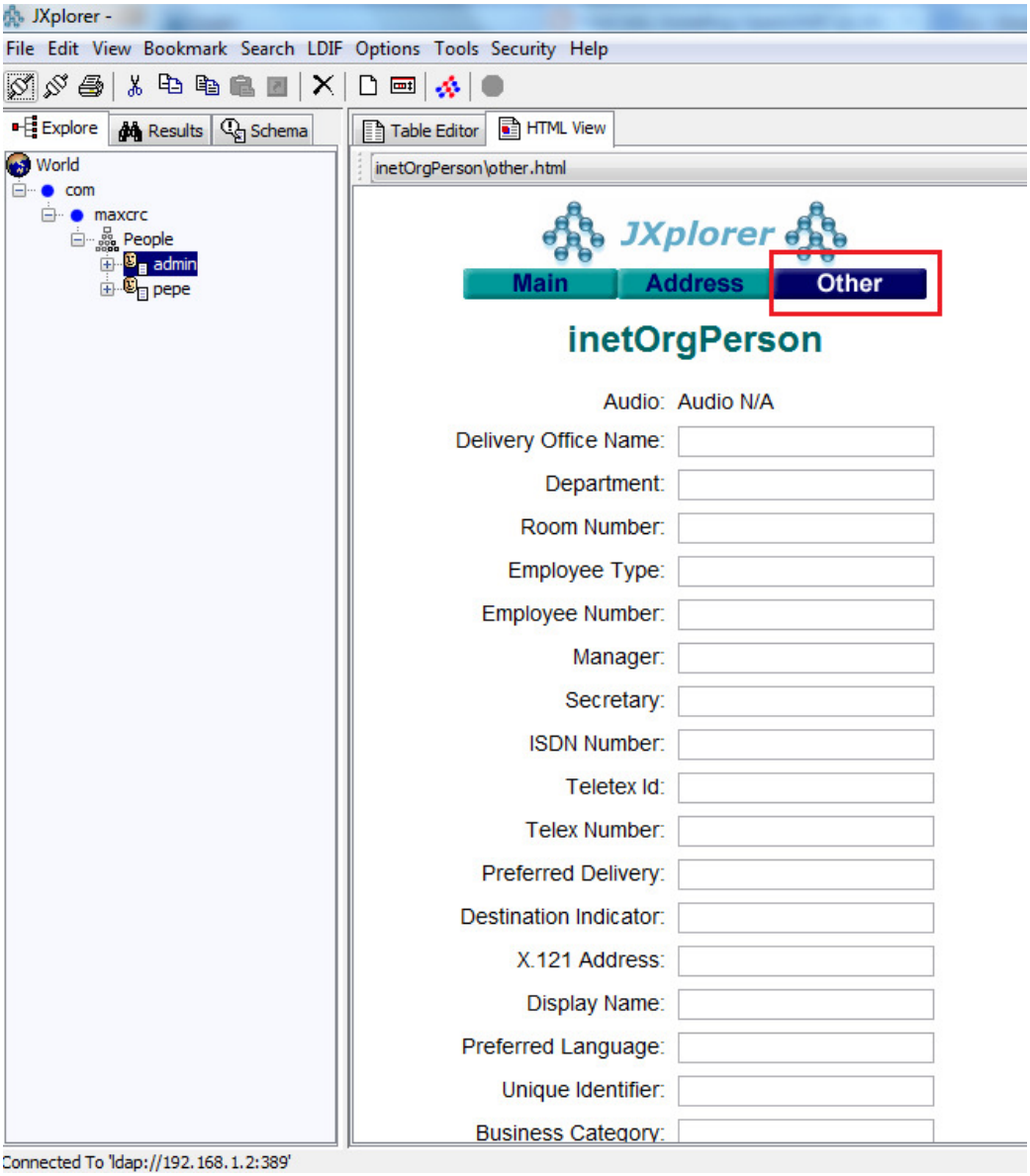


Figura 38. Vista de atributos del usuario *Admin* (Otros).

Fuente: Propia

3.6 Configuración de Firewall Juniper SG 550.

Como se mostró en el capítulo anterior, se realizó un esbozo de la topología de red, en base la cual se realizó las configuraciones del firewall.

En primer lugar se configuró la tarjeta de red de mi equipo, con la dirección IP192.168.1.2/24, la cual va ser la dirección IP, del servidor LDAP, para realizar la consulta de los usuarios que puedan tener acceso a la red y tener acceso a la GUI del Juniper, para realizar las configuraciones pertinentes.

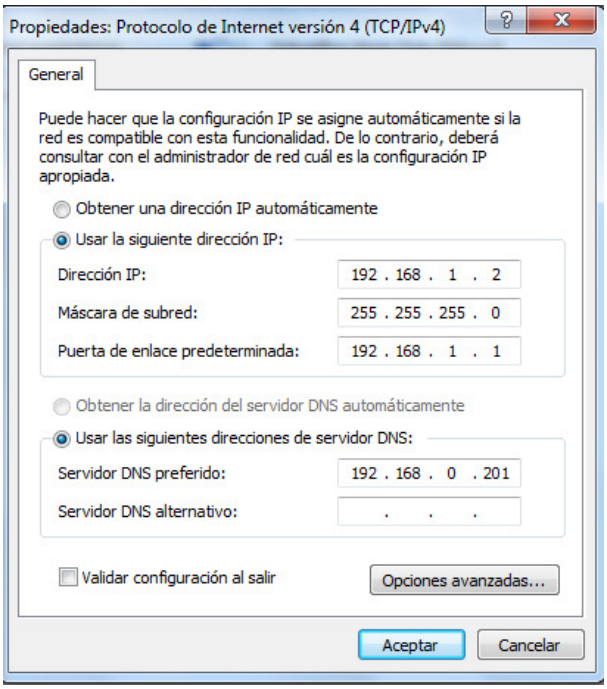


Figura 39. Definición de IP.

Fuente: Propia.

Como primer paso se realizó un *Reset* del equipo, para tener el dispositivo sin ninguna configuración previa, con la configuración por defecto de fábrica. Este procedimiento se realizó presionando por un espacio de 5 segundos el *Reset Config Button*. Su ubicación puede observarse en el Anexo C, donde se muestra el panel frontal del equipo. Una vez realizado el *Reset*, procedemos a iniciar sesión a través de la interfaz gráfica, llamada WebUI, para ello debemos configurar en nuestra tarjeta de red una dirección ip que este en la misma subred, por ejemplo, la IP 192.168.1.1/24. Además debemos conectarnos al puerto 0/0 (ethernet0 / 0 interfaz), que viene ya enlazado a la zona de seguridad Trust.

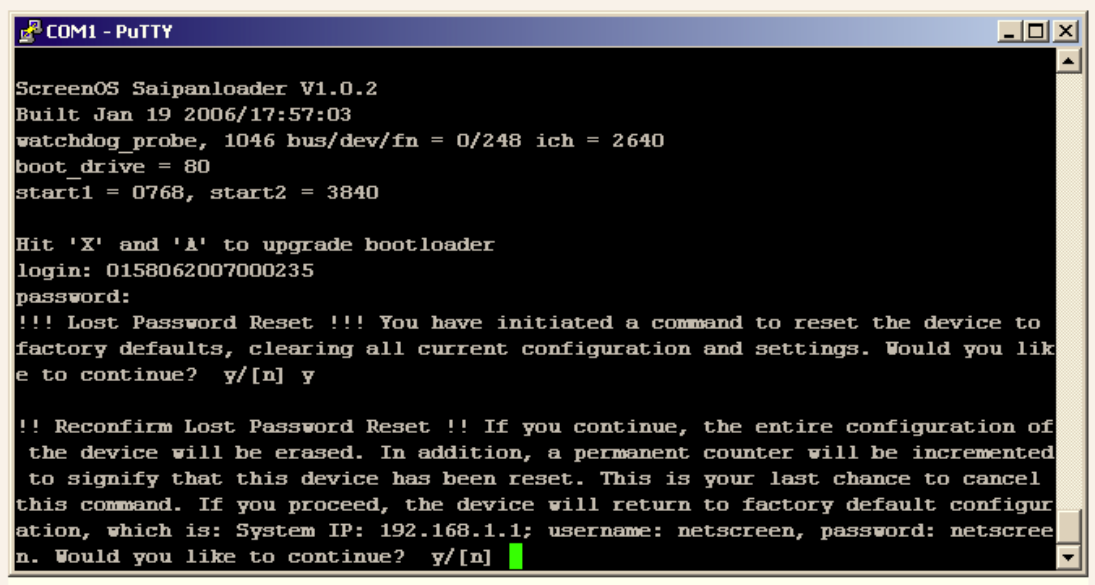


Figura 40. Reinicio de equipo Juniper.

Fuente: Propia.

Al iniciar un navegador colocando la dirección IP 192.168.1.1/24, se puede tener acceso a la GUI del equipo y empezar a configurarlo, como se mencionó anteriormente. Puede denotarse una configuración inicial.

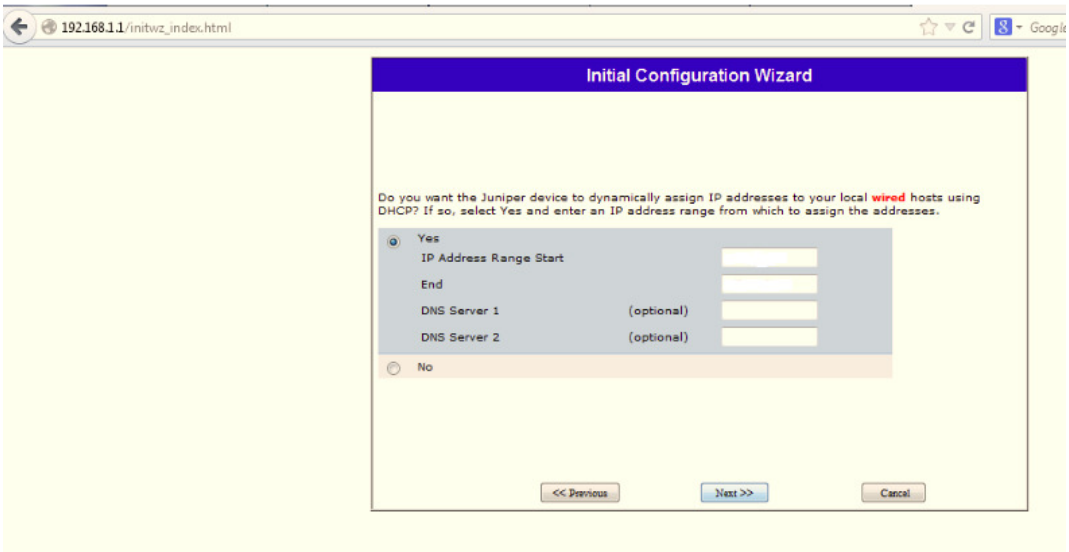


Figura 41. Configuración inicial por Wizard.

Fuente: Propia.

Una vez configurado el Wizard, se procede a entrar a la GUI propiamente dicha, iniciamos sesión con el usuario y contraseña que viene por defecto que este caso es ***netscreen***. Una vez dentro del panel de configuración la cambiamos por una de nuestra preferencia, tanto para el usuario y la contraseña se utilizó *admin*.

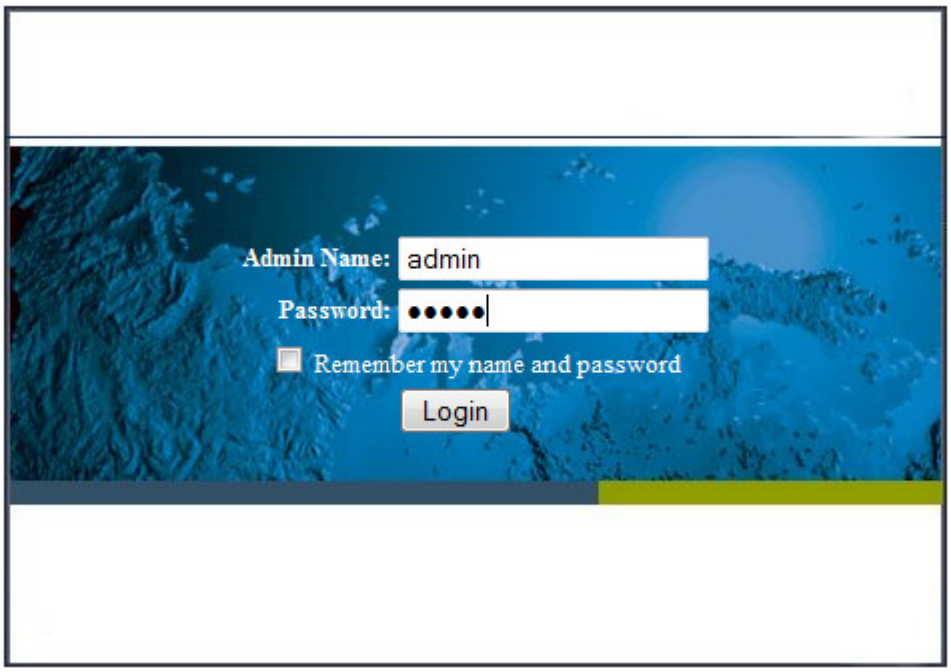


Figura 42. Inicio de sesión por GUI.

Fuente: Propia.

Una vez dentro nos encontramos con el siguiente panel, con información por ejemplo del consumo de los recursos del equipo, por ejemplo de la CPU, memoria, una vista de la información del equipo, como la versión del hardware, el firmware, el número de serial y el nombre del host. También nos permite observar una vista de las interfaces VPN y una muy importante como es el detalle de alarmas y eventos.

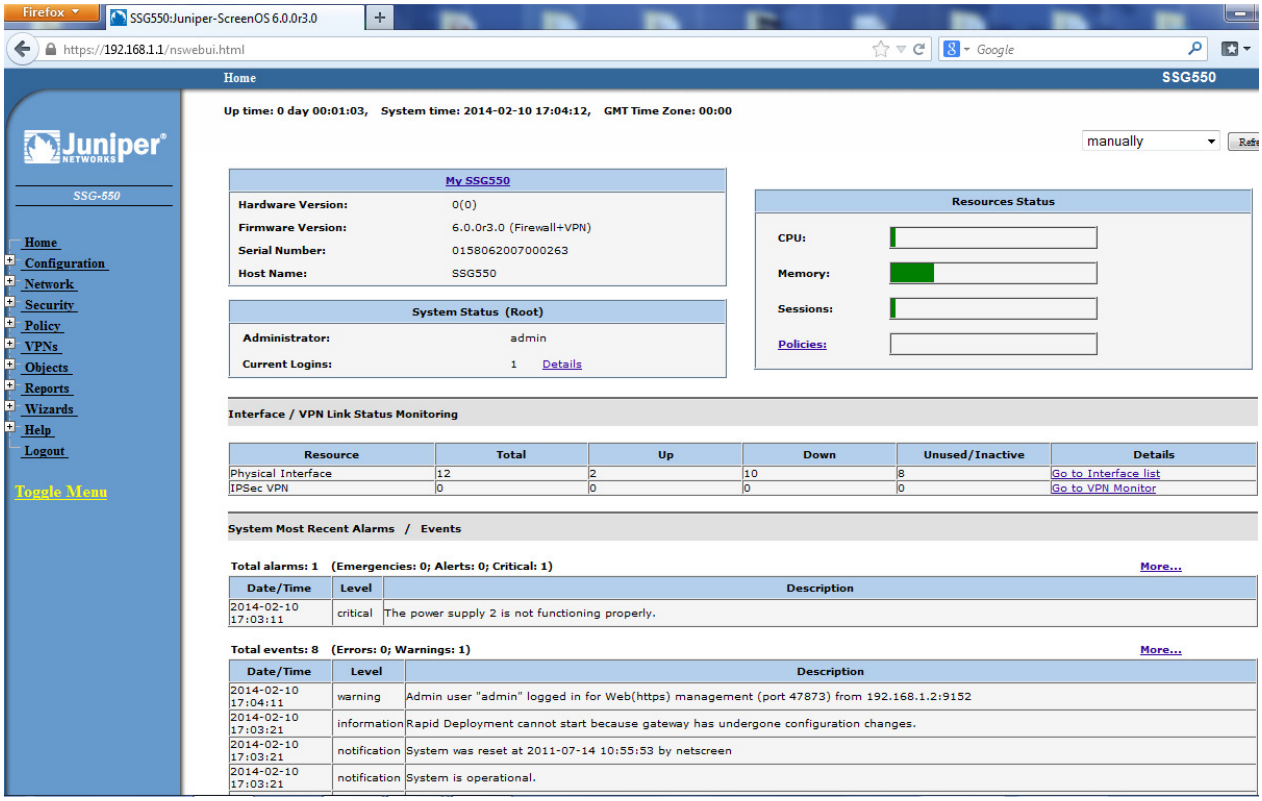
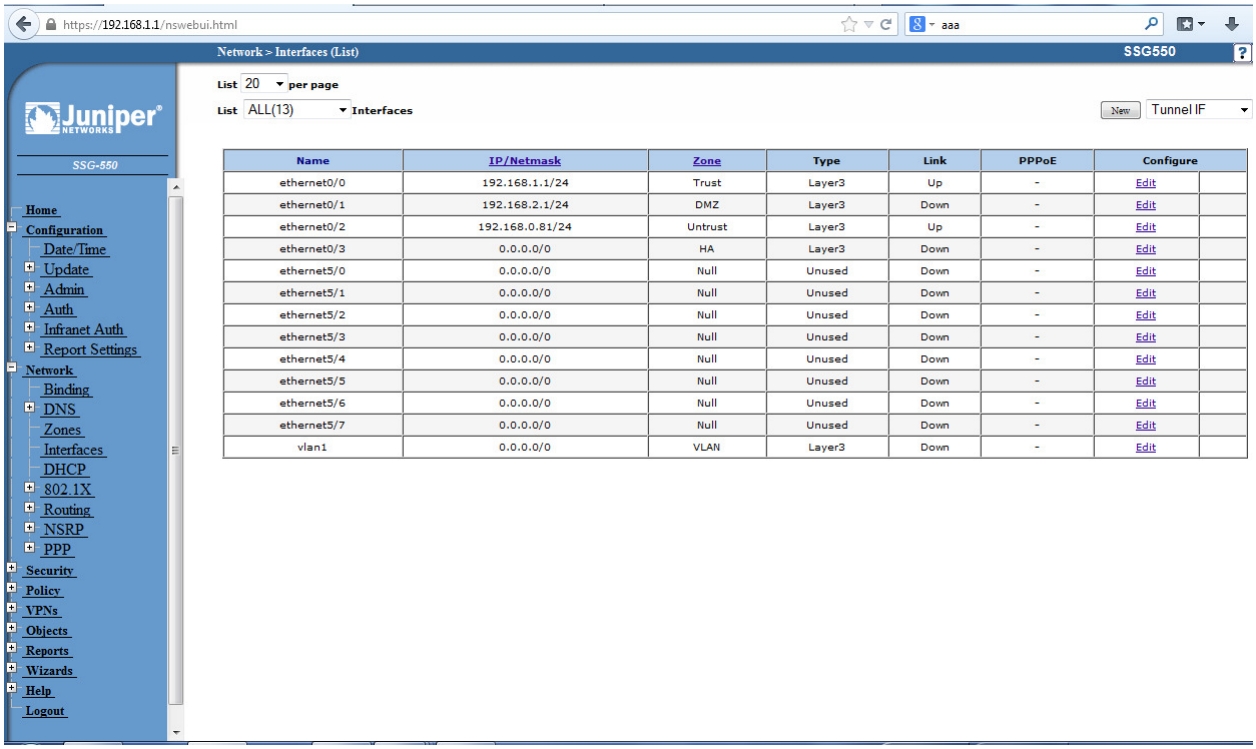


Figura 43. Vista Panel Principal.

Fuente: Propia.

Si detallamos en panel izquierdo, esta nos muestra una serie de funcionalidades del equipo, por ejemplo, si nos colocamos en *Network* → *Interfaces*, podemos enumerar las interfaces que nos presenta el equipo.



Name	IP/Netmask	Zone	Type	Link	PPPoE	Configure
ethernet0/0	192.168.1.1/24	Trust	Layer3	Up	-	Edit
ethernet0/1	192.168.2.1/24	DMZ	Layer3	Down	-	Edit
ethernet0/2	192.168.0.81/24	Untrust	Layer3	Up	-	Edit
ethernet0/3	0.0.0.0/0	HA	Layer3	Down	-	Edit
ethernet5/0	0.0.0.0/0	Null	Unused	Down	-	Edit
ethernet5/1	0.0.0.0/0	Null	Unused	Down	-	Edit
ethernet5/2	0.0.0.0/0	Null	Unused	Down	-	Edit
ethernet5/3	0.0.0.0/0	Null	Unused	Down	-	Edit
ethernet5/4	0.0.0.0/0	Null	Unused	Down	-	Edit
ethernet5/5	0.0.0.0/0	Null	Unused	Down	-	Edit
ethernet5/6	0.0.0.0/0	Null	Unused	Down	-	Edit
ethernet5/7	0.0.0.0/0	Null	Unused	Down	-	Edit
vlan1	0.0.0.0/0	VLAN	Layer3	Down	-	Edit

Figura 44. Vista de interfaces de red.

Fuente: Propia.

Donde por ejemplo la interfaz ETH0/0 tiene configurada la IP 192.168.1.1 /24, y es la que nos permite llegarle a la interfaz GUI del equipo, y la ETH0/2 con la IP 192.168.0.80/ 24, por DHCP, es la que nos permite salir a internet a través de NAT, enrutando los paquetes por el Gateway de esta, la cual se detallara su configuración más adelante

En la siguiente vista podemos detallar la configuración de la interfaz ETH0/0 tiene configurada la IP de gestión 192.168.1.1 /24, configurada en la zona Trust, de manera estática. Entre los servicios habilitados podemos nombrar ICMP, SSH, SNMP, Telnet, etc.

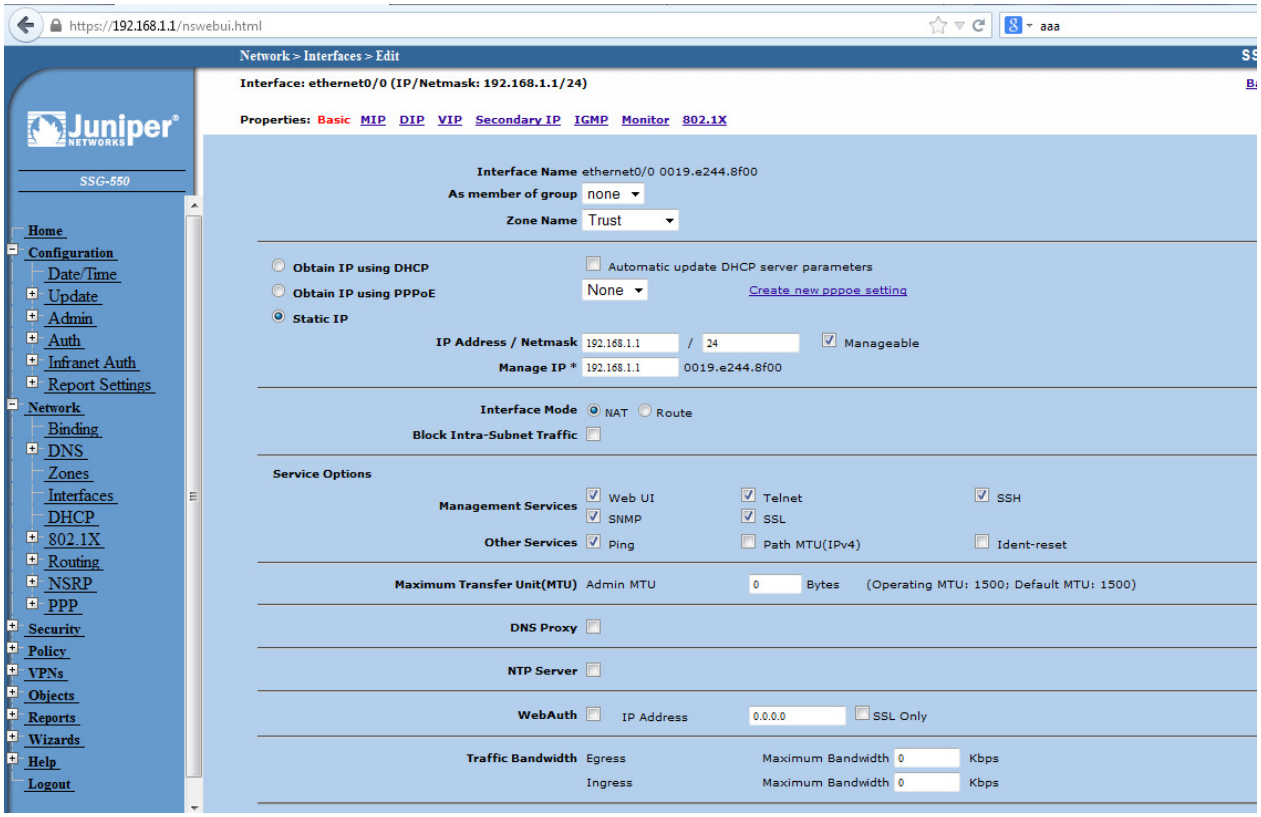


Figura 45. Vista de interfaz eth0/0.

Fuente: Propia.

Para la interfaz ETH0/2 tiene configurada la IP 192.168.0.81 /24, obtenida por DHCP, y se encuentra en la zona Untrust, (no segura). Como dije con anterioridad a través de esta interfaz se enrutará todo el tráfico de la ETH0/0

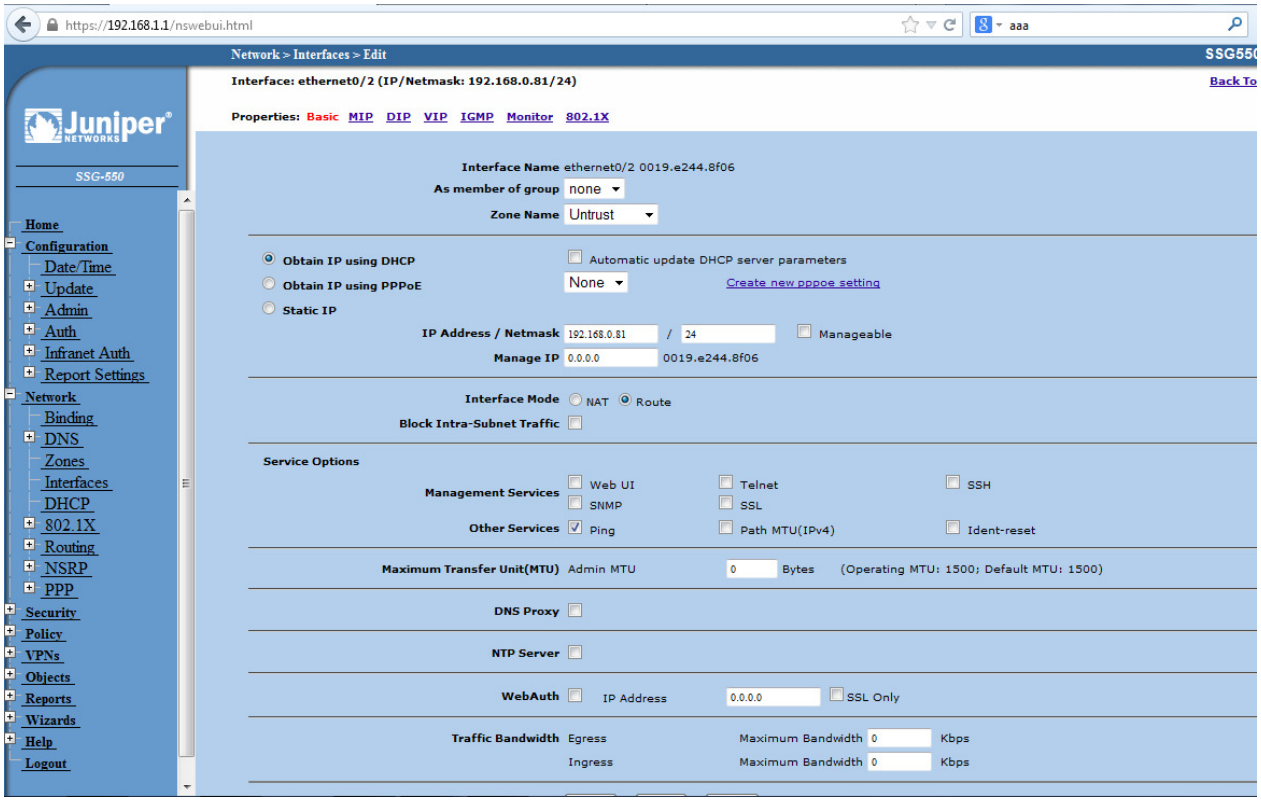


Figura 46. Vista de interfaz eth0/2.

Fuente: Propia.

En la siguiente imagen, se observa una tabla de enrutamiento creada, con el objetivo de enrutar los paquetes o tráfico que provengan de toda red, y dirigirlos a través del gatway 192.168.0.201 por la ETH0/2.

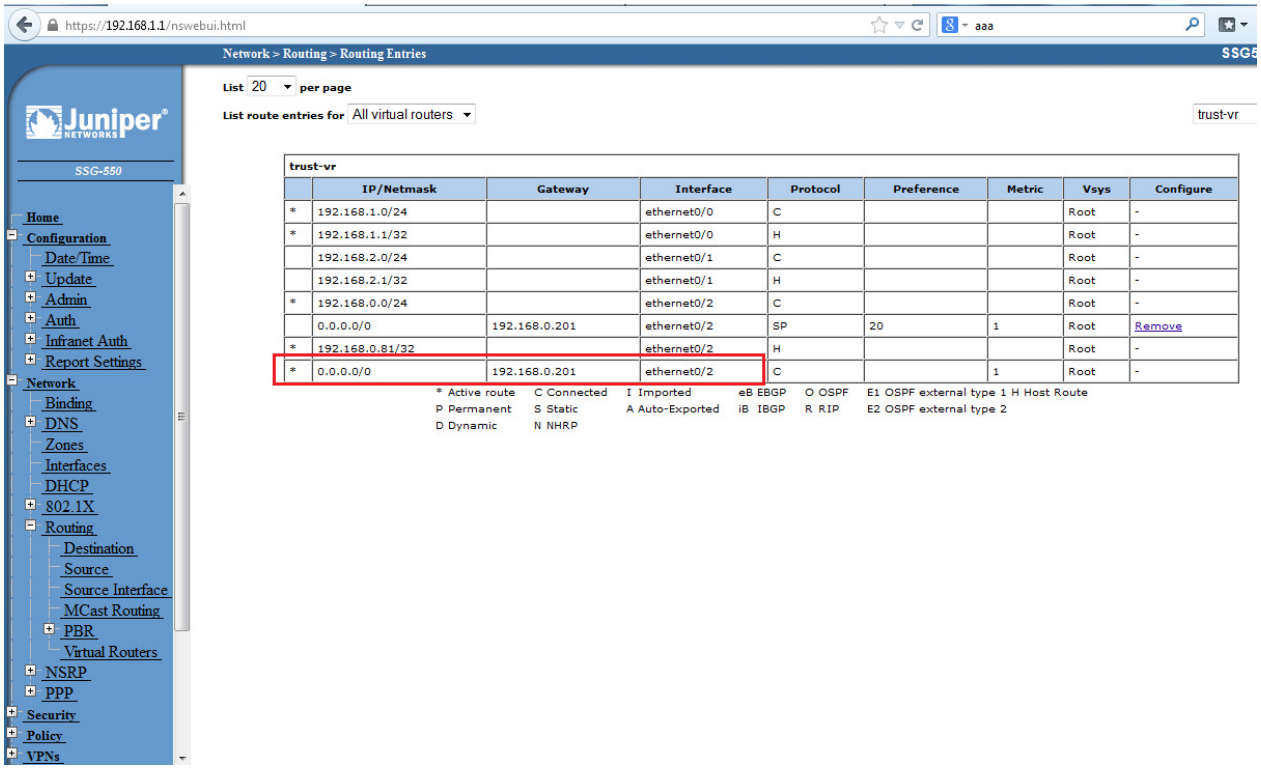


Figura 47. Vista de tabla de enrutamiento.

Fuente: Propia.

En la elaboración de las pruebas de enrutamiento, además se configuró un ruta que establecía como origen del tráfico, desde la interface ETH 0/0, es decir, todo tráfico que provenga de la red 192.168.1.0 /24, enrutarlo a través del Gateway 192.168.0.201 por la ETH0/2, como se expresa en la configuración.

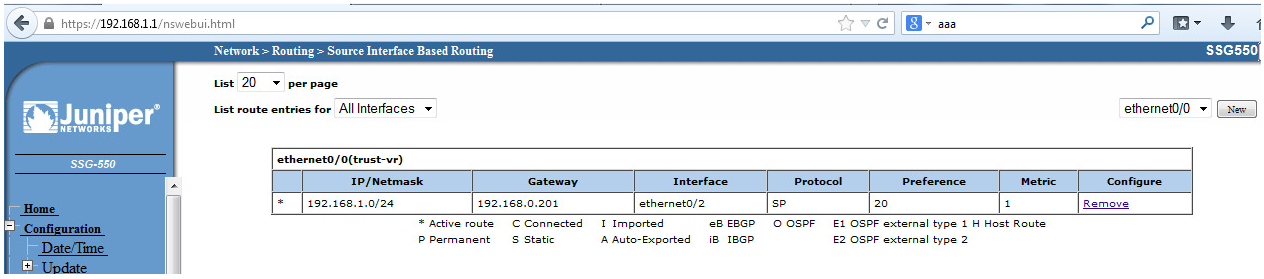


Figura 48. Vista de tabla de enrutamiento de interface eth0/0.

Fuente: Propia.

Siguiendo en la configuración del firewall, se procedió a establecer Políticas, entre las que se configuró al principio destaca la siguiente:

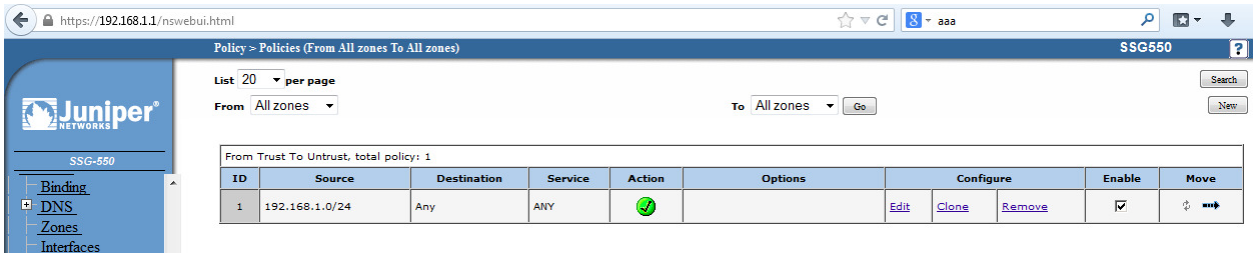


Figura 49. Configuración de regla.

Fuente: Propia.

Esta política permite que todo tráfico que provenga de la red 192.168.1.0 /24 sea destinada a cualquier red, al igual que permite cualquier servicio. Una vez implementada la política, realizando pruebas, aparece la siguiente ventana de validación:

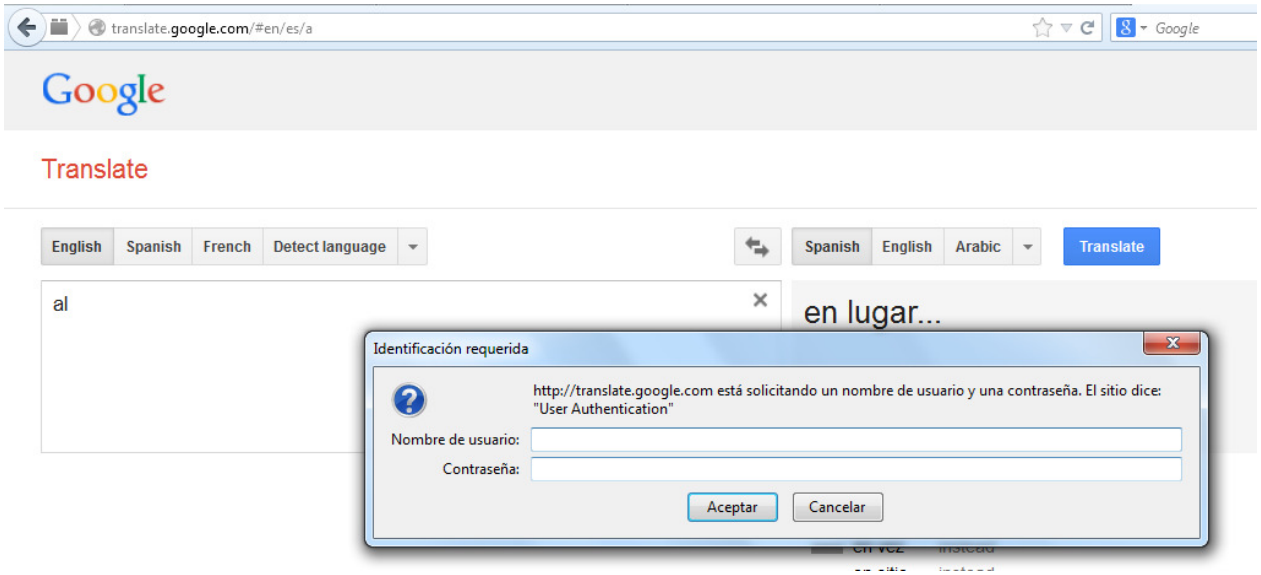


Figura 50. Vista de solicitud de credenciales para autenticación.

Fuente: Propia.

A continuación, en la siguiente captura procedemos a configurar la información del servidor de autenticación LDAP, que en este caso será mi ordenador, con la IP configurada 192.168.1.2 /24.

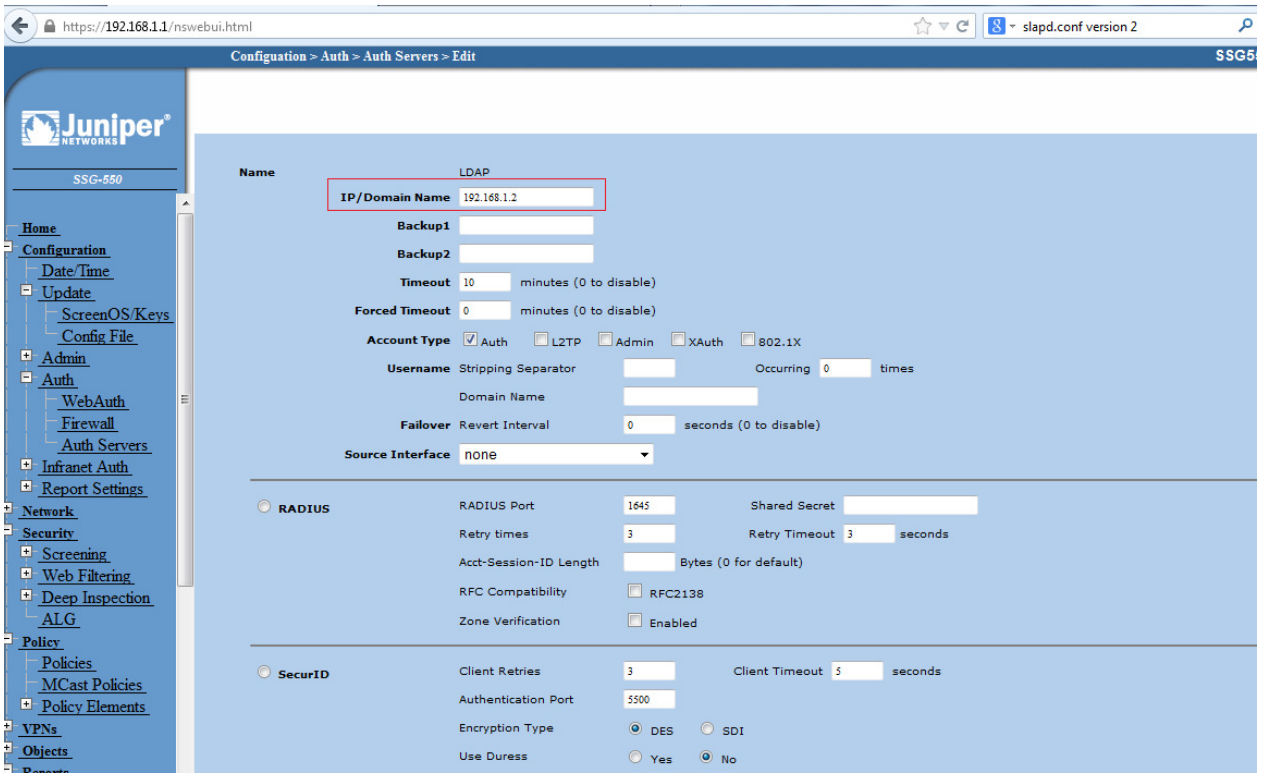


Figura 51. Configuración de la IP servidor LDAP.

Fuente: Propia

Y en la siguiente vista procedemos a colocar la información LDAP, como el puerto TCP 389, el Common Name Identifier (CN), y el Distinguished Name (dn), que para este piloto se configuró el siguiente en el servidor LDAP, *OU=People, dc=maxcrc, dc=com,* para así realizar la búsqueda en el árbol de usuarios.

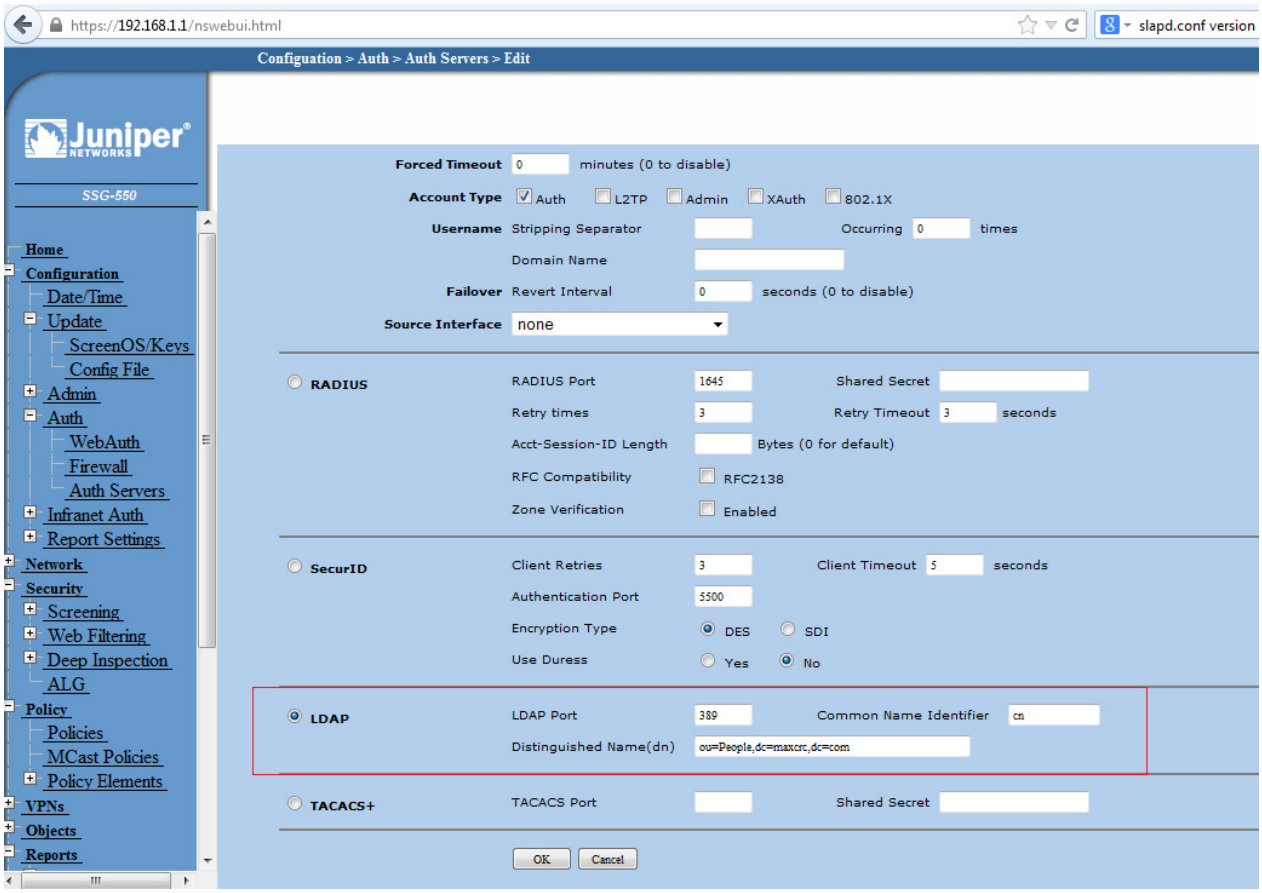


Figura 52. Configuración del (Puerto,CN,DN) en protocolo LDAP.

Fuente: Propia

En la fig.49, en cuanto a la configuración de las Políticas, solo se colocó una de ellas, realizando las pruebas respectivas, se agregó otra, la cual se diferencia de la primera que acepta los servicios, HTTP, HTTP-EXT y HTTPS, es decir, todo el tráfico que provenga de la red 192.168.1.0/24, hacia cualquier destino, permitir la apertura de los puertos HTTP, HTTP-EXT y HTTPS.

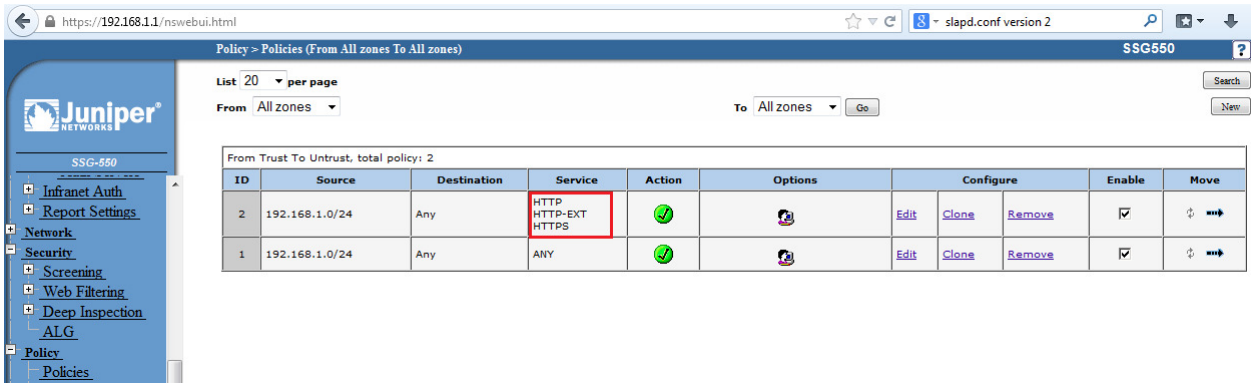


Figura 53. Configuración de los servicios de la regla.

Fuente: Propia

Además de permitir los servicios, HTTP nombrados anteriormente en la regla creada, en la configuración de las políticas, hay un segmento de *Configuración Avanzada de Política*, la cual la realizamos en las dos ya creadas anteriormente, donde se tildaron las casilla de *Authentication*, *el Auth Server*, se eligió *LDAP*, además que permitiese a cualquier usuario. Puede verse la configuración en la siguiente captura:

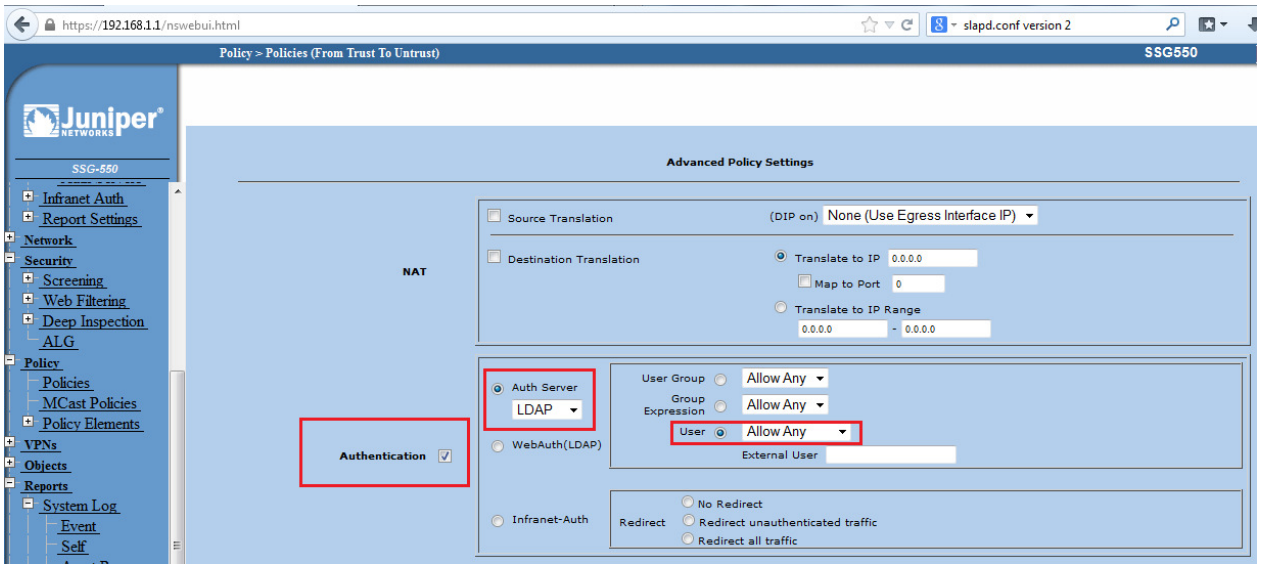


Figura 54. Configuración de la autenticación y usuarios.

Fuente: Propia

4. Plan de Pruebas del Sistema de control de acceso.

4.1 Ejecución de pruebas del piloto.

Finalizado el proceso de configuración del sistema de control de acceso, con todos los componentes software, se procedió a realizar una serie de pruebas para demostrar el funcionamiento del piloto, es decir, realizar la autenticación de manera correcta de un usuario creado en LDAP, para poder tener acceso a una red de comunicación, en la misma se evaluará los log, que permitirán analizar de una forma más completa su funcionamiento.

En la siguiente captura, al iniciar nuestro servidor LDAP, y tener nuestro firewall en funcionamiento, con sus políticas, se procedió a realizar una consulta para navegar en internet, la respuesta de esta petición se muestra en una ventana de autenticación de usuario, esto quiere decir, que al realizar dicha petición con nuestro usuario (*admin*), esta va contra nuestro repositorio en LDAP y realizar la autenticación por el firewall.

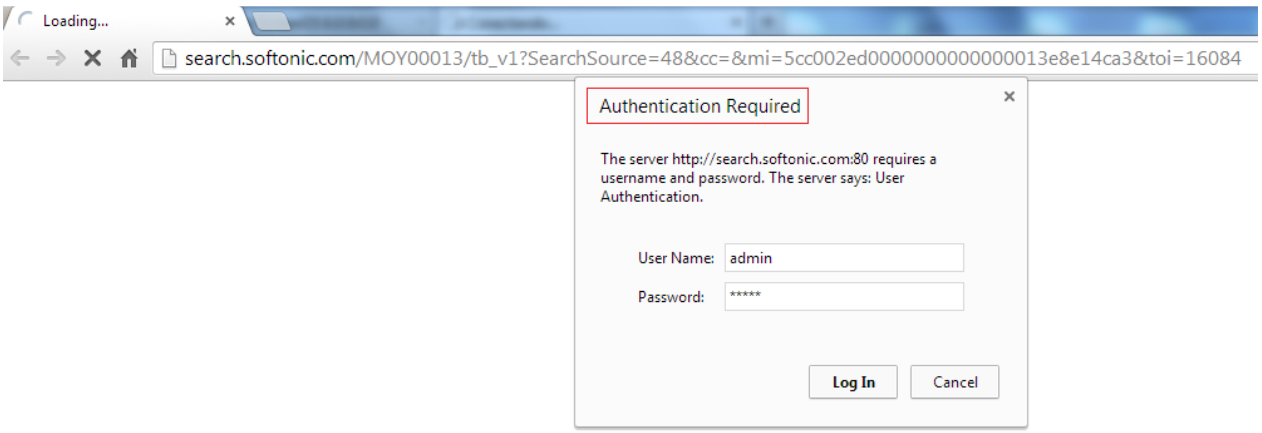


Figura 55. Vista de ventana solicitando credenciales de autenticación (*Admin*).

Fuente: Propia

Siguiendo con las pruebas, se realizó la misma con el usuario *pepe*, para realizar otra consulta, de igual manera se realiza dicha petición, que busca en nuestro repositorio LDAP, y autentica a través de nuestro firewall.

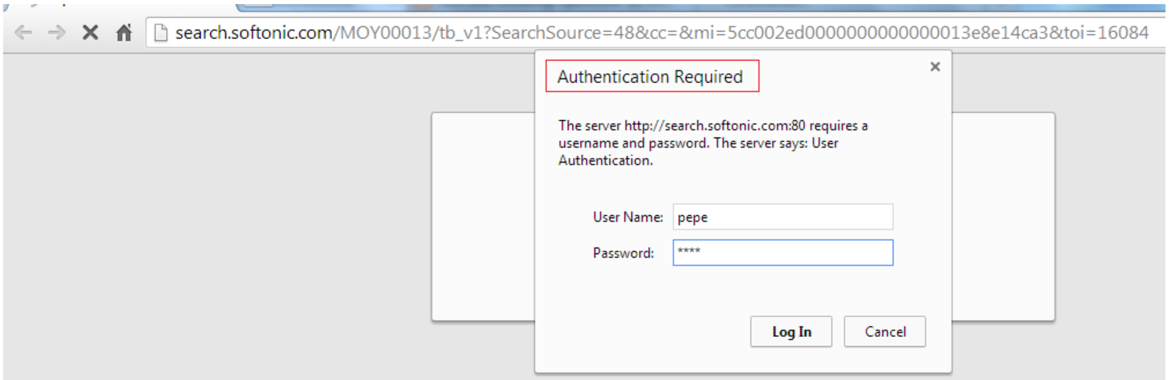


Figura 56. Vista de ventana solicitando credenciales de autenticación (*pepe*).

Fuente: Propia

La validación de nuestros usuarios *admin*, puede verse que se realizó de manera satisfactoria en los log del sistema, que a continuación se mostrará en la siguiente imagen.

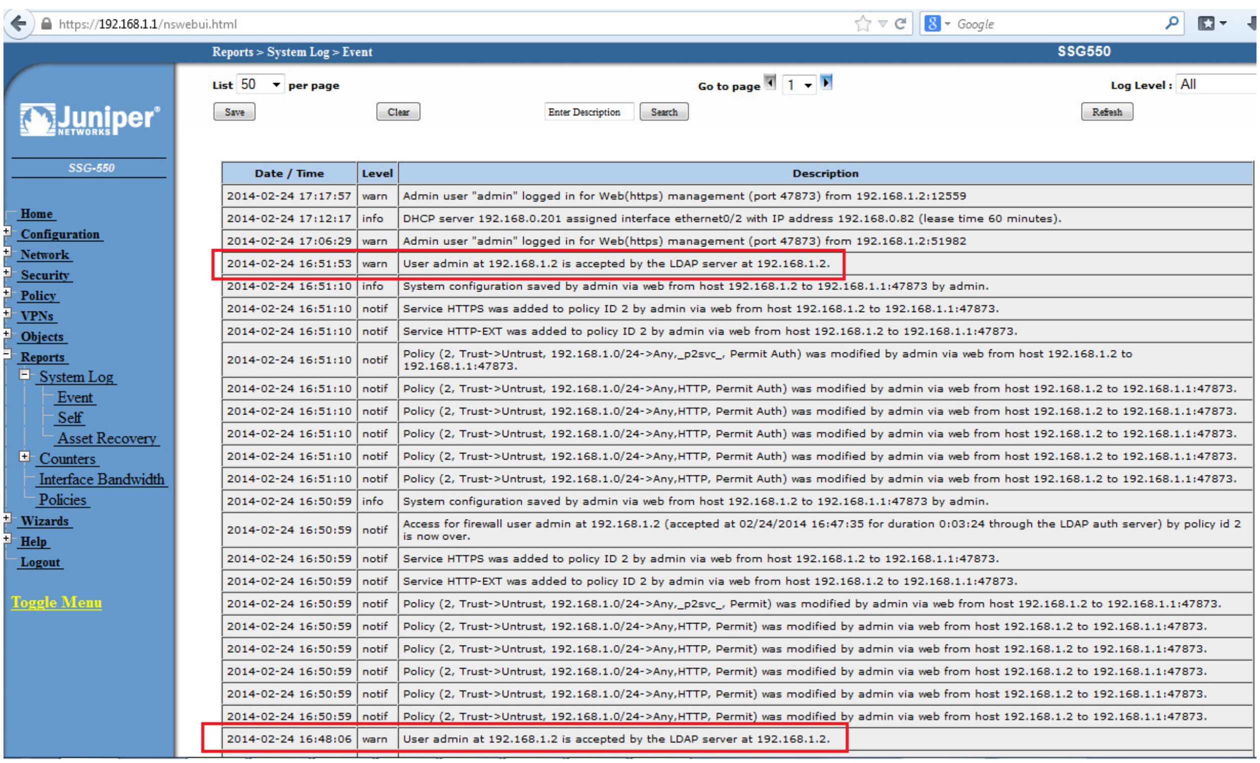


Figura 57. Vista de eventos o logs verificando la autenticación de usuarios a través del Juniper.

Fuente: Propia

En esta otra imagen puede verificarse la validación de nuestro usuario *pepe*, puede observarse que se realizó de manera satisfactoria en los log del sistema. Otro de los mensajes de interés para su análisis, son los que están señalados en el recuadro azul, donde indica el acceso por el firewall del usuario (*admin, pepe*) en el servidor 192.168.1.2, el día como la hora y la duración en minutos desde que inicia sesión el usuario hasta su desconexión, a través de la política inidentificada número 2.

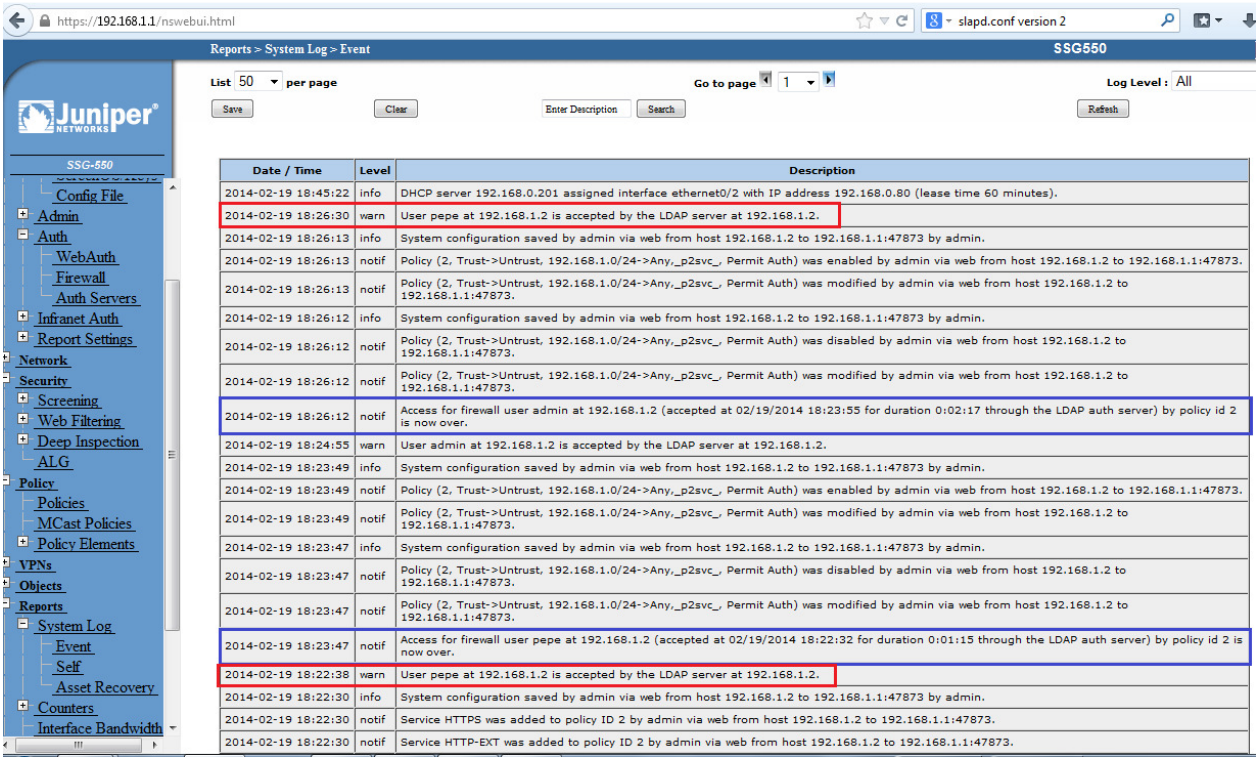


Figura 58. Vista de eventos o logs verificando la autenticación a través del Juniper y las reglas aplicadas.

Fuente: Propia

4.2 Captura y Análisis de tráfico.

En la siguiente sección, mediante la herramienta Wireshark, se procedió a realizar un análisis de tráfico de nuestro sistema, para observar en principio o detectar por que no se realizaba la autenticación a través del protocolo LDAP contra el servidor. En principio, al realizar las

pruebas de análisis de tráfico, en uno de los errores que se capturó se hizo mención de la versión de LDAP en el servidor, esto dio pie a realizar cambios en la versión de LDAP.

Siguiendo con el protocolo de pruebas, se realizó varias de ellas, donde se puede visualizar el correcto funcionamiento del sistema, es decir, al capturar dicho tráfico, podemos observar el protocolo en funcionamiento, que en este caso es LDAP, como también el usuarios y la estructura Common Name del usuario que se autentica.

Por ejemplo, en esta primera captura, podemos visualizar, en principio, que se realiza una petición satisfactoria del usuario *pepe* desde la ip 192.168.1.1 al servidor 192.168.1.2, a través del protocolo LDAP, puerto 389. Esta petición de autenticación es realizada de manera satisfactoria. La estructura del Common Name es cn=pepe,ou=People,dc=maxcrc,dc=com, la cual es la que se observa en la imagen.

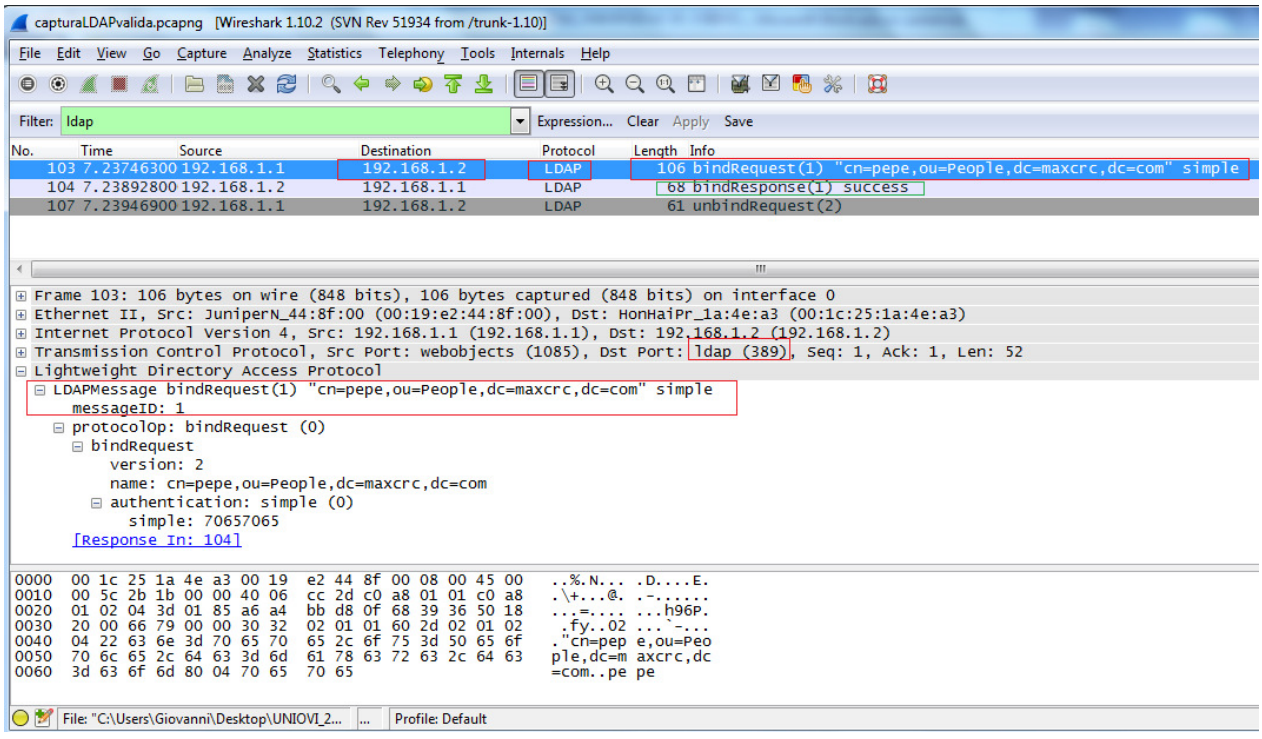


Figura 59. Captura de tráfico visualizando IP del servidor y protocolo LDAP.

Fuente: Propia

Siguiendo con las pruebas, se realiza el mismo proceso de autenticación, pero con el usuario *admin*, igualmente se realiza una petición del usuario *admin* desde la ip 192.168.1.1 al servidor 192.168.1.2, a través del protocolo LDAP, puerto 389. Esta petición de autenticación es realizada de manera errónea con la contraseña *gmazzei* la primera vez, para verificar que realiza la validación y la búsqueda en el árbol LDAP, en el segundo intento se realiza de manera satisfactoria.

La estructura del Common Name es *cn=admin,ou=People,dc=maxcrc,dc=com*.

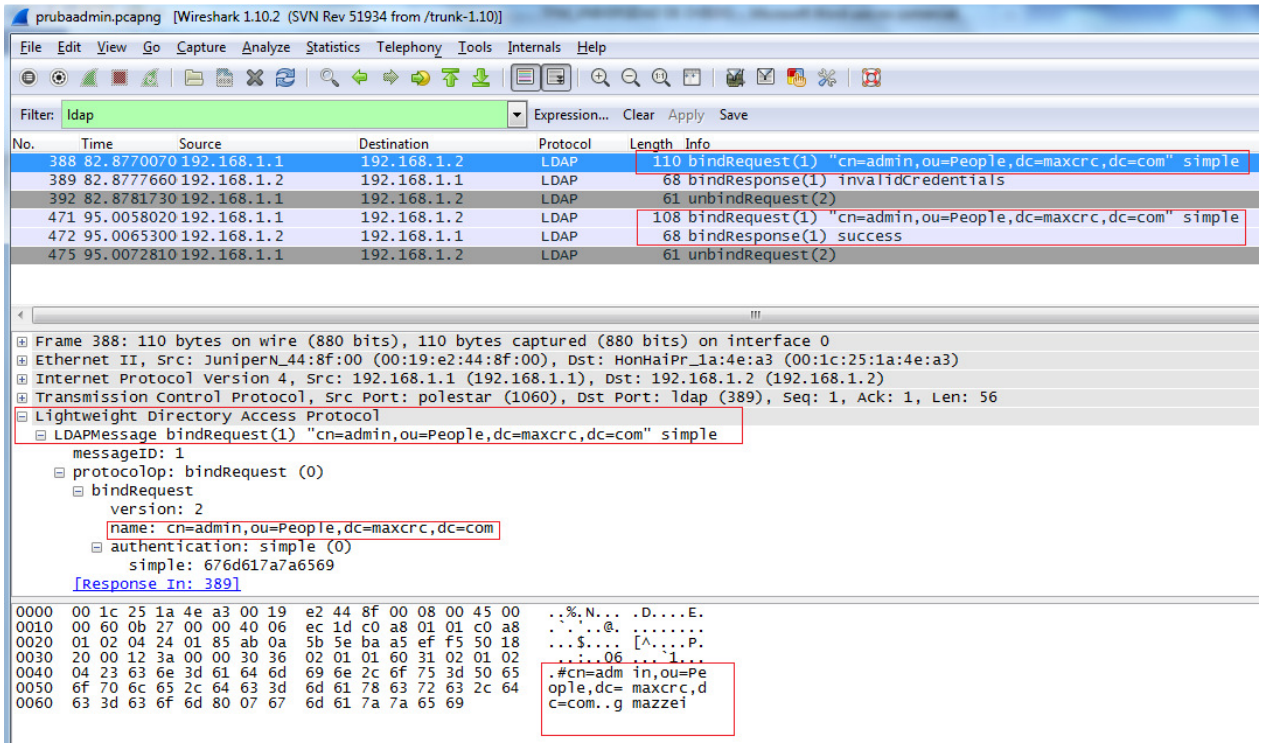


Figura 60. Captura de tráfico visualizando el proceso de autenticación (Invalido y Satisfactorio).

Fuente: Propia

4.3 Problemas presentados.

En la realización y ejecución del presente trabajo, surgieron varios problemas a considerar, entre estos problemas podemos citar los siguientes:

- ✓ En principio, al realizar las pruebas de análisis de tráfico, en uno de los errores que se capturó se hizo mención de la versión de LDAP en el servidor, esto permitió realizar

cambios en la versión de LDAP, específicamente en el archivo de configuración. En una configuración inicial, se colocó LDAPv3, pero a raíz de las pruebas y análisis de captura de tráfico, se cambión a LDAPv2 en el archivo *slapd.conf*. En la captura de pantalla puede verse, la línea de código numero 26 referente a la versión del LDAP, en la que se agregó *allow bind_v2*.

```
1  # BDB Backend configuration file
2  # See slapd.conf(5) for details on configuration options.
3  # This file should NOT be world readable.
4  ucdata-path ./ucdata
5  include     ./schema/core.schema
6  include     ./schema/cosine.schema
7  include     ./schema/nis.schema
8  include     ./schema/inetorgperson.schema
9  include     ./schema/openldap.schema
10 include     ./schema/dyngroup.schema
11
12
13 pidfile     ./run/slapd.pid
14 argsfile    ./run/slapd.args
15
16
17 # Enable TLS if port is defined for ldaps
18
19
20 TLSVerifyClient never
21 TLSCipherSuite HIGH:MEDIUM:-SSLv2
22 TLSCertificateFile ./secure/certs/server.pem
23 TLSCertificateKeyFile ./secure/certs/server.pem
24 TLSCACertificateFile ./secure/certs/server.pem
25
26 allow bind_v2
27
28 ########################################################################
29 # bdb database definitions
30 ########################################################################
31
32
33 database    bdb
34 suffix      "dc=maxcrc,dc=com"
35 rootdn      "cn=Manager,dc=maxcrc,dc=com"
```

Figura 61. Vista del archivo de configuración del *slapd.conf*.

Fuente: Propia

- ✓ El segundo problema detectado en la realización de la presente demo, se hace referencia a la contraseña que se coloca para autenticar al usuario a la red, ya que esta va en texto plano o simple. En una de las pruebas de captura y análisis de tráfico, se puede verificar la contraseña colocada por un usuario. Por ejemplo en la captura puede verse la contraseña colocada para el usuario *admin*, que también su contraseña es *admin.*

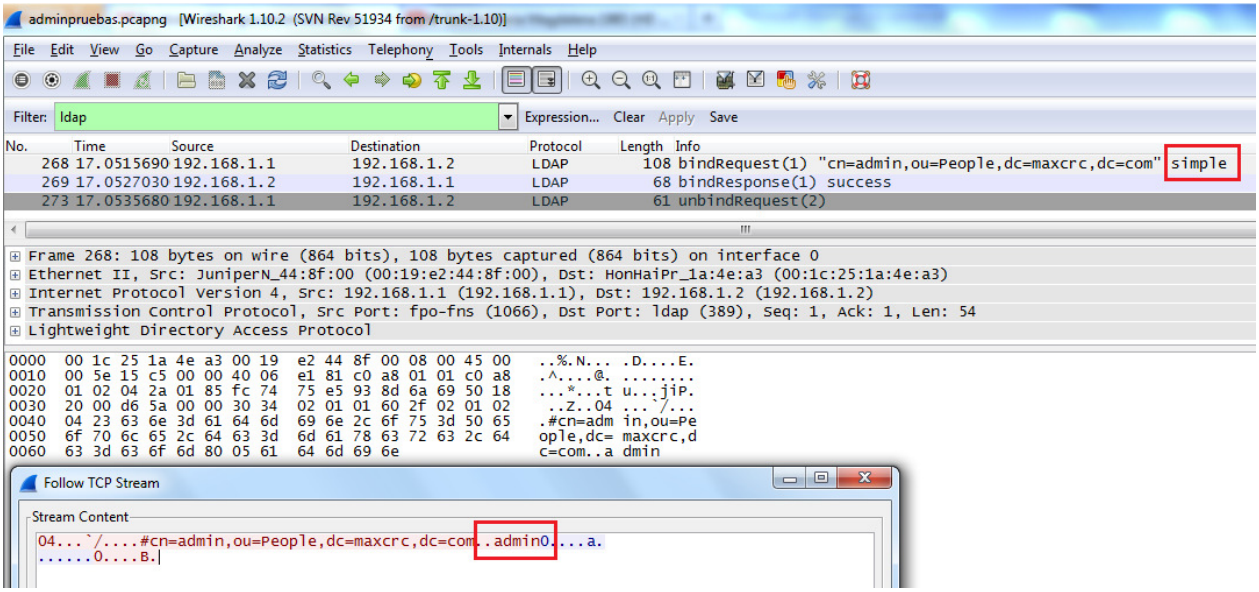


Figura 62. Captura de tráfico que visualiza la contraseña en texto plano.

Fuente: Propia

5.- Resultados y Conclusiones.

5.1 Resultados.

Mediante la realización y finalización de las pruebas de manera satisfactoria, se han detallado varios elementos a tener en consideración, en primer plano, evaluar la eficacia del control de acceso mediante el protocolo LDAP, además de las prestaciones del firewall de Juniper, que permitió mediante su configuración establecer las políticas pertinentes para aceptar o no a un usuario en una red. En las pruebas realizadas, se configuraron dos usuarios, sobre las cuales se probó la autenticación para el acceso a la red de la Universidad de Oviedo.

El resultado final de las pruebas, permitieron autenticar a dos usuarios configurados en LDAP, en este caso, el usuario *Admin* y *Pepe*, como se explicó en el marco teórico, la configuración del árbol de usuarios pueden ser en grupos o usuarios particulares, como es en esta prueba. Mediante la captura a continuación puede observarse mediante el analizador de tráfico el usuario autenticado de manera correcta, como la desconexión con el servidor LDAP, dando así satisfactoria la autenticación. Al principio de las pruebas, hubo dificultad a la hora de que el protocolo LDAP fuese reconocido, en la documentación se estableció por un error en la versión del mismo, esto puedo detectarse en un principio mediante el analizador de tráfico, una vez que se reconfiguró el archivo *slapd.conf*, se realizó de manera correcta la autenticación y el respectivo acceso a la red.

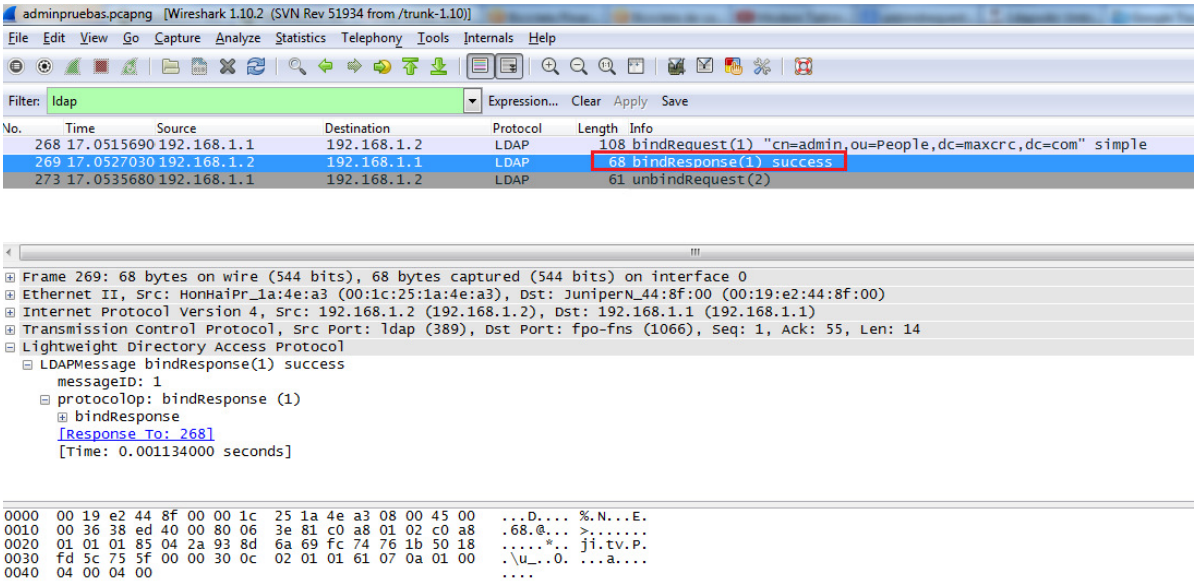


Figura 63. Captura de tráfico que visualiza la conexión y desconexión del usuario en el servidor LDAP.

Fuente: Propia

Otro resultado que se obtuvo es que mediante la herramienta Jxplore, establecemos una contraseña en texto plano, esto es un punto no favorable, ya que compromete de manera contundente la seguridad, esto se observó, mediante el análisis de tráfico, donde se podía establecer la contraseña configurada en la herramienta. Esto implica que cualquier persona que esté escaneado la red, pueda capturar una contraseña y vulnerar el sistema. A la hora de autenticar para tener acceso a la red, hubo errores, pero dichos errores fueron a causa de colocación de usuarios y contraseñas de manera incorrecta.

5.2 Conclusiones.

Las conclusiones extraídas del presente trabajo la expresamos de la siguiente manera:

A lo largo de este Trabajo Fin de Máster, he hecho un análisis de todos los aspectos de seguridad que implica a la hora de acceder a una red, vía conexión alámbrica, especialmente se ha hecho énfasis al protocolo LDAP, a través del equipo de red firewall de Juniper, para el acceso a una red, en este caso, se tomó en consideración el acceso a la red de la Universidad de Oviedo. Al

principio del presente trabajo, se realizó mención de la seguridad que presenta la infraestructura de acceso a la red en un punto cualquiera de la universidad vía alámbrico. Como prueba de la misma, me he conectado a internet con mi laptop en un laboratorio, donde no se me pedía credenciales a diferencia de la red inalámbrica.

Podemos concluir para el presente trabajo lo siguiente:

- ✓ Actualmente existen numerosas alternativas en cuanto a la seguridad de acceso a redes, ya sea tanto alámbrica como inalámbrica, el acceso a la misma por vía alámbrica se ve comprometida, por la ausencia de algún protocolo de autenticación a la red, a diferencia de la inalámbrica (WiFi), la cual para poder acceder a la intranet como a una red externa se nos pide las credenciales respectivas.

- ✓ El estudio del marco teórico, específicamente del desarrollo de los métodos de autenticación AAA, como los de soporte ha permitido comprender cada uno de ellos, en especial, su funcionamiento y su interacción en un entorno empresarial.

- ✓ Mediante la elaboración y puesta en marcha de una demo, se ha podido mostrar, el funcionamiento correcto de la autenticación de un usuario, a través de los log mostrados por la herramienta, se ha hecho seguimiento de su funcionamiento, como también de la puesta en marcha del protocolo LDAP, así de las políticas o reglas configuradas en la misma, para que esta realizase dicha función.

- ✓ El estudio del diseño de una conexión segura en una red alámbrica mediante la autenticación a través del firewall Juniper, ha permitido cerrar un poco esa brecha que existe en las redes alámbricas, y hacerlas más seguras, dando así confiabilidad e integridad a los usuarios que acceden a redes, ya sean comerciales o domésticas, y que a la vez se garantice la integridad de la información y los datos que por allí circula.

- ✓ Una de las desventajas del uso de herramientas basadas en software libre, es la carencia de empresas que den soporte y servicio especializado a estas soluciones, teniendo en muchos casos que ser asumido por la empresa que los implante.

- ✓ Finalmente, todo mecanismo de protección de información en una red debe estar enmarcado dentro de una política de seguridad adecuada. El seguimiento de una política consistente evita que las medidas de protección se vuelvan un obstáculo para el trabajo habitual con los sistemas de información, y garantiza la calidad y confidencialidad de la información presente en los sistemas de la empresa.

6.- Recomendaciones.

En la elaboración del presente trabajo en conjunto con las conclusiones, podemos establecer ciertas recomendaciones para futuras implementaciones de estos sistemas de control de acceso a redes alámbricas, entre las que podemos nombrar tenemos:

- ✓ Realizar test de intrusión aplicando las diferentes metodologías, ente las que se encuentran OSSTMM (Open Source Security Testing Methodology Manual), ISSAF (Information Systems Security Assessment Framework) y OWASP (Open Web Application Security Project).

- ✓ Evitar el uso de contraseñas que viajan en texto plano, ya que comprometen la seguridad en la red, ya que pueden ser descifrados a través de sniffer o analizadores de tráfico. Evaluando las posibles alternativas, las deducimos en primer lugar realizar una actualización del firmware de la versión actual 6.0.0r3.0 a la recomendada en la página del fabricante de Juniper, la cual establece la versión 6.2.0r3. Realizar una actualización de la versión de LDAP, de la versión 2 a la versión 3. A pesar que se realizó el trabajo con la versión anterior, en la realización de las pruebas, tuvo que configurarse con la versión 2, ya que al realizar pruebas con la versión 3, no realizaba la autenticación del usuario de prueba, al realizar el análisis del tráfico, establecía error de versión. Por tanto, se reconfiguró el archivo de configuración *slapd.conf*, con la versión 2. En cuanto a realizar contraseñas más fuertes de romper, podemos utilizar el comando *slappasswd* que es una herramienta muy útil que nos permite generar passwords utilizables en los atributos userPassword de nuestros usuarios almacenados en ldap o en la directiva de configuración *rootpwd*, que sirve para definir un password de root para ldap en el archivo de configuración del servidor ldap.

  Otra opción factible es que el servidor openLDAP puede ser configurado para que utilice las prestaciones de cifrado que ofrece OpenSSL. Normalmente las consultas al servidor LDAP se realizan por el puerto 389 (protocolo ldap) pero dichas consultas se transmiten sin cifrar. Para realizar consultas seguras cifrando los datos con SSL, es necesario utilizar

el puerto 636 (protocolo ldaps o protocolo ldap seguro). Para ello, el servidor debe disponer de un certificado firmado por una entidad certificadora (CA) y habrá que configurar slapd para que utilice los certificados.

Bibliografía.

[1] Casamor, Antonio Salavert. (2003).Los Protocolos en las redes de ordenadores. Ediciones Universidad Politécnica de Catalunya.

[2] España Boquera, María Carmen. (2003). Servicio Avanzado de Telecomunicaciones. Ediciones Diaz de Santos S.A. Madrid, España. Págs. 222-229

[3] Glenn, Walter. (2005). Linksys Networks. The Oficial Guide Mc-Grow Hill.

[4] Habaken, Joe. (2006). Home Wireless Networking in a Snap. Sams Publisher. Indianápolis, Indiana USA.

[5] Hansen, Yago Fernández; Varón, Antonio Ramos; García-Morán, Jean Paul. RADIUS /AAA / 802.1X.Sistemas basados en la autenticación en Windows y GNU/Linux. Ediciones RA-MA. Madri, España.Págs.21-116, 505-522.

[6] Mathon, Philippe. (2004). VPN Implementación en Windows Server. Ediciones Eni. Barcelona, España. Págs. 160-175.

[7] Mathon, Philippe. (2001). TCP IP/ entorno Windows 2000.Ediciones Eni. Barcelona, España.

[8] Roger, Jean Marc. (2004). Seguridad en la Informática de la Empresa. Riesgos, amenazas, prevención y soluciones. Ediciones Eni .Barcelona, España.

[9] Yang Xiao, Jie Li Yi Pan. (2005). Wireless Networks and Mobile Computing. Security and Rounting in Wireless Networks .Ediciones Nova.

[10] Mendillo, Vincenzo. Universidad Central de Venezuela. Seguridad en Informática y Comunicaciones. [DVD].

[11] Castro Gil Manuel Alonso, Díaz Orueta Gabriel, Alzórriz Armendáriz Ignacio, Sancristóbal Ruiz Elio. PROCESOS Y HERRAMIENTAS PARA LA SEGURIDAD DE REDES. Págs.289-290.

[12] Cisco. http://www.cisco.com/

[13] Jxplore. http://jxplorer.org/

[14] OpenLDAP. http://www.openldap.org/

[15] Juniper. www.juniper.net/

[16] Kerberos. http://www.kerberos.org/

[17] Microsoft. http://www.microsoft.com/

[18] Centro Nacional de Tecnología de la Información. http://www.cnti.gob.ve/

[19] TACACS. http://www.tacacs.net

[20] ZeroShell. http://www.zeroshell.org/

ANEXOS A.

A.1 Configuración CLI Juniper SSG 550

El presente anexo presenta la configuración completa del equipo Juniper, en general puede detallarse, la configuración de las políticas, los servicios, la configuración de las interfaces de red, las direcciones IP, como también el routing. Este nos permite a su vez, tener una mejor visión de la configuración del equipo, ya que en caso de presentarse un error o algún problema en la configuración, su revisión es mucho más productiva a diferencia de la GUI.

```
set clock timezone 0
set vrouter trust-vr sharable
set vrouter "untrust-vr"
exit
set vrouter "trust-vr"
unset auto-route-export
exit
set auth-server "Local" id 0
set auth-server "Local" server-name "Local"
set auth-server "LDAP" id 1
set auth-server "LDAP" server-name "192.168.1.2"
set auth-server "LDAP" account-type auth
set auth-server "LDAP" type ldap
set auth-server "LDAP" ldap cn "cn"
set auth-server "LDAP" ldap dn "ou=People,dc=maxcrc,dc=com"
set auth default auth server "LDAP"
set auth radius accounting port 1646
set admin name "admin"
set admin password "nH/vDirbE5GBcjdGoslAEBBtHFA6En"
set admin http redirect
set admin auth timeout 10
set admin auth server "Local"
set admin format dos
set zone "Trust" vrouter "trust-vr"
set zone "Untrust" vrouter "trust-vr"
set zone "DMZ" vrouter "trust-vr"
set zone "VLAN" vrouter "trust-vr"
set zone "Untrust-Tun" vrouter "trust-vr"
```

```
set zone "Trust" block
unset zone "Trust" tcp-rst
set zone "Untrust" block
unset zone "Untrust" tcp-rst
set zone "MGT" block
set zone "DMZ" tcp-rst
set zone "VLAN" block
unset zone "VLAN" tcp-rst
set zone "Untrust" screen tear-drop
set zone "Untrust" screen syn-flood
set zone "Untrust" screen ping-death
set zone "Untrust" screen ip-filter-src
set zone "Untrust" screen land
set zone "V1-Untrust" screen tear-drop
set zone "V1-Untrust" screen syn-flood
set zone "V1-Untrust" screen ping-death
set zone "V1-Untrust" screen ip-filter-src
set zone "V1-Untrust" screen land
set interface "ethernet0/0" zone "Trust"
set interface "ethernet0/1" zone "DMZ"
set interface "ethernet0/2" zone "Untrust"
set interface ethernet0/0 ip 192.168.1.1/24
set interface ethernet0/0 nat
unset interface vlan1 ip
set interface ethernet0/1 ip 192.168.2.1/24
set interface ethernet0/1 nat
set interface ethernet0/2 ip 192.168.0.82/24
set interface ethernet0/2 nat
unset interface vlan1 bypass-others-ipsec
unset interface vlan1 bypass-non-ip
set interface ethernet0/0 ip manageable
set interface ethernet0/1 ip manageable
unset interface ethernet0/2 ip manageable
set interface ethernet0/2 manage ping
set interface vlan1 manage mtrace
set interface ethernet0/2 dhcp client enable
unset interface ethernet0/2 dhcp client settings update-dhcpserver
set interface ethernet0/0 dhcp server service
set interface ethernet0/0 dhcp server enable
set interface ethernet0/0 dhcp server option gateway 192.168.1.1
set interface ethernet0/0 dhcp server option netmask 255.255.255.0
```

```
set interface ethernet0/0 dhcp server option dns1 192.168.0.201
set interface ethernet0/0 dhcp server ip 192.168.1.33 to 192.168.1.126
unset interface ethernet0/0 dhcp server config next-server-ip
unset flow no-tcp-seq-check
set flow tcp-syn-check
unset flow tcp-syn-bit-check
set flow reverse-route clear-text prefer
set flow reverse-route tunnel always
set webauth server "LDAP"
set pki authority default scep mode "auto"
set pki x509 default cert-path partial
set dns host dns1 0.0.0.0
set dns host dns2 0.0.0.0
set dns host dns3 0.0.0.0
set address "Trust" "192.168.1.0/24" 192.168.1.0 255.255.255.0
set address "Untrust" "192.168.0.0/24" 192.168.0.0 255.255.255.0
set address "Untrust" "192.168.0.81/24" 192.168.0.81 255.255.255.0
set user "pepe" uid 1
set user "pepe" type  auth
set user "pepe" hash-password "02bmrQcpuWNKN+pN/QfNOFAJ+7MmAxhbC6ZZM="
set user "pepe" "enable"
set ike respond-bad-spi 1
unset ike ikeid-enumeration
unset ike dos-protection
unset ipsec access-session enable
set ipsec access-session maximum 5000
set ipsec access-session upper-threshold 0
set ipsec access-session lower-threshold 0
set ipsec access-session dead-p2-sa-timeout 0
unset ipsec access-session log-error
unset ipsec access-session info-exch-connected
unset ipsec access-session use-error-log
set vrouter "untrust-vr"
exit
set vrouter "trust-vr"
exit
set url protocol websense
exit
set policy id 2 name "LDAP_HTTP" from "Trust" to "Untrust"  "192.168.1.0/24" "Any"
"HTTP" permit auth
set policy id 2
```

```
set service "HTTP-EXT"
set service "HTTPS"
exit
set policy id 1 name "LDAP" from "Trust" to "Untrust"  "192.168.1.0/24" "Any" "ANY"
permit auth server "LDAP"
set policy id 1
exit
set nsmgmt bulkcli reboot-timeout 60
set ssh version v2
set config lock timeout 5
unset license-key auto-update
set snmp port listen 161
set snmp port trap 162
set vrouter "untrust-vr"
set router-id 156.35.171.150
exit
set vrouter "trust-vr"
set router-id 192.168.1.1
set source-routing enable
unset add-default-route
set route source in-interface ethernet0/0 192.168.1.0/24 interface ethernet0/2 gateway
192.168.0.201 preference 20 permanent
exit
set vrouter "untrust-vr"
exit
set vrouter "trust-vr"
exit
```

ANEXO B

B.1 JXplore.

JXplorer es una aplicación Java de código abierto que permite navegar y buscar cualquier directorio LDAP. Esa herramienta permite detallar la estructura de los datos del directorio como una vista de árbol en el panel izquierdo, y los datos de una entrada particular en el directorio en el panel de la derecha.

JXplorer incluye una serie de funciones de utilidad relacionados con directorios, tales como conectividad segura SSL, lectura y escritura de archivos LDIF, copiar, pegar y borrar, entre otros. JXplorer se puede ejecutar en Windows, Solaris, Linux, y Macintosh OSX.

Entre las ventajas que podemos encontrar en esta herramienta, nombraremos algunas de ellas:

- ✓ Podemos conectarnos a cualquier directorio que admite LDAP y navegar, buscar y modificar el directorio.
- ✓ Leer el esquema del directorio directamente, en lugar de confiar en el esquema archivos de configuración.
- ✓ Visualizar, cortar, pegar, y editar el contenido del directorio de los árboles, incluyendo arrastrar y pegar en plataformas Windows.
- ✓ Importación y exportación de archivos LDIF de un directorio e incluso verlos sin conexión.
- ✓ Permite configurar el explorador de muchas maneras, incluyendo su apariencia y la información para realizar looging o inicio de sesión. Por ejemplo, se puede configurar el aspecto del navegador a un estándar de la compañía mediante el uso de iconos específicos de la empresa para el directorio y gráficas de la compañía dentro de las plantillas HTML.
- ✓ Tiene como opción llenar los formularios de información de los usuarios como de los de la organización en plantillas HTML configurables mediante un sencillo extensión del lenguaje HTML.
- ✓ Se ejecuta en una amplia variedad de sistemas operativos, ya que JXplorer está escrito en el lenguaje de programación Java.
- ✓ Implementa SSL para comunicarse de forma segura, y SASL para la seguridad basada en certificados.

ANEXO C.

C.1 Configuración del archivo slapd.conf.

```
# BDB Backend configuration file
# See slapd.conf(5) for details on configuration options.
```

```
# This file should NOT be world readable.
ucdata-path     ./ucdata
include         ./schema/core.schema
include         ./schema/cosine.schema
include         ./schema/nis.schema
include         ./schema/inetorgperson.schema
include         ./schema/openldap.schema
include         ./schema/dyngroup.schema


pidfile         ./run/slapd.pid
argsfile        ./run/slapd.args


# Enable TLS if port is defined for ldaps


TLSVerifyClient never
TLSCipherSuite HIGH:MEDIUM:-SSLv2
TLSCertificateFile ./secure/certs/server.pem
TLSCertificateKeyFile ./secure/certs/server.pem
TLSCACertificateFile ./secure/certs/server.pem

allow bind_v2

########################################################################
# bdb database definitions
########################################################################


database        bdb
suffix          "dc=maxcrc,dc=com"
rootdn          "cn=Manager,dc=maxcrc,dc=com"
# Cleartext passwords, especially for the rootdn, should
# be avoid.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
rootpw    {SSHA}dIwc2LiDOjpNzpqA/5zbmUR7oK/FvOnq

# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd and slap tools.
# Mode 700 recommended.
directory ./data
dirtyread
searchstack 20
# Indices to maintain
index mail pres,eq
index objectclass pres
```

index default eq,sub
index sn eq,sub,subinitial
index telephonenumber
index cn

ANEXO D.

D.1 Vista del Panel frontal del dispositivo de red Juniper SG 550.

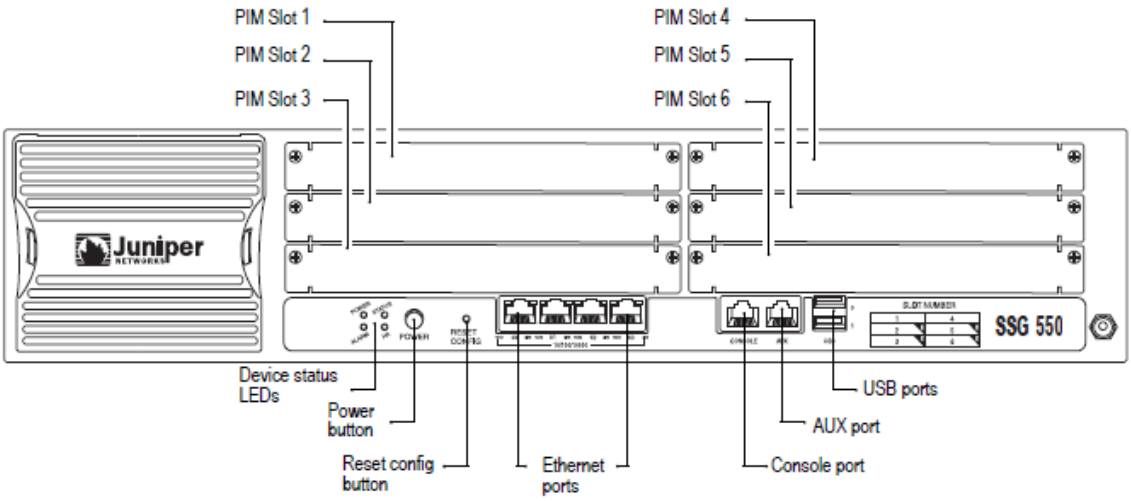


Figura 64. Descripción del Panel frontal del firewall Juniper SSG 550.

Fuente: Juniper.