UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN ELECTRÓNICA
Y REDES DE COMUNICACIÓN

TEMA:

MÉTODOS Y PROTOCOLOS PARA LA NEGOCIACIÓN SEGURA DE CLAVE DE CIFRADO UTILIZANDO IKE EN UNA VPN Y CONTROL DE TRÁFICO EN LA UNIVERSIDAD TÉCNICA DEL NORTE

TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y REDES DE COMUNICACIÓN

AUTOR: RONALD LUIS MENA MARTÍNEZ

DIRECTOR: MSc. EDGAR MAYA

Ibarra – Ecuador

2017

UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN

A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La Universidad Técnica del Norte dentro del proyecto Repositorio Digital Institucional, determinó la necesidad de disponer de textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la Universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información.

DATOS DEL CONTACTO	
Cédula de identidad	1003340278
Apellidos y Nombres	Mena Martínez Ronald Luis
Dirección	Sucre 313 y Borrero
E-mail	ron90mena@yahoo.es
Teléfono fijo	062-606-050
Teléfono móvil	0997432626
DATOS DE LA OBRA	
Título	"MÉTODOS Y PROTOCOLOS PARA LA NEGOCIACIÓN SEGURA DE CLAVE DE CIFRADO UTILIZANDO IKE EN UNA VPN Y CONTROL DE TRAFICO EN LA UNIVERSIDAD TÉCNICA DEL NORTE"
Autor	Ronald Luis Mena Martínez
Fecha	Julio del 2016
Programa	Pregrado
Título	Ingeniero en Electrónica y Redes de Comunicación
Director	MSc. Edgar Maya

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Ronald Luis Mena Martínez, con cédula de identidad Nro. 100334027-8, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad de material y como apoyo a la educación, investigación y extensión, en concordancia con la ley de Educación Superior Artículo 144.

3. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrollo, sin violar los derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que se asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Firma:……

Nombre: Ronald Mena

Cédula: 100334027-8

Ibarra, Julio del 2016

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Yo, Ronald Luis Mena Martínez, con cédula de identidad número 100334027-8 manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador artículos 4, 5 y 6, en calidad de la autora del trabajo de grado con el tema: MÉTODOS Y PROTOCOLOS PARA LA NEGOCIACIÓN SEGURA DE CLAVE DE CIFRADO UTILIZANDO IKE EN UNA VPN Y CONTROL DE TRAFICO EN LA UNIVERSIDAD TÉCNICA DEL NORTE.
Que ha sido desarrollado con el propósito de obtener el título de Ingeniera en Electrónica y Redes de Comunicación de la Universidad Técnica del Norte, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia suscribo en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Técnica del Norte.

Ronald Luis Mena Martínez
100334027-8
Ibarra, julio del 2016

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

MSc. EDGAR MAYA, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN CERTIFICA

Que, el presente Trabajo de Titulación: "MÉTODOS Y PROTOCOLOS PARA LA NEGOCIACIÓN SEGURA DE CLAVE DE CIFRADO UTILIZANDO IKE EN UNA VPN Y CONTROL DE TRÁFICO EN LA UNIVERSIDAD TÉCNICA DEL NORTE"
Ha sido desarrollado por el Señor Ronald Luis Mena Martínez bajo mi supervisión.

Es todo en cuanto puedo certificar en honor a la verdad.

MSc. Edgar Maya
DIRECTOR

AGRADECIMIENTO

Agradeciendo primeramente al rey de reyes y señor de señores el único que nunca nos abandona y siempre está con nosotros mi sabio Dios “Jehová”, por este logro en mi vida, ya que sin él nada de esto hubiese sido posible; toda la gloria y la honra se ha para Dios.

Agradezco a mis padres por su apoyo incondicional y por el amor y la educación que me han brindado siempre, a mi madrecita que siempre ha estado conmigo en todo momento ayudándome en todo lo que necesite, a mi hermano que siempre ha estado apoyándome brindándome su apoyo en todo lo que necesite y en todo momento.

A mi novia Yessenia F. por su apoyo incondicional y ser mi ayuda idónea en todo momento y por ser fuente de alegría en mi vida.

Agradezco a todos los Ingenieros por su enseñanza y educación que fueron un instrumento valioso en la formación de mi educación y un pilar de enseñanza, a mi tutor de tesis al MSc Edgar Maya por su apoyo y enseñanza, a los Ingenieros miembros de mi tribunal, al Ingeniero Vinicio Guerra por su apoyo y ayuda.

Ronald Mena

DEDICATORIA

Dedico este presente documento al creador de los cielos, al único y justo que es Dios, que siempre está con nosotros en todo momento.

También para mis padres por todo lo que me han brindado y me han educado por el camino correcto, por todos los consejos sabios que han brindado y por todas las experiencias que me han enseñado a ver que la vida no es fácil pero se la puede llenar de amor y felicidad si uno se lo propone, dedico también mi hermano que ha sido mi mano derecha que ha estado hay siempre apoyándome en mi vida.

Agradezco en especial a mi madrecita María Elena por todo su apoyo y en especial por su inmenso amor y comprensión en todo momento que ha sido mi ayuda incondicionalmente, que ha estado hay en todo momento apoyándome y cuidándome todos los días mi vida, muchas gracias mamá.

Ronald Mena

RESUMEN

En el documento se detalla una Red Privada Virtual basado en software libre a través de GNU/LINUX con el sistema operativo CentOS para solventar la demanda de una buena comunicación y poder solucionar los problemas de conexión entre el Antiguo Hospital San Vicente de Paul (AHSVP) y el Edificio Central de la Universidad Técnica del Norte, a través de un túnel por donde circula el tráfico de red el cual garantizará una mejor conexión y sobre todo una comunicación segura de tal manera que cuente con servicios, métodos y protocolos de seguridad para una buena comunicación sin ataques de autenticación o pérdidas de la información.

En la VPN, cuenta con dos servidores que establecen la conexión a través de políticas de firewall para habilitar el túnel VPN, obteniendo un mejor control de los servicios de red y dando como resultado una conexión virtual segura sin altos gastos económicos como son las líneas dedicadas.

El túnel cuenta con seguridad IPSec que garantiza la conexión y así contar con un protocolo de seguridad denominado Internet Key Exchange (IKE), para la negociación y las asociaciones del protocolo IPSec, ya que por el túnel transitará el tráfico que avala un beneficio óptimo de conexión entre el AHSVP y el Edificio Central.

ABSTRACT

In this document a Virtual Private Network based on free software by GNU / Linux with the CentOS operating system to address the demand for good communication and to troubleshoot connection between the Old Hospital and the Central Building is presented Technical University of the North, through a tunnel through which circulate the network traffic which would ensure a better connection and especially secure communication so that services are provided, methods and security protocols for secure communication without attacks authentication or loss of information.

This tunnel has IPSec security guaranteeing the connection and to have a secure protocol such as Internet Key Exchange (IKE) for negotiation and IPSec security associations for this tunnel pass traffic ensuring optimal benefit of connection between these two stations work.

Account both servers with firewall policies to enable the VPN tunnel and have better control of network services and achieve a secure virtual connection and especially without high economic costs such as leased lines.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO I

1. Introducción

1.1 Problema

La Universidad Técnica del Norte posee una red de enlace vía microonda entre el edificio Central UTN y el Antigua Hospital San Vicente de Paúl (AHSVP) que actualmente no cuenta con la suficiente carga para solventar las demandas de comunicación y la solución de los problemas de conexión entre estas.

Las conexiones de comunicación del Edificio Central UTN y el AHSVP no puede tener un funcionamiento óptimo debido a que no existe un diseño o una implementación que garantice los recursos, como son una buena comunicación segura libre de intrusos, además que permita obtener un control de Tráfico a través de un túnel VPN proporcionando una mejor conexión entre los servicios prestados que ofrece este túnel.

Al no existir un diseño de red segura entre el AHSVP y el edifico Central, se programa crear una VPN que garantice la conexión a través de un protocolo seguro como Internet Key Exchange (IKE) para la negociación de las asociaciones de seguridad IPSec (SA). Este proceso requiere que los sistemas de IPSec primero deben autenticarse entre sí y establecer ISAKMP (IKE), por medio de claves compartidas a través del túnel VPN, permitiendo establecer soluciones viables.

1.2 Objetivos

1.2.1 Objetivo General

Garantizar la confiabilidad a través de autenticación y negociación de las redes entre la UTN, el Hospital Viejo y los recursos de red controlando el tráfico.

1.2.2 Objetivos Específicos

- Obtener parámetros e intercambio seguro de claves para la comunicación de redes de comunicación.
- Establecer un canal seguro usando el método de Diffie-Hellman para definir una clave secreta de sesión.
- Establecer una Asociación de Seguridad (SA), en protocolo IPSEC.
- Controlar los diferentes tipos de tráfico con la utilización de la Red Privada Virtual VPN, para obtener un control de tráfico de la red.

1.3 Alcance

En la Universidad Técnica del Norte se implementará un conjunto de métodos y protocolos; asegurando la confidencialidad e integridad del flujo de datos y una verificación de la comunicación por parte de los servidores ubicados en el Edifico Central y el AHSVP.

Se implementará una solución a la deficiencia de la conexión para brindar autenticación segura por medio de los servicios otorgados por el protocolo IKE (Internet Key Exchange), en la Red Privada Virtual VPN que se plantea desarrollar para controlar el tipo de tráfico

dentro del Túnel VPN manipulando los servicios y aplicaciones de la Universidad Técnica del Norte.

Se establecerá un canal seguro libre de intrusos mediante los métodos y conjuntos de protocolos, usando el Protocolo IPSEC y algoritmos de clave segura denominados Diffie-Hellman que se caracterizan por su alta compatibilidad hacia el desarrollo de una VPN. De esta manera que garantizará una conexión segura entre los puntos de acceso UTN-Hospital.

Dentro del manejo seguro de la comunicación establecida en el túnel de la VPN UTN-Hospital el protocolo IKE es una alternativa al intercambio manual de claves parte fundamental del desarrollo, donde su objetivo es la negociación de una Asociación de Seguridad para el complemento de IPSEC.

Al implementarse se proporcionará una mejora en cuanto al tráfico de la red de la Universidad Técnica del Norte, ya que se podrá controlar el flujo de la información que se maneja en la VPN y recorra por el túnel de comunicación entre las dos estaciones.

1.4 Justificación

A través de la implementación de métodos de seguridad por medio de protocolos que permita mejorar el tráfico de la red de la Universidad Técnica del Norte permitirán obtener un mejoramiento entre la conexión de la UTN y el AHSVP, garantizando el rendimiento de la red.

Este conjunto de protocolos y métodos asegura además la verificación de los extremos de la comunicación obteniendo un beneficio en el ámbito de la seguridad y logrando obtener una red de comunicaciones libre de intrusos y de usuarios que no pertenezcan a la red virtual. Contar de así con este tipo de seguridad de la información en las Redes Privadas Virtuales (VPN) e integración de los datos para la Universidad Técnica del Norte, con mejor prestación de los recursos de la Red.

Al implementar los métodos de seguridad que abarcan el manejo del protocolo Internet key exchange (IKE), es un indicador de mejoramiento en la seguridad, control y administración de los recursos de la red ya que emplea un intercambio secreto de claves de tipo Diffie-Hellman para establecer cifrado la sesión compartida.

1.5 Antecedentes

En el ámbito del manejo de la información para proveer un mejor servicio en base a la seguridad y control de los recursos de la red se ha planteado el uso de diferentes medios, métodos o protocolos que aporten a la manipulación control y desempeño de la red, ya que la información o datos transmitidos son de alta relevancia, para mantenerlos vulnerables o de ser el caso perder la información.

En la actualidad se ha desarrollado múltiples procesos como es el caso de los siguientes temas a detallar como pautas del Ingeniero Cosme MacArthur Ortega B, quien optó emplear una metodología para la implementación de Redes Privadas Virtuales, donde se menciona las Redes Privadas Virtuales tecnología que nos permitirá conectar redes

distantes geográficamente, de manera segura y a bajos costos, utilizando redes públicas como medio de enlace o transmisión.

Aspectos de relevancia tomados del proyecto como son el analizar los diferentes protocolos de túnel que se pueden utilizar para la implementación de VPN, entre los que tenemos PPTP (Point to Point Tunneling Protocol), el protocolo L2F (Layer Two Forwarding) y el protocolo L2TP (Layer Two Tunneling Protocol) protocolo que reemplazo al L2F, el protocolo PPP y el IPSec, que abarcan el protocolo IKE que son empleados en diferentes tipos de redes.

Según el análisis del uso de protocolos de seguridad ah brindado un gran aporte al manejos de las redes en diversas instituciones, dentro del Ecuador y el de emplear VPN con seguridad aportando al desarrollo y manipulación de la información e incluso aportando con la seguridad a través de protocolos que resguardan la información.

Según el proceso de desarrollo que ha sufrido el manejo de la información, el interés en proteger y mejorar ha incursionado en áreas amplias de seguridad y extender campos hasta llegar al punto de almacenar la información en el Cloud.

Por lo tanto, a través de los estudios realizados el uso o implementación de una Red Privada Virtual, permite garantizar la conexión entre dos estaciones con métodos, protocolos seguros de autenticación sumando a este proceso el uso de IKE para garantizar el control de tráfico que va a circular desde el túnel VPN.

Dando lugar al planteamiento del proyecto en función del uso de un túnel VPN que se encarga de llevar todo este tráfico que pertenezca a la Universidad Técnica Del Norte para garantizar una mejor conexión y de prestación de este tipo de servicio.

CAPÍTULO II

2. Introducción de la Red Privada Virtual (VPN)

2.1 Definición

Las Redes Privadas Virtuales surgen debido a que las redes privadas son muy costosas a comparación con las redes públicas y es por ello que es una gran idea montar una red privada en una red pública.

El nacimiento de las redes privadas virtuales, han ofrecido muchas ventajas en el ámbito de las telecomunicaciones, además de la reducción de costos de instalación para las compañías, empresas y el mantenimiento de una forma significativa. (Morales, Redes Privadas Virtuales, 2010)

Una Red Privada Virtual se la define de la siguiente manera:

Una Red Privada Virtual (VPN, Virtual Private Network) es una red que usa la infraestructura de una red pública para poder transmitir información.

Se las denomina privadas debido a que se establece entre el emisor y el receptor, y virtuales porque no es necesario de un medio físico entre la comunicación.

Estas redes VPN extienden la red de una empresa a sus oficinas distantes, podría ser el caso que en lugar de alquilar líneas dedicadas a un costo muy elevado, se utiliza el internet.

Al usar las VPN, se crea una conexión privada, segura a través de las redes públicas como internet. De esta manera los usuarios pueden realizar la llamada local a internet y no usar llamadas de larga distancia. (Morales, Redes Privadas Virtuales, 2010)

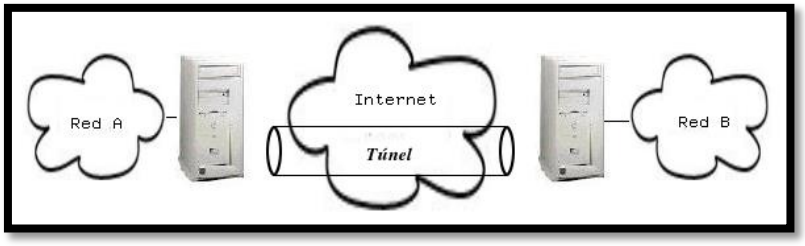


Figura 1. Esquema de una VPN a través de Internet
Fuente: (Crespo, 2015)

El esquema de una VPN a través de Internet como se muestra en la figura 1, utiliza un medio de acceso público que permite la interconexión de las dos redes *A* y *B* de forma que se unan en una única red *virtual* cuyo acceso siga siendo restringido.

2.2 Componentes de una VPN

Los componentes que tiene una VPN son:

- Servidor VPN
- Cliente VPN
- Túnel
- Conexión VPN
- Red pública

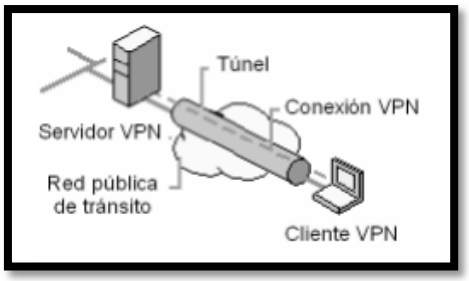


Figura 2. Componente de una VPN
Fuente: (Morales, 2011)

Para que exista un vínculo punto a punto en una VPN, los datos deben ser encapsulados con un encabezado que contenga la información de enrutamiento y que los datos puedan recorrer la red pública hasta alcanzar a su destino. Para realizar un vínculo privado, se cifran para asegurar la confidencialidad. Gracias a esto los paquetes interceptados en la red pública no se pueden descifrar si no se dispone de las claves para descifrarlas. Cuando los datos privados se encapsulan se las denomina túnel. La parte de la conexión en la que se encapsulan y se cifran los datos en la red privada se llama VPN, que se observa en la figura 2. (Ñacato, 2007)

2.3 Características de las VPN

Las características de una VPN más relevante es su integridad, confidencialidad y por supuesto la seguridad que propone para encriptar los datos.

Además del bajo costo, sencilla de usar, control de acceso basado en políticas de seguridad, brinda acceso remoto, los algoritmos de compresión que optimizan el tráfico para los clientes.

2.4 Funcionamiento de una VPN

Básicamente el funcionamiento de una VPN es la de encriptar los datos para que no estén vulnerables en la red pública o Internet, cuenta con un firewall lo que realiza la función de protector para los intrusos, luego los datos llegan al Internet donde se genera un túnel dedicado especialmente para nuestros datos, lo que hace este túnel es de que los datos viajen a una velocidad y un ancho de banda garantizado y lleguen al destino remoto. Estas redes VPN se pueden enlazar a otras redes, como son la de usuarios móviles a través de protocolos como Internet, IPSec, ATM, Frame Relay. (GOUJON, 2012)

2.5 Arquitectura de una VPN

Actualmente, dentro de la arquitectura de una VPN, las más empleadas y recomendadas por sus características de rendimiento, seguridad y control son las de Acceso Remoto y las de Sitio a Sitio.

Las redes VPN de acceso remoto se las divide en extranet e intranet y las redes de sitio a sitio se dividen en Dial-up y directas.

2.5.1 VPN de acceso remoto

Consiste en usuarios que se conectan con la empresa desde sitios remotos utilizando la Internet, servicio que proporciona una conexión segura a la red pudiendo acceder a los servicios o recursos que disponga la empresa.

Este servicio es proporcionado mediante el establecimiento de una VPN (red privada virtual) y requiere que el usuario disponga de una conexión a Internet en su ubicación remota.

Como se muestra en la figura 3 el usuario remoto accede a la VPN a través del túnel VPN.

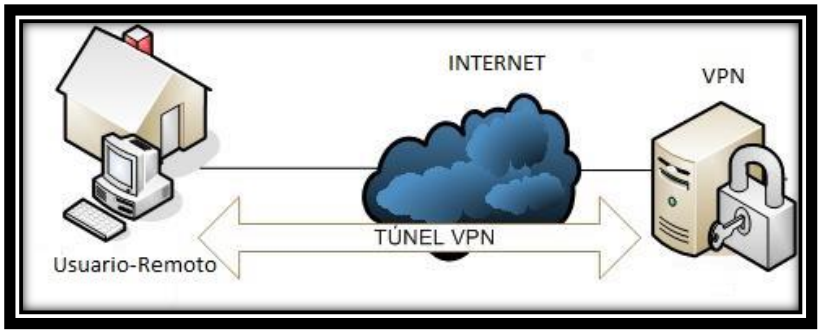


Figura 3. Topología de Acceso Remoto-VPN
Referencia: Elaboración propia

2.5.2 VPN de sitio a sitio

Estas conectan oficinas remotas con la sede central de la organización. El servidor VPN, que posee un vínculo permanente a Internet, acepta las conexiones vía Internet provenientes de los sitios y establece el túnel VPN. (Alejandro, 2012)

Conectan la red de una sucursal a la red de la oficina central de una empresa. En el pasado, se requería una conexión de línea arrendada o de Frame Relay para conectar sitios, pero dado que en la actualidad la mayoría de las empresas tienen acceso a Internet, estas conexiones se pueden reemplazar por VPN de sitio.

En la figura 4 se muestra una conexión donde los terminales de red conocen la configuración VPN con anticipación, para que usuarios que tengan permisos o las claves por parte de los dispositivos puedan ingresar sin ningún problema.

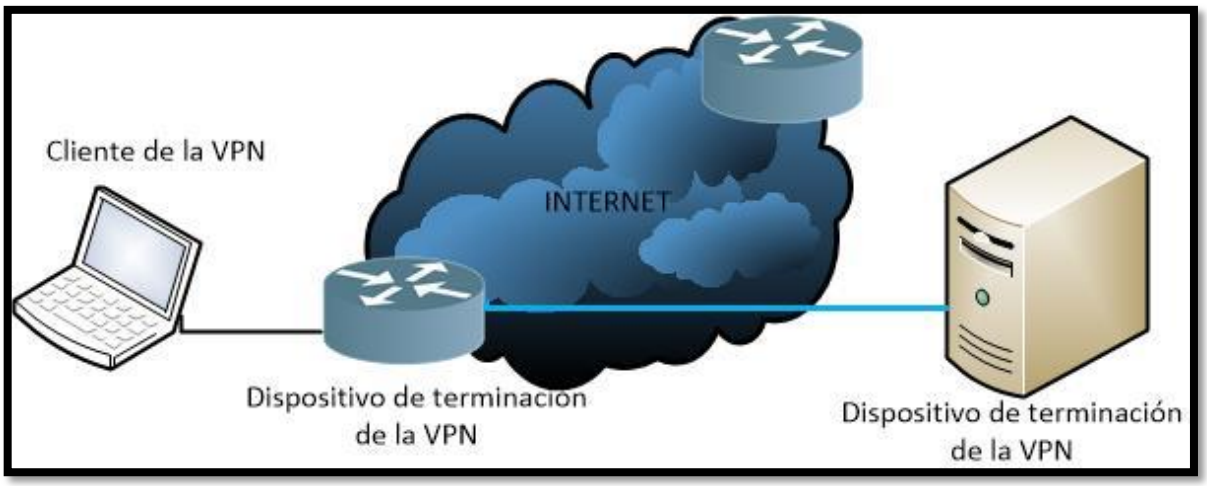


Figura 4. Topología de una VPN sitio a sitio
Referencia: Elaboración propia

2.5.3 VPN interna WLAN

Es una variante del tipo acceso remoto, lo que cambia es utilizar Internet como medio de conexión, ya que emplea la misma red de área local LAN de la empresa.

Como se muestra en la figura 5 el dispositivo o el cliente accede a la VPN que se encuentra en la misma area local, es decir en la misma red o en la misma ubicación.

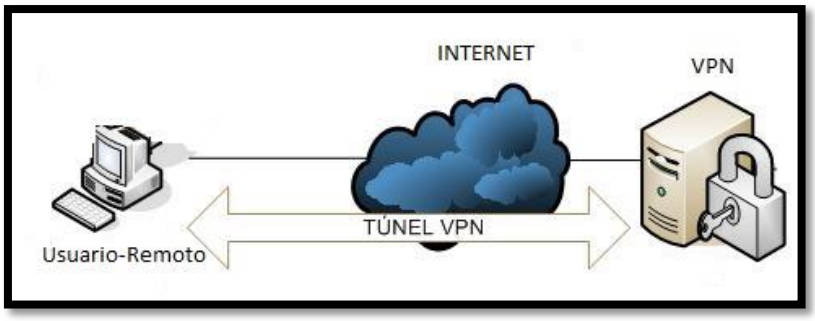


Figura 5. Topología de una VPN sitio a sitio
Referencia: Elaboración propia

2.6 Tipos de conexión VPN

A continuación se presentaran los tipos de conexión que existen para realizar una VPN.

2.6.1 VPN con firewall

Una VPN con firewall es muy común en la actualidad, las organizaciones que se encuentran a una conexión vía a Internet utilizan un tipo de firewall o algún cortafuegos, de esta manera obtener una VPN de mejor seguridad lo que ayuda a tener una VPN con una mejor implementación de seguridad. (Enders, 2012)

La variedad de proveedores donde se puede elegir un servicio en función de una VPN basada en un firewall, es amplia y está disponible para trabajar en cualquier plataforma.

El sistema operativo, donde se va a desarrollar la VPN es un aspecto importante porque es donde se va a montar el firewall, ya que de lo contrario los problemas y las

vulnerabilidades se verán presentes en el desarrollando la VPN, cabe mencionar que ningún dispositivo va estar cien por ciento seguro. (Enders, 2012)

Al configurar una VPN con firewall se debe considerar el protocolo con el cual se va a configurar la VPN se puede utilizar PPTP, L2TP, IPSec.

En el firewall a implementar en la VPN se ejecuta en los niveles dos y tres del modelo de referencia OSI, para el proxy es ejecutado en el nivel siete del modelo de referencia OSI y el filtrado examina el paquete.

La tecnología en función de VPN se ejecuta en los niveles más bajos de la pila de OSI, el cortafuego también debe hacerlo, o de lo contrario podría caer en problemas de desempeño. (Enders, 2012)

2.6.2 VPN de acceso remoto

Este acceso remoto se refiere que una persona de afuera está tratando de ingresar a la VPN, lo que significa que envía un flujo de datos. Debido a esto el túnel creado por la VPN puede venir de la Internet o de la línea de marcación.

En la figura 6 los usuarios intentan conectarse a una maquina remota la que desea establecer una conexión a través del túnel cifrado hacia el servidor interno de la red o desde una línea de acceso por marcación al servidor de autenticación. (Tanenbaum, Redes de Computadoras, 2009)

Lo que proporciona este tipo de autenticación por acceso remoto es que reduce considerablemente los costos, evita que se contrate las costosas líneas rentadas dedicadas y las de marcación remota.

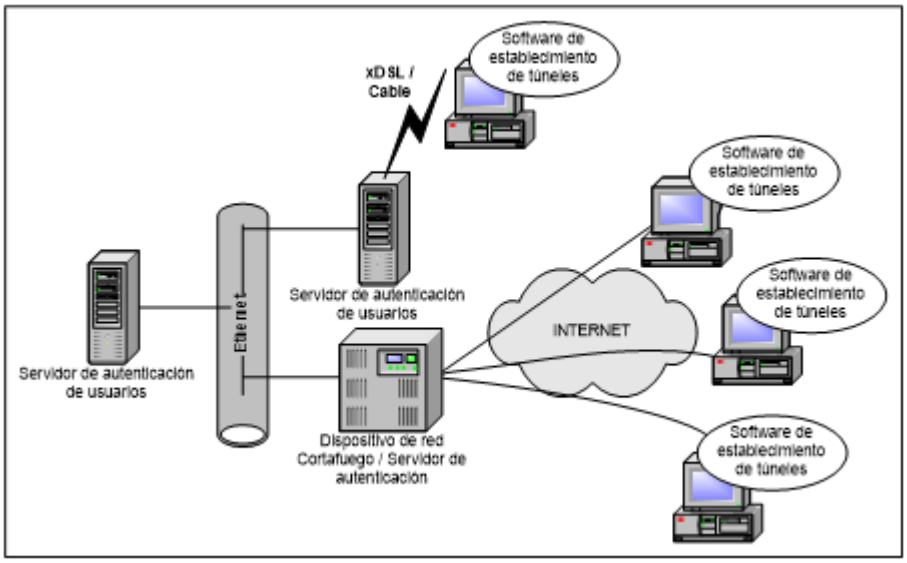


Figura 6. Acceso Remoto
Fuente: (Tanenbaum, Redes de Computadoras, 2009)

2.6.3 VPN de router a router

Las empresas que están en el ámbito de las redes de comunicación tales como 3COM, INTEL, Cisco y demás empresas de redes de comunicación ofrecen servicios para crear las VPN. Tanto el router como el concentrador soportan y se encuentran diseñadas para crear una VPN las cuales pueden ser sitio a sitio o las redes privadas virtuales de acceso remoto. (Microsoft, 2016)

Es importante ya que posee los métodos de seguridad como son las de cifrado, de autenticación para la correcta y segura transmisión de los datos.

2.7 Requerimientos de una VPN

Dentro de una VPN se requieren de requerimientos para que una VPN tenga confidencialidad y una seguridad de datos, una Red Privada Virtual debe de contener lo siguiente:

- ✓ Autenticación de usuarios
- ✓ Control de Acceso
- ✓ Administración de direcciones
- ✓ Encriptación de datos
- ✓ Administración de claves
- ✓ Ancho de banda

2.7.1 Autenticación de usuarios

Lo principal de esta autenticación es que solo usuarios ingresados o autorizados pueden tener acceso a la VPN de tal manera que verifica la identidad de los usuarios que deseen tener acceso a la VPN.

La autenticación es algo muy importante que da el acceso a la VPN por medio de claves públicas Public Key Infraestructure (PKI), el cual es un sistema de autenticación por medio de certificados, de esta manera que cada usuario se autentica de tal manera para poder intercambiar las claves públicas y ser garantizado por una autoridad de certificación. (Microsoft, 2016)

2.7.2 Control de Acceso

Este control se define como un régimen de técnicas para acceder a la red y de esta manera validar estas técnicas de control de acceso, que solo personal autorizado que cuenten con las pólizas de autenticación puedan entrar a la red.

Las VPN se ponen en funcionamiento para poder obtener el inicio de una sesión, permitir el uso a los usuarios de los recursos que puede tener la red o impedir los recursos y hasta terminar la sesión.

Los métodos y el conjunto de normas y reglas para ingresar a la VPN o disponer de un acceso a la red se la denomina una póliza de Control de Acceso, para este vínculo se puede requerir de un servidor denominado RADIUS el cual puede administrar el control de acceso basándose en la póliza de la red. (Aplicaciones Tecnologicas, 2013)

Una VPN realmente pretende es lograr de una manera económica, sencilla, fácil de instalar, administrar, es principalmente que tenga su acceso seguro a la red VPN y además permitir el uso o no de los diferentes recursos de la red.

Cabe destacar que con un buen sistema de protección de cifrado y autenticación pero sin un control de acceso los recursos de la red quedan desprotegidos para posibles ataques de usuarios no autorizados.

- Integridad del tráfico transmitido y evita usuarios no autorizados (VPN)
- Protege los recursos de la Red (Control de Acceso)

2.7.2.1 Administración de direcciones

El servidor donde se ubica la VPN asigna las direcciones a los clientes VPN y asegurarse que permanezca esa dirección privada, debido a que el Protocolo de Internet (IP) es un protocolo no confiable y vulnerable nada seguro, se debe ocultar la dirección privada dentro de una red pública. (Mario, 2013)

Para permitir este logro de ocultar la IP se debe manejar algún tipo de mecanismo, es por eso que aparece el Tunneling. La cualidad de este tipo es que oculta de tal manera que encapsula los datos con la dirección destino privada, dando seguridad a través de la red pública

2.7.2.2 Encriptación de datos

Esto hace mención a que los datos que viajan por la red pública, deben ser ilegibles para los usuarios que no se encuentren autorizados en la VPN. El motivo para que los datos se encuentren cifrados es para brindar seguridad de los mismos y evitar que usuarios no autorizados accedan a la información.

Las redes privadas virtuales necesitan de una cierta longitud para su clave de tal manera que hace imposible descifrar los datos a través de la VPN, que se encuentra encriptado a través de ciertos algoritmos para que resulte difícil encontrar la clave, existe un problema en el tamaño de la longitud de la clave debido a que si se es muy larga afecta el rendimiento

del procesador, en este caso se hace el uso de las claves simétricas y asimétricas. (Luz S. d., 2010)

El funcionamiento de las claves simétricas es bastante simple ya que se usa la misma clave tanto para cifrar como para descifrar, como se observa en la figura 7.

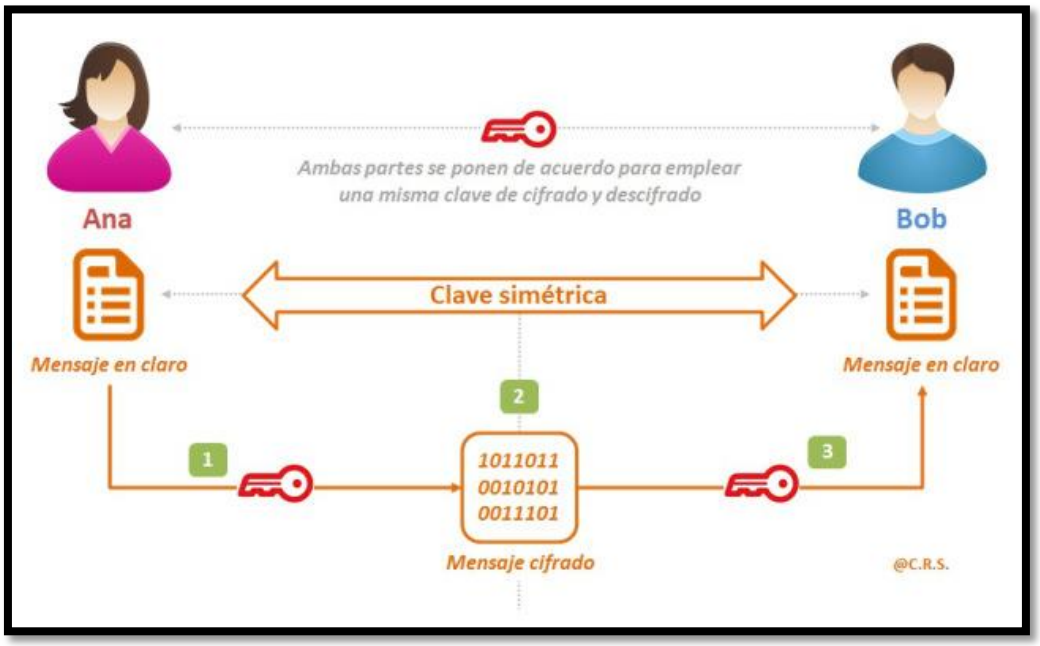


Figura 7. Clave simétrica
Fuente: **(Sandoval, 2012)**

El uso de la clave asimétrica es diferente ya que utiliza dos claves una para cifrar y otra para descifrar a diferencia de las claves simétricas que solo emplea una clave.

Una de las claves es conocida solamente por el administrador o por el usuario denominada clave privada, la otra clave es denominada clave pública ya que puede ser vista por todos los usuarios, que se detallada en la figura 8.

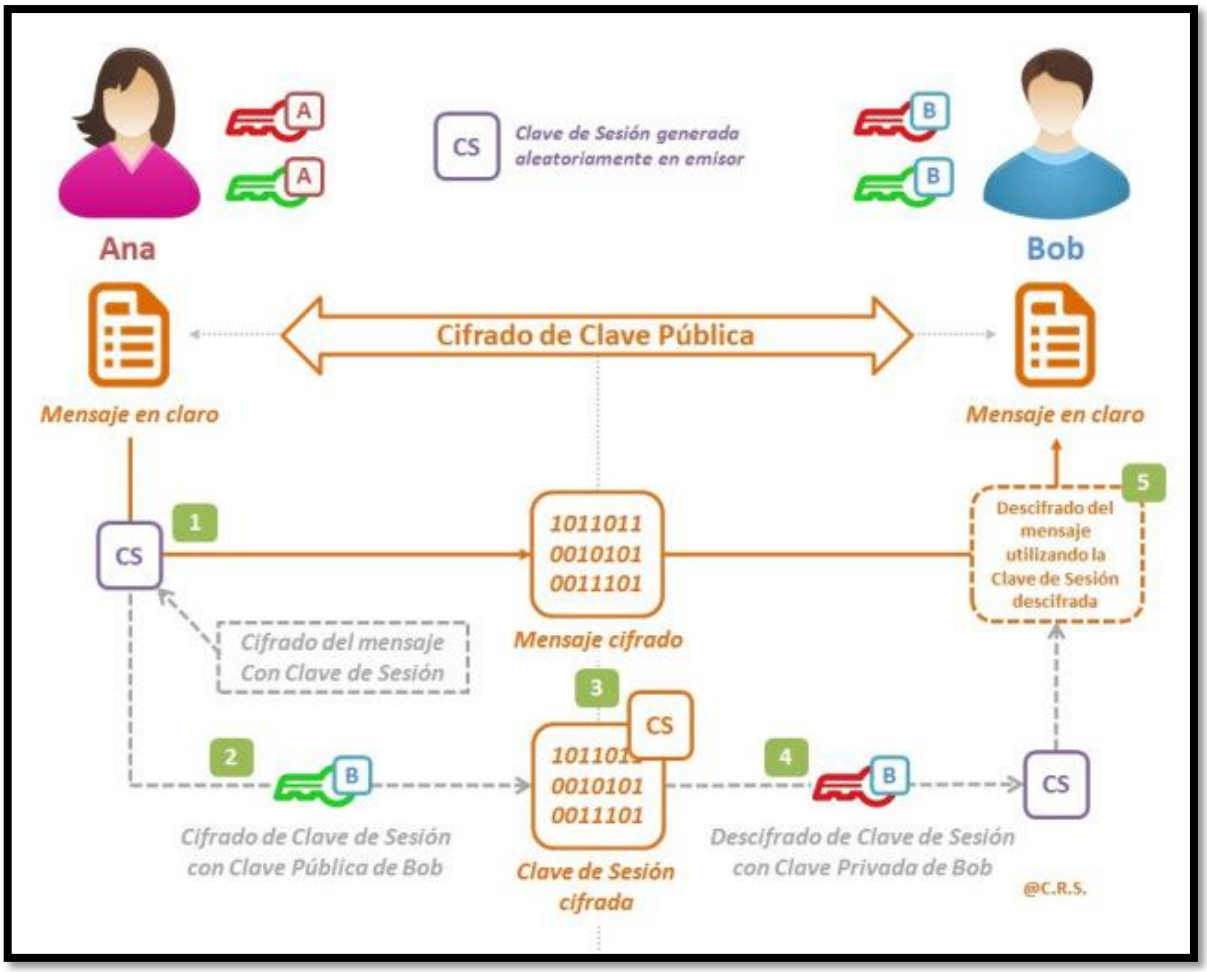


Figura 8. Claves simétricas
Fuente: (Sandoval, 2012)

2.7.2.3 Administración de claves

La admiración de claves es importante en una VPN para asegurar la integridad de la clave pública, la cual se publica con un certificado denominado Autoridad de Certificación (CA), lo cual el CA firma el certificado con la clave privada.

2.7.2.4 Ancho de banda

El ancho de banda en una VPN es para determinar principalmente que los datos fluyan de manera eficiente, lo que conlleva a que tenga una calidad de servicio (QoS), la cual es importante en una Red Privada Virtual.

La VPN deberá proporcionar formas para el control del ancho de banda.

2.7.3 VPN Tunneling

2.7.3.1 Definición

El Tunneling no es más que un camino o el trayecto para la transferencia de los datos de una red a otra, que encapsula las tramas en una cabecera adicional, lo cual este proporciona una información de enrutamiento para de esta manera los datos que fueron transferidos pueden viajar a través de la red intermedia. (Luz S. d., 2010)

Los protocolos que se pueden usarse para crear un Tunneling son los siguientes:

- DLSW (Data Link Switching)
- MPLS (Multi Protocol Label Swtching)
- PPTP (Point to Point Tunneling Protocol)
- L2TP (Layer 2 Tunneling Protocol)
- L2F (Layer 2 Forwarding)
- IPSec (Internet Protocol Security)

2.7.3.2 Funcionamiento

El Tunneling se basa en tres protocolos los cuales se detallan a continuación:

- **El protocolo del carrier**

Básicamente este protocolo es el que va transportando la información a través de la red.

- **En protocolo del encapsulamiento (empaquetamiento)**

Este protocolo es el que envuelve al paquete que se envió a través de la red y dependiendo del protocolo será más confiable, los cuales pueden ser GRE, L2F, PPTP, L2TP, IPSec, siendo este último el más seguro para la red.

- **El protocolo pasajero**

Es el protocolo del paquete de información se envía dentro del envoltorio, es decir, el paquete original de información. Los "protocolos pasajero" habituales son IPX, NetBeui e IP.

El Tunneling tiene aspectos como la de trasportar los paquetes que no se han o no soporten por la Internet, un ejemplo el de NetBeui que corresponde a Microsoft envuelto por un paquete IP que si puede ser enviado por la Internet. (Luz S. d., 2010)

El protocolo GRE (Generic Routing Encapsulation) para el túnel VPN de punto a punto, con la finalidad de poder pasar el paquete envuelto en un paquete IP para poder enviarlo por la Internet.

Para una mejor seguridad encapsulamiento seguro se usa el protocolo IPSec, tanto para las conexiones de usuario y de encriptar la información además autentica a los usuarios.

Para comprender mejor el proceso de Tunneling, realiza lo siguiente:

- Encapsulamiento
- Trasmite
- Desencapsulación

2.7.3.3 Protocolo internos del túnel de una VPN.

Protocolo pasajero.- Este representa el protocolo que debe encapsularse, los cuales pueden ser PPP, SLIP.

Protocolo encapsulador.- Estos pueden ser L2F, L2TP, PPTP, para la creación, mantenimiento y destrucción del túnel.

Protocolo portador.- Es el que se encarga el trasporte del encapsulamiento como lo es el protocolo IP que maneja amplias capacidades de direccionamiento.

2.7.4 Tunneling y VPN

En las VPN se usa la tunelización con el fin de manejar mecanismos seguros del trasporte de la información (datos) y de esta manera involucra la tunelización en tres áreas principales:

- Encapsulación
- Protección de direcciones privadas
- Integridad y confidencialidad de los datos

Los protocolos PPP y L2TP son usados para encapsular las tramas a nivel de capa de enlace de datos y los protocolos a nivel de capa de red como lo son IP sobre IP y el protocolo IPSec.

Existen mecanismos de integridad y de confidencialidad que garantizan que ningún usuario que no esté previamente autorizado sea capaz de poder ingresar y mucho menos de alterar los datos que se encuentran en el túnel VPN. (Luz S. d., 2010)

Además el Tunneling proporciona de una manera opcional proteger la integridad de la cabecera IP, cuando se usa IPSec existen los protocolos de autenticación como lo son AH y ESP los cuales proporcionan autenticación a los datos trasmitidos.

2.7.4.1 Tipos de Tunneling

Existen estos dos tipos de túneles, los cuales se detallaran a continuación:

2.7.4.2 Túnel voluntario

El túnel voluntario se produce cuando una estación de trabajo o un enrutador utilizan software de cliente de túnel para crear una conexión VPN con el servidor de túnel de destino. Un buen ejemplo de esto es el usuario de acceso telefónico a Internet que, para crear un túnel a través de Internet, debe llamar primero a un ISP. (Microsoft, 2016)

El túnel voluntario no es diferente de otros tipos de acceso a la red e IAS se puede utilizar para la autenticación, autorización y administración de cuentas.

2.7.4.3 Túnel obligatorio

Un servidor de acceso de marcación capaz de soportar una VPN configura y crea un túnel obligatorio. La computadora del usuario deja de ser un punto terminal del túnel. Otro dispositivo, el servidor de acceso remoto, entre la computadora del usuario y el servidor del túnel, es el punto terminal del túnel y actúa como el cliente del mismo. (Brown)

2.7.5 Seguridad de las VPN

Se detalla los aspectos relevantes en función de la seguridad de la VPN.

2.7.5.1 Necesidad de seguridad en una VPN

Se crearon las VPN con el motivo de que solo personal directamente autorizado tenga el acceso a las aplicaciones y servidores, no cualquier usuario puede obtener los datos de una manera sin que se autentique en la red, los datos deben permanecer seguros, de esta manera se obtiene la facilidad de administración, la configuración debe ser directa y el mantenimiento y actualización deben estar asegurados. (netdatanetworks, 2011)

La mejor seguridad para establecer una VPN es la del protocolo IPSec, la autenticación de los usuarios y la seguridad al encriptar los datos son muy seguros.

2.8 Tipos de amenazas a las redes VPN

Las amenazas pueden estructurarse de la siguiente manera:

Amenazas no estructuradas.- Pueden ser ocasionadas por personas que deseen hacer daño y las cuales puede ser gravemente afectada la red.

Amenazas estructuradas.- A diferencia de las anterior estas personas poseen un conocimiento sobre las redes de comunicaciones, con el propósito de hacer dinero, realizando robos de datos o de información.

Amenazas internas y externas.- Las amenazas ocasionadas internamente son causadas por parte del personal que se encuentran o tienen acceso a la red y está en ellos si desean

hacer daño a la red, y las externas son amenazas de usuarios que se encuentran fuera de la red, que no están autorizadas a la red.

2.8.1 Tipos de ataques a las redes

Existen varios ataques a las redes privadas virtuales las cuales pueden ser:

- Integridad de los datos
- Ataques de contraseña
- Sniffers
- DoS
- Spoofing
- Ataque de clave comprometida

Integridad de los datos.- Este ataque es muy grave debido a que si el maleante logra acceder a los datos en el transcurso de la transmisión puede ser demasiado perjudicial debido a que quedan expuestos los datos para modificarlos, alterarlos e incluso los datos pueden ser eliminados, sin que se entere el que trasmitió la información ni el receptor de que los datos han sido alterados. (Luz S. D., Redes@zone, 2010)

Ataques de contraseña.- Esto se da cuando las contraseñas o los nombres de usuarios no se encuentran encriptados al enviarlos a la red, y de esta manera quedan expuestos por intrusos y se pueden hacer pasar por legítimos usuarios.

Sniffers.- Este tipo de ataques de las denomina Sniffers debido a que personal no autorizado usan herramientas de software, como programas para descifrar las claves de los

usuarios, este programa capta los datos que se encuentran circulando por la red y de esta manera descifrar los password y los nombres de usuario; esto depende también si los datos se encuentran cifrados o no y de que algoritmo se use para cifrar los datos, haciendo de esta manera más difícil para el Sniffers descifrar los datos. (BUSTAMANTE, Diseño e Implementación de una infraestructura de servicios de red y resguardo de servidores, 2012)

DoS.- Este es un ataque de denegación de servicio, el cual el atacante tiene acceso a los servidores de dicha compañía y de esta manera denegar el servicio a los usuarios.

Spoofing.- Esta técnica se da por medio de las direcciones IP, usando IP falsas por medio de ciertos programas para de esta manera acceder a los datos y poder realizar daños a la empresa.

Ataque de clave comprometida.- Este modelo de ataque es cuando el intruso logra su cometido y conoce las claves de acceso y entra a la red, a esto se la denomina clave comprometida, ya que el intruso puede acceder a la información de manera tranquila pudiendo lograr grandes daños económicos a la empresa.

Seguridad en los datos.- La seguridad de los datos es una parte muy importante que se debe tomar en consideración, se debe proporcionar confidencialidad, integridad y autenticación.

2.8.2 Criptografía y criptoanálisis

La criptografía permite que la información en un determinado mensaje sea más fácil de entender para los usuarios que poseen la clave y que puedan acceder a la red. En cuanto al criptoanálisis trata de obtener la información sin la clave. De esta manera se acceda a una técnica criptográfica debe demanda mayor complejidad. (Luz S. D., Redes@zone, 2010)

2.8.2.1 Cifrado simétrico o de clave privada

Este modelo de seguridad de encriptación de los datos comparte una sola clave, la cual es la misma clave para tanto para cifrar como para descifrar los datos, como se observa en la figura 9.

Figura 9. Cifrado simétrico
Fuente: (Amazoni, 2013)

Este tipo de cifrado contiene ciertos elementos los cuales son:

- **Texto Nativo.-** Estos son los datos de origen que van a ser llevados a través e de red.

- **Algoritmo de cifrado.**- Este algoritmo hace varias trasformaciones al texto sin cifrar para de esta manera ser trasmitido por la red y de alguna manera poder tener seguridad de los datos, para que no se encuentren en texto plano.

- **Clave secreta.-** Esta es la clave que tiene el texto cifrado, para tener acceso a los datos.

- **Texto cifrado.-** Este es el texto que se encuentra cifrado por algún algoritmo de encriptación de datos.

- **Algoritmo de descifrado.-** Este es el proceso para descifrar los datos.

2.8.2.2 Algoritmos de cifrado simétrico

- **Algoritmo DES**

Este algoritmo es de 64 bits que emplea claves de 56 bits, consta de 16 rondas, más dos permutaciones, una que se aplica al principio y otra al final de esta manera la última es la inversa de la primera.- Su desventaja es que su longitud es demasiado corta, lo cual hace que con el avance actual de los ordenadores de hoy en día, los ataques pueden llegar a su acometido por parte de los hackers. (Luz S. D., Redes@zone, 2010)

- **DES Múltiple**

Este algoritmo consiste en replicar algunas veces el algoritmo DES con diferentes tipos de claves al mensaje, el algoritmo Triple DES, se emplea con mayor frecuencia.

Este algoritmo de Triple, lo que realiza es que codifica con una subclave 1, nuevamente codifica con una clave 2, y vuelve a codificar con la subclave 1, el resultado de este tipo de clave nos da una longitud de 112 bits. (Luz S. D., Redes@zone, 2010)

- **Idea (International Data Encryption Algorithm)**

Este algoritmo es de una longitud de 128 bits lo que resulta que codifica bloques de 64 bits, lo que resulta ser una fuente bastante fuerte para los usuarios no autorizados, emplea el mismo algoritmo de DES tanto para cifrar como para descifrar.

- **Algoritmo de Rijndael (AES)**

Es el sucesor de DES, AES es un sistema de cifrado por bloques, diseñado para manejar longitudes de clave y de bloque variables, ambas comprendidas entre los 128 y los 256 bits.

2.8.2.3 Cifrado asimétrico o de clave pública

Este cifrado asimétrico fue propuesto por DIffie Hellman en 1976, este tipo de cifrado es más robusto que el de cifrado por clave simétrica, en este tipo de cifrado existen dos tipos de claves una llamada clave pública que es conocida por todos, e incluso por los atacantes de la red, ya que esta clave pública viaja a través de la Internet, y otra clave que es la de clave privada que solo conoce el dueño y por ningún motivo debe darla a conocer. (Luz S. D., Redes@zone, 2010)

Los elementos de este tipo de cifrado son los mismos que los de la clave privada, a diferencia que este posee dos claves una publica y otra privada.

El modo de operación para cifrar la información es como se muestra en la figura 10.

Figura 10. Cifrado Asimétrico
Fuente: (Catalina Gaviria Carrillo, 2012)

- Cada usuario genera las dos claves tanto una pública como la privada.

- Los dos usuarios publican sus claves públicas para dar a conocer, de esta manera cada usuario sabe la clave pública del otro usuario.

- De esta manera si un usuario uno desea enviarle datos a un usuario dos, se cifra el mensaje con la clave pública del usuario dos.

- Para descifrar el mensaje el usuario dos lo descifra el mensaje con la clave privada del mismo usuario 2, ya que solo él sabe la clave para descifrarla, de esta manera es más seguro y confiable que las claves simétricas.

2.8.2.4 Funciones (hash)

Estas funciones son muy adecuadas para la autenticación de datos, de usuarios y para las firmas digitales, estas funciones realizan y verifican que el mensaje sea autentico y que no haya sido alterado.

Una función hash es método para generar claves o llaves que representen de manera casi unívoca a un documento o conjunto de datos. Es una operación matemática que se realiza sobre este conjunto de datos de cualquier longitud, y su salida es una huella digital, de tamaño fijo e independiente de la dimensión del documento original. El contenido es ilegible. (Luz S. D., Criptografía : Algoritmos de autenticación (hash), 2010)

En resumen estos algoritmos a diferencia del criptoanálisis simétrica y asimétrica, cumplen con la función de asegurar que los datos transmitidos en la red no se encuentren alterados por ningún usuario que no tenga ningún acceso legal a la red, de esta manera haciendo ilegible una contraseña o firmar digitalmente un documento.

2.8.2.5 Firmas digitales

Las firmas digitales usan las claves públicas, lo que hace una firma digital es que el documento presenta una codificación de seguridad, este proceso de firma digital realiza una trasformación y guarda algunos datos del autor en una firma.

Una firma electrónica posee el mismo valor que una firma manuscrita y cumplen con el mismo propósito, esta firma digital no puede repetirse en cada documento, debe ser

diferente en cada tipo de documento, de lo contrario podría ser usada y alterada en cualquier documento que tenga esta firma digital. (Luz S. D., Redes@zone, 2010)

Ventajas de la firma digital es que aumenta la seguridad, de esta manera se elimina el fraude por un impostor que desee firmar el documento. “Integridad del mensaje: Teniendo firma digital se demuestra la validez del documento. Aseguras al receptor que el documento es válido y libre de información falsa.” (Luz S. D., Redes@zone, 2010)

Existen requerimientos legales para usar una firma digital, para logara satisfacer documentos legales en el proceso de gestión.

Aplicaciones de las Firmas Digitales:

- Correo seguro
- Mensajes con autenticidad asegurada
- Contratos comerciales electrónicos
- Factura _ electrónica
- Desmaterialización de documentos
- Transacciones comerciales electrónicas
- Invitación electrónica
- Dinero electrónico
- Notificaciones judiciales electrónicas
- Voto electrónico
- Decretos ejecutivos (gobierno)
- Créditos de seguridad social
- Contratación pública
- Sellado de tiempo

2.8.3 Formas de autenticación de usuarios

Existen muchas formas de autenticar a los usuarios para de esta forma estar seguros de que no puedan ingresar intrusos en la red, todo esto se logra con un servidor remoto que identifica a los usuarios una vez que se hayan ingresado y que legalicen con sus claves de acceso quien dicen ser; para una Red Privada Virtual se recomienda usar el sistema de autenticación denominado PKI. (Galarza, 2013)

2.8.4 Autenticación basada en contraseña

La autenticación es la técnica por el cual se verifica que su compañero de comunicación sea quien debe ser y no es un impostor, la autenticación verifica la identidad de un proceso remoto para brindar mayor confiabilidad de acceso.

Cada método de autenticación dispone de sus ventajas y de contra parte en lo que se refiere a la seguridad, para ello se recomienda usar métodos de autenticación basado en certificados para todos los procesos de acceso a la red.

Es importante destacar que el procedimiento de autenticación determina que métodos de autenticación son compatibles, un método de acceso a la red que se puede instalar es un Servidor Radius.

Este método basado en contraseñas es fácil de implementar debido que el servidor aloja la contraseña y verifica con la clave de acceso del usuario, si coinciden las claves de acceso podrá entrar a la red, es recomendable que el usuario cambie de contraseñas para posibles

amenazas de robo de contraseña, esto lo hacen a través de Sniffers una persona que tiene conocimiento de redes de comunicación. (Dias, 2015)

2.8.5 Kerberos

Kerberos es un protocolo de autenticación mejorado de comparación con los demás protocolos, este protocolo guarda a los clientes y sus respectivas claves privadas en una base de datos, la clave privada es conocida sólo por Kerberos y al cliente que le pertenece, los servicios de red que requieren la autenticación se registran con el Kerberos, al igual que los clientes que desean utilizar esos servicios. (Henao, 2012)

Este protocolo pone a funcionamiento tres niveles de protección:

- El programador de la aplicación es el que asume y determina lo que apropiado para la aplicación.

- Kerberos proporciona los mensajes seguros, cuidando que el mensaje no este divulgado por la red.

- Este protocolo proporciona un alto nivel de seguridad para los mensajes, Kerberos realiza que cada mensaje se autentique y además se cifre.

2.8.5.1 Funcionamiento de Kerberos

Kerberos se basa en protocolo de distribución de claves de Needham y del Schroeder. Cuando las peticiones del usuario o un servicio, su identidad deben ser establecidas. Hay tres fases a la autenticación con el Kerberos:

- La primera fase es que el usuario obtiene la forma de entrar o las credenciales para tener acceso a los servicios que presenta la red.

- La segunda fase de Kerberos es la autenticación del usuario para un servicio determinado o especifico.

- En la tercera fase y fase final el usuario muestra al servidor de dichas credenciales.

Las credenciales de Kerberos son las siguientes:

La primera es la de boletos, que se basa en la encriptación de claves privadas, pero estas se cifran usando diferentes tipos de claves, estas se usan para tener la confianza y la seguridad de que un usuario se ha quien dice ser, además este boleto puede pasar información que el servidor puede pasar al usuario para asegurarse de quien dice ser en realidad. (Henao, 2012)

La segunda credencial es del autenticador, al igual que la credencial de boleto se basa en la criptografía de claves privadas, esta contiene una información adicional la cual es que prueba que el cliente que presenta el boleto es el mismo al cual el boleto fue publicado.

2.8.6 PKI

Esta infraestructura PKI no es más que una Clave pública que permite a una compañía o empresa que cuente con un sistema de seguridad, que se refiere a un buen control de acceso, confidencialidad, este protocolo usa la tecnología tal como lo es las firmas digitales

y los certificados digitales de tal manera es una fuente segura este sistema de claves pública (PKI). (Henao, 2012)

Este protocolo se basa sobre todo en la cifra RSA, describe los procesos para la gestión de certificados digitales en base a las claves públicas para el buen intercambio de información, ya que este protocolo hace posible que el usuario firme digitalmente un documento electrónico.

2.8.6.1 Elementos de un PKI.

PKI se basa en el cifrado de clave pública y está formada por:

- Certificados Digitales, por ejemplo X509.
- Una estructura jerárquica para la generación y verificado de estos certificados formada por las Agencias de Certificación (CA) y las Autoridades de Registro (RA).
- Directorios de certificados, que son el soporte software adecuado (bases de datos) para el almacenamiento de los certificados.
- Un sistema de administración de certificados, que es el programa que utiliza la Agencia de Certificación o la empresa donde se ha instalado la PKI para que realice la comprobación, la generación, la revocación de certificados.

2.8.7 Servidores Radius

Un servidor RADIUS gestiona el acceso a las redes. Se utiliza principalmente por los proveedores de servicios de Internet para gestionar acceso a Internet a sus clientes. El nombre RADIUS es en realidad un acrónimo de "Remote Authentication Dial In User Service" (Dial de autenticación remoto para acceso a servicios). El protocolo no sólo logra acceso a la red, sino también a la gestión de cuentas del usuario. (Cooper)

Las funciones de un servidor RADIUS se resumen con las siglas "AAA" que significan: Autenticación, Autorización y Anotación. Los hacedores de servidores no reciben conexiones directas de los clientes sino que interactúan con las aplicaciones del cliente en otros equipos de la red.

2.9 IPSEC

IPsec (abreviatura de Internet Protocol security) es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre el Protocolo de Internet (IP) autenticando y/o cifrando cada paquete IP en un flujo de datos. IPsec también incluye protocolos para el establecimiento de claves de cifrado.

IPsec proporciona varios servicios necesarios para que la comunicación sea segura, estos servicios son los de confidencialidad, integridad, y autenticación. Gracias a estos servicios, la seguridad de las comunicaciones está garantizada. (Luz S. D., Redes Zone, 2013)

2.9.1 Servicios que proporciona IPsec:

Confidencialidad: Requiere que la información sea accesible únicamente a las entidades autorizadas

Integridad: Incluye códigos detectores de errores y que la información no se vea modificada. IPsec permite al host receptor verificar que los campos de cabecera del datagrama y la carga útil cifrada no han sido modificados mientras el datagrama estaba en ruta hacia el destino.

Autenticación: El usuario es realmente quien dice ser. Cuando el host recibe un datagrama IPsec de un origen, el host está seguro de que la dirección IP de origen del datagrama es el origen real del mismo.

Permite el acceso remoto a ordenadores en distintos lugares como si estuviéramos en la misma red local (redes privadas virtuales). Gracias a esta característica podremos tener redes privadas comunicando diferentes sedes de empresas en Internet, sin necesidad de redes físicas privadas con el coste que estas redes contienen. (Luz S. D., Redes Zone, 2013)

Negociación del cifrado: Mecanismos que permiten a los dos host que están se están comunicando acordar las claves y algoritmos de cifrado.

Cuando dos host han establecido una sesión IPsec, los segmentos TCP y UDP enviados entre ellos son cifrados y autenticados. Por tanto, IPsec garantiza la seguridad de las comunicaciones.

2.9.2 AH

Esta cabecera proporciona autenticación e integridad a los datos transmitidos, para proporcionar esta característica IPsec hace uso de las huellas digitales HMAC, calculará funciones HASH al contenido del paquete IP, como SHA-1 o MD5 y una clave secreta compartida.

Esta cabecera no proporciona confidencialidad porque no está cifrada.

Esta cabecera se integra entre la cabecera IP y la carga útil, se puede transmitir mediante TCP o UDP.

2.9.3 ESP

ESP se lo denomina carga de seguridad encapsulada. Ofrece autenticación, integridad y confidencialidad de los datos transmitidos a través de IPsec. Para conseguir éstas características de seguridad, se hace un intercambio de llaves públicas (Algoritmos de cifrado asimétrico).

La función principal del protocolo ESP es proporcionar confidencialidad a los datos, para poder hacerlo, ESP define el cifrado y la forma en la que se ubicarán los datos en un nuevo datagrama IP. Para proporcionar autenticación e integridad, ESP usa mecanismos parecidos a AH. (Alejandro, 2012)

Los datos se pueden transmitir vía TCP, UDP o un datagrama IP completo (lo mismo ocurría con AH). La cabecera del paquete IP no está protegida por ESP (si utilizamos el modo túnel, la protección será a todo el paquete IP interno).

2.9.4 IKE

Este protocolo se utiliza para generar y administrar las claves necesarias para establecer las conexiones AH (Cabecera de autenticación) y ESP (Carga de Seguridad Encapsulada).

Esta configuración se podrá hacer de forma manual a ambos extremos del canal, o a través de un protocolo (el protocolo IKE) para que se encargue de la negociación automática de los participantes (SA = Asociación de Seguridad).

El protocolo IKE no sólo se encarga de la gestión y administración de las claves sino también del establecimiento de la conexión entre los participantes correspondientes. IKE no sólo está en IPsec sino que puede ser usado en los distintos algoritmos de enrutamiento como OSPF o RIP. (Almacen Informatico, 2016)

Fases de la negociación IKE

Establecimiento del canal seguro usando un algoritmo asimétrico de intercambio de claves como Diffie-Hellman para cifrar la comunicación IKE. Esta negociación se realiza mediante un único SA bidireccional. La autenticación puede ser mediante PSK (clave compartida) o con otros métodos como firmas digitales, o cifrados de clave pública.

Usando el canal seguro que se ha creado, se negociará la asociación de seguridad de IPsec (u otros servicios).

Algunas características de IKE

Compatibilidad con NAT transversal, aunque uno o los dos participantes estén detrás de una NAT, la conexión se podrá realizar.

Utilización de números de secuencia y ACK's para proporcionar confiabilidad, también incluye sistema de procesamiento de errores.

Resistente a ataques de denegación de servicio. IKE no realiza ninguna acción hasta que determina si el extremo que realiza la petición realmente existe, de esta forma se protege contra ataques desde direcciones IP faltas. (Corrales)

2.9.5 Diffie-Hellman

No es un algoritmo simétrico propiamente dicho, se usa para generar una clave privada simétrica a ambos extremos de un canal de comunicación inseguro. Se emplea para obtener

la clave secreta con la que posteriormente cifrar la información, junto con un algoritmo de cifrado simétrico.

Su seguridad radica en la dificultad de calcular el logaritmos discreto de números grandes.

CAPÍTULO III

3. Metodología para la implementación de una VPN

Se detalla aspectos fundamentales de la metodología empleada para la implementación una Red Privada Virtual.

3.1 Pasos de la metodología para la implementación de una VPN

La metodología a emplear consiste en el análisis de parámetros que se detallan a continuación, durante el proceso:

- Selección de Hardware para la implementación VPN
- Selección de Software para la implementación de la VPN
- Parámetros de una VPN para la implementación.
- Arquitectura y topología física de red
- Topología de Red de la VPN

3.1.1 Selección de Hardware para la implementación VPN

Se selecciona el Hardware, es decir el computador u ordenador, que permite satisfacer las necesidades para la implementación de la VPN. Por lo tanto, se establece características:

Estas características del ordenador es proporcionado por el Departamento de Desarrollo Tecnológico e Informático (DDTI)

Tabla 1. Requerimiento de Hardware

Modelo del procesador	i7-6700
Frecuencia del procesador	3,4 GHz
Memoria interna	16 GB
Tipo de memoria interna	DDR4-SDRAM
Velocidad de memoria del reloj	2133 MHz
Capacidad total de almacenaje	1128 GB
Sistema operativo instalado	LINUX-CentOS 6.6
Número de filamentos de procesador	8
Caché del procesador	8 MB

Referencia. Elaboración Propia

En función de la selección del hardware se estableció características básicas que permitan satisfacer la implementación de la VPN y de acuerdo a esto se obtuvo los equipos proporcionados desde el Departamento de Desarrollo Tecnológico e Informático (DDTI) por parte de la Universidad Técnica del Norte.

Se recomienda que el ordenador cuente con las siguientes características:

Tabla 2. Requerimientos de Software

Capacidad de disco duro libre	40 GB
Sistema operativo	LINUX-CentOS 6.6 o superior
Memoria RAM	4 GB o superior
Modelo del procesador	i3,i5.i7

Referencia. Elaboración Propia

3.1.2 Selección de software para la implementación de la VPN

En la elección del software para la VPN, donde se va a desarrollar la VPN, se establece características que cumplan a cabalidad la función de la VPN. Por lo tanto se proporcionó por el DDTI las siguientes características.

Tabla 3. Selección de Software para la implementación de la VPN

Sistema Operativo	UNIX-CentOS
Versión	CentOS 6.6
Kernel	2.6.32-573.22.1.el6.x86_64
Características	software libre
Programa de servidor	VPN-OPENVPN

Referencia. Elaboración Propia

Se recomienda tener las siguientes características.

Tabla 4. Recomendaciones de Software

Sistema Operativo	**UNIX-CentOS**
Versión	CentOS 6.6 o superior
Kernel	x86_64 de 64 bits

Referencia. Elaboración Propia

También es necesario mencionar que las Redes Privadas Virtuales pueden ser construidas tanto por software, como por hardware o por la combinación de ambas.

Se implementa en software libre por motivos de mayor seguridad, para que los datos en el túnel VPN pasen de una manera encriptada y no en texto plano.

Es necesario recalcar que para fines de implementar servidores es recomendable usar UNIX-LINUX, debido a su mejor rendimiento del sistema operativo, no se necesita instalar un antivirus, a su rapidez y recursos es ampliamente más rápido, necesitando menos recursos en cuanto a hardware.

3.1.3 Parámetros de una VPN para la implementación.

Estos parámetros son de importancia ya que se presenta el porqué de realizar una Red Privada Virtual dentro de una Institución, entre ellos la reducción de costos, su alta seguridad con los diferentes protocolos de autenticación y encriptación de los datos a nivel de capa 3 IP.

3.1.3.1 Importancia del desarrollo de una Red Privada Virtual

El tener una Red Privada Virtual entre el AHSVP de la ciudad de Ibarra y el edificio Central de la Universidad Técnica del Norte, permite conectar a los usuarios de las diferentes áreas que forman parte de la estructura física de la Universidad Técnica del Norte, de forma segura y confiable ante ataques externos con fines antiéticos y de acciones fraudulentas que vulneran la información personal administrativa, proporcionando por medio de la VPN garantizar la confidencialidad y seguridad de los datos que atraviesan por el túnel a implementar.

Los usuarios serán solo personal administrativo que se encuentre configurado e ingresado en el servidor VPN con sus respectivas credenciales y contraseñas logrando de esta manera una confiabilidad dentro de la red.

3.1.3.2 Descripción de la comunicación entre el AHSVP y la Universidad Técnica del Norte a través de la VPN.

Se establece la comunicación entre los servidores VPN que se encuentra alojados bajo software libre, donde se emplea Linux-CentOS 6.6, para una comunicación satisfactoria.

En el edificio central de la Universidad Técnica del Norte se instalará el primer servidor VPN en un computador de características idóneas, con el cual se va realizar la comunicación de extremo a extremo, con el segundo servidor VPN que está ubicado en el AHSVP, para obtener la comunicación privada entre ellas.

3.1.3.3 Seguridad a implementar en la VPN

La VPN va a estar en función de IPSec logrando de esta manera una seguridad en la comunicación al momento de trasferir información a través de una red IP. Este protocolo brinda privacidad e integridad de los paquetes IP, ya que permite agregar encriptado y autenticación de las comunicaciones IP.

Para la autenticación de los clientes se generan certificados y llaves tanto del servidor como del cliente lograr así una confiabilidad y garantizar una completa seguridad de la red.

3.1.3.4 Estudio y análisis

En este estudio se desarrollará los parámetros que deben cumplir las Redes Privadas Virtuales que se va a implementar en la Universidad Técnica del Norte con el AHSVP que se encuentra geográficamente alojado, que puedan comunicarse de una manera segura, con las restricciones que van a ser implementadas entre ellas el tipo de seguridad que se va a configurar en la VPN.

Durante el proceso de estudio y análisis también se deben estudiar los parámetros que se va a implementar la VPN, entre estos se involucra el tipo de software y hardware para la implementación de la Red Privada Virtual.

En el caso de la VPN se configura bajo software libre LINUX, que se establecerá la conexión Servidor a Servidor en CentOS, dando lugar a la comunicación entre los dos servidores VPN de extremo a extremo.

3.1.4 Arquitectura y topología física de red

Dentro de la implementación de los servidores VPN se determina la arquitectura física y topología que posee actualmente la red de la Universidad Técnica del Norte.

Se detallan los equipos de la topología como se observa en la figura 8, tanto como servidores, equipos de red, de seguridad, servidores de telefonía IP, switchs, Firewall y los servidores VPN entre otros equipos que componen la red, que se especificarán en la topología de la Universidad Técnica del Norte.

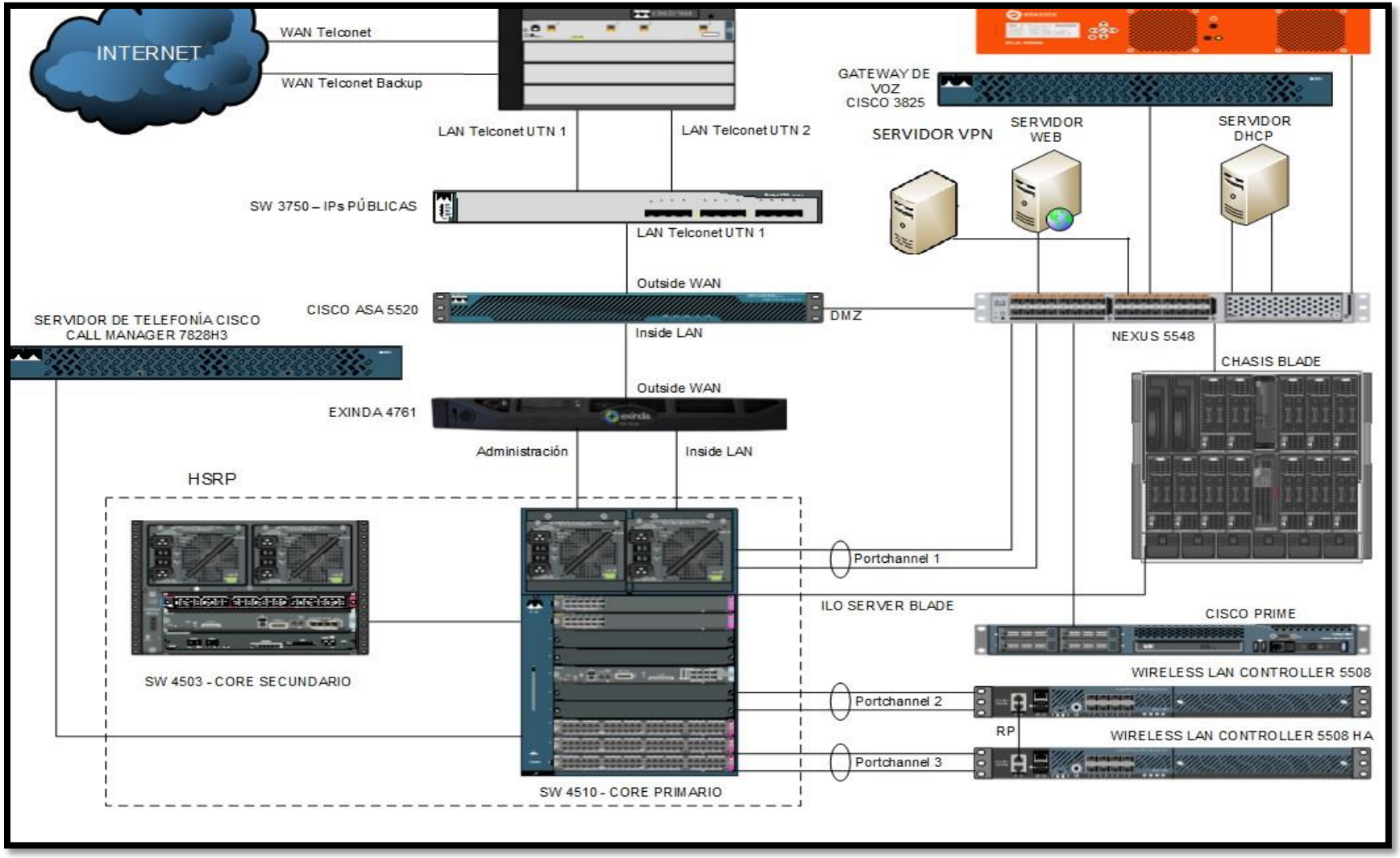


Figura 11. Topología de la Red de Universidad Técnica del Norte
Referencia: Elaboración propia

En la figura 11, se observa que el servidor VPN entre otros servidores de la UTN están conectados al switch Nexus y este a la DMZ que permite la seguridad a la red internan donde se alojan la red de servidores para mayor seguridad y respaldo de la información.

3.1.5 Topología de Red de la VPN

En la topología de red que se observa en la figura 9 de la VPN, tanto del Hospital Viejo como del edificio Central, establece una comunicación segura entre las estaciones de trabajo, con la seguridad denominada IPSec que es de capa 3 parar lograr fiabilidad en las redes sobre el protocolo IP y así autenticar y cifrar cada paquete ip en un flujo de datos.

Asignados a su vez a cada uno de los servidores que intervienen en la arquitectura de la VPN una ip pública pro con sus respectivas características como se observa en la figura 12.

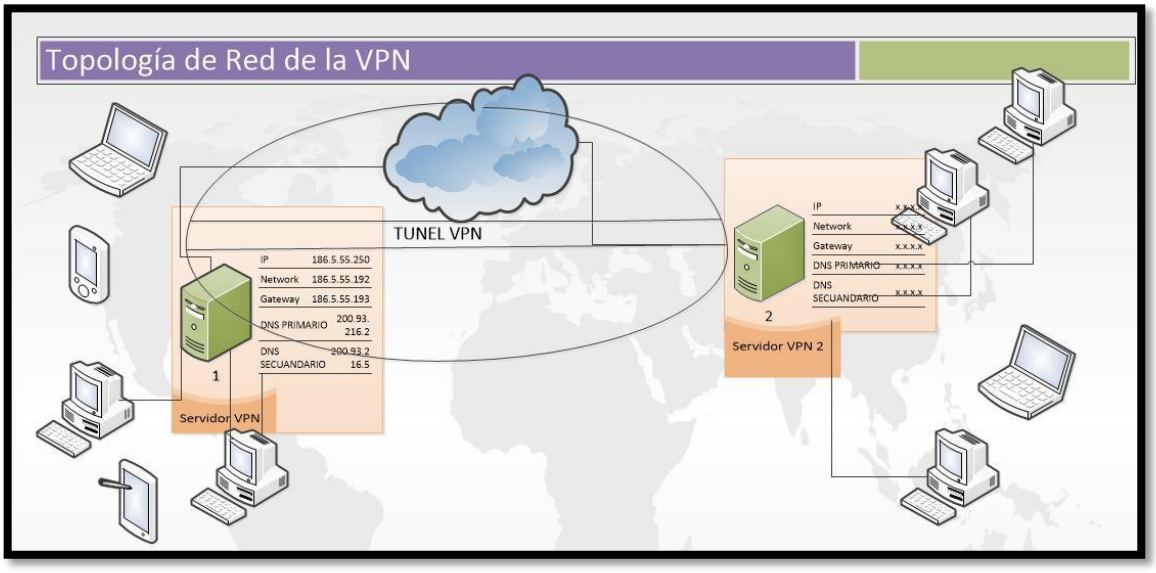


Figura 12. **Topología de Red de la VPN**
Referencia. Elaboración Propia

3.1.6 Topología General de Red del Antiguo Hospital San Vicente de Paúl (AHSVP)

En la figura 13, se observa la topología establecida en el AHSVP donde se especifica los dispositivos que intervienen para la implementación de al VPN, al igual se observa el servidor VPN A y el servidor VPN B, en la topología establecida e incluso los equipos de red que intervienen.

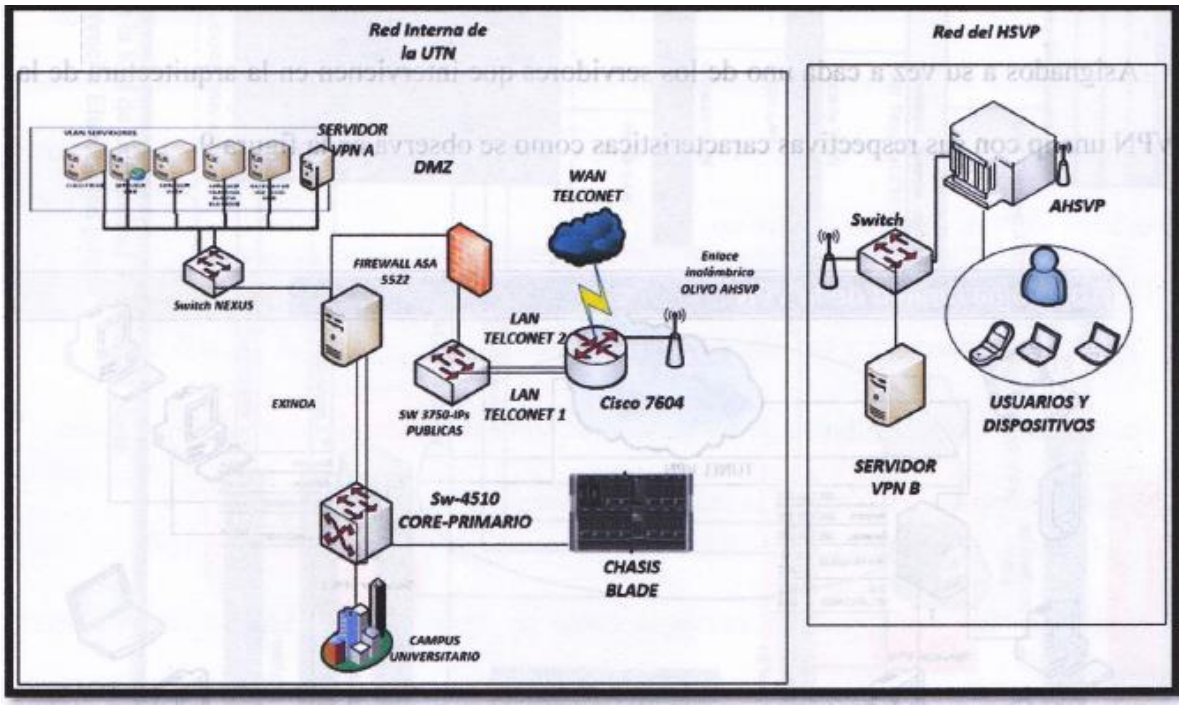


Figura 13. Topología de red del Hospital Antiguo

Referencia: Elaboración propia

CAPÍTULO IV

4. Implementación y Configuración de una VPN

Se realiza el proceso para la implementación y configuración de la VPN, en relación de los diferentes tipos de protocolos para la funcionalidad correcta de los servidores VPN.

4.1 Configuración de los servidores VPN

Se establece la conexión entre el Edificio Central y el Hospital antiguo con la implementación de la VPN, como se observa en la figura 14, donde los servidores VPN se encuentran ubicados en cada una de las áreas de trabajo que pertenecen a la Universidad Técnica del Norte. Además que la información que se transmita en el túnel y se encripta brindando mayor seguridad.

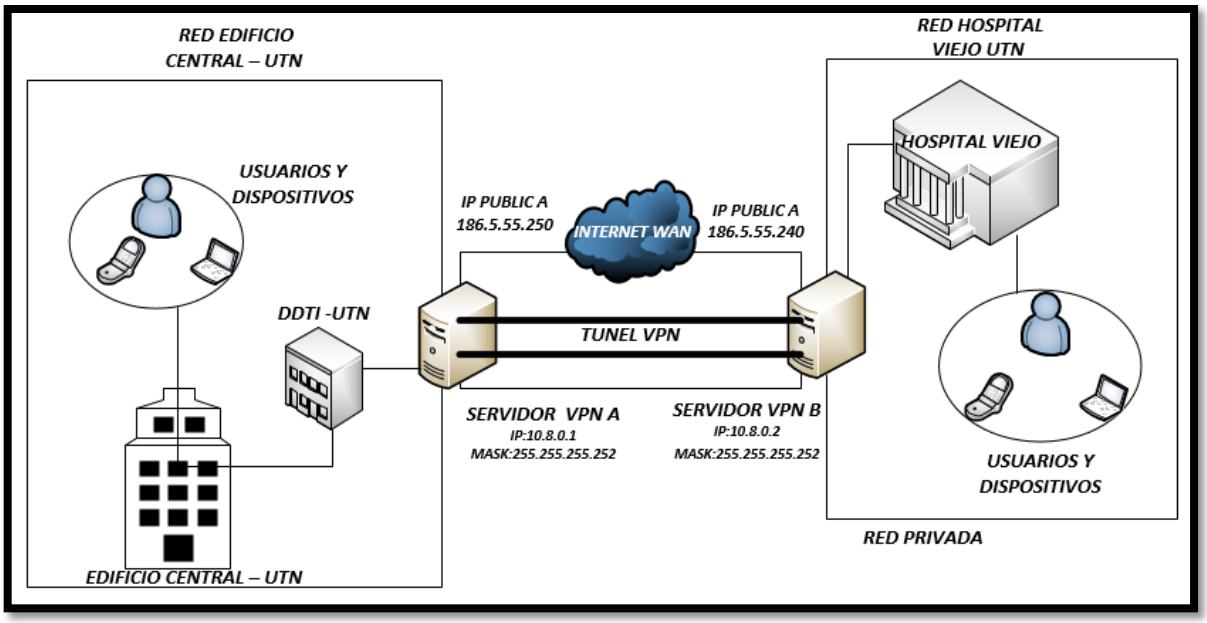


Figura 14. Topología general de la implementación de la Red Privada Virtual, entre los servidores VPN.
Referencia: Elaboración propia

Por lo tanto, se realiza la configuración de los servidores A que se encuentra en el Edificio Central y del servidor B que se encuentra en el AHSVP, que pertenecen a la Universidad Técnica del Norte.

4.2 Configuración del Servidor A del edificio Central UTN

Se realiza el proceso para la implementación y configuración de la VPN en el servidor A.

El servidor A está asignado con el direccionamiento proporcionado por el Departamento de Desarrollo Tecnológico e Informático (DDTI) que cuenta con:

- IP Pública: 186.5.55.250
- Mascara: 255.255.255.192
- Gateway: 186.5.55.193
- DNS Primario: 200.93.192.148
- DNS Secundario: 200.93.192.161

Instalación del Protocolo SSH

La instalación del protocolo SSH permite la comunicación remota del servidor para la manipulación por parte del administrador, la configuración completa se encuentra en el **Anexo A.**

Con el siguiente comando se instala el servidor ssh.

- yum install openssh-server

Una vez instalado el servidor ssh se procede a configurar con el comando.

- nano /etc/ssh/sshd_config

En el archivo sshd_config, se procede a habilitar el puerto 22, a continuación se valida el servicio con el comando respectivo:

- # Port 22
- service sshd restart

4.2.1 Instalación de OpenVPN

Se procede a la instalación y configuración del servidor OpenVPN en CentOS 6.6, en los dos servidores. También a configurar los clientes en sus diferentes sistemas operativos (Windows, Linux), para conectarse.

Antes de comenzar, se requiere tener los paquetes necesarios para Enterprise Linux (EPEL) y repositorios habilitados. Se trata de un repositorio que ofrece el Proyecto Fedora que proporcionará el paquete OpenVPN. Véase **Anexo B**

4.2.1.1 Instalación de Repositorios

Acceder en modo de súper usuarios e ingresar la clave, para instalar OpenVPN, como se detalla respectivamente, Véase **Anexo B**.

- su

- wget http://dl.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm
- rpm –Uvh epel-release-6.8.noarch.rpm

Con este repositorio el cual proporciona complementos de alta calidad de software para la distribución de LINUX, de código abierto para diferentes propósitos de red.

Por ùltimo se instala el repositorio que construye paquetes rpm.

- yum install gcc make rpm-build autoconf.noarch zlib-devel pam-devel openssl-devel -y

Instalación de OpenVPN Software

Se instala la aplicación openvpn de EPEL con el siguiente comando, que es donde se va alojar el servidor VPN, Véase **Anexo C**.

- yum install openvpn -y

4.2.1.2 Configuración de Open VPN

Por consiguiente de haber realizado la instalación de repositorios, se configura como se indica a continuación, configuración completa véase **Anexo D.**

Mover el directorio openvpn a la carpeta de OpenVPN server.conf/etc/openvpn con el siguiente comando:

- cp /usr/share/doc/openvpn-*/sample/sample-config-files/server.conf /etc/openvpn

Abrir el archivo, que se encuentra el servidor openvpn, en la ubicación adecuada, para configurar el server.conf, donde se descomenta el parámetro de "push" que es el encargado del tráfico en el sistema de clientes a enrutar a través de OpenVPN, con el uso de los comandos respectivamente.

- nano -w /etc/openvpn/server.conf
- push "redirect-gateway def1 bypass-dhcp"

Cambiar la sección referente a las consultas de ruta con el DNS asignado a las IP públicas.

- push "dhcp-option DNS x.x.x.x"
- push "dhcp-option DNS x.x.x.x"

Para mejorar la seguridad, se debe asegurar de retirar el comentario de "usuario" relevante y líneas de "grupo"

- user nobody
- group nobody

4.2.2 Instalación de Easy-RSA

Culminada la instalación de OpenVPN y de modificar el archivo de configuración, se obtiene las llaves y certificados necesarios. Véase **Anexo E.** Se procede a instalar el paquete Easy-RSA.

- yum install openvpn easy-rsa

4.2.2.1 Configuración de Easy-RSA

Al igual que con el archivo de configuración, OpenVPN coloca los scripts requeridos en la carpeta de documentación de forma predeterminada. Véase **Anexo F.**

Crear la carpeta requerida y copiar los archivos a través de:

- mkdir -p /etc/openvpn/easy-rsa/keys

Copiar la carpeta easy-rsa a la ruta donde se encuentra openvpn

- cp –rf /usr/share/easy-rsa/2.0/* /etc/openvpn/easy-rsa/

Verificar el contenido a través ls en la carpeta que contiene las claves, una vez instalado Openvpn dirigirse a la carpeta de OpenVPN.

- cd /etc/openvpn

Descargar el paquete easy-rsa con el siguiente comando e instalar el repositorio easy-rsa, con los siguientes comandos respectivamente:

- https://github.com/OpenVPN/easy-rsa/releases/download/2.2.2/EasyRSA-2.2.2.tgz
- wget https://github.com/OpenVPN/easy-rsa/releases/download/2.2.2/EasyRSA-2.2.2.tgz

Revisar con el comando ll, para verificar si se encuentra instalado el paquete easy-rsa.

- drwxr-xr-x. 3 root 4096 abr 13 12:49 easy-rsa

Después de haber descargado y almacenado el archivo se procede a descomprimir

- tar –zxf EasyRSA-2.2.2.tgz

Verificar que se ha descomprimido el paquete EasyRSA-2.2.2 y mover los datos de la carpeta easy-rsa en donde se encuentran los datos que se van a generan para la autenticación a la carpeta Openvpn.

- mv EasyRSA-2.2.2 easy-rsa

Ejecutar el comando **ll keys** para observar las llaves de seguridad para la posterior autenticación como se observa en la figura 15.

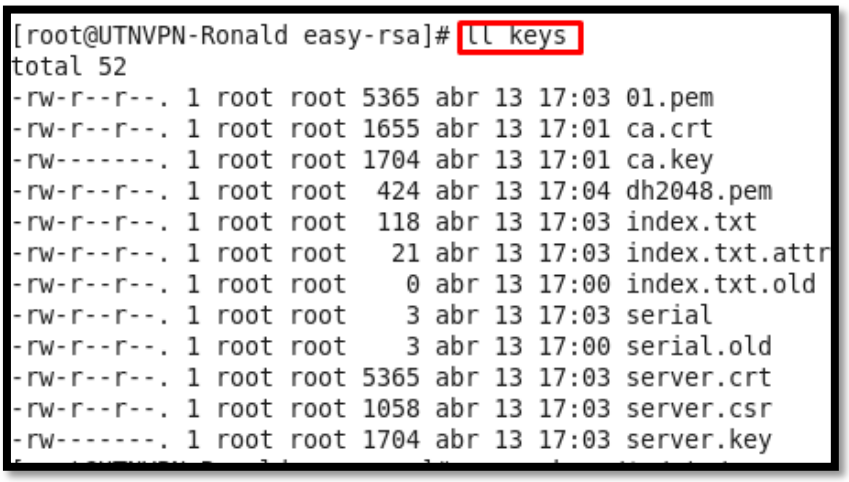


Figura 15. Llave de seguridad creada por el servidor
Fuente: Servidor OpenVPN

Copiar las llaves a la carpeta openvpn, después dirigirse a la carpeta que contiene las llaves.

- cp –a keys/* /etc/openvpn
- cd easy-rsa/
- ls

4.2.2.2 Generación de claves y certificados con Easy-RSA

Dentro de este proceso se generan los certificados del servidor para la implementación de la VPN, ver la configuración completa en el **Anexo G.**

Editar el siguiente archivo /etc/openvpn/easy-rsa /vars

- vi / etc / openvpn / easy-rsa / vars

Cambiar los valores que con el país, estado, ciudad, identificación del correo correspondiente, como la figura 16.

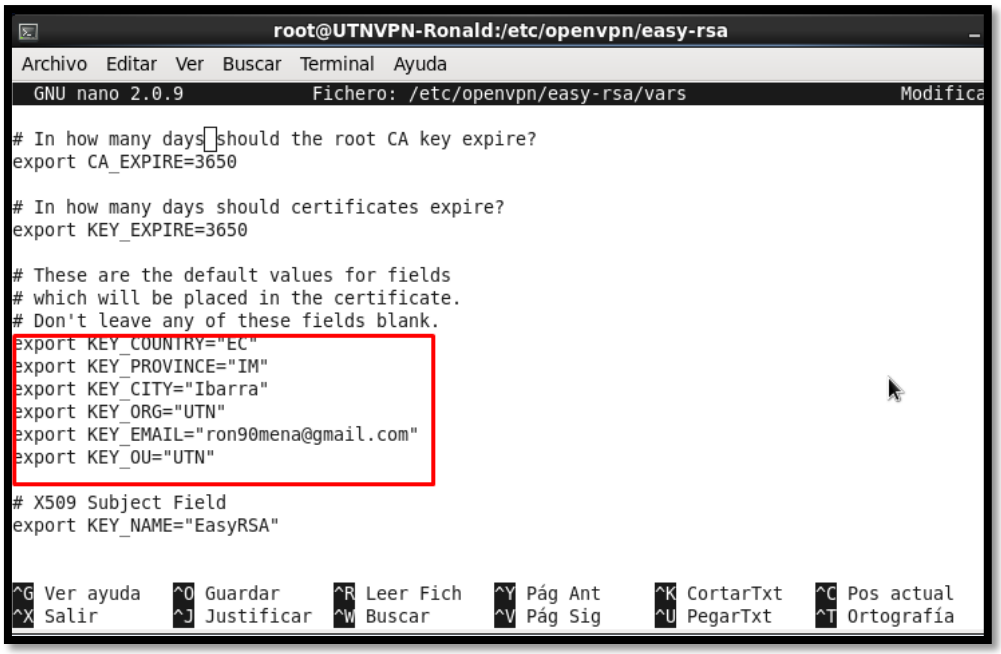


Figura16. Creando los datos para el certificado del servidor
Fuente: Elaboración Propia

OpenVPN podría no detectar correctamente la versión de OpenSSL en CentOS 6. Como medida de precaución, copiar manualmente el archivo de configuración requerido OpenSSL, que se puede observar en la figura 17.

- ✓ cp /etc/openvpn/easy-rsa/openssl-1.0.0.cnf /etc/openvpn/easy-rsa/openssl.cnf

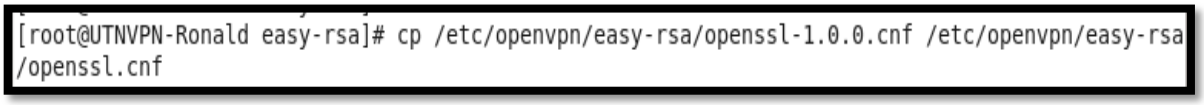


Figura 17. Detección de OpenSSL para OpenVPN
Fuente: Servidor VPN

Construir la entidad emisora de certificados o CA, con base en la información proporcionada anteriormente.

- source ./vars

Lo siguiente es limpiar toda la inicialización posteriormente antes efectuada de la autoridad de certificación, para generar los certificados.

- ./clean-all

4.2.3 Certificado para el servidor OpenVPN

Para obtener el certificado del servidor OpenVPN, se debe ejecutar el procedimiento adecuado que se detalla y la configuración completa se encuentra en el **Anexo H.**

Se ejecuta el siguiente comando para generar el certificado CA y la clave de CA:

- ./build-ca

Se completa la información que se va a incorporar en la solicitud para el certificado.

- Country Name (2 letter code) [EC]:
- State or Province Name (full name) [IM]:
- Locality Name (eg, city) [Ibarra]:
- Organization Name (eg, company) [UTN]:
- Organizational Unit Name (eg, section) [UTN]:
- Common Name (eg, your name or your server's hostname) [UTN CA]: server
- Name [EasyRSA]:

- Email Address [ron90mena@gmail.com]:

Al obtener el certificado CA, se crea un certificado para el servidor OpenVPN. Se presenta la interrogante de aceptar la clave-servidor, en este el administrador acepta la petición, con el siguiente comando:

- ./build-key-server server

4.2.3.1 Parámetro Diffie Hellman

Se requiere para generar los archivos de intercambio de claves Diffie-Hellman a través del archivo build-dh, al obtener el resultado se dirige a la carpeta keys y se copia el archivo keys para ser trasladado a la carpeta openvpn con los comandos respectivamente, la configuración completa se encuentra en el **Anexo I.**

- ./build-dh
- cd /etc/openvpn/easy-rsa/keys
- cp dh1024.pem ca.crt server.crt server.key /etc/openvpn

Las claves y certificados necesarios se generan en el directorio / etc / openvpn / easy-rsa / keys / directorio, donde se copia el siguiente archivos de certificados y claves para el /etc/openvpn / directorio.

- ca.crt
- dh 2048 .pem
- server.crt
- server.key

4.2.3.2 Certificado para el cliente OpenVPN

Para que los clientes se autentiquen, se necesita crear certificados de cliente. Puede repetirse este proceso si es necesario para generar un certificado único y una clave para cada cliente o dispositivo. Véase en el **Anexo J**

- ./build-key client

Se configura los parámetros del cliente Openvpn, como se observa en la figura 18.

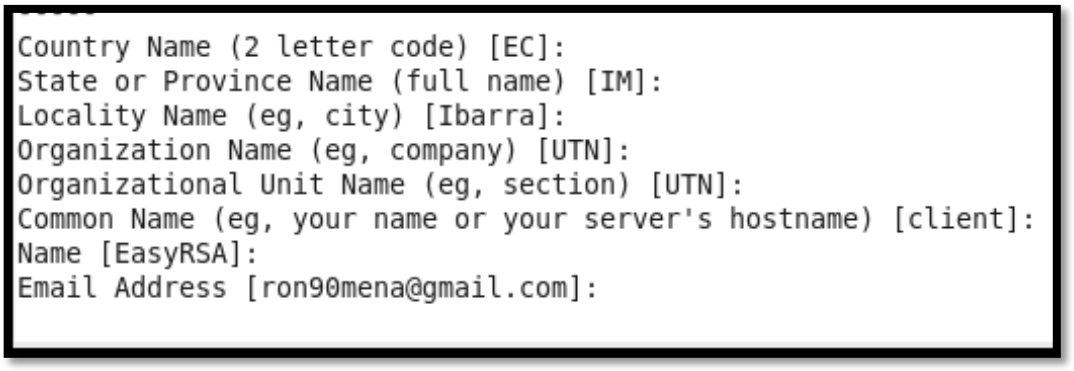


Figura 38. Parámetros para el certificado del cliente
Fuente: Servidor VPN

Se procede a aceptar el certificado del cliente.

Ejecutar el siguiente comando para verificar si se han creado las llaves de los certificados tanto del servidor como para el cliente, como esta en la figura 19.

- ll keys

Por ultimo copiar las claves de los certificados a la carpeta donde se encuentra Openvpn

- cp –a keys/* /etc/openvpn

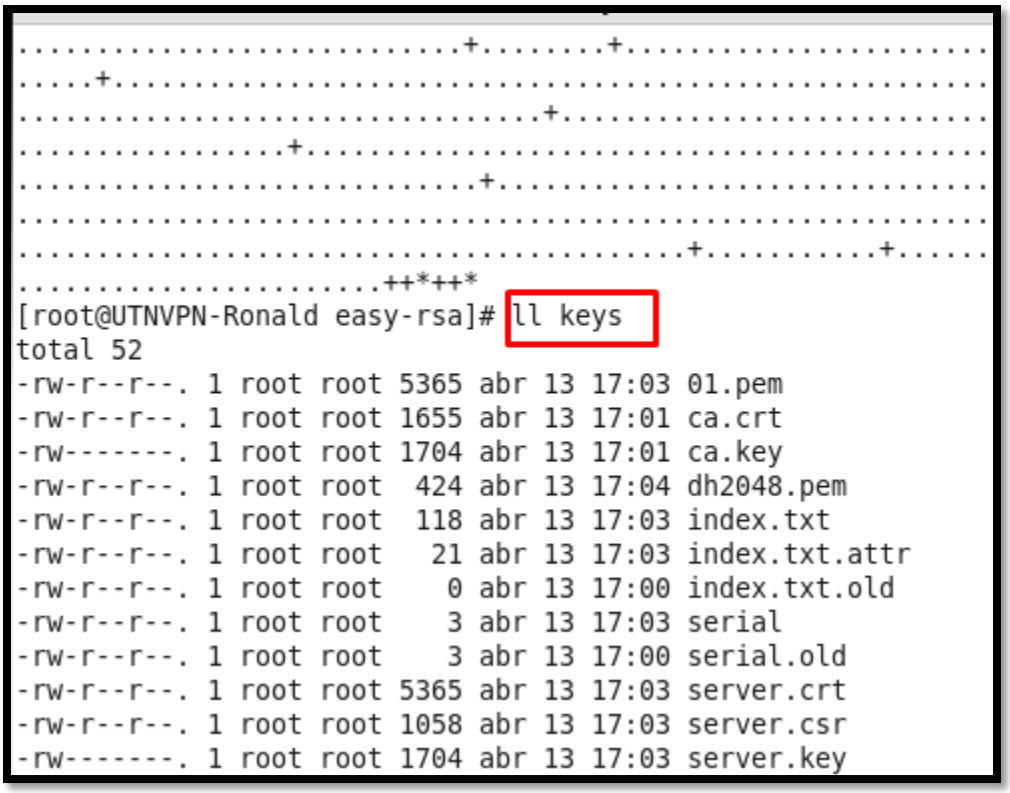


Figura 19. Llaves de servidor-cliente
Fuente: Servidor VPN

4.2.4 Servidor OPEN VPN

Con los siguientes comandos se inicia el servidor VPN y respectivamente se verifica el túnel creado por este servidor, como se observa en la figura 20, posterior al proceso realizado.

- Service openvpn restart
- ifconfig

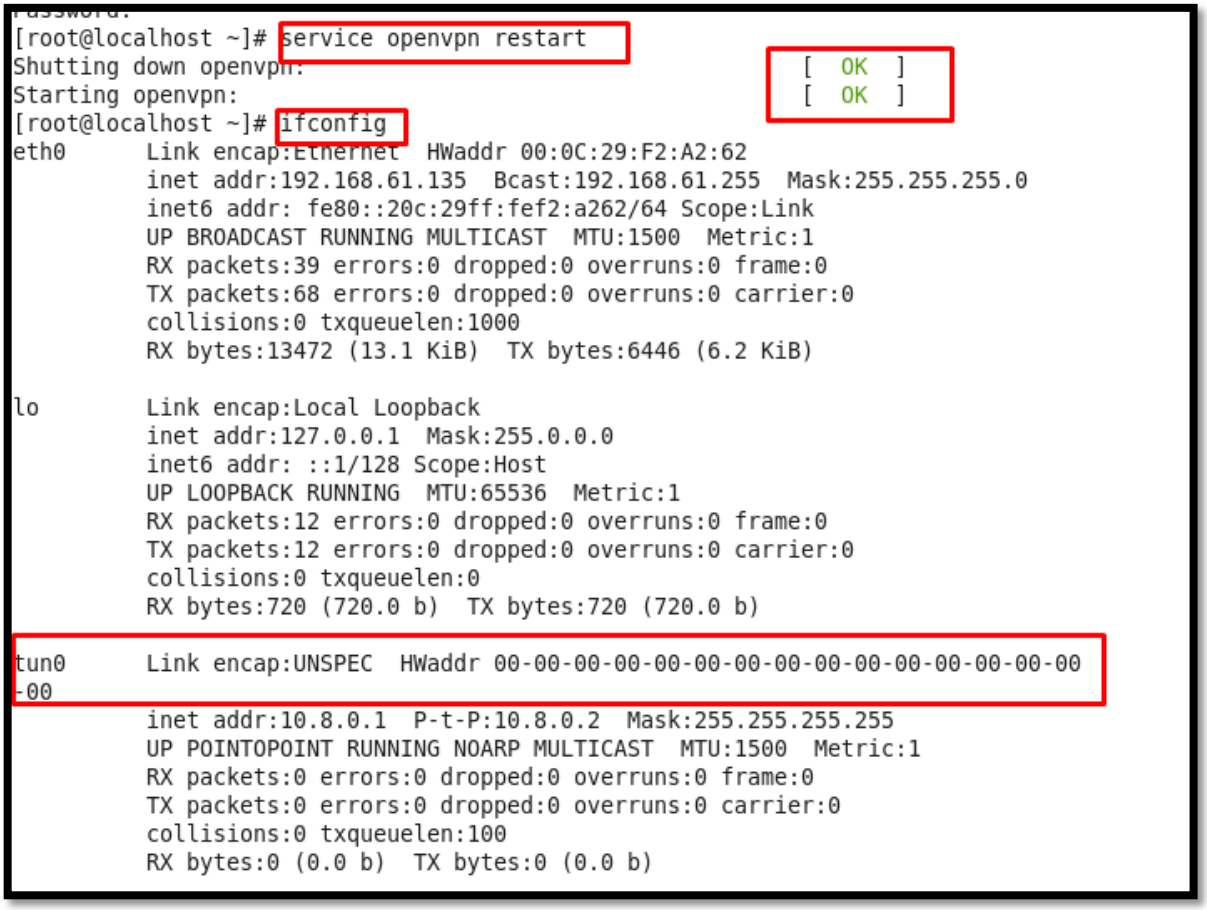


Figura 20. Inicialización del servidor VPN
Fuente: Servidor VPN

Se observa en la figura 20, la dirección virtual ip del servidor VPN, para el servidor A.

Dirección IP servidor A: 10.8.0.1

4.2.5 Instalación OpenVPN para los clientes del servidor A

En la configuración de OpenVPN, se dirige a la página oficial y se descarga OpenVPN para software Windows e instalar el cliente OPENVPN. Véase en el **Anexo L**

- https://openvpn.net/index.php/open-source/downloads.html

Al finalizar la instalación se observa el icono en el escritorio, que es el indicador de que el archivo de OpenVPN para cliente está disponible.

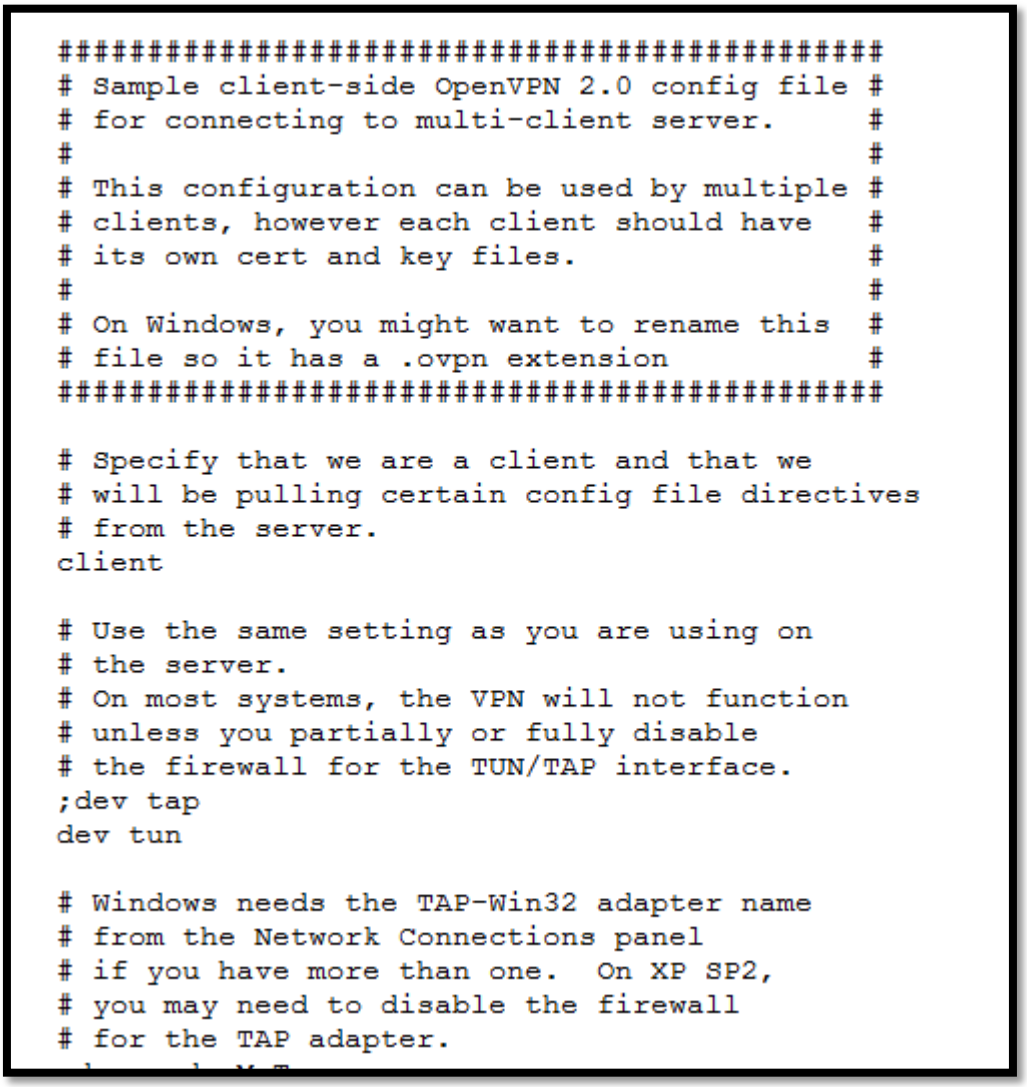
4.2.5.1 Configuración del Cliente en Windows

Una vez instalado OpenVPN, se realiza los siguientes pasos que se detallan a continuación. Véase el **Anexo M.**

- Dirigirse al Disco local C o la ubicación que se encuentre instalado OpenVPN
- Después a archivos de programa
- Dirigirse a OpenVPN y carpeta config donde se debe pegar los certificados del cliente y del servidor.
- Se configura el cliente, con el nombre de los certificados antes realizados.

Nota: Abrir como un bloc de notas de preferencia en WordPad.

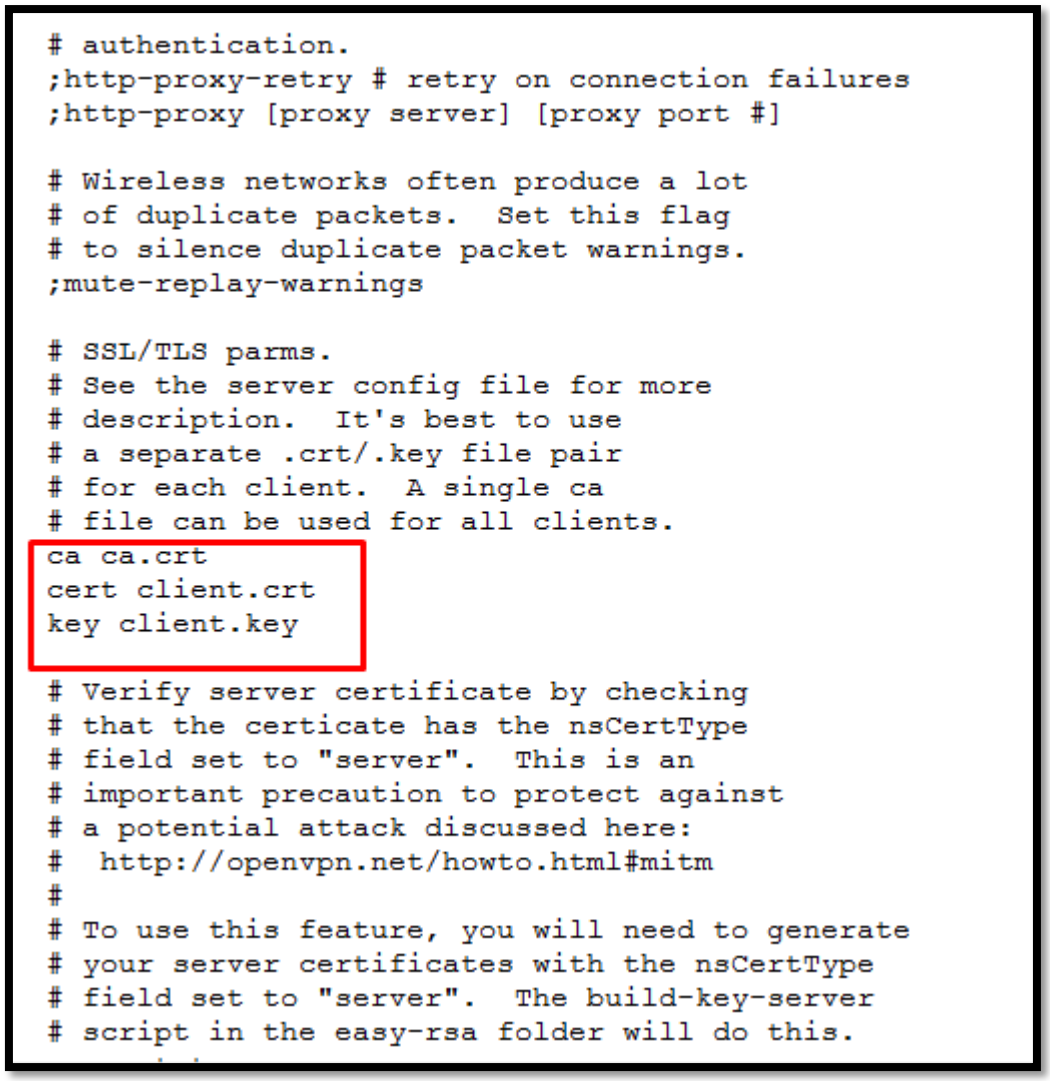
En la figura 21, se detalla que debe estar activado la línea dev tun, para el funcionamiento idóneo del túnel con los clientes dentro de la VPN. Además de ser la visualización y descripción inicial del archivo para la configuración del cliente.



```
##############################################
# Sample client-side OpenVPN 2.0 config file #
# for connecting to multi-client server.     #
#                                            #
# This configuration can be used by multiple #
# clients, however each client should have   #
# its own cert and key files.                #
#                                            #
# On Windows, you might want to rename this  #
# file so it has a .ovpn extension           #
##############################################

# Specify that we are a client and that we
# will be pulling certain config file directives
# from the server.
client

# Use the same setting as you are using on
# the server.
# On most systems, the VPN will not function
# unless you partially or fully disable
# the firewall for the TUN/TAP interface.
;dev tap
dev tun

# Windows needs the TAP-Win32 adapter name
# from the Network Connections panel
# if you have more than one.  On XP SP2,
# you may need to disable the firewall
# for the TAP adapter.
```

Figura 21. Configuración inicial del cliente
Fuente: Servidor OpenVPN

Es de mucha importancia recalcar el proceso ya que se requiere cambiar el nombre del certificado, con el mismo nombre que se configuro en el servidor OpenVPN, como se observa en la figura 22.



```
# authentication.
;http-proxy-retry # retry on connection failures
;http-proxy [proxy server] [proxy port #]

# Wireless networks often produce a lot
# of duplicate packets.  Set this flag
# to silence duplicate packet warnings.
;mute-replay-warnings

# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
ca ca.crt
cert client.crt
key client.key

# Verify server certificate by checking
# that the certicate has the nsCertType
# field set to "server".  This is an
# important precaution to protect against
# a potential attack discussed here:
#  http://openvpn.net/howto.html#mitm
#
# To use this feature, you will need to generate
# your server certificates with the nsCertType
# field set to "server".  The build-key-server
# script in the easy-rsa folder will do this.
```

Figura 22. Certificados del servidor al cliente
Fuente: Servidor OpenVPN

Una vez logrado se conecta a la VPN el cliente, lo que da paso a la conexión del servidor VPN con la IP pública del servidor. Y al finalizar se muestra un indicador de que la conexión fue satisfactoria.

En el símbolo del sistema o cmd de Windows del cliente se ejecuta un ipconfig para ver la IP del cliente VPN, como resultado de la conexión exitosa, formando parte de los clientes de la red VPN establecida con la conexión al servidor A donde su IP asignada es la 10.8.0.1 y automáticamente asignada la dirección IP virtual al cliente por parte del servidor que es 10.8.0.6, como se observa en la figura 23.

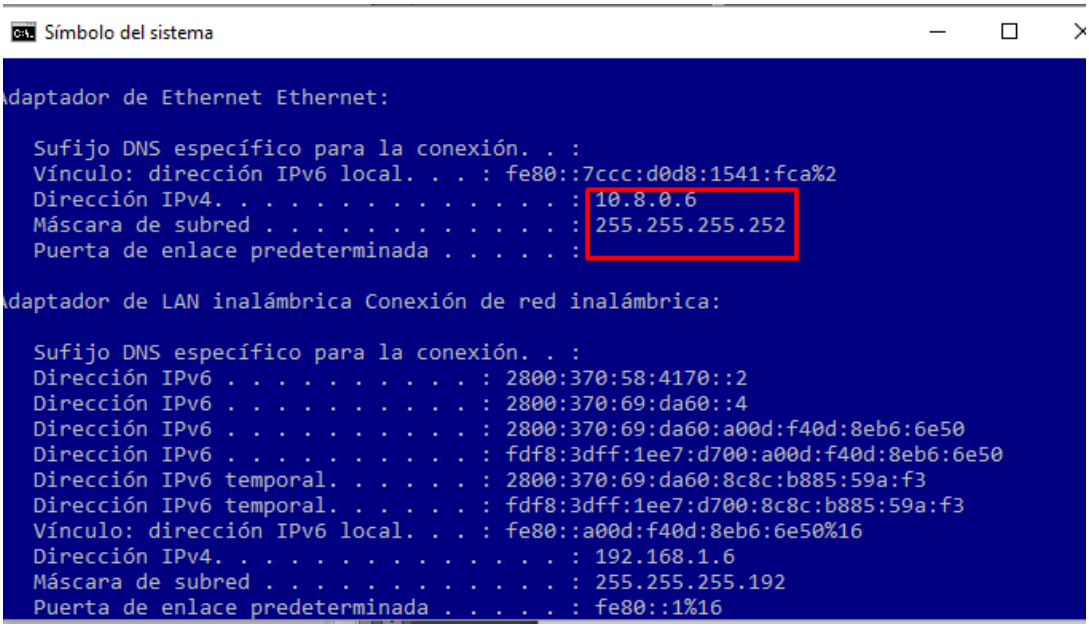


Figura 43. Ip nueva del cliente proveniente del servidor VPN
Fuente: Cliente de Servidor OpenVPN

Prueba de funcionamiento del cliente realizando un ping al servidor A OpenVPN, donde la conexión es exitosa, como se observa en la figura 24. Realizando un ping a la ip del servidor A con la dirección ip 10.8.0.1.

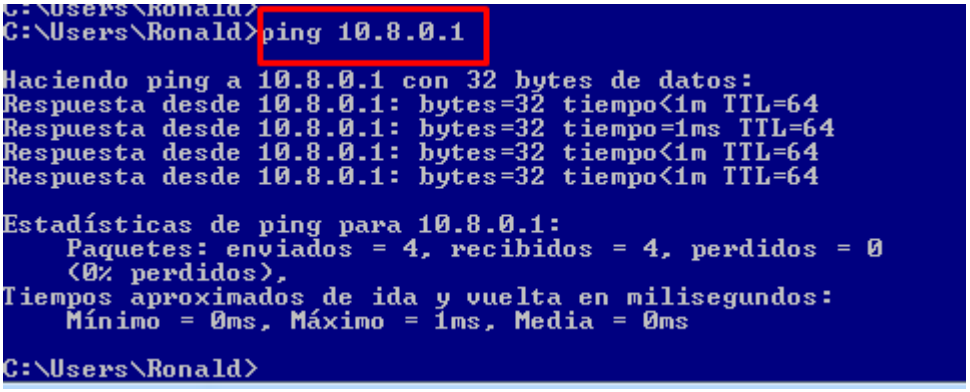


Figura 24. Prueba de funcionamiento del cliente al servidor
Fuente: Cliente OpenVPN

4.2.5.2 Configuración de un Cliente en un Sistema Android

Para configurar un cliente VPN, se realiza los siguientes pasos. Véase el **Anexo N**

- Disponer de un dispositivo móvil o Smartphone con sistema Android
- Instalar la aplicación Openvpn connect en el Smartphone
- Aceptar y acceder a los archivos de configuración de los certificados del cliente y seleccionar el archivo .ovpn.
- Se abre la aplicación Openvpn connect, una vez instalada.
- Se configura el cliente openvpn en el sistema operativo Android, donde se transfiere los certificados al dispositivo móvil.
- Dirigirse al menú y escoger la opción Import, que se encarga de importar los certificados.
- Seleccionar los certificados generados por el servidor OpenVPN, que se han generado.
- Seleccionar la memoria del Smartphone en este caso sdcard, para copiar en esta los certificados.
- Los certificados realizados previamente por el Servidor OpenVPN, deben estar alojados en el móvil y trasladarse a la carpeta VPN.

- Elegir el certificado del cliente .ovpn en este proceso denominado client.ovpn
- Una vez importado el certificado del cliente se visualiza la aprobación para conectar con el servidor VPN.
- Aceptar el acuerdo para que la aplicación interprete todo el tráfico de la red y autenticar con el servidor VPN.
- Una vez que se autentique con el servidor se podrá conectar al servidor VPN
- Se puede ingresar y navegar de una manera segura a través de una Red Privada Virtual al finalizar el proceso.

4.3 Configuración del Servidor B en el Hospital Antiguo

Se realiza el proceso para la implementación y configuración de la VPN en el servidor B.

El servidor B está asignado con el direccionamiento proporcionado por el Departamento de Desarrollo Tecnológico e Informático (DDTI) que cuenta con:

- IP Pública: 190.95.196.206
- Mascara: 255.255.255.224
- Gateway: 190.95.196.193
- DNS Primario: 200.93.216.2
- DNS Secundario=200.93.216.5

Instalación del Protocolo SSH

La instalación del protocolo SSH permite la comunicación remota del servidor para la manipulación por parte del administrador, la configuración completa se encuentra en el **Anexo A.**

Con el siguiente comando se instala el servidor ssh.

- yum install openssh-server

Una vez instalado el servidor ssh se procede a configurar con el comando.

- nano /etc/ssh/sshd_config

En el archivo sshd_config, se procede a habilitar el puerto 22, a continuación se valida el servicio con el comando respectivo:

- # Port 22
- service sshd restart

4.3.1 Instalación de OpenVPN

Se procede a la instalación y configuración del servidor OpenVPN en CentOS 6.6, en los dos servidores. También a configurar los clientes en sus diferentes sistemas operativos (Windows, Linux), para conectarse.

Antes de comenzar, se requiere tener los paquetes necesarios para Enterprise Linux (EPEL) y repositorios habilitados. Se trata de un repositorio que ofrece el Proyecto Fedora que proporcionará el paquete OpenVPN. Véase **Anexo B**

4.3.1.1 Instalación de Repositorios

Acceder en modo de súper usuarios e ingresar la clave, para instalar OpenVPN, como se detalla respectivamente, Véase **Anexo B**.

- su
- wget http://dl.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm
- rpm –Uvh epel-release-6.8.noarch.rpm

Con este repositorio el cual proporciona complementos de alta calidad de software para la distribución de LINUX, de código abierto para diferentes propósitos de red.

Por ùltimo se instala el repositorio que construye paquetes rpm.

- yum install gcc make rpm-build autoconf.noarch zlib-devel pam-devel openssl-devel -y

Instalación de OpenVPN Software

Se instala la aplicación openvpn de EPEL con el siguiente comando, que es donde se va alojar el servidor VPN, Véase **Anexo C**.

- yum install openvpn -y

4.3.1.2 Configuración de Open VPN

Por consiguiente de haber realizado la instalación de repositorios, se configura como se indica a continuación, configuración completa véase **Anexo D.**

Mover el directorio openvpn a la carpeta de OpenVPN server.conf/etc/openvpn con el siguiente comando:

- cp /usr/share/doc/openvpn-*/sample/sample-config-files/server.conf /etc/openvpn

Abrir el archivo, que se encuentra el servidor openvpn, en la ubicación adecuada, para configurar el server.conf, donde se descomenta el parámetro de "push" que es el encargado del tráfico en el sistema de clientes a enrutar a través de OpenVPN, con el uso de los comandos respectivamente.

- nano -w /etc/openvpn/server.conf
- push "redirect-gateway def1 bypass-dhcp"

Cambiar la sección referente a las consultas de ruta con el DNS asignado a las IP públicas.

- push "dhcp-option DNS x.x.x.x"
- push "dhcp-option DNS x.x.x.x"

Para mejorar la seguridad, se debe asegurar de retirar el comentario de "usuario" relevante y líneas de "grupo"

- user nobody
- group nobody

4.3.2 Instalación de Easy-RSA

Culminada la instalación de OpenVPN y de modificar el archivo de configuración, se obtiene las llaves y certificados necesarios. Véase **Anexo E.**

Se procede a instalar el paquete Easy-RSA.

- yum install openvpn easy-rsa

4.3.2.1 Configuración de Easy-RSA

Al igual que con el archivo de configuración, OpenVPN coloca los scripts requeridos en la carpeta de documentación de forma predeterminada. Véase **Anexo F.**

Crear la carpeta requerida y copiar los archivos a través de:

- mkdir -p /etc/openvpn/easy-rsa/keys

Copiar la carpeta easy-rsa a la ruta donde se encuentra openvpn

- cp –rf /usr/share/easy-rsa/2.0/* /etc/openvpn/easy-rsa/

Verificar el contenido a través ls en la carpeta que contiene las claves, una vez instalado Openvpn dirigirse a la carpeta de OpenVPN.

- cd /etc/openvpn

Descargar el paquete easy-rsa con el siguiente comando e instalar el repositorio easy-rsa, con los siguientes comandos respectivamente:

- https://github.com/OpenVPN/easy-rsa/releases/download/2.2.2/EasyRSA-2.2.2.tgz
- wget https://github.com/OpenVPN/easy-rsa/releases/download/2.2.2/EasyRSA-2.2.2.tgz

Revisar con el comando ll, para verificar si se encuentra instalado el paquete easy-rsa.

- drwxr-xr-x. 3 root 4096 abr 13 12:49 easy-rsa

Después de haber descargado y almacenado el archivo se procede a descomprimir

- tar –zxf EasyRSA-2.2.2.tgz

Verificar que se ha descomprimido el paquete EasyRSA-2.2.2 y mover los datos de la carpeta easy-rsa en donde se encuentran los datos que se van a generan para la autenticación a la carpeta Openvpn.

- mv EasyRSA-2.2.2 easy-rsa

Ejecutar el comando **ll keys** para observar las llaves de seguridad para la posterior autenticación como se observa en la figura 25.

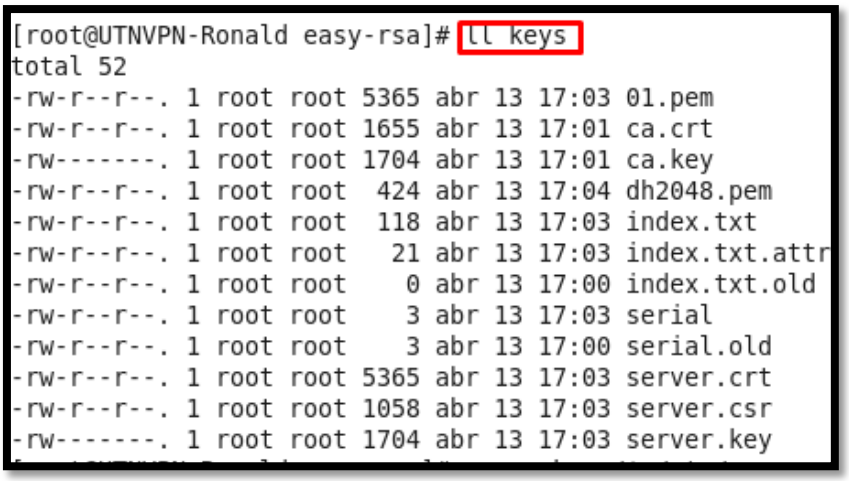


Figura 25. Llave de seguridad creada por el servidor
Fuente: Servidor OpenVPN

Copiar las llaves a la carpeta openvpn, después dirigirse a la carpeta que contiene las llaves.

- cp –a keys/* /etc/openvpn
- cd easy-rsa/
- ls

4.3.2.2 Generación de claves y certificados con Easy-RSA

Dentro de este proceso se generan los certificados del servidor para la implementación de la VPN, ver la configuración completa en el **Anexo G.**

Editar el siguiente archivo /etc/openvpn/easy-rsa /vars

- vi / etc / openvpn / easy-rsa / vars

Cambiar los valores que con el país, estado, ciudad, identificación del correo correspondiente, como la figura 26.

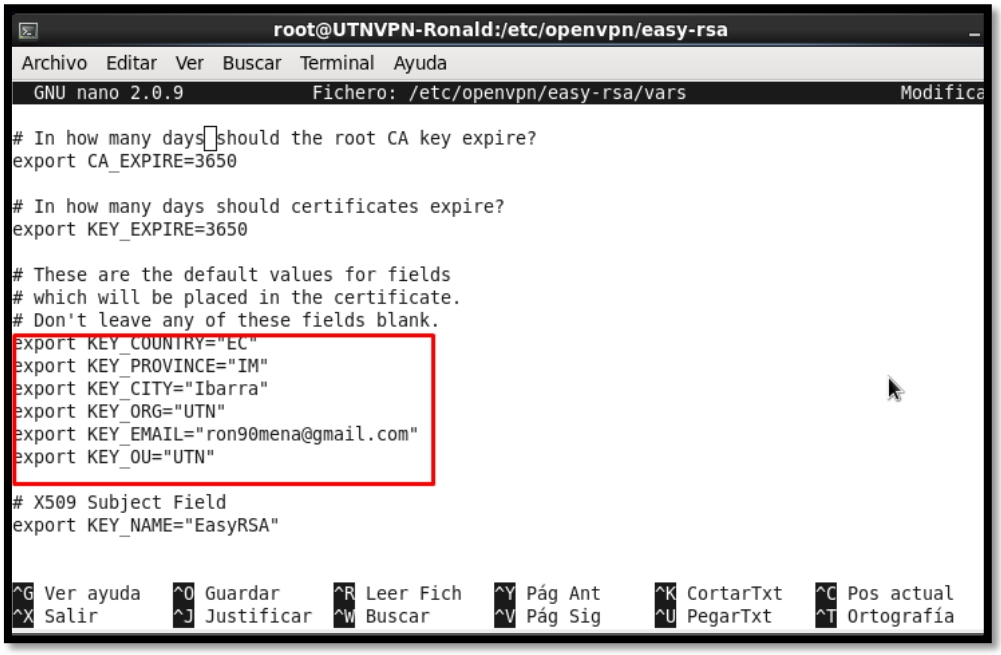


Figura 26. Creando los datos para el certificado del servidor
Fuente: Elaboración Propia

OpenVPN podría no detectar correctamente la versión de OpenSSL en CentOS 6. Como medida de precaución, copiar manualmente el archivo de configuración requerido OpenSSL, que se puede observar en la figura 27.

- ✓ cp /etc/openvpn/easy-rsa/openssl-1.0.0.cnf /etc/openvpn/easy-rsa/openssl.cnf

Figura 27. Detección de OpenSSL para OpenVPN
Fuente: Servidor VPN

Construir la entidad emisora de certificados o CA, con base en la información proporcionada anteriormente.

- source ./vars

Lo siguiente es limpiar toda la inicialización posteriormente antes efectuada de la autoridad de certificación, para generar los certificados.

- ./clean-all

4.3.3 Certificado para el servidor OpenVPN

Para obtener el certificado del servidor OpenVPN, se debe ejecutar el procedimiento adecuado que se detalla y la configuración completa se encuentra en el **Anexo H.**

Se ejecuta el siguiente comando para generar el certificado CA y la clave de CA:

- ./build-ca

Se completa la información que se va a incorporar en la solicitud para el certificado.

- Country Name (2 letter code) [EC]:
- State or Province Name (full name) [IM]:
- Locality Name (eg, city) [Ibarra]:
- Organization Name (eg, company) [UTN]:
- Organizational Unit Name (eg, section) [UTN]:

- Common Name (eg, your name or your server's hostname) [UTN CA]: server
- Name [EasyRSA]:
- Email Address [ron90mena@gmail.com]:

Al obtener el certificado CA, se crea un certificado para el servidor OpenVPN. Se presenta la interrogante de aceptar la clave-servidor, en este el administrador acepta la petición, con el siguiente comando:

- ./build-key-server server

4.3.3.1 Parámetro Diffie Hellman

Se requiere para generar los archivos de intercambio de claves Diffie-Hellman a través del archivo build-dh, al obtener el resultado se dirige a la carpeta keys y se copia el archivo keys para ser trasladado a la carpeta openvpn con los comandos respectivamente, la configuración completa se encuentra en el **Anexo I.**

- ./build-dh
- cd /etc/openvpn/easy-rsa/keys
- cp dh1024.pem ca.crt server.crt server.key /etc/openvpn

Las claves y certificados necesarios se generan en el directorio / etc / openvpn / easy-rsa / keys / directorio, donde se copia el siguiente archivos de certificados y claves para el /etc/openvpn / directorio.

- ca.crt
- dh 2048 .pem
- server.crt
- server.key

4.3.3.2 Certificado para el cliente OpenVPN

Para que los clientes se autentiquen, se necesita crear certificados de cliente. Puede repetirse este proceso si es necesario para generar un certificado único y una clave para cada cliente o dispositivo. Véase en el **Anexo J**

- ./build-key client

Se configura los parámetros del cliente Openvpn, como se observa en la figura 28.

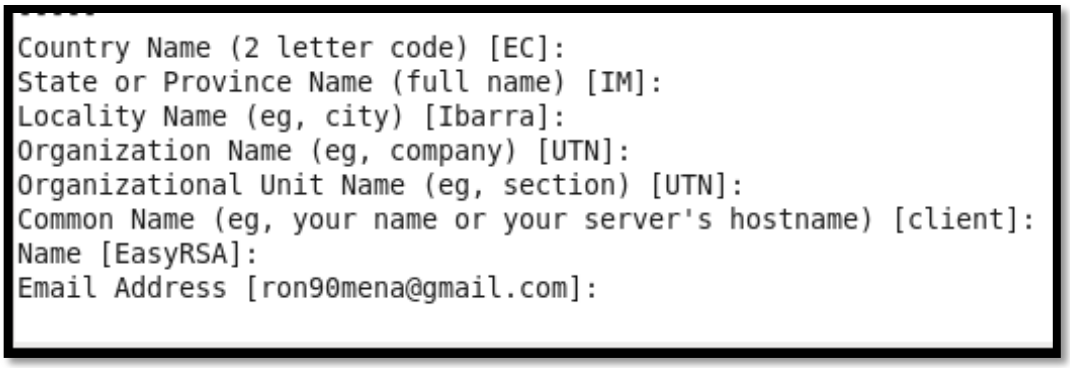


Figura 28. Parámetros para el certificado del cliente
Fuente: Servidor VPN

Se procede a aceptar el certificado del cliente.

Ejecutar el siguiente comando para verificar si se han creado las llaves de los certificados tanto del servidor como para el cliente, como esta en la figura 29.

- ll keys

Por ultimo copiar las claves de los certificados a la carpeta donde se encuentra Openvpn

- cp –a keys/* /etc/openvpn

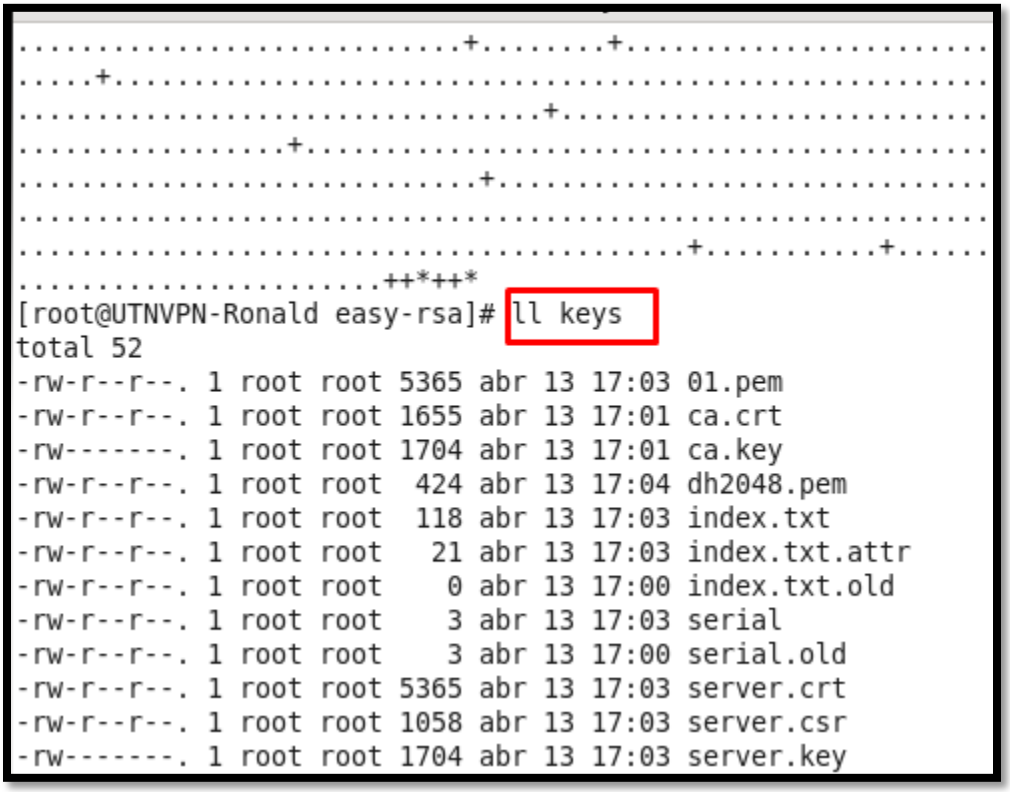


Figura 29. Llaves de servidor-cliente
Fuente: Servidor VPN

4.3.4 Servidor OPEN VPN

Con los siguientes comandos se inicia el servidor VPN y respectivamente se verifica el túnel creado por este servidor, como se observa en la figura 30, posterior al proceso realizado.

- Service openvpn restart
- ifconfig

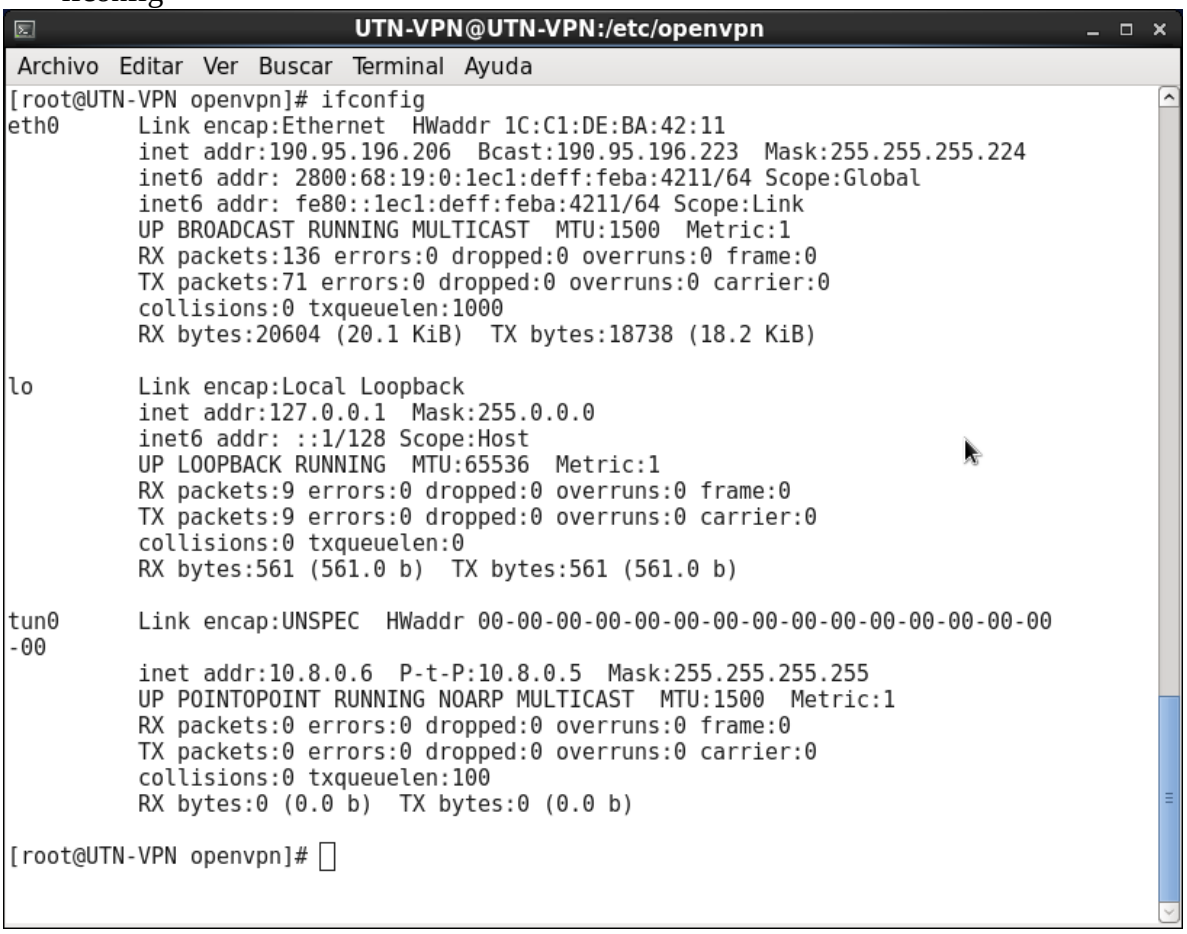


Figura 30. Inicialización del servidor VPN
Fuente: Servidor VPN

Se observa en la figura 27, las direcciones ip del servidor VPN, tanto para el servidor A como para el B.

Dirección IP servidor A: 10.8.0.1

Dirección IP servidor B: 10.8.0.6

4.3.5 Instalación OpenVPN para los clientes

En la configuración de la OpenVPN, se dirige a la página oficial y se descarga OpenVPN para software Windows e instalar el cliente OPENVPN. Véase en el **Anexo L**

- https://openvpn.net/index.php/open-source/downloads.html

Al finalizar la instalación se observa el icono en el escritorio, que es el indicador de que el archivo de OpenVPN para cliente está disponible.

4.3.5.1 Configuración del Cliente en Windows

Una vez instalado OpenVPN, se realiza los siguientes pasos que se detallan a continuación. Véase el **Anexo M.**

- Dirigirse al Disco local C o la ubicación que se encuentre instalado OpenVpn
- Después a archivos de programa
- Dirigirse a OpenVPN y carpeta config donde se debe pegar los certificados del cliente y del servidor.
- Se configura el cliente, con el nombre de los certificados antes realizados. \

Nota: Abrir como un bloc de notas de preferencia en WordPad.

En la figura 31, se detalla que debe estar activado la línea dev tun, para el funcionamiento idóneo del túnel con los clientes dentro de la VPN. Además de ser la visualización y descripción inicial del archivo para la configuración del cliente.

```
##############################################
# Sample client-side OpenVPN 2.0 config file #
# for connecting to multi-client server.     #
#                                            #
# This configuration can be used by multiple #
# clients, however each client should have   #
# its own cert and key files.                #
#                                            #
# On Windows, you might want to rename this  #
# file so it has a .ovpn extension           #
##############################################

# Specify that we are a client and that we
# will be pulling certain config file directives
# from the server.
client

# Use the same setting as you are using on
# the server.
# On most systems, the VPN will not function
# unless you partially or fully disable
# the firewall for the TUN/TAP interface.
;dev tap
dev tun

# Windows needs the TAP-Win32 adapter name
# from the Network Connections panel
# if you have more than one.  On XP SP2,
# you may need to disable the firewall
# for the TAP adapter.
```

Figura 31. Configuración inicial del cliente
Fuente: Servidor OpenVPN

Es de mucha importancia recalcar el proceso ya que se requiere cambiar el nombre del certificado, con el mismo nombre que se configuro en el servidor OpenVPN, como se observa en la figura 32.

```
# authentication.
;http-proxy-retry # retry on connection failures
;http-proxy [proxy server] [proxy port #]

# Wireless networks often produce a lot
# of duplicate packets.  Set this flag
# to silence duplicate packet warnings.
;mute-replay-warnings

# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
ca ca.crt
cert client.crt
key client.key

# Verify server certificate by checking
# that the certicate has the nsCertType
# field set to "server".  This is an
# important precaution to protect against
# a potential attack discussed here:
#  http://openvpn.net/howto.html#mitm
#
# To use this feature, you will need to generate
# your server certificates with the nsCertType
# field set to "server".  The build-key-server
# script in the easy-rsa folder will do this.
```

Figura 32. Certificados del servidor al cliente
Fuente: Servidor OpenVPN

Una vez logrado se conecta a la VPN el cliente, lo que da paso a la conexión del servidor VPN con la IP pública del servidor B. Y al finalizar se muestra un indicador de que la conexión fue satisfactoria.

En el símbolo del sistema o cmd de Windows se ejecuta un ipconfig para ver la IP del cliente VPN, como resultado de la conexión exitosa, formando parte de los clientes de la red VPN establecida con la conexión al servidor donde su IP asignada es la 10.8.0.6 y automáticamente asignada la dirección IP virtual al cliente por parte del servidor que es 10.8.0.16, como se observa en la figura 33.

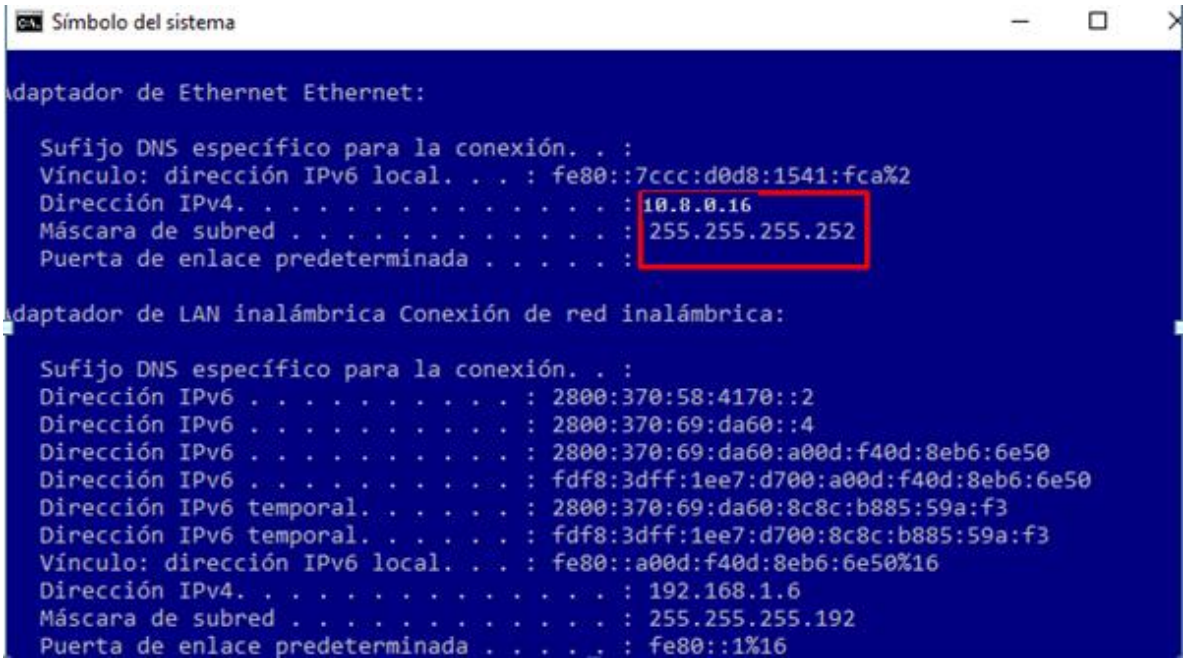


Figura 33. Ip nueva del cliente proveniente del servidor VPN
Fuente: Cliente de Servidor OpenVPN

Prueba de funcionamiento del cliente realizando un ping al servidor OpenVPN, donde la conexión es exitosa, como se observa en la figura 34. Realizando un ping a la ip del servidor B con la dirección ip 10.8.0.6.

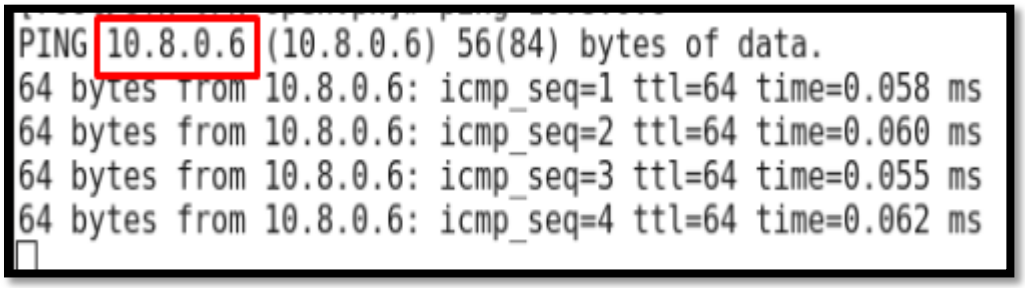


Figura 34. Prueba de funcionamiento del cliente al servidor B
Fuente: Cliente OpeVPN

4.3.5.2 Configuración de un Cliente en un Sistema Android

Para configurar un cliente VPN, se realiza los siguientes pasos. Véase el **Anexo N**

- Disponer de un dispositivo móvil o Smartphone con sistema Android
- Instalar la aplicación Openvpn connect en el Smartphone
- Aceptar y acceder a los archivos de configuración de los certificados del cliente y seleccionar el archivo .ovpn.
- Se abre la aplicación Openvpn connect , una vez instalada.
- Se configura el cliente openvpn en el sistema operativo Android, donde se transfiere los certificados al dispositivo móvil.
- Dirigirse al menú y escoger la opción Import, que se encarga de importar los certificados.

- Seleccionar los certificados generados por el servidor OpenVPN, que se han generado.
- Seleccionar la memoria del Smartphone en este caso sdcard, para copiar en esta los certificados.
- Los certificados realizados previamente por el Servidor OpenVPN, deben estar alojados en el móvil y trasladarse a la carpeta VPN.
- Elegir el certificado del cliente .ovpn en este proceso denominado client.ovpn
- Una vez importado el certificado del cliente se visualiza la aprobación para conectar con el servidor VPN.
- Aceptar el acuerdo para que la aplicación interprete todo el tráfico de la red y autenticar con el servidor VPN.
- Una vez que se autentique con el servidor se podrá conectar al servidor VPN
- Se puede ingresar y navegar de una manera segura a través de una Red Privada Virtual al finalizar el proceso.

4.4 Seguridad en una VPN en el servidor A y servidor B

En este tema se implementa la seguridad de la VPN en los dos servidores con los protocolos de autenticación y cifrado.

4.4.1 Configuración en CentOS - OpenSwan IPSec VPN

Instalar el tcpdump ntsysv en los dos servidores que posteriormente colabora para observar el encriptado y la autenticación con los comandos detallados en el proceso. Véase el **Anexo O**

- yum -y install vim wget bind-utils lsof tcpdump ntsysv

Proceder a actualizar los paquetes provenientes de los servidores para que no exista ningún problema de compilación del kernel y se reinicia el sistema.

- yum -y update
- reboot

Instalar el repositorio Webmin repo que ayuda para la instalación de IPSec, para ello se aplica los siguientes comandos.

- wget http://ftp.riken.jp/Linux/fedora/epel/6 … noarch.rpm
- yum -y install ./epel-release-6-8.noarch.rpm

Para instalar IPSec se debe configurar webmin para que no existan problemas o dificultades posteriores.

- vim /etc/yum.repos.d/webmin.repo

Dentro de este archivo se coloca los siguientes parámetros:

- [Webmin]
- name =Webmin Distribution Neutral
- #baseurl=http://download.webmin.com/download/yum
- mirrorlist=http://download.webmin.com/download/yum/mirrorlist
- enabled =1

Instalar el paquete que contiene webmin, y proceder a instalar el repositorio necesario para importar webmin.

- wget http://www.webmin.com/jcameron-key.asc
- rpm --import jcameron-key.asc
- yum -y install webmin

Una vez instalado se procede a restaurar el servicio de webmin.

- /etc/init.d/webmin restart

Proceder a ejecutar el comando para inicializar el arranque desde el sistema.

- chkconfig webmin on

Configurar el kernel para IPSec de la siguiente manera.

-nano /etc/sysctl.conf

Donde se visualiza la configuración y se habilita el enrutamiento de paquetes IP llamado "forwarding".

- net.ipv4.ip_forward = 1
- net.ipv4.conf.all.accept_redirects = 0
- net.ipv4.conf.all.send_redirects = 0

Verificar los cambios con el comando.

- sysctl -p

Configurar Selinux para que no existan inconvenientes a la hora de iniciar IPSec.

- echo 0 > /selinux/enforce
- vim /etc/sysconfig/selinux

Donde se muestra el archivo de texto, modificando el valor expuesto a continuación:

- SELINUX=disabled

Una vez configurado el Selinux se procede a escribir el comando que realizara un acuerdo de la configuración de envió y reenvió de paquetes ip por el túnel.

- for s in /proc/sys/net/ipv4/conf/*; do echo 0 > $s/send_redirects; echo 0 > $s/accept_redirects; done

Se procede a configurar el rc de manera local y se escribe el mismo comando.

- vim /etc/rc.local
- for s in /proc/sys/net/ipv4/conf/*; do echo 0 > $s/send_redirects; echo 0 > $s/accept_redirects; done

Ahora se procede a configurar el Firewall del servidor para IPSec.

- iptables -F -t nat
- iptables -t nat -A POSTROUTING -j MASQUERADE
- iptables -L -t nat

Se guarda y se reinicia las iptables.

- /etc/init.d/iptables save
- /etc/init.d/iptables restart

4.4.2 Instalación de OpenSwan VPN

Instalar OpenSwan VPN con el comando y se procede a dar paso del arranque desde el sistema con:

- yum -y install openswan
- chkconfig ipsec on

Se reinicia el servicio ipsec.

- /etc/init.d/ipsec restart

Copiar el archivo de configuración de ipsec y ipsec secrets a la ruta etc/ipsec.conf_org y etc/ipsec.secrets_org respectivamente.

- cp /etc/ipsec.conf /etc/ipsec.conf_org

Se escribirán las siguientes sentencias:

- echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
- echo 0 > /proc/sys/net/ipv4/conf/default/send_redirects
- echo 0 > /proc/sys/net/ipv4/conf/eth0/send_redirects
- echo 0 > /proc/sys/net/ipv4/conf/eth1/send_redirects
- echo 0 > /proc/sys/net/ipv4/conf/lo/send_redirects
- echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects
- echo 0 > /proc/sys/net/ipv4/conf/default/accept_redirects
- echo 0 > /proc/sys/net/ipv4/conf/eth0/accept_redirects
- echo 0 > /proc/sys/net/ipv4/conf/eth1/accept_redirects
- echo 0 > /proc/sys/net/ipv4/conf/lo/accept_redirects

En este proceso se crean las claves para la autenticación entre los dos servidores.

- ipsec newhostkey -- configdir /etc/ipsec.d --output /etc/ipsec.secrets --bits 4096

Ahora se configura el nombre de cada servidor que tendrá la VPN para colocar sus respectivas claves de autenticación y de encriptación a nivel IP.

Comando es para el servidor A

- ipsec showhostkey –left

Comando es para el servidor B

- ipsec showhostkey –right

El comando para dar inicio del túnel en este caso es mytunnel.conf de configuración de IPs y de claves de cifrado con el uso de los protocolos de la misma.

- touch /etc/ipsec.d/mytunnel.conf

Se procede a configurar el túnel con el siguiente comando, para los dos servidores.

- nano /etc/ipsec.d/mytunnel.conf

Dentro de este archivo de configuración se colocar lo siguiente en el archivo:

- conn mytunnel
- authby=rsasig
- auto=start
- left=x.x.x.x
- leftrsasigkey=
- right=x.x.x.x

- rightrsasigkey=

En el parámetro left se coloca la IP pública del primer servidor, así como en right se coloca la IP para del segundo servidor B.

En la parte de leftrsasigkey y rightrsasigkey se coloca las llaves de autenticación elaboradas por los protocolos de autenticación.

Se debe instalar **WinSCP** en los servidores, para de esta manera poder copiar las claves de seguridad. Una vez ingresado correctamente al servidor A se debe dirigir a:

- A la carpeta etc
- Buscar donde se encuentra ipsec.secrets
- Abrir el editor de texto y proceder a copiar el pubkey generado por parte del primer servidor 1, como se observa en la figura 35.

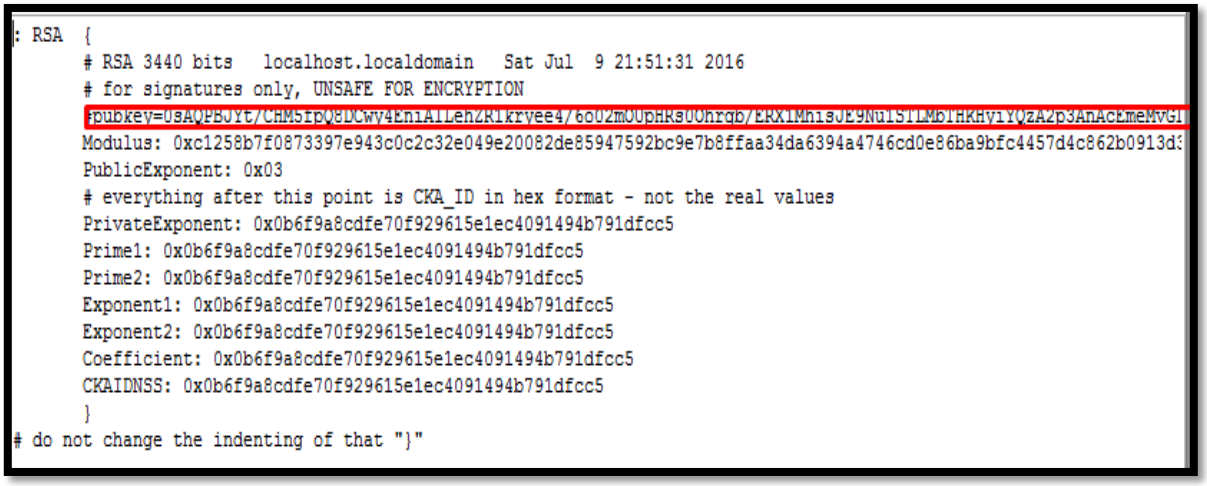


Figura 35. Clave generada por el protocolo de seguridad IPSec
Fuente: Elaboración propia Servidor Open VPN

Una vez copiado la clave del servidor A se procede a realizar lo siguiente:

- Dentro de la carpeta etc, buscar ipsec.d
- Una vez dentro de esta carpeta, abrir mytunnel.conf
- Es aquí en este archivo de texto donde se copiara la clave de seguridad

De la misma manera dirigirse al servidor B para poder copiar la clave para la autenticación de cifrado y que IPSec esté en funcionamiento.

Con el siguiente comando ver las llaves generadas por cada uno de los servidores

- cat /etc/ipsec.d/mytunnel.conf cat /etc/ipsec.d/mytunnel.conf

Se procede a ver el estado de configuración de ipsec, con el siguiente comando.

- nano /etc/ipsec.conf

Dentro de esta configuración observar que no se encuentre comentada la línea del archivo:

- include /etc/ipsec.d/*.conf

Proceder y reiniciar el servicio IPSec en el servidor, con el comando.

- service ipsec restart

Una vez iniciado verificar IPSec si se encuentra con algún error ya sea de Kernel o de algún otro parámetro de configuración.

- ipsec verify

Con los siguientes comandos IPSec se verifica si está habilitado y que se encuentre el protocolo de seguridad dentro del túnel VPN.

- setenforce 0
- ipsec auto --add mytunnel
- ipsec auto --up mytunnel

Una vez terminado de configurar IPSec se procede a verificar con un ping desde un servidor a otro servidor de la siguiente manera.

- El primer servidor A ejecutar el icmp con el comando ping.
- Al segundo servidor B ejecutamos el tcpdump -n -i eth0 esp para de esta manera observar el funcionamiento de los protocolos de seguridad, IPSec.

4.5 Configuración para la conexión de los servidores VPN con OpenVPN

La conexión se la realiza entre los servidores A y B de las dependencias respectivamente:

El servidor A se encuentra ubicado en la Universidad con sus respectivas direcciones:

- IP pública: 186.5.55.250
- IP privada del servidor 10.8.0.1

El servidor B se encuentra ubicado en el Hospital Viejo tenemos:

- IP pública: 190.95.196206
- IP privada del servidor 10.8.0.6

4.5.1 Generación de clave SSL

4.5.1.1 Servidor VPN A

Ahora se genera la clave SSL que asegurará las comunicaciones a través del túnel en el servidor A.

- openvpn --genkey --secret clave.key

De forma confidencial se coloca una clave para conectar con el servidor B.

Esta clave se debe copiar a /etc/openvpn en los dos servidores a conectar. Para copiar al segundo servidor emplear la herramienta scp:

- scp clave.key tuservidor.com:/etc/openvpn

Finalmente queda el archivo de configuración, en el primer servidor crear un archivo con el nombre del segundo para identificar fácilmente qué conexión es colocada.

- nano servidorB.conf

En este archivo se coloca lo siguiente, el puerto, el modo de trasporte que trabaja con udp, IP pública del servidor B con su respectivo puerto, las IPs privadas del servidor B y del servidor A en el archivo y para autenticarse los servidores requieren una clave que crea

el administrador, donde obligatoriamente requieren conocer los dos servidores A y B respectivamente.

- port 1194
- proto udp
- remote IP publica del servidor B 1194
- dev tun
- ifconfig 10.8.0.6 10.8.0.1
- secret clave.key
- comp-lzo
- keepalive 10 60
- ping-timer-rem
- persist-tun
- persist-key
- log /var/log/vpn.log

4.5.1.2 Servidor VPN B

Ahora se genera la clave SSL que asegurará las comunicaciones a través del túnel en el servidor B.

- nano servidorA.conf

En este archivo se coloca lo siguiente, el puerto, el modo de trasporte que trabaja con udp, IP pública del servidor A con su respectivo puerto, las IPs privadas del servidor A y del servidor B en el archivo y para autenticarse los servidores requieren una clave que crea el administrador, donde obligatoriamente requieren conocer los dos servidores B y A respectivamente.

- port 1194
- proto udp
- remote servidor1.com
- dev tun
- ifconfig 10.8.0.1 10.8.0.6
- secret clave.key
- comp-lzo
- keepalive 10 60
- ping-timer-rem
- persist-tun
- persist-key
- log /var/log/vpn.log

4.6 Pruebas de los servidores VPN

Pruebas de conexión entre Servidor A y Servidor B.

Las IP son virtuales de cada uno de los servidores y se observa la conexión entre ellos por medio del ping, que prueba su conectividad, como se observan en las figuras 36 y 37. Logrando de esta manera una comunicación entre los servidores A y B.

IP Servidor A: 10.8.0.1

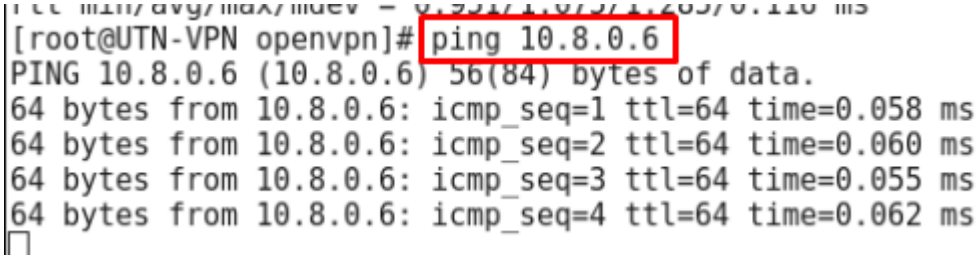


Figura 36. Ping de conectividad del servidor A al servidor B
Fuente: Elaboración propia

IP Servidor B: 10.8.0.6

```
[root@UTN-VPN openvpn]# ping 10.8.0.1
PING 10.8.0.1 (10.8.0.1) 56(84) bytes of data.
64 bytes from 10.8.0.1: icmp_seq=1 ttl=64 time=0.991 ms
64 bytes from 10.8.0.1: icmp_seq=2 ttl=64 time=0.951 ms
64 bytes from 10.8.0.1: icmp_seq=3 ttl=64 time=1.10 ms
64 bytes from 10.8.0.1: icmp_seq=4 ttl=64 time=1.17 ms
64 bytes from 10.8.0.1: icmp_seq=5 ttl=64 time=1.04 ms
64 bytes from 10.8.0.1: icmp_seq=6 ttl=64 time=1.04 ms
64 bytes from 10.8.0.1: icmp_seq=7 ttl=64 time=0.951 ms
64 bytes from 10.8.0.1: icmp_seq=8 ttl=64 time=0.982 ms
64 bytes from 10.8.0.1: icmp_seq=9 ttl=64 time=1.22 ms
64 bytes from 10.8.0.1: icmp_seq=10 ttl=64 time=1.28 ms
^C
```

Figura 37. Ping de conectividad del servidor B al servidor A
Fuente: Elaboración propia

La prueba realizada en esta ocasión se realizó en función de la conexión entre el cliente y al servidor OpenVPN a través de un ping, por medio de la IP pública, como se muestra en la figura 38.

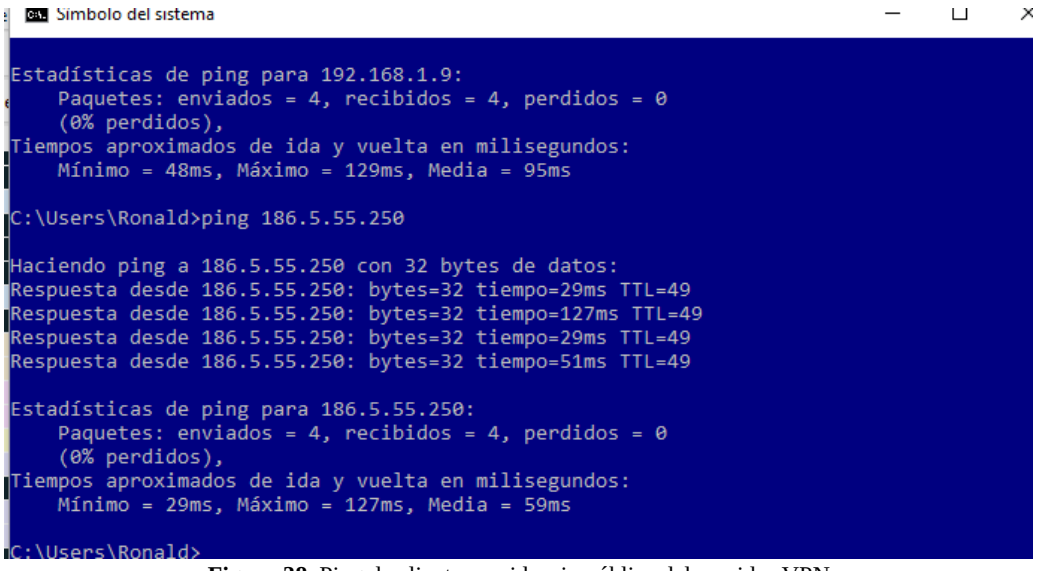


Figura 38. Ping de cliente-servidor, ip pública del servidor VPN
Fuente: Elaboración propia

Verificación de Protocolo OpenVPN y encriptación de paquetes

La prueba se realizó a través de la herramienta denominada Wireshark, para capturar el tráfico UDP, TCP, durante la ejecución del protocolo VPN, como se muestra en la figura 39.

En la figura 39, se observa parámetros como el origen, destino y la ejecución del protocolo OpenVPN con la información correspondiente, observada a través de la herramienta que captura del tráfico ejecutado.Protocolo: OpenVPN

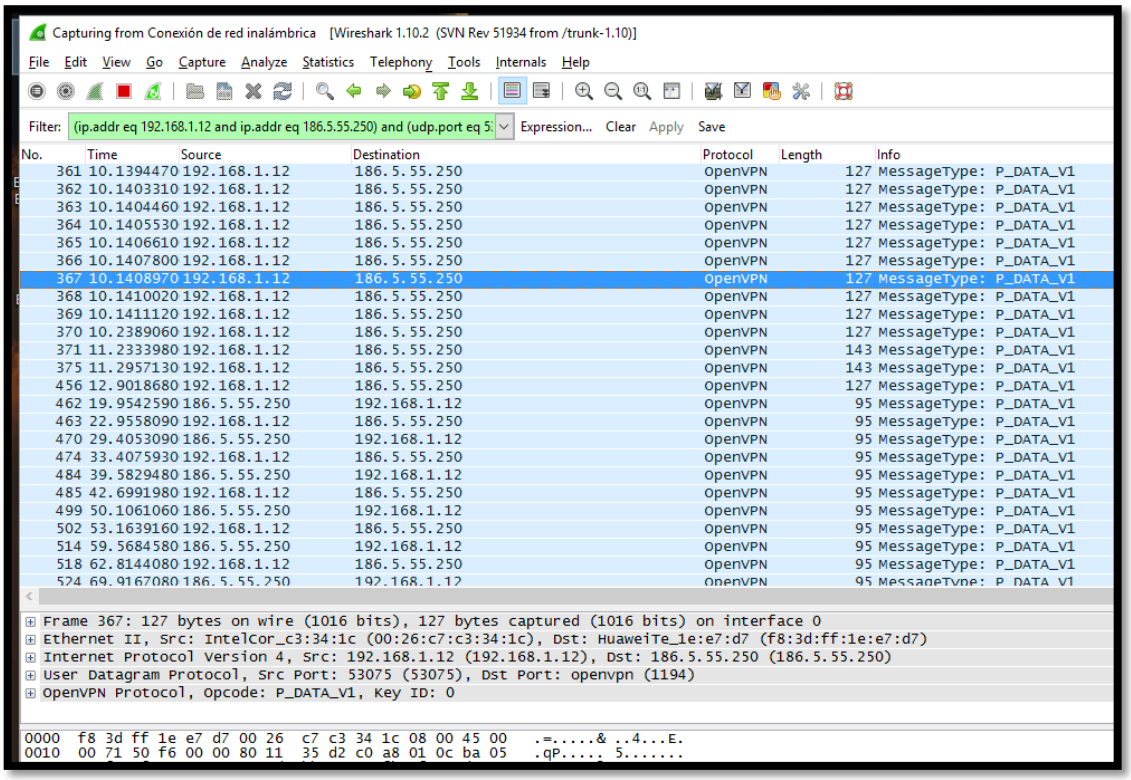


Figura 39. Corriendo el protocolo VPN en el analizador wireshark
Fuente: Elaboración propia

Se puede observar que el servidor VPN está funcionando correctamente y se analiza el flujo UDP de la red a la que se encuentra conectado el cliente VPN, como se observa en la figura 40, dando lugar a la verificación del tráfico dentro del servidor.

Al generar tráfico se captura determinada información que circula por la Red establecida para el servidor VPN, por lo tanto, se realiza un filtrado de paquetes udp.

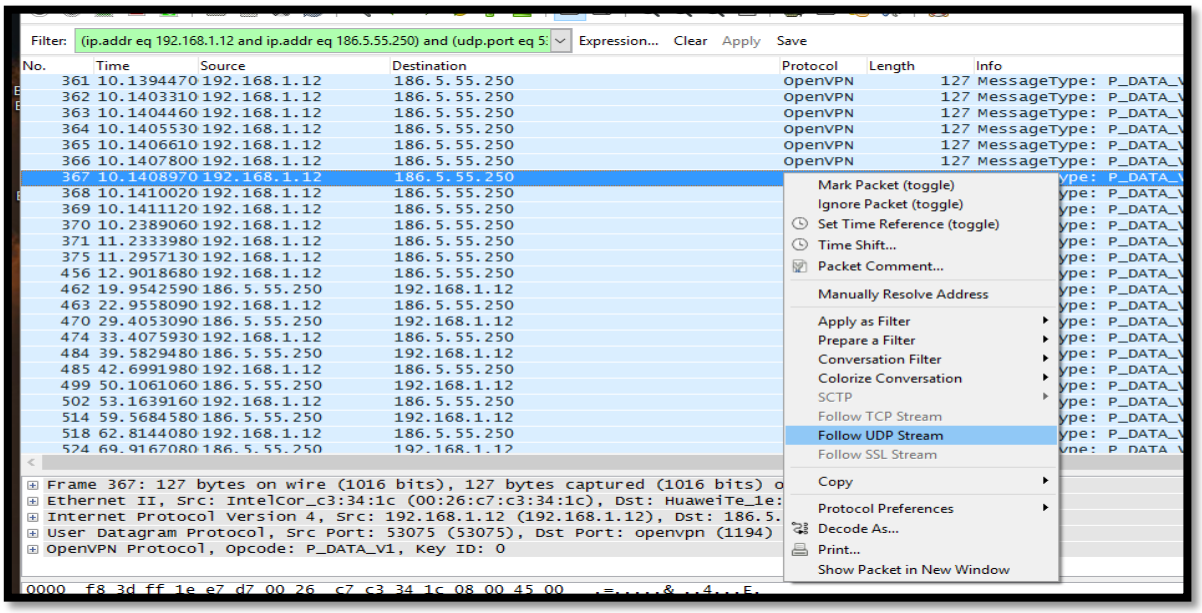


Figura 40. Trafico UDP del servidor VPN con el analizador wireshark
Fuente: Elaboración propia

En la selección de uno de los paquetes UDP capturados por medio de wireshark que se transmite por el túnel VPN, se observa que la información que se encuentra encriptada, como se señala en la figura 41.

Representa los datos que se encuentran encriptados por los protocolos de seguridad, generados por los servidores como Diffie Hellman.

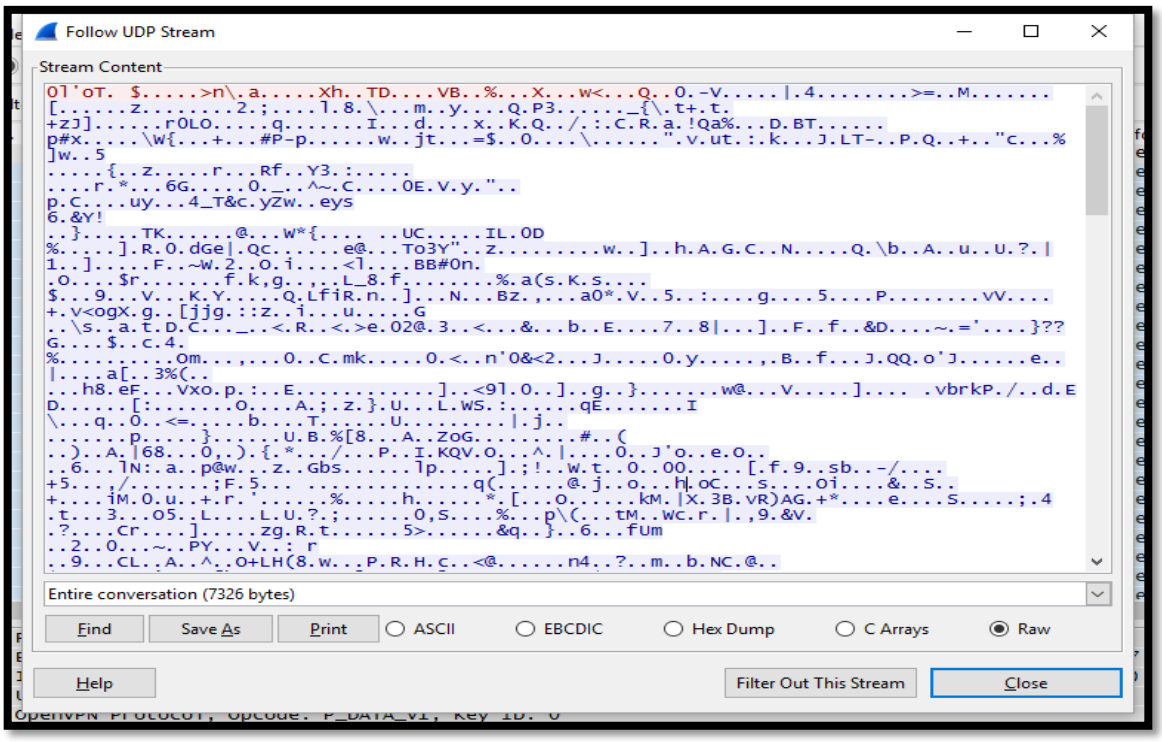


Figura 41. Encriptado todo el tráfico que pasa por el túnel VPN
Fuente: Elaboración propia

La prueba realizada en este caso está en función del tráfico generado, en función de diferentes protocolos como: OpenVPN, TCP, SSL, ARP entre otros creado por los clientes que pasan a través del túnel del servidor y se observa en la herramienta Wireshark, detallada en la figura 42.

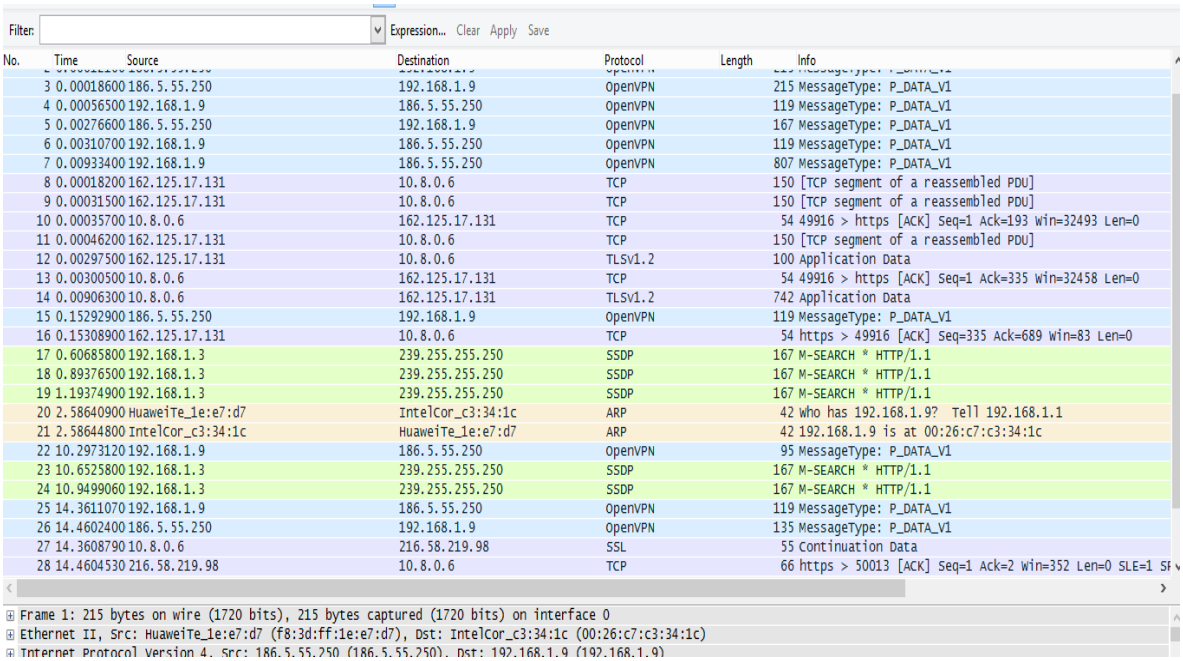


Figura 42. Tráfico que pasa por el túnel VPN
Fuente: Elaboración propia

CAPÍTULO V

5. Control de los diferentes tipos de tráfico con el uso de la Red Privada Virtual VPN

Se detalla a través de la implementación de la VPN, el control de tráfico establecido entre el edificio Central y el Hospital Viejo.

Asignación y configuración de servidores en la Red de la UTN.

Dentro de la configuración se detallan características en función de la seguridad que se aplica en la implementación de la VPN, Véase en el **Anexo P.**

Interfaces del servidor 1:

- LOCAL_IFACE="eth1" # Conexión a la red local
- INET_IFACE="eth0" # Conexión a Internet (IP PÚBLICA)
- PRIVATE=10.8.0.1/24 # IP del servidor VPN
- IP ”eth1” 172.16.4.0
- IP ”eth0” 186.5.55.250

Interfaces del servidor 2:

- LOCAL_IFACE="eth1" # Conexión a la red local
- INET_IFACE="eth0" # Conexión a Internet (IP PÚBLICA)
- PRIVATE=10.8.0.6/24 # IP del servidor VPN
- IP ”eth1” 172.16.16.0
- IP ”eth0” 190.95.196.206

Para el Firewall de los servidores de la Red Privada Virtual se activa el reenvió de paquetes Packet forwarding.

- net.ipv4.ip_forward = 1

Se configura el firewall en los servidores VPN con iptables que viene integrado en el kernel de Linux, con el siguiente comando verificamos las reglas que se encuentran configuradas en los servidores.

- iptables –L

Aceptar el puerto OPENVPN en los servidores para aceptar el tráfico que viaja a través del túnel.

- iptables -A INPUT -i eth0 -m state --state NEW -p udp --dport 1194 -j ACCEPT

Para interfaces de tipo túnel se trasmita a través de otras interfaces.

- iptables -A FORWARD -i tun+ -j ACCEPT
- iptables -A FORWARD -o tun+ -j ACCEPT
- iptables -A OUTPUT -o tun+ -j ACCEPT
- iptables -A FORWARD -i tun+ -o eth0 -m state --state RELATED,ESTABLISHED -j ACCEPT
- iptables -A FORWARD -i eth0 -o tun+ -m state --state RELATED,ESTABLISHED -j ACCEPT

Para el NAT del tráfico de cliente VPN al internet.

- iptables -t nat -A POSTROUTING -s 192.168.0.0/24 -o eth0 -j MASQUERADE

Esta regla para que acepte el tráfico que proviene del túnel a la dirección IP de nuestro servidor.

- iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -j SNAT --to-source 186.5.55.250

Para poder controlar donde se envian los paquetes dentro de una LAN o para permitir el reenvío a la LAN se configura las reglas siguientes:

- iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
- iptables -A FORWARD -s 10.8.0.0/24 -j ACCEPT
- iptables -A FORWARD –j REJECT

Con estas reglas se configura los puertos de seguridad en los servidores, como lo son el puerto 53 TCP/UDP Sistema de nombres de dominio (DNS), puerto 500 que es el puerto udp de seguridad IPSec, ISAKMP, Autoridad de Seguridad Local, el puerto 4500 udp que es el IPSec NAT Traversal. A INPUT -p udp -m state --state NEW --dport 53 -j ACCEPT, aceptar el tráfico icmp, interfaz local, el puerto 22 tcp ssh.

- A INPUT -p tcp -m state --state NEW --dport 53 -j ACCEPT
- A INPUT -p udp --sport 500 --dport 500 -j ACCEPT
- A INPUT -p udp --sport 4500 --dport 4500 -j ACCEPT
- A OUTPUT -p udp --sport 500 --dport 500 -j ACCEPT
- A OUTPUT -p udp --sport 4500 --dport 4500 -j ACCEPT
- A INPUT -m state --state ESTABLISHED, RELATED -j ACCEPT
- A INPUT -p icmp -j ACCEPT
- A INPUT -i lo -j ACCEPT
- A INPUT -m state --state NEW -m tcp -p --dport 22 -j ACCEPT
- A INPUT -j REJECT --reject-with icmp-host-prohibited
- A FORWARD -j REJECT --reject-with icmp-host-prohibited
- nano /etc/sysconfig/iptables

Políticas predeterminadas

- iptables -P OUTPUT ACCEPT
- iptables -P INPUT LOG_DROP
- iptables -P FORWARD LOG_DROP

Evitar que los paquetes externos utilicen la interface loopback

- iptables -A INPUT -i $INET_IFACE -s $LOOP -j LOG_DROP
- iptables -A FORWARD -i $INET_IFACE -s $LOOP -j LOG_DROP
- iptables -A INPUT -i $INET_IFACE -d $LOOP -j LOG_DROP
- iptables -A FORWARD -i $INET_IFACE -d $LOOP -j LOG_DROP

Eliminamos los paquetes de clases de direcciones reservadas

- iptables -A FORWARD -i $INET_IFACE -s 192.168.0.0/16 -j LOG_DROP
- iptables -A FORWARD -i $INET_IFACE -s 172.16.0.0/12 -j LOG_DROP
- iptables -A FORWARD -i $INET_IFACE -s 10.0.0.0/8 -j LOG_DROP
- iptables -A INPUT -i $INET_IFACE -s 192.168.0.0/16 -j LOG_DROP
- iptables -A INPUT -i $INET_IFACE -s 172.16.0.0/12 -j LOG_DROP
- iptables -A INPUT -i $INET_IFACE -s 10.0.0.0/8 -j LOG_DROP

Bloquear el trafico NetBios saliente hacia Internet debido a que Netbios es una interfaz de programación de aplicaciones que pueden utilizar los programas en una red de área local (LAN).

- iptables -A FORWARD -p tcp --sport 137:139 -o $INET_IFACE -j LOG_DROP
- iptables -A FORWARD -p udp --sport 137:139 -o $INET_IFACE -j LOG_DROP
- iptables -A OUTPUT -p tcp --sport 137:139 -o $INET_IFACE -j LOG_DROP
- iptables -A OUTPUT -p udp --sport 137:139 -o $INET_IFACE -j LOG_DROP

Validación de la dirección origen de los paquetes salientes

- iptables -A FORWARD -s ! $PRIVATE -i $LOCAL_IFACE -j LOG_DROP

Permitir accesos locales a la interface loopback

- iptables -A INPUT -s $LOOP -j ACCEPT
- iptables -A INPUT -d $LOOP -j ACCEPT

ICMP aceptados (pueden ser desactivados según necesidades de seguridad)

- iptables -A INPUT -p icmp --icmp-type destination-unreachable -j ACCEPT
- iptables -A INPUT -p icmp --icmp-type source-quench -j ACCEPT
- iptables -A INPUT -p icmp --icmp-type time-exceeded -j ACCEPT
- iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
- iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT

Permitir servicios de Correo, Web, DNS y SSH

- iptables -A INPUT -p tcp --dport http -j ACCEPT
- iptables -A INPUT -p tcp --dport https -j ACCEPT
- iptables -A INPUT -p tcp --dport ssh -j ACCEPT
- iptables -A INPUT -p tcp --dport smtp -j ACCEPT
- iptables -A INPUT -p udp --dport domain -j ACCEPT
- iptables -A INPUT -p tcp --dport domain -j ACCEPT

PARAMETROS EXCLUSIVOS OPENVPN

Permitir paquetes entrantes desde el túnel OpenVPN

- iptables -A INPUT -p udp --dport 1194 -j ACCEPT

Permitir paquetes desde los dispositivos TUN/TAP

- iptables -A INPUT -i tun+ -j ACCEPT
- iptables -A FORWARD -i tun+ -j ACCEPT
- iptables -A INPUT -i tap+ -j ACCEPT
- iptables -A FORWARD -i tap+ -j ACCEPT

Permitir paquetes desde la red privada

- iptables -A INPUT -i $LOCAL_IFACE -j ACCEPT
- iptables -A FORWARD -i $LOCAL_IFACE -j ACCEPT

Mantener estado de las conexiones desde la red local

- iptables -A OUTPUT -m state --state NEW -o $INET_IFACE -j ACCEPT
- iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
- iptables -A FORWARD -m state --state NEW -o $INET_IFACE -j ACCEPT
- iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT

Enmascarar la red local

- iptables -t nat -A POSTROUTING -s $PRIVATE -o $INET_IFACE -j MASQUERADE

Se guarda y se reinician las iptables.

- service iptables save
- service iptables restart

Borrar reglas antiguas y bloquear todo el tráfico

- iptables –F
- iptables -P OUTPUT DROP
- iptables -P INPUT DROP
- iptables -P FORWARD DROP

CAPÍTULO VI

6.1 CONCLUSIONES

- Se realizó la implementación de una Red Privada Virtual en la Universidad Técnica del Norte que comunicara el Antiguo Hospital con el Edificio Central de la Universidad, logrando de esta manera una comunicación segura a través de la VPN y de protocolos a nivel de IP, obteniendo seguridad, confidencialidad e integridad de los datos que hoy en día es de vital importancia en las instituciones u organizaciones.

- Se configuró el tipo de seguridad en el túnel de la Red Privada Virtual con el protocolo Internet Protocol Security - IPSec el cual asegura las comunicaciones sobre el Protocolo de Internet IP autenticando y cifrando cada paquete IP, logrando de esta manera una comunicación segura entre las redes remotas en un canal inseguro.

- Para el caso de la autenticación de los clientes que logren conectarse de una manera segura se empleó protocolos de seguridad como Diffie-Hellman y los certificados previos del servidor que ayudaron a fortalecer la autenticación de cliente-servidor y obtener una comunicación segura a través de la red privada virtual.

- Con la Red Privada Virtual VPN se logró mejorar la comunicación entre el Antiguo Hospital y el Edificio Centro de la Universidad Técnica del Norte, permitiendo de esta manera conectarse desde la comodidad de sus casas a la red privada virtual y poder acceder a la red desde cualquier lugar que se encuentren sobre un canal seguro.

- El protocolo IPSec es el encargado de brindar la seguridad a modo de túnel entre los dos servidores VPN que se encuentran configurados con software libre GNU/LINUX con el sistema operativo CentOS, con el cual se conecta a través de las ip públicas de manera segura.

- Las redes privadas virtuales representan un enlace de comunicación confidencial, seguro que aporta la confidencialidad e integridad de los datos obteniendo un control de tráfico a través de políticas de firewall implementadas en los dos servidores VPN obteniendo de tal manera una seguridad aún mayor en la red.

6.2 RECOMENDACIONES

- Las instituciones deberían contar con una Red Privada Virtual VPN, ya que su costo es considerablemente bajo debido a que se lo puede implementar en una plataforma como lo es software libre y lo mejor de todo convertirlo en un canal

seguro para acceder a la red de la institución es este caso a la red de la Universidad Técnica del Norte.

- Los servidores deben de tener características buenas como por ejemplo que tenga su memoria RAM mayor a 1Gb, velocidades de procesador mayores a 2Ghz, sus discos duros mayores a 40 Gb, verificar que su tarjeta de red funcione correctamente para obtener el mayor servicio de estos servidores VPN,

- Es recomendable observar la velocidad con la que cuenta cada servidor de ip públicas para ver de esta manara si la tarjeta de red está funcionando de manera correcta y evaluar al proveedor de internet el cual es de suma importancia en el rendimiento de la Red Privada Virtual VPN.

- Se recomienda cuantificar el ancho de banda ya que van a pasar datos importantes por este canal seguro y se recomienda tener un buen ancho de banda que pueda soportar el tráfico y administrarlo de la mejor manera, de esta manera logrando al máximo el rendimiento de los servidores que se encuentran en el Hospital Antiguo y en el edificio de la Universidad Técnica del Norte.

BIBLIOGRAFÍA

Alejandro. (Febrero de 2012). *Mas de Redes*. Obtenido de Mas de Redes: http://enredajo.blogspot.com/2009/03/que-es-una-vpn-y-tipos-de-vpn.html

Aplicaciones Tecnologicas. (2013). *Una VPN es una red VIRTUAL PRIVADA a la cual se puede tener acceso a través de INTERNET*. Obtenido de Una VPN es una red VIRTUAL PRIVADA a la cual se puede tener acceso a través de INTERNET: http://www.aplicacionestecnologicas.com/Computadoras_Redes/Una_VPN_es/index.html

Brown, S. I. (s.f.). *Fundamentos de las Redes Privadas Virtuales (VPN)*. Obtenido de Fundamentos de las Redes Privadas Virtuales (VPN): http://virtual.uaeh.edu.mx/repositoriooa/paginas/Vpn/bibliografa.html

BUSTAMANTE, L. G. (2012). DISEÑO E IMPLEMENTACIÓN DE UNA INFRAESTRUCTURA DE SERVICIOS DE RED Y RESGUARDO DE SERVIDORES. En L. G. BUSTAMANTE, *DISEÑO E IMPLEMENTACIÓN DE UNA INFRAESTRUCTURA DE SERVICIOS DE RED Y RESGUARDO DE SERVIDORES.* Quito.

BUSTAMANTE, L. G. (Mayo de 2012). *FACULTAD DE SISTEMAS Y TELECOMUNICACIONES*. Obtenido de FACULTAD DE SISTEMAS Y TELECOMUNICACIONES: http://repositorio.uisek.edu.ec/jspui/bitstream/123456789/534/1/TESIS%20FINAL%20LUIS%20GUILLERMO%20LE%C3%93N%20BUSTAMANTE.pdf

Ciberaula-Linux. (2015). *Ciberaula-Linux*. Obtenido de Ciberaula-Linux: http://linux.ciberaula.com/articulo/que_es_linux/

Cooper, S. B. (s.f.). *eHOW*. Obtenido de eHOW: http://www.ehowenespanol.com/servidor-radius-info_376327/

Corrales, J. M. (s.f.). *Tuneles VPN*. Obtenido de Tuneles VPN: http://informatica.gonzalonazareno.org/proyectos/2009-10/jmc.pdf

Crespo, J. P. (09 de 02 de 2500). *http://blackspiral.org/docs/pfc/itis/node5.html*. Obtenido de http://blackspiral.org/docs/pfc/itis/node5.html.

Dias, J. (22 de Enero de 2015). *10incobe_Trabajando por la confianza digital*. Obtenido de 10incobe_Trabajando por la confianza digital: https://www.incibe.es/blogs/post/Seguridad/BlogSeguridad/Articulo_y_comentarios/autenticacion_passwords

Diaz, J. (22 de Enero de 2015). *Autenticación basada en contraseñas.* Obtenido de Autenticación basada en contraseñas:

https://www.incibe.es/blogs/post/Seguridad/BlogSeguridad/Articulo_y_comentarios/autenticacion_passwords

Enders, R. (Noviembre de 2012). *Search DataCenter*. Obtenido de Search DataCenter: http://searchdatacenter.techtarget.com/es/respuesta/Como-asegurarme-de-que-mi-VPN-trabajara-con-nuestro-firewall

Galarza, L. M. (s.f.). *FUNDAMENTOS DE COMPUTACION VPN*.

GOUJON, A. (10 de Septiembre de 2012). *welivesecurity*. Obtenido de welivesecurity: http://www.welivesecurity.com/la-es/2012/09/10/vpn-funcionamiento-privacidad-informacion/

Grupo de Sistemas Operativos DATSI FI UPM. (2016). *Protocolo IPsec*. Obtenido de Protocolo IPsec: http://laurel.datsi.fi.upm.es/proyectos/teldatsi/teldatsi/protocolos_de_comunicaciones/protocolo_ipsec

Hall, J. C. (s.f.). Obtenido de http://datateca.unad.edu.co/contenidos/233011/233011Exe/leccin_1_criptografa_de_clave_simtrica.html

Henao, D. P. (Diciembre de 2012). *Diseño E Implementacion de una Red Privada Segura basada en Linux.* Obtenido de Diseño E Implementacion de una Red Privada Segura

basada en Linux: http://dspace.ups.edu.ec/bitstream/123456789/5087/1/UPS-ST000997.pdf

Luz, S. D. (9 de 11 de 2010). *Criptografía : Algoritmos de autenticación (hash)*. Obtenido de Criptografía : Algoritmos de autenticación (hash): http://www.redeszone.net/2010/11/09/criptografia-algoritmos-de-autenticacion-hash/

Luz, S. d. (9 de Noviembre de 2010). *Redes@Zone*. Obtenido de Redes@Zone: http://www.redeszone.net/2010/11/09/criptografia-algoritmos-de-autenticacion-hash/

Luz, S. D. (9 de noviembre de 2010). *Redes@zone*. Obtenido de Redes@zone: http://www.redeszone.net/2010/11/09/criptografia-algoritmos-de-autenticacion-hash/

Mario, L. (2013). *FUNDAMENTOS DE COMPUTACION*. Obtenido de FUNDAMENTOS DE COMPUTACION: https://www.google.com.ec/url?sa=t&rct=j&q=&esrc=s&source=web&cd=6&ved=0ahUKEwiYnYS91pfMAhWM7yYKHR7ZCwMQFgg6MAU&url=http%3A%2F%2Fwww.ecotec.edu.ec%2Fdocumentacion%255Cinvestigaciones%255Cestudiantes%255Ctrabajos_de_clases%2F1584_TRECALDE_0034.doc&usg=AFQj

Microsoft. (2016). *Microsoft*. Obtenido de Microsoft: https://msdn.microsoft.com/es-ec/library/cc736357(v=ws.10).aspx

Microsoft. (2016). *Microsoft*. Obtenido de Microsoft: https://msdn.microsoft.com/es-es/library/cc782786(v=ws.10).aspx

Microsoft. (2016). *Microsoft-IAS-Tuneles*. Obtenido de Microsoft-IAS-Tuneles: https://msdn.microsoft.com/es-es/library/cc776016(v=ws.10).aspx

Microsoft. (2016). *Protocolo de túnel punto a punto (PPTP)*. Obtenido de Protocolo de túnel punto a punto (PPTP): https://msdn.microsoft.com/es-es/library/cc739465(v=ws.10).aspx

Microsoft. (2016). *Protocolos de túnel VPN*. Obtenido de Protocolos de túnel VPN: https://technet.microsoft.com/es-es/library/cc771298(v=ws.10).aspx

Morales, A. G. (s.f.). Obtenido de http://www.uaeh.edu.mx/docencia/Tesis/icbi/licenciatura/documentos/Redes%20privadas%20virtuales.pdf

Morales, A. G. (2010). Redes Privadas Virtuales. En A. G. Morales, *Redes Privadas Virtuales.* Pachuca.

netdatanetworks. (30 de Agosto de 2011). *Necesidades de Comunicación y Seguridad: VPN (IPSec/SSL)*. Obtenido de Necesidades de Comunicación y Seguridad: VPN

(IPSec/SSL): https://netdatanetworks.wordpress.com/2011/08/30/necesidades-de-comunicacion-y-seguridad-vpn-ipsecssl/

NETDATANETWORKS. (Agosto de 2011). *Necesidades de Comunicación y Seguridad: VPN (IPSec/SSL).* Obtenido de Necesidades de Comunicación y Seguridad: VPN (IPSec/SSL): https://netdatanetworks.wordpress.com/2011/08/30/necesidades-de-comunicacion-y-seguridad-vpn-ipsecssl/

NETDATA-NETWORKS. (30 de Agosto de 2011). *Necesidades de Comunicación y Seguridad: VPN (IPSec/SSL).* Obtenido de Necesidades de Comunicación y Seguridad: VPN (IPSec/SSL): https://netdatanetworks.wordpress.com/2011/08/30/necesidades-de-comunicacion-y-seguridad-vpn-ipsecssl/

Ñacato, M. A. (2007). Diseño e Implementación de una Red Privada Virtual para la Empresa Hato Telecomunicaciones. En M. A. Ñacato, *Diseño e Implementación de una Red Privada Virtual para la Empresa Hato Telecomunicaciones.* Quito.

RUEDA, I. G. (2011). *"Diseño de una red telemática con VPN e internet para la Dirección Provincial de Salud de Loja".* Obtenido de "Diseño de una red telemática con VPN e internet para la Dirección Provincial de Salud de Loja": http://dspace.ucuenca.edu.ec/bitstream/123456789/2551/1/tm4470.pdf

Sandoval, B. (s.f.). Obtenido de https://securitcrs.wordpress.com/criptografia/criptografia-de-clave-privada/

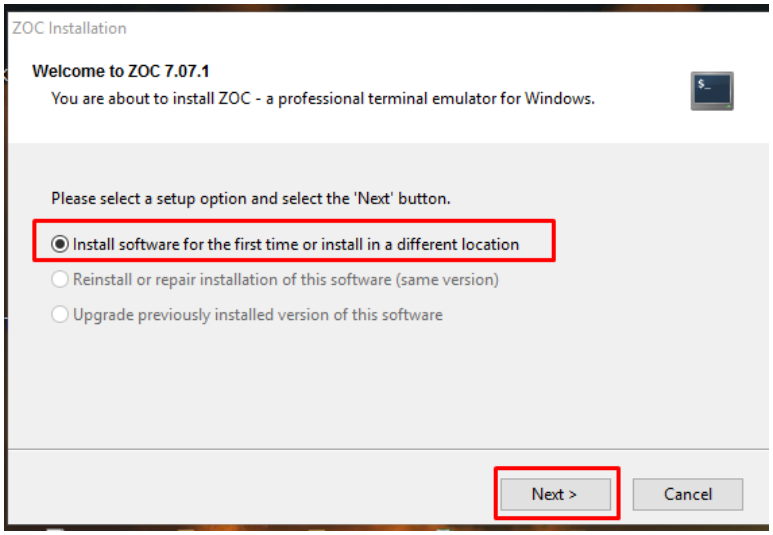
Tanenbaum, A. S. (2009). *Redes de Computadoras.* Mexico: Person-Educación.

Tanenbaum, A. S. (2010). *Redes de Computadoras.* Mexico: Pearson Educación.

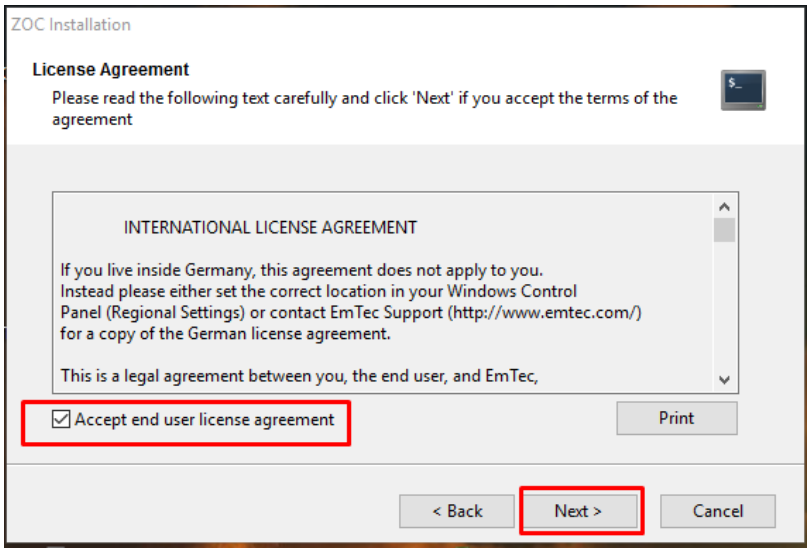
ANEXOS

Anexo A. Instalación del Protocolo SSH

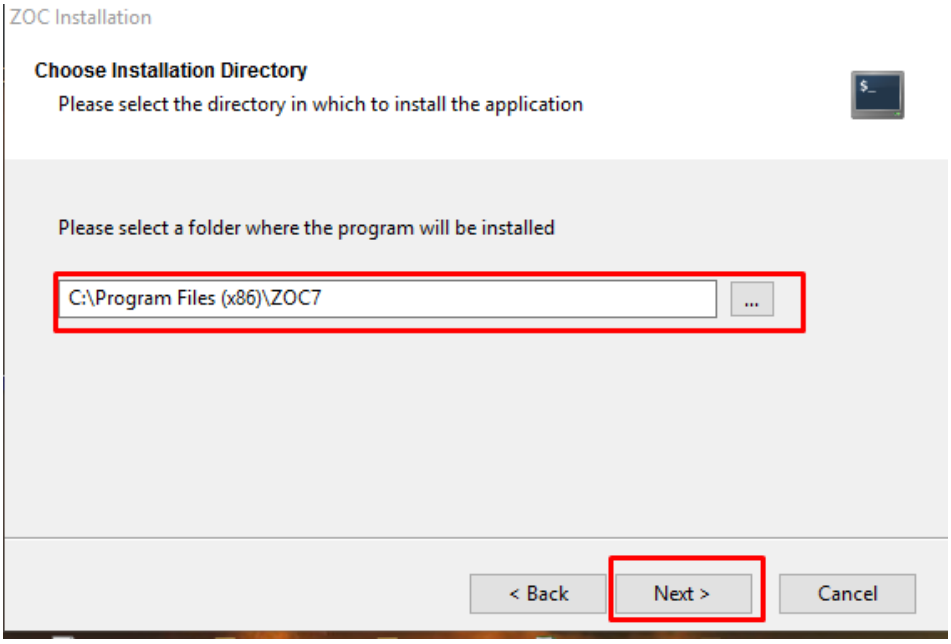
En la siguiente figura se instala el Software ZOC 7.07.1, de la siguiente manera.



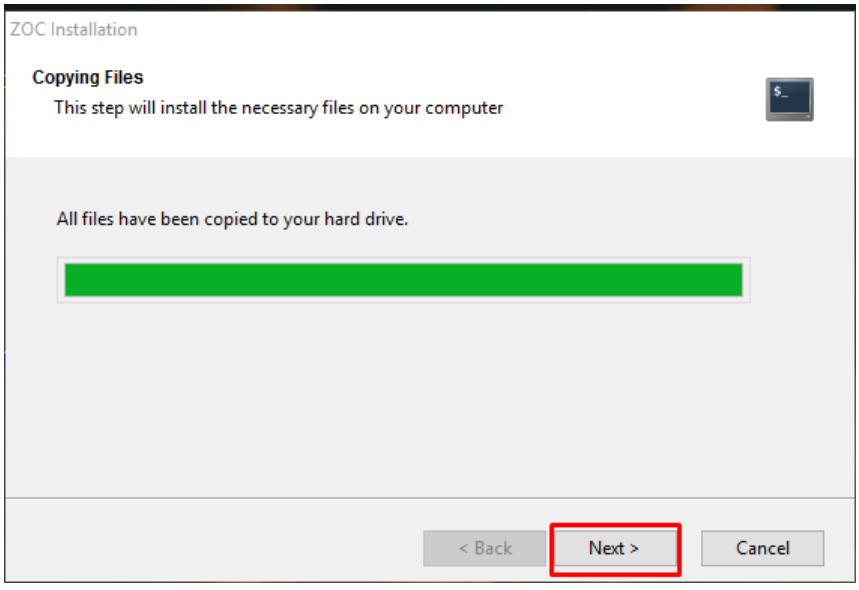
Aceptamos el acuerdo de licencia y procedemos a dar en siguiente.



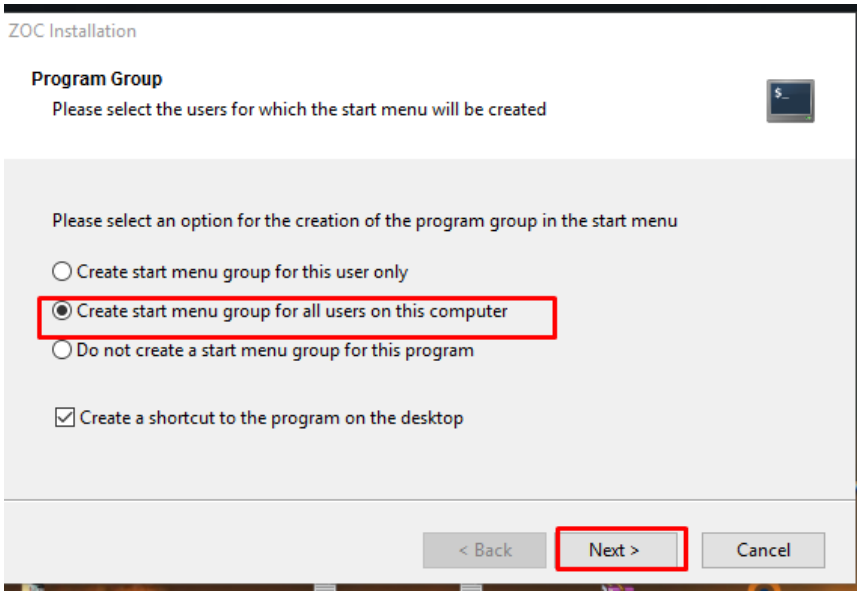
Aquí escogemos la ubicación donde se va a instalar el software, por defecto se deja en la misma ubicación.



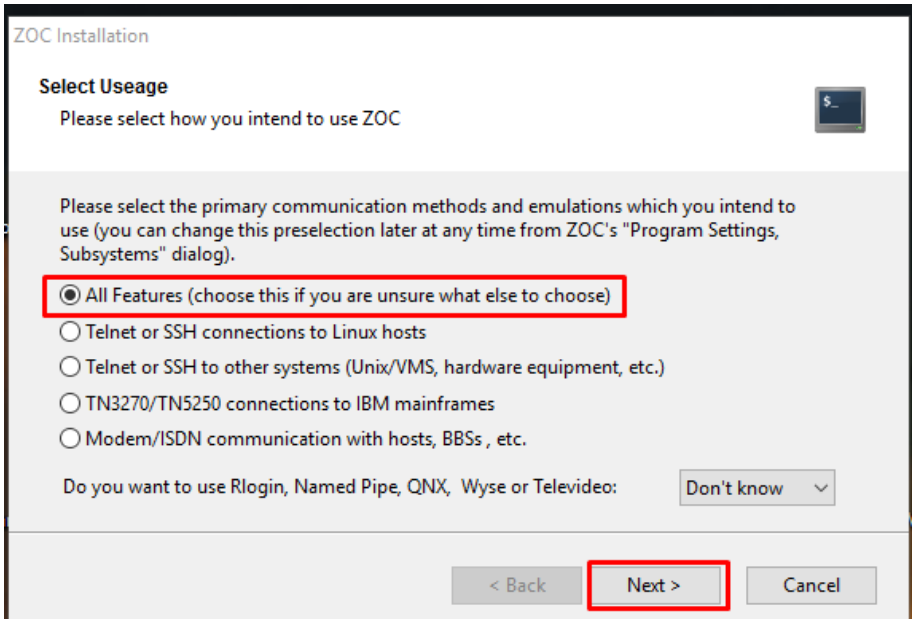
Esperamos a que se instale en programa.



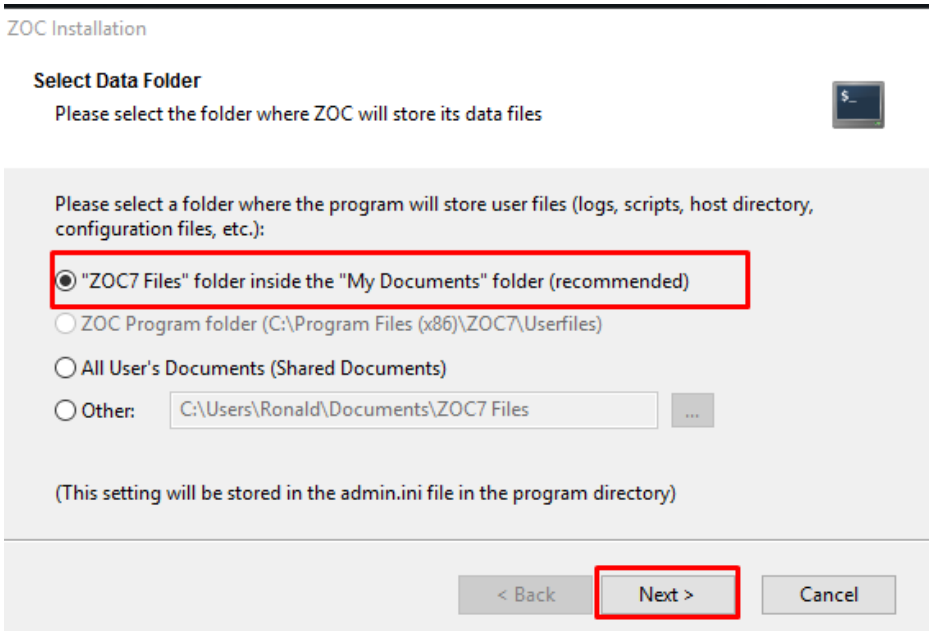
Creamos un grupo de usuarios para esta máquina para que puedan ingresan a través de ssh a diferente servidores.



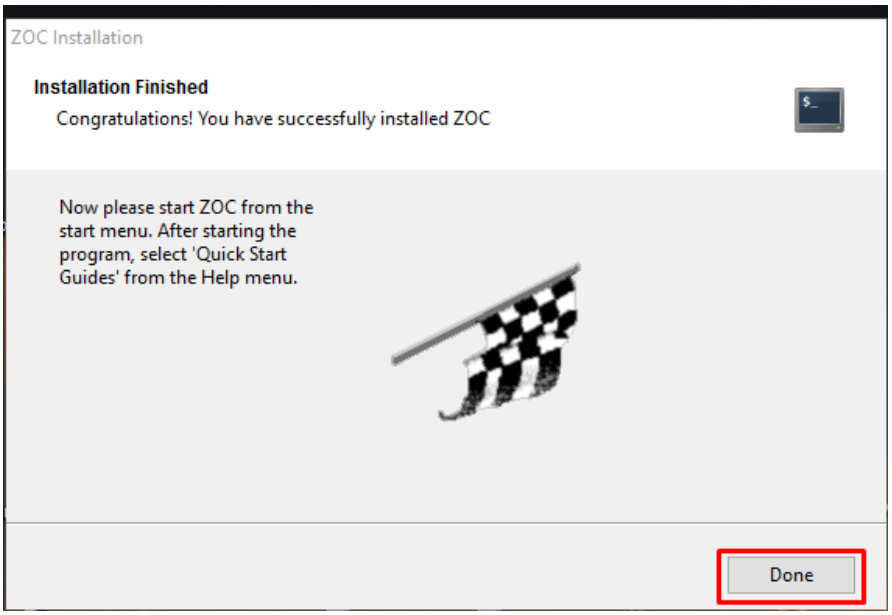
Escogemos todas las características del programa,

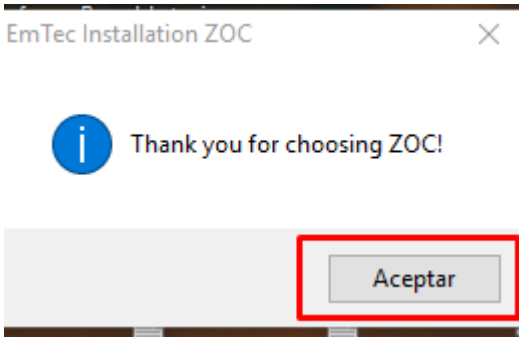


Se crea una carpeta del programa Zoc.

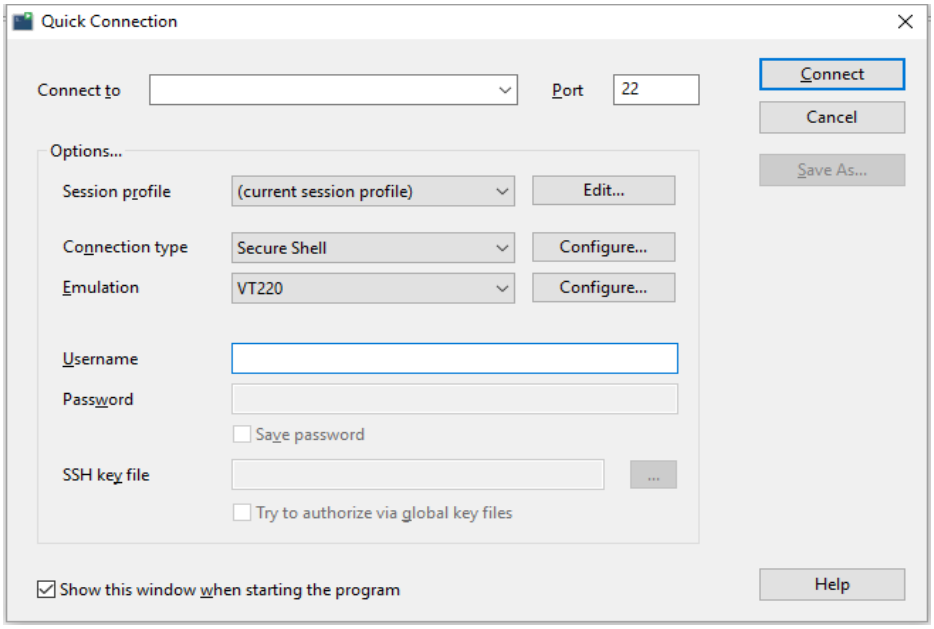


Terminamos la instalación.

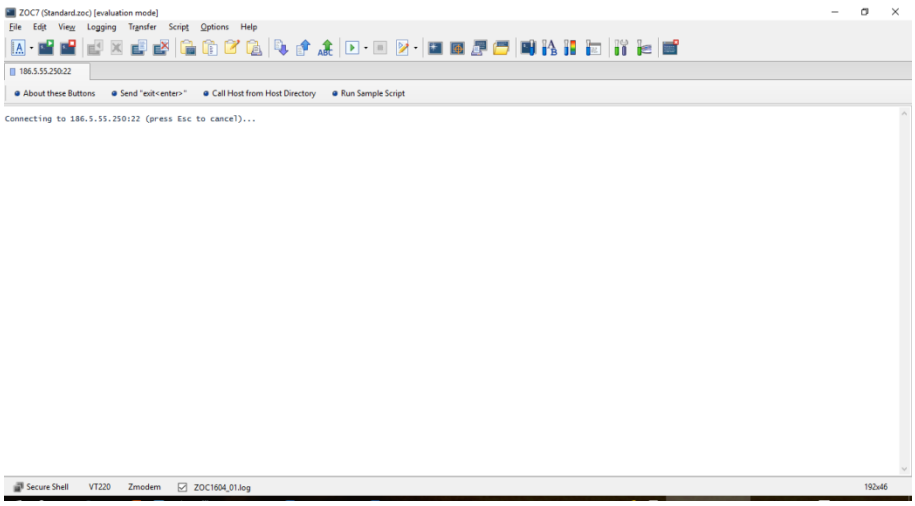




Por ultimo configuramos el Software Zoc con la IP pública del servidor, su usuario, su password, el puerto con el cual se va a comunicar y damos clic en conectar.



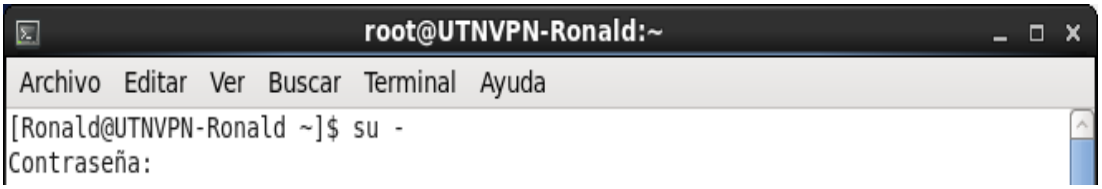
Una vez configurado todo nos saldrá la siguiente ventana, que nos indica que estamos en dentro de nuestro servidor, listo para poder configurarlo a través de una conexión segura.


ZOC7 (Standard.zoc) [evaluation mode]
File Edit View Logging Transfer Script Options Help
186.5.55.250:22
About these Buttons
Send "exit<enter>"
Call Host from Host Directory
Run Sample Script
Connecting to 186.5.55.250:22 (press Esc to cancel)...
Secure Shell
VT220
Zmodem
ZOC1604_01.log
192x46

Anexo B. Instalación de Repositorios

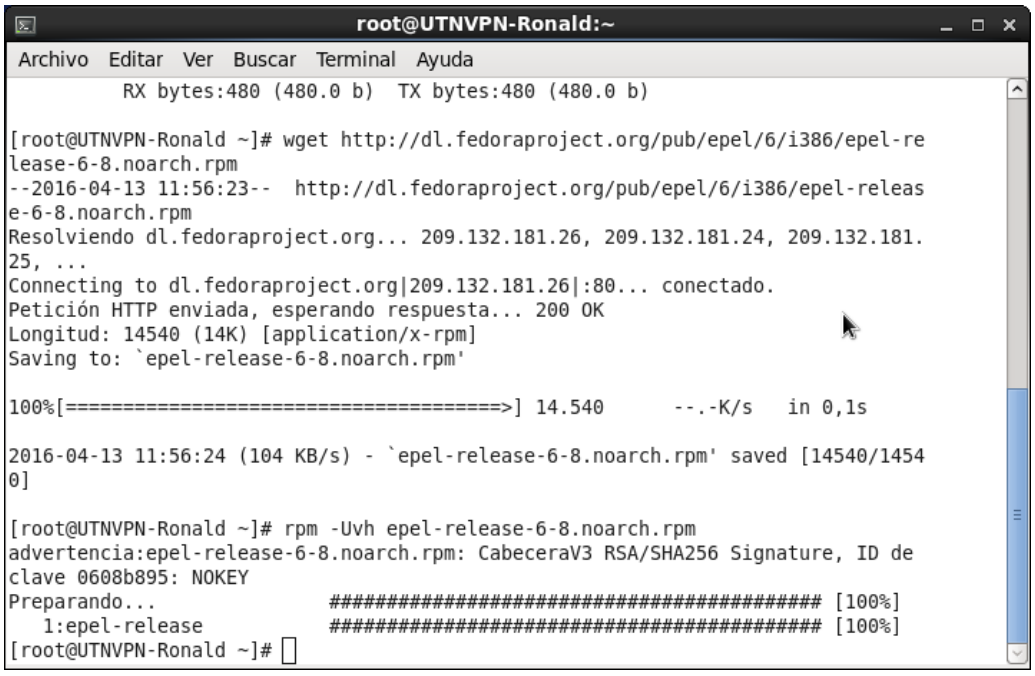
Nos ingresamos como súper usuarios con el siguiente comando:

- su -
  Ingresamos la clave para acceder como súper usuarios:



Para instalar OpenVPN se necesita instalar el siguiente repositorio:

- wget http://dl.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm



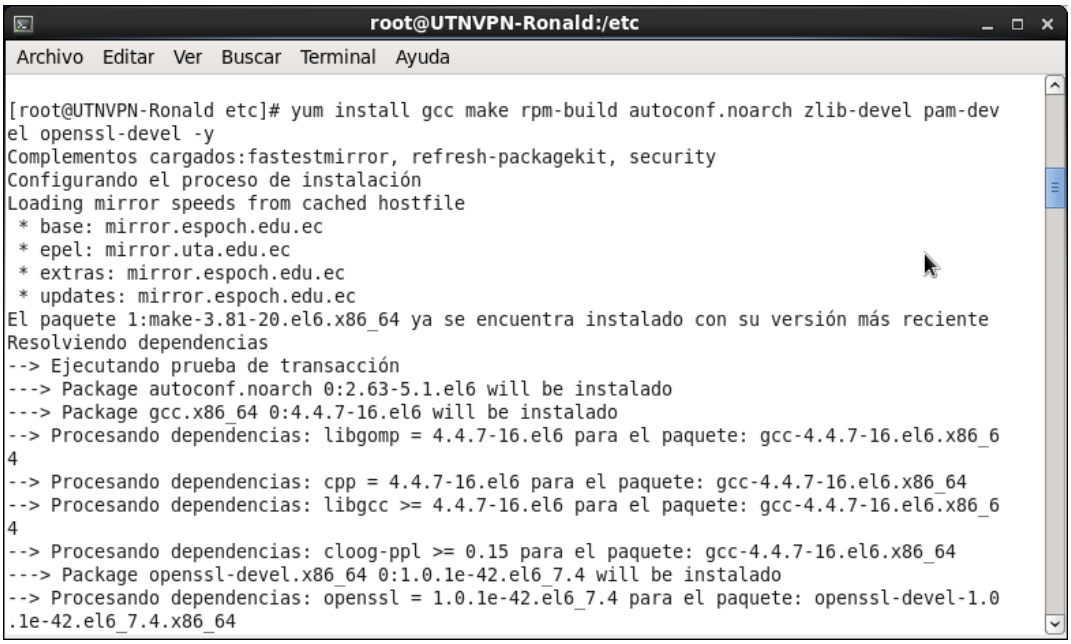
Ahora el repositorio:

- rpm –Uvh epel-release-6.8.noarch.rpm

```
[root@UTNVPN-Ronald ~]# rpm -Uvh epel-release-6-8.noarch.rpm
advertencia:epel-release-6-8.noarch.rpm: CabeceraV3 RSA/SHA256 Signature, ID de
clave 0608b895: NOKEY
Preparando...                ########################################### [100%]
   1:epel-release            ########################################### [100%]
[root@UTNVPN-Ronald ~]#
```

Por ultimo instalamos el repositorio

- yum install gcc make rpm-build autoconf.noarch zlib-devel pam-devel openssl-devel -y



Anexo C. Instalación de OpenVPN Software

✓ Instalamos la aplicación openvpn de EPEL con el siguiente comando.

- yum install openvpn -y

```
[root@UTNVPN-Ronald ~]# rpm -Uvh epel-release-6-8.noarch.rpm
advertencia:epel-release-6-8.noarch.rpm: CabeceraV3 RSA/SHA256 Signature, ID de
clave 0608b895: NOKEY
Preparando...                ########################################### [100%]
   1:epel-release            ########################################### [100%]
[root@UTNVPN-Ronald ~]# yum install openvpn -y
Complementos cargados:fastestmirror, refresh-packagekit, security
Bloqueo existente en /var/run/yum.pid: otra copia se encuentra en ejecución como pid 10
988.
Otra aplicación tiene retenido el bloqueo de Yum; esperándolo para salir...
  La otra aplicación es: PackageKit
    Memoria :  34 M RSS (356 MB VSZ)
    Iniciado: Wed Apr 13 11:57:19 2016 - 00:56 atrás
    Estado  : Durmiendo, pid: 10988
Otra aplicación tiene retenido el bloqueo de Yum; esperándolo para salir...
  La otra aplicación es: PackageKit
    Memoria :  34 M RSS (356 MB VSZ)
    Iniciado: Wed Apr 13 11:57:19 2016 - 00:58 atrás
    Estado  : Durmiendo, pid: 10988
Otra aplicación tiene retenido el bloqueo de Yum; esperándolo para salir...
  La otra aplicación es: PackageKit
    Memoria :  34 M RSS (356 MB VSZ)
    Iniciado: Wed Apr 13 11:57:19 2016 - 01:00 atrás
    Estado  : Durmiendo, pid: 10988
Otra aplicación tiene retenido el bloqueo de Yum; esperándolo para salir...
  La otra aplicación es: PackageKit
    Memoria :  34 M RSS (356 MB VSZ)
```

Anexo D. Configuración de Open Vpn

- ✓ Cambiamos el siguiente directorio a la carpeta de Openvpn con el siguiente comando:

- ✓ cp /usr/share/doc/openvpn-*/sample/sample-config-files/server.conf /etc/openvpn

```
[root@UTNVPN-Ronald ~]# cp /usr/share/doc/openvpn-*/sample/sample-config-files/s
erver.conf /etc/openvpn
```

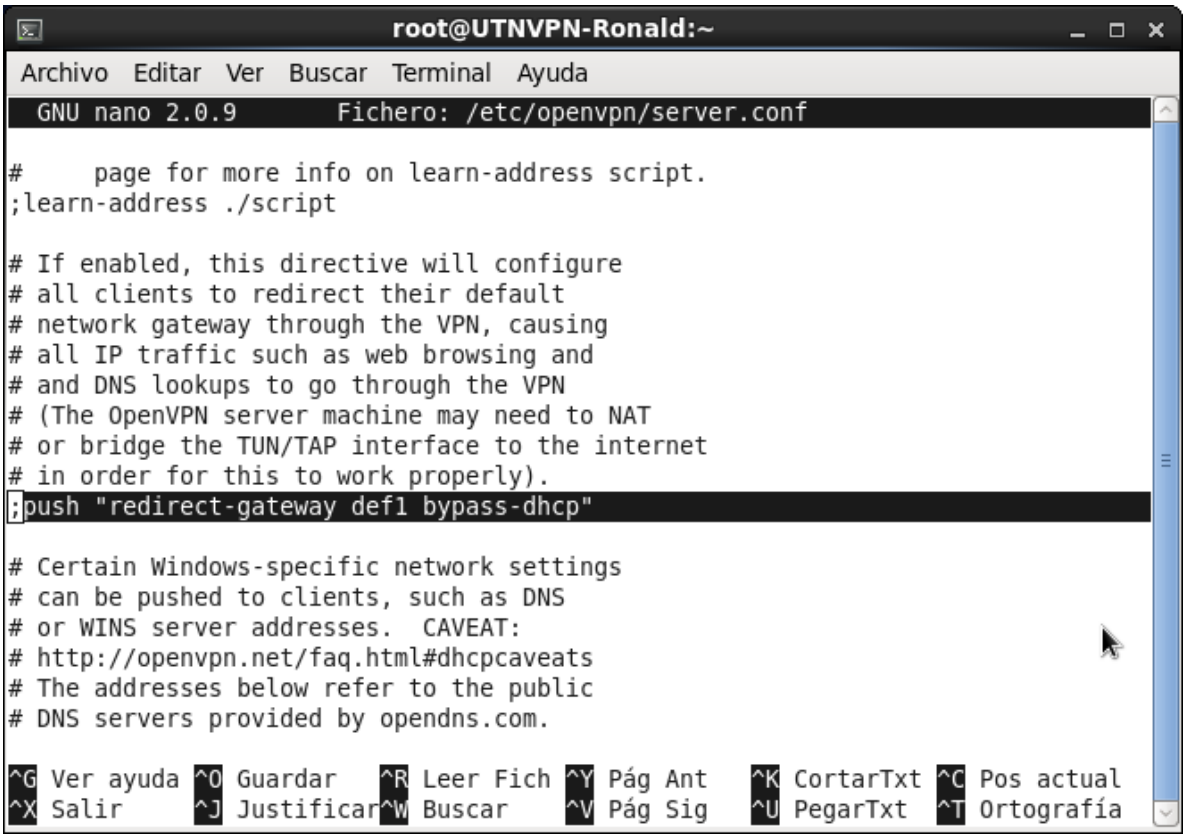
- ✓ Nos dirigimos al archivo en la ubicación adecuada, para configurar el server.conf

- ✓ nano -w /etc/openvpn/server.conf

```
[root@UTNVPN-Ronald ~]# nano -w /etc/openvpn/server.conf
[root@UTNVPN-Ronald ~]#
```

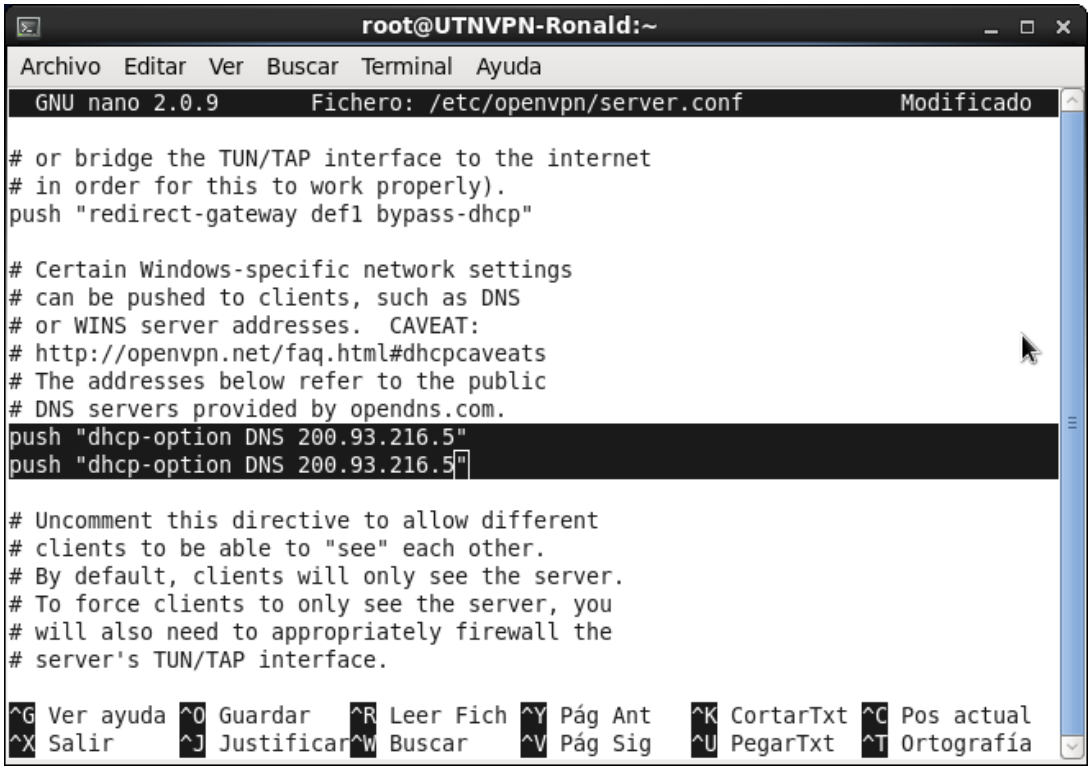
- ✓ Debemos eliminar el comentario el parámetro de "push" el cual es el que provoca el tráfico en nuestros sistemas cliente a ser enrutado a través de OpenVPN.

- ✓ push "redirect-gateway def1 bypass-dhcp"



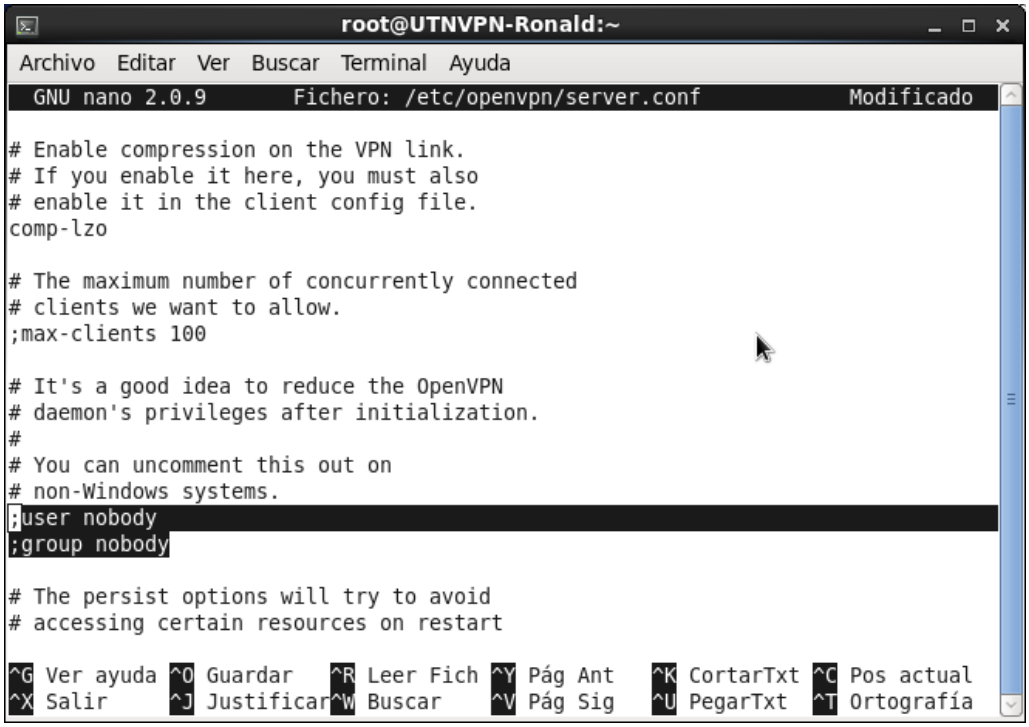
Cambiamos la sección que sigue referente a las consultas de ruta DNS a los servidores DNS.

- push "dhcp-option DNS x.x.x.x"
- push "dhcp-option DNS x.x.x.x"

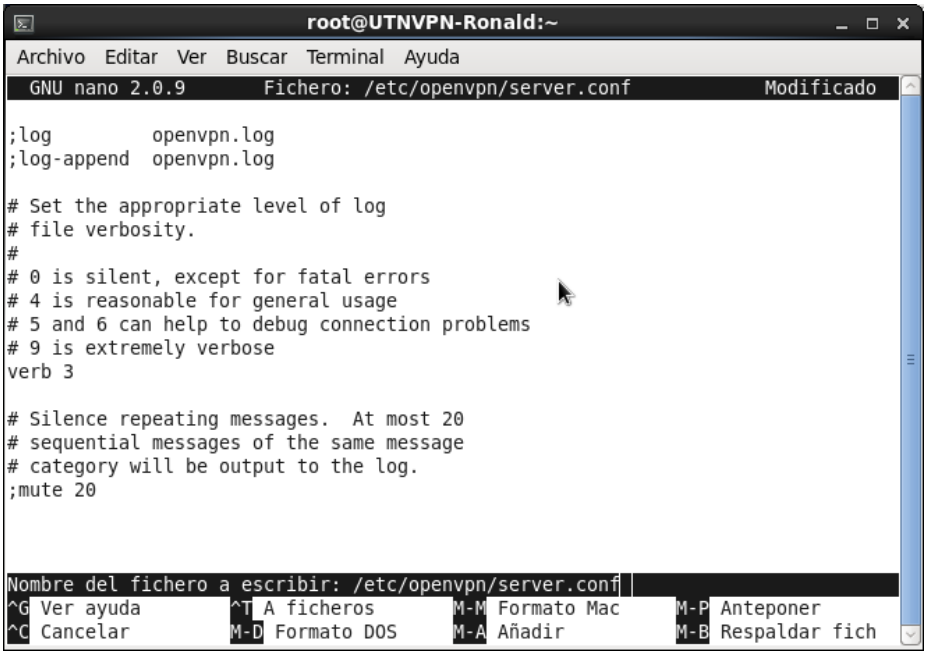


Para mejorar la seguridad, se debe asegurarse de eliminar el comentario de "usuario" relevante y líneas de "grupo"

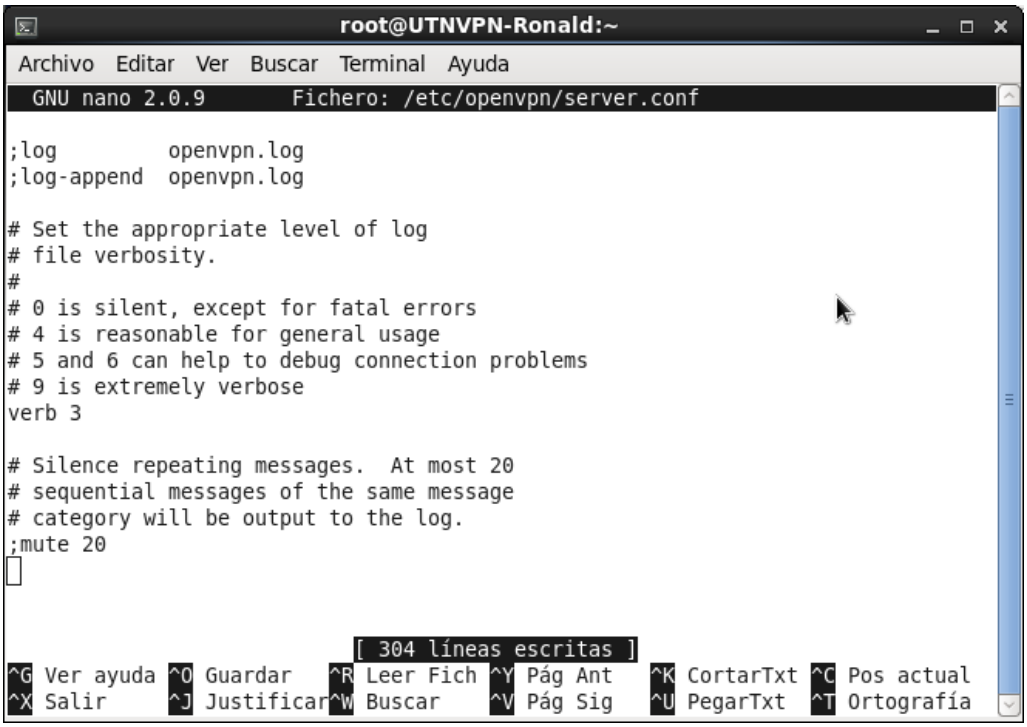
- user nobody
- group nobody



- Guardamos con control O y control x para salir:
- Control O



Control X:


root@UTNVPN-Ronald:~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.0.9 Fichero: /etc/openvpn/server.conf
;log openvpn.log
;log-append openvpn.log
Set the appropriate level of log
file verbosity.
#
0 is silent, except for fatal errors
4 is reasonable for general usage
5 and 6 can help to debug connection problems
9 is extremely verbose
verb 3
Silence repeating messages. At most 20
sequential messages of the same message
category will be output to the log.
;mute 20
[304 líneas escritas]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía

Anexo E. Instalación de Easy-RSA

Ahora que hemos terminado de modificar el archivo de configuración, generaremos las llaves y certificados necesarios.

Primeramente instalamos el paquete Easy-RSA.

- yum install openvpn easy-rsa

```
[root@UTNVPN-Ronald etc]# yum install openvpn easy-rsa
Complementos cargados:fastestmirror, refresh-packagekit, security
Configurando el proceso de instalación
Loading mirror speeds from cached hostfile
 * base: mirror.espoch.edu.ec
 * epel: mirror.uta.edu.ec
 * extras: mirror.espoch.edu.ec
 * updates: mirror.espoch.edu.ec
El paquete openvpn-2.3.10-1.el6.x86_64 ya se encuentra instalado con su versión más reciente
Resolviendo dependencias
--> Ejecutando prueba de transacción
---> Package easy-rsa.noarch 0:2.2.2-1.el6 will be instalado
--> Resolución de dependencias finalizada

Dependencias resueltas
```

Anexo F. Configuración de Easy-RSA

Al igual que con el archivo de configuración, OpenVPN coloca los scripts requeridos en la carpeta de documentación de forma predeterminada. Crear la carpeta requerida y copiar los archivos a través de.

- mkdir -p /etc/openvpn/easy-rsa/keys

```
[root@UTNVPN-Ronald openvpn]# mkdir -p /etc/openvpn/easy-rsa/keys
```

Copiar la carpeta easy-rsa a la ruta donde se encuentra openvpn

```
[root@UTNVPN-Ronald openvpn]# cp -rf /usr/share/easy-rsa/2.0/* /etc/openvpn/easy-rsa/
```

Verificamos con el comando

-ls

```
[root@UTNVPN-Ronald openvpn]# ls
easy-rsa  server.conf
```

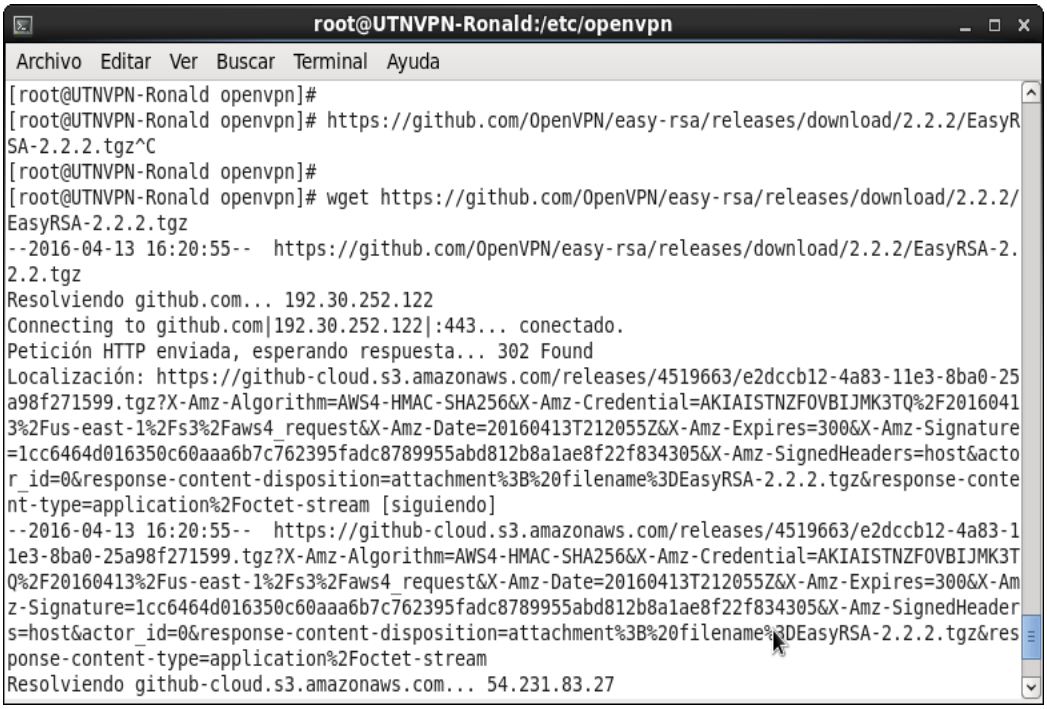
Una vez instalado Openvpn nos dirigimos a la carpeta donde se encuentra:

- cd /etc/openvpn

```
[root@UTNVPN-Ronald ~]# cd /etc/openvpn
```

Ahora descargamos el paquete easy-rsa con el siguiente comando:

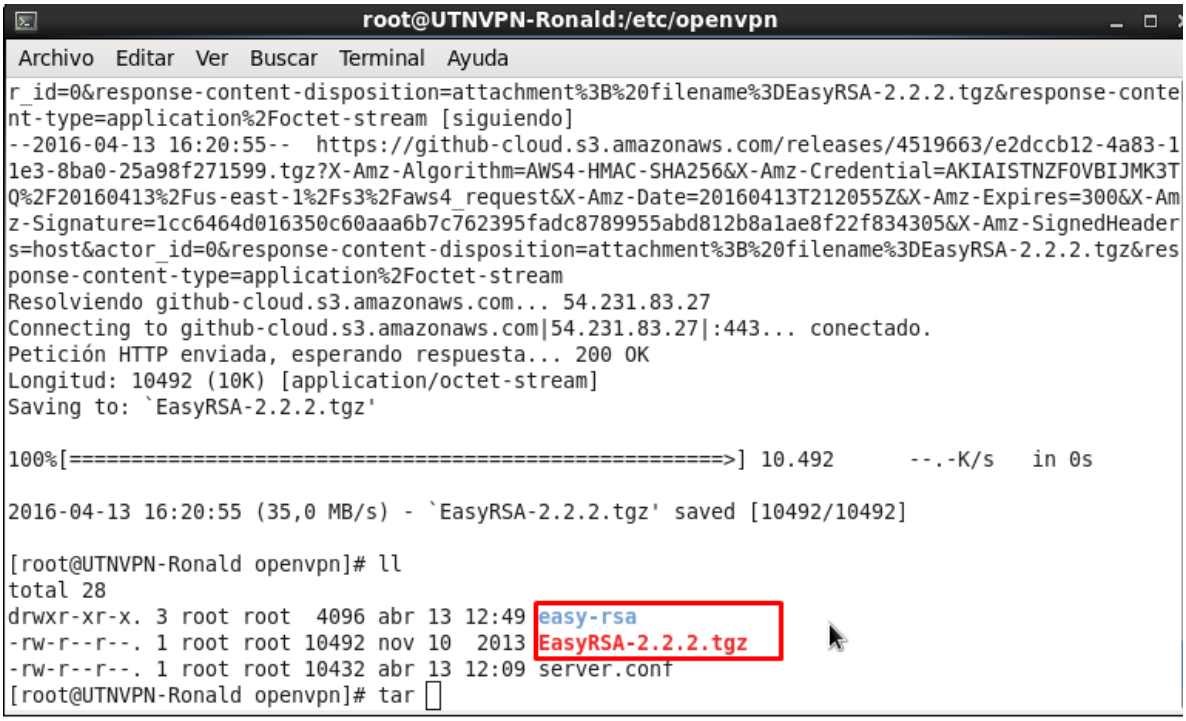
- https://github.com/OpenVPN/easy-rsa/releases/download/2.2.2/EasyRSA-2.2.2.tgz



Ahora instalamos el repositorio easy-rsa con el siguiente comando:

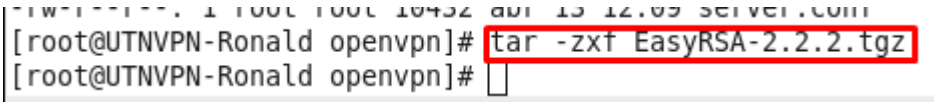
- wget https://github.com/OpenVPN/easy-rsa/releases/download/2.2.2/EasyRSA-2.2.2.tgz

Revisamos con el comando ll, para verificar si se encuentra instalado el paquete easy-rsa.



Ahora procedemos a descomprimir el archivo con el comando:

- tar –zxf EasyRSA-2.2.2.tgz



Verificamos q se ha descomprimido el paquete EasyRSA-2.2.2

```
[root@UTNVPN-Ronald openvpn]# ll
total 32
drwxr-xr-x. 3 root root   4096 abr 13 12:49 easy-rsa
drwxrwxr-x. 2  501 games  4096 nov  8  2013 EasyRSA-2.2.2
-rw-r--r--. 1 root root  10492 nov 10  2013 EasyRSA-2.2.2.tgz
-rw-r--r--. 1 root root  10432 abr 13 12:09 server.conf
[root@UTNVPN-Ronald openvpn]#
```

Mover los datos de la carpeta easy-rsa en donde se encuentran los datos que se van a generan para la autenticación a la carpeta Openvpn.

- mv EasyRSA-2.2.2 easy-rsa

```
[root@UTNVPN-Ronald openvpn]# mv EasyRSA-2.2.2 easy-rsa
[root@UTNVPN-Ronald openvpn]#
```

Ejecutamos el comando ll keys para observar las llaves de seguridad para la posterior autenticación.

```
[root@UTNVPN-Ronald easy-rsa]# ll keys
total 52
-rw-r--r--. 1 root root 5365 abr 13 17:03 01.pem
-rw-r--r--. 1 root root 1655 abr 13 17:01 ca.crt
-rw-------. 1 root root 1704 abr 13 17:01 ca.key
-rw-r--r--. 1 root root  424 abr 13 17:04 dh2048.pem
-rw-r--r--. 1 root root  118 abr 13 17:03 index.txt
-rw-r--r--. 1 root root   21 abr 13 17:03 index.txt.attr
-rw-r--r--. 1 root root    0 abr 13 17:00 index.txt.old
-rw-r--r--. 1 root root    3 abr 13 17:03 serial
-rw-r--r--. 1 root root    3 abr 13 17:00 serial.old
-rw-r--r--. 1 root root 5365 abr 13 17:03 server.crt
-rw-r--r--. 1 root root 1058 abr 13 17:03 server.csr
-rw-------. 1 root root 1704 abr 13 17:03 server.key
```

Copiamos las llaves a la carpeta openvpn con el siguiente comando.

- cp –a keys/* /etc/openvpn

```
[root@UTNVPN-Ronald easy-rsa]# cp -a keys/* /etc/openvpn
```

- ✓ Con los siguientes comandos nos dirigimos a la carpeta openvpn y con el comando Ls observamos que ya se encuentran los keys.

- cd ..
- ls

```
[root@UTNVPN-Ronald easy-rsa]# cd ..
[root@UTNVPN-Ronald openvpn]# ls
01.pem  dh2048.pem         index.txt       serial       server.crt
ca.crt  easy-rsa           index.txt.attr  serial.old   server.csr
ca.key  EasyRSA-2.2.2.tgz  index.txt.old   server.conf  server.key
```

- ✓ Cambiamos de directorio con el comando

- ✓ cd easy-rsa/
- ✓ ls

```
[root@UTNVPN-Ronald openvpn]# cd easy-rsa/
[root@UTNVPN-Ronald easy-rsa]# ls
build-ca      build-key-pass    build-req-pass  list-crl          pkitool      whichopensslcnf
build-dh      build-key-pkcs12  clean-all       openssl-0.9.6.cnf revoke-full
build-inter   build-key-server  inherit-inter   openssl-0.9.8.cnf sign-req
build-key     build-req         keys            openssl-1.0.0.cnf vars
[root@UTNVPN-Ronald easy-rsa]#
```

Anexo G.Generación de claves y certificados con Easy-RSA

Editar el siguiente archivo /etc/openvpn/easy-rsa /vars

- vi / etc / openvpn / easy-rsa / vars

```
[root@UTNVPN-Ronald easy-rsa]# vi /etc/openvpn/easy-rsa/vars
[root@UTNVPN-Ronald easy-rsa]# nano  /etc/openvpn/easy-rsa/vars
```

Cambiar los valores que con su país, estado, ciudad, identificación del correo.

OpenVPN podría no detectar correctamente la versión de OpenSSL en CentOS 6. Como medida de precaución, copie manualmente el archivo de configuración requerido OpenSSL

✓ cp /etc/openvpn/easy-rsa/openssl-1.0.0.cnf /etc/openvpn/easy-rsa/openssl.cnf

```
[root@UTNVPN-Ronald easy-rsa]# cp /etc/openvpn/easy-rsa/openssl-1.0.0.cnf /etc/openvpn/easy-rsa
/openssl.cnf
```

Ahora vamos a construimos la entidad emisora de certificados o CA, con base en la información proporcionada anteriormente.

- source ./vars

```
[root@UTNVPN-Ronald easy-rsa]# cd /etc/openvpn/easy-rsa
[root@UTNVPN-Ronald easy-rsa]# source ./vars
NOTE: If you run ./clean-all, I will be doing a rm -rf on /etc/openvpn/easy-rsa/keys
[root@UTNVPN-Ronald easy-rsa]#
```

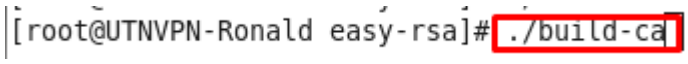
Lo siguiente es limpiar toda la inicialización posteriormente antes efectuada de la autoridad de certificación

```
[root@UTNVPN-Ronald easy-rsa]# ./clean-all
```

Anexo H. Certificado para el servidor OpenVPN

Se ejecuta el siguiente comando para generar el certificado CA y la clave de CA:

- ./build-ca




```
[root@UTNVPN-Ronald easy-rsa]# ./build-ca
Generating a 2048 bit RSA private key
.....................................................................................+++
.....................................+++
writing new private key to 'ca.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [EC]:
State or Province Name (full name) [IM]:
Locality Name (eg, city) [Ibarra]:
Organization Name (eg, company) [UTN]:
Organizational Unit Name (eg, section) [UTN]:
Common Name (eg, your name or your server's hostname) [UTN CA]:server
Name [EasyRSA]:
Email Address [ron90mena@gmail.com]:
[root@UTNVPN-Ronald easy-rsa]#
[root@UTNVPN-Ronald easy-rsa]#
```

Aquí se va realizar la información que se va a incorporar en la solicitud de certificado.

Country Name (2 letter code) [EC]: ----> **Press Enter**

State or Province Name (full name) [IM]: ----> **Press Enter**

Locality Name (eg, city) [Ibarra]: ----> **Press Enter**

Organization Name (eg, company) [UTN]: ----> **Press Enter**

Organizational Unit Name (eg, section) [UTN]: ----> **Press Enter**

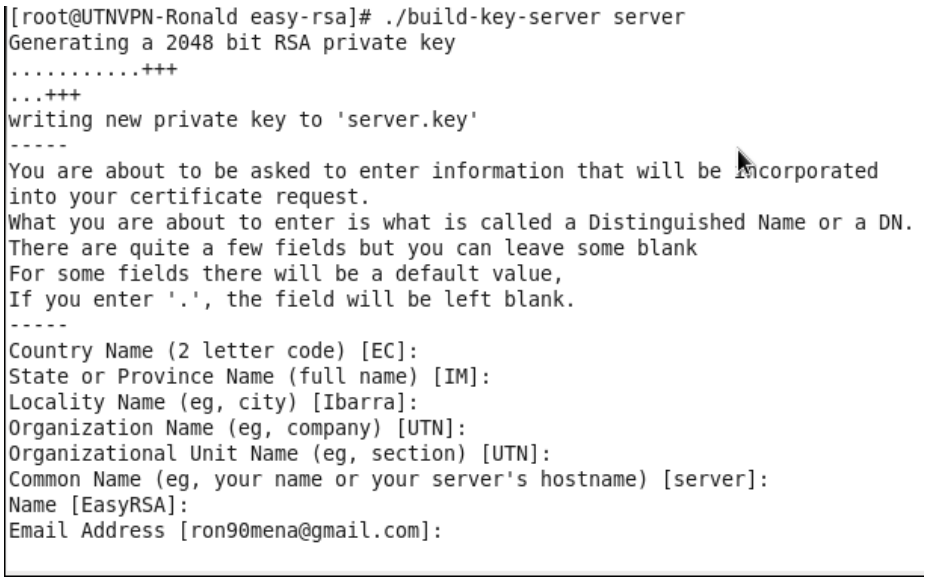
Common Name (eg, your name or your server's hostname) [UTN CA]: server----> **Press Enter**

Name [EasyRSA]: ----> **Press Enter**

Email Address [ron90mena@gmail.com]: ----> **Press Enter**

Ahora que tenemos nuestra CA, vamos a crear nuestro certificado para el servidor OpenVPN. Cuando se le preguntó por la acumulación de clave-servidor, responder sí a comprometerse, con el siguiente comando:

- ✓ ./build-key-server server



Responder sí a comprometerse.

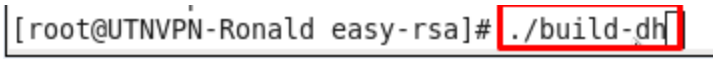
```
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from /etc/openvpn/easy-rsa/openssl-1.0.0.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'EC'
stateOrProvinceName   :PRINTABLE:'IM'
localityName          :PRINTABLE:'Ibarra'
organizationName      :PRINTABLE:'UTN'
organizationalUnitName:PRINTABLE:'UTN'
commonName            :PRINTABLE:'server'
name                  :PRINTABLE:'EasyRSA'
emailAddress          :IA5STRING:'ron90mena@gmail.com'
Certificate is to be certified until Apr 11 17:54:55 2026 GMT (3650 days)
Sign the certificate? [y/n]:y
```

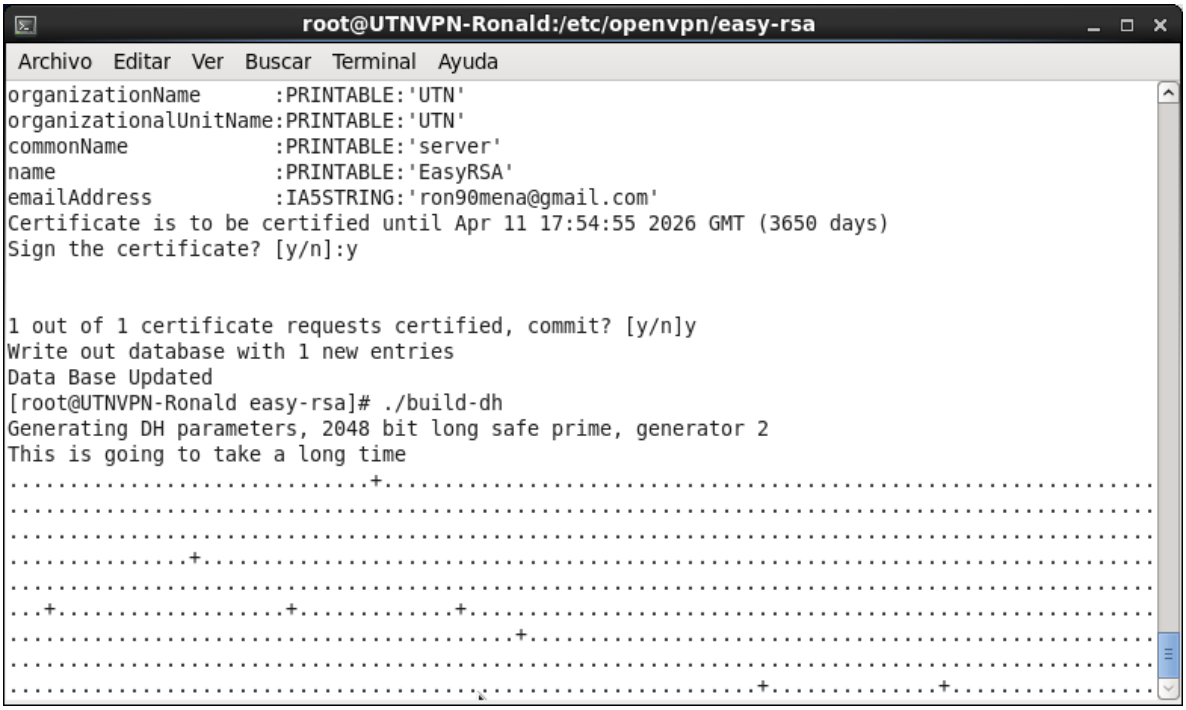
```
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
[root@UTNVPN-Ronald easy-rsa]#
```

Anexo I. Parámetro Diffie Hellman

También vamos a necesitar para generar nuestros archivos de intercambio de claves Diffie-Hellman con el script de la acumulación de dh y copiar todos nuestros archivos en / etc / openvpn.

- ✓ ./build-dh
- ✓ cd /etc/openvpn/easy-rsa/keys
- ✓ cp dh1024.pem ca.crt server.crt server.key /etc/openvpn





Las claves y certificados necesarios se generan en el directorio / etc / openvpn / easy-rsa / keys / directorio. Copia el siguiente archivos de certificados y claves para el /etc/openvpn / directorio.

- ca.crt
- dh 2048 .pem
- server.crt
- server.key

Anexo J. Certificado para el cliente OpenVPN

Para que los clientes se autentican, necesitaremos crear certificados de cliente. Puede repetir este si es necesario para generar un certificado único y una clave para cada cliente o dispositivo.

- ✓ ./build-key client

```
[root@UTNVPN-Ronald easy-rsa]# ./build-key client
Generating a 2048 bit RSA private key
.................+++
.....................................................................+++
writing new private key to 'client.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [EC]:
```

- ✓ Configuramos los parámetros del cliente Openvpn.

```
-----
Country Name (2 letter code) [EC]:
State or Province Name (full name) [IM]:
Locality Name (eg, city) [Ibarra]:
Organization Name (eg, company) [UTN]:
Organizational Unit Name (eg, section) [UTN]:
Common Name (eg, your name or your server's hostname) [client]:
Name [EasyRSA]:
Email Address [ron90mena@gmail.com]:
```

- ✓ Aceptamos el certificado del cliente. (y).

```
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
[root@UTNVPN-Ronald easy-rsa]#
```

- ✓ Ejecutamos el siguiente comando para ver si se han creado las llaves de los certificados tanto del servidor como para el cliente,

- ll keys

```
...........................+........+.....................
.....+....................................................
................................+.........................
................+.........................................
............................+.............................
..........................................................
........................................+...........+.....
......................++*++*
[root@UTNVPN-Ronald easy-rsa]# ll keys
total 52
-rw-r--r--. 1 root root 5365 abr 13 17:03 01.pem
-rw-r--r--. 1 root root 1655 abr 13 17:01 ca.crt
-rw-------. 1 root root 1704 abr 13 17:01 ca.key
-rw-r--r--. 1 root root  424 abr 13 17:04 dh2048.pem
-rw-r--r--. 1 root root  118 abr 13 17:03 index.txt
-rw-r--r--. 1 root root   21 abr 13 17:03 index.txt.attr
-rw-r--r--. 1 root root    0 abr 13 17:00 index.txt.old
-rw-r--r--. 1 root root    3 abr 13 17:03 serial
-rw-r--r--. 1 root root    3 abr 13 17:00 serial.old
-rw-r--r--. 1 root root 5365 abr 13 17:03 server.crt
-rw-r--r--. 1 root root 1058 abr 13 17:03 server.csr
-rw-------. 1 root root 1704 abr 13 17:03 server.key
```

- ✓ Por ultimo copiamos las claves de los certificados a la carpeta donde se encuentra Openvpn.

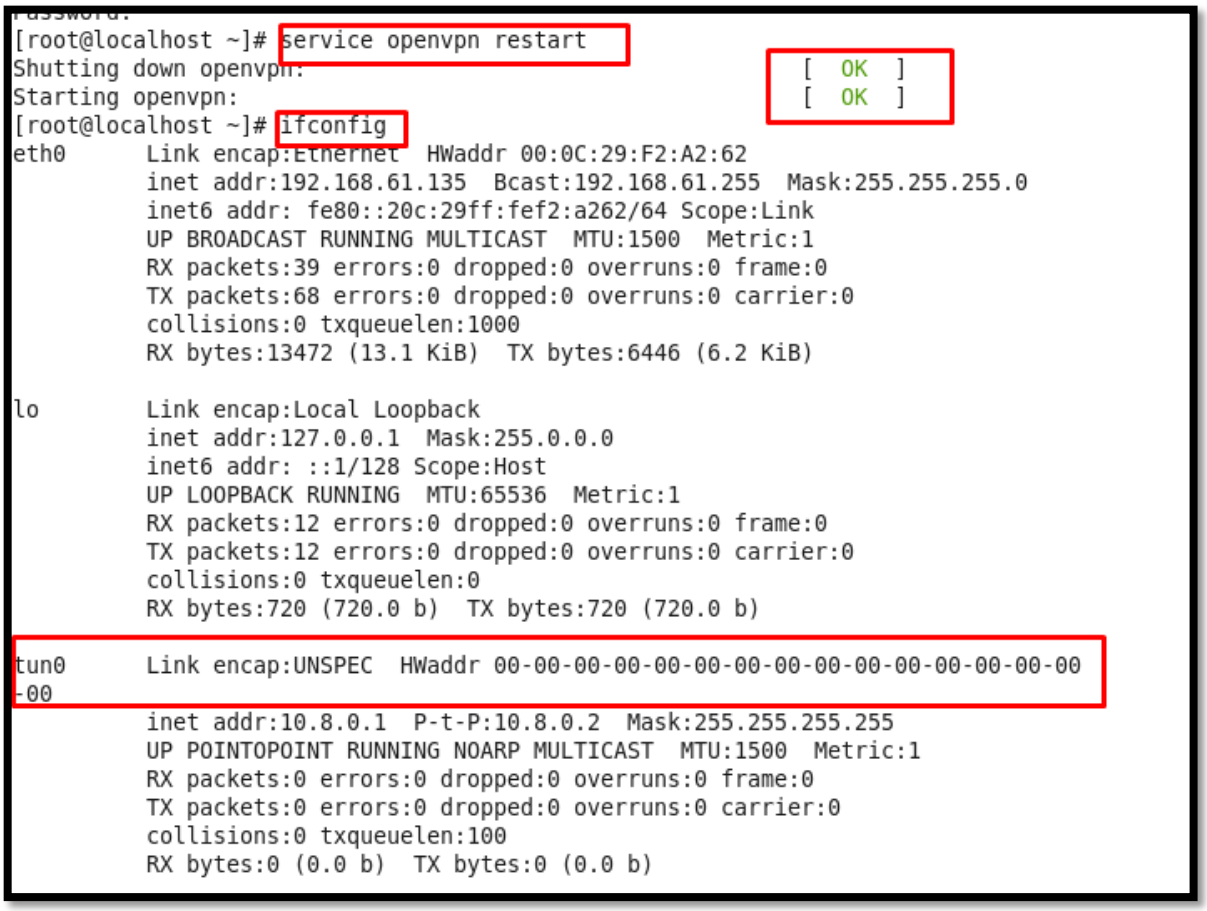
- cp –a keys/* /etc/openvpn

```
[root@UTNVPN-Ronald easy-rsa]# cp -a keys/* /etc/openvpn
```

Anexo K. Servidor OPEN VPN

Con los siguientes comandos iniciaremos el servidor VPN y verificamos el túnel creado por este servidor.

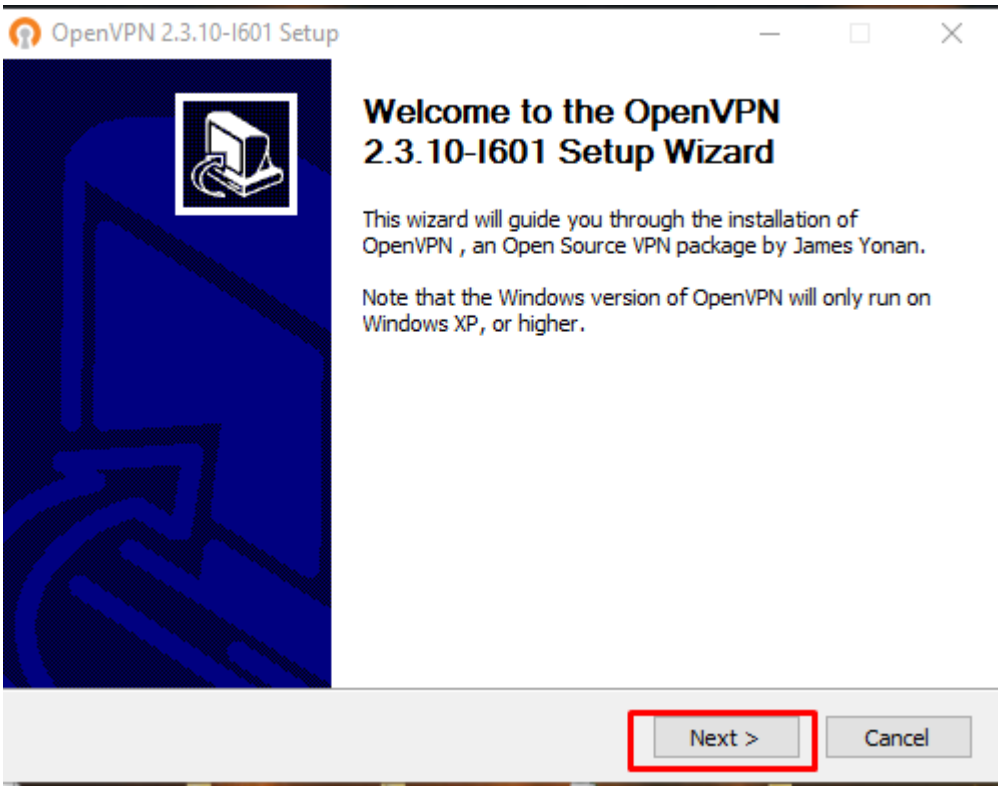
- Service openvpn restart
- ifconfig



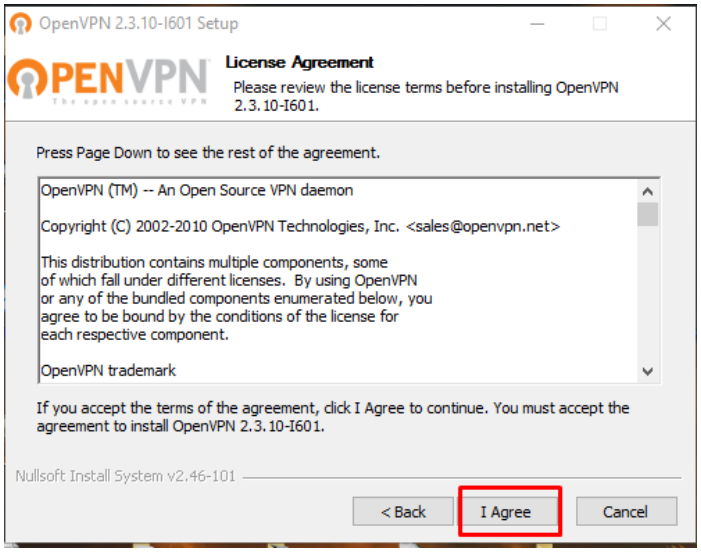
Anexo L. Instalación OpenVPN del cliente en Windows

Nos dirigimos a la página oficial de OpenVpn y descargamos OpenVPN para software Windows.

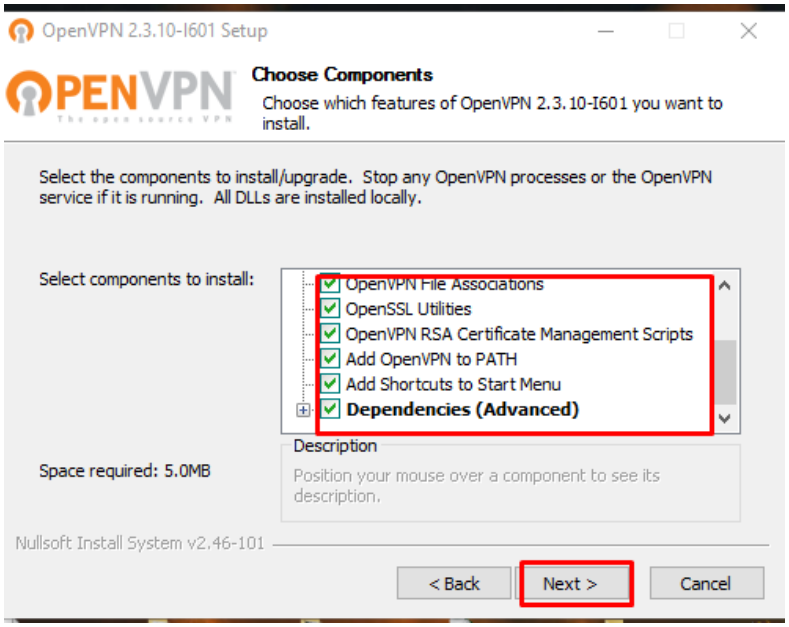
- https://openvpn.net/index.php/open-source/downloads.html
- Y se procederá a instalarla la aplicación OpenVPN.



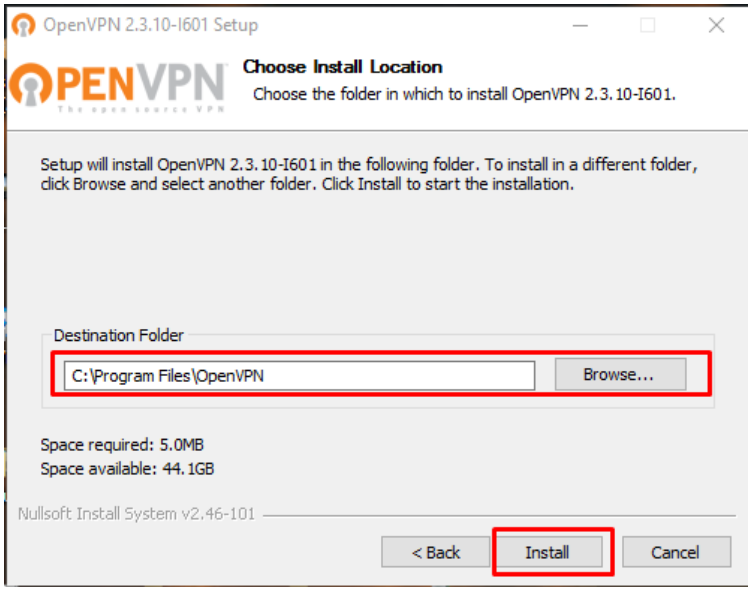
Aceptamos el acuerdo de licencia y procedemos a aceptar.



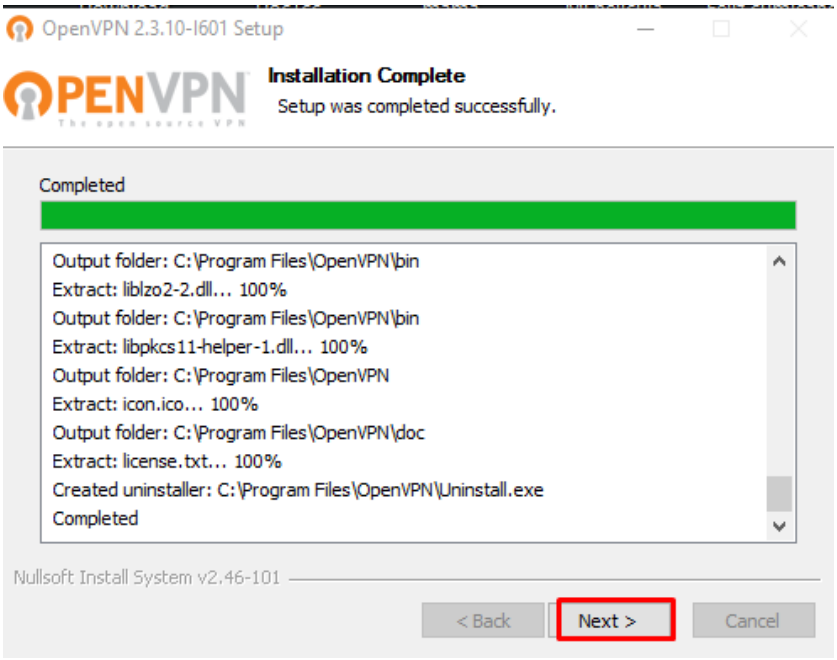
Seleccionamos todas los componentes para la instalación del software OpenVPN.



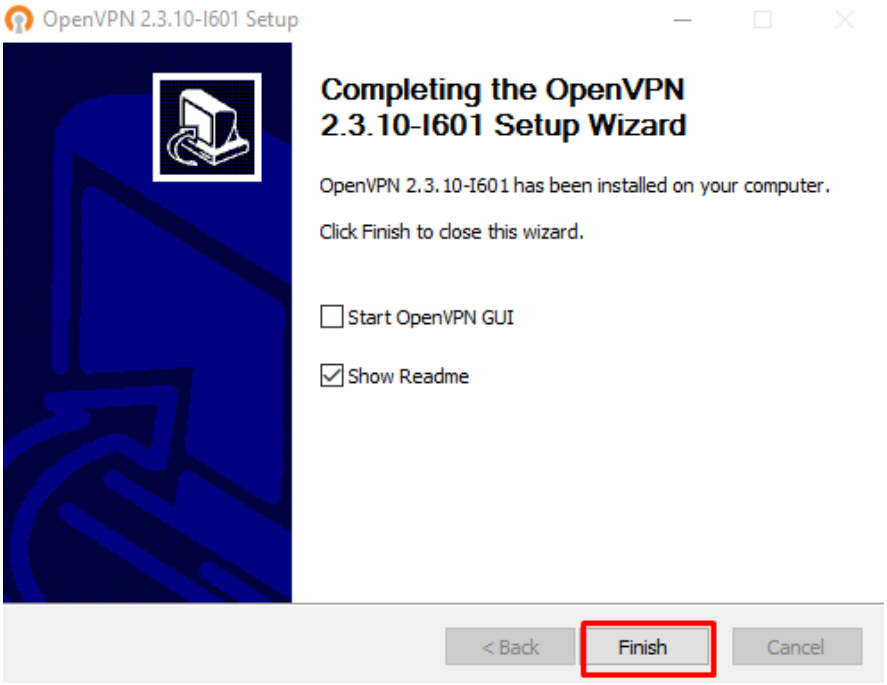
Seleccionamos la ubicación donde se va guardar el programa.



Una vez aceptado esperamos la instalación que se complete y procedemos a dar en siguiente.



Por ultimo procedemos a terminar la instalación en finish.

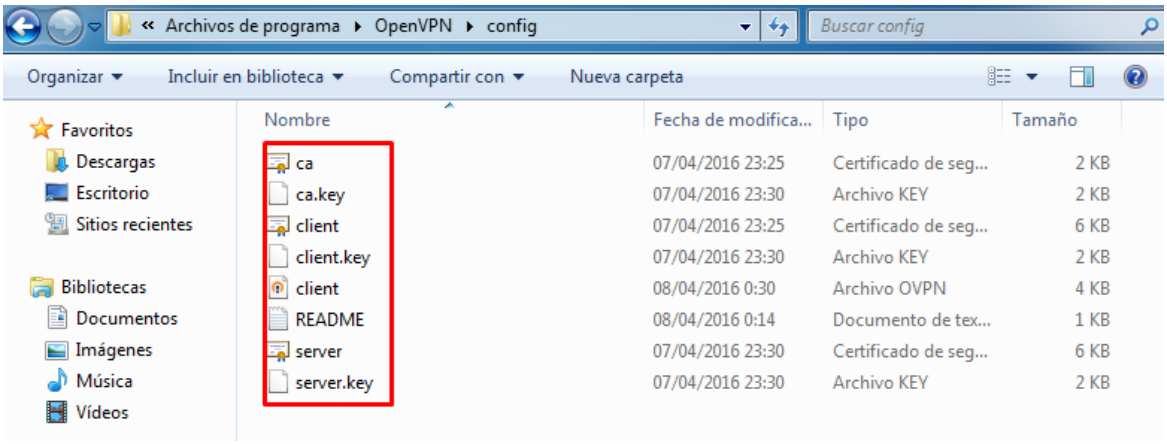


Al finalizar la instalación nos saldrá este icono en el escritorio, hacemos doble clic en el icono.

Y aparecerá este icono pequeño en la barra inferior derecha.

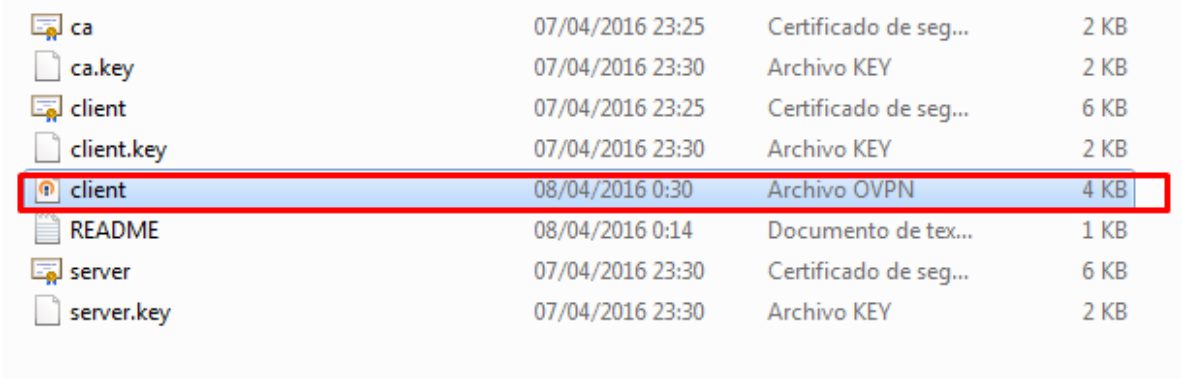
Anexo M. Configuración del Cliente en Windows

Una vez instalado OpenVPN nos dirigimos al Disco local C o donde se encuentre instalado OpenVpn, luego a archivos de programa, OpenVPN, a la carpeta config y aquí pegamos los certificados del cliente y del servidor.



Ahora se configura el cliente, con el nombre de los certificados antes realizados.

Nota: Lo abrimos como un bloc de notas de preferencia en WordPad

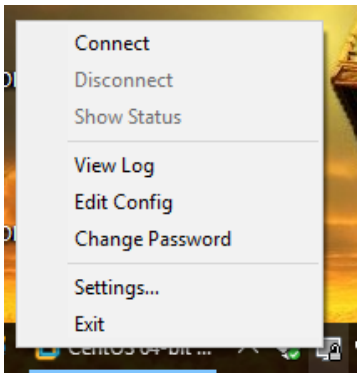


En esta parte es de mucha importancia ya que debemos cambiar el nombre del certificado, con el mismo nombre que se configuro en el servidor OpenVPN.

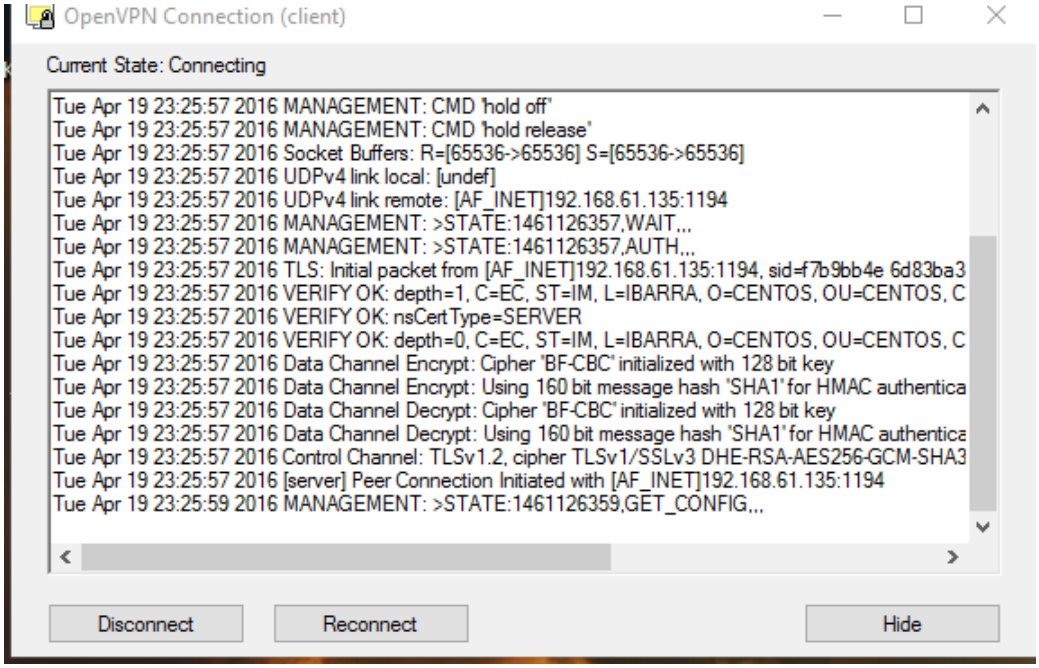
```
##############################################
# Sample client-side OpenVPN 2.0 config file #
# for connecting to multi-client server.     #
#                                            #
# This configuration can be used by multiple #
# clients, however each client should have   #
# its own cert and key files.                #
#                                            #
# On Windows, you might want to rename this  #
# file so it has a .ovpn extension           #
##############################################

# Specify that we are a client and that we
# will be pulling certain config file directives
# from the server.
client

# Use the same setting as you are using on
# the server.
# On most systems, the VPN will not function
# unless you partially or fully disable
# the firewall for the TUN/TAP interface.
;dev tap
dev tun

# Windows needs the TAP-Win32 adapter name
# from the Network Connections panel
# if you have more than one.  On XP SP2,
# you may need to disable the firewall
# for the TAP adapter.
```

```
# authentication.
;http-proxy-retry # retry on connection failures
;http-proxy [proxy server] [proxy port #]

# Wireless networks often produce a lot
# of duplicate packets.  Set this flag
# to silence duplicate packet warnings.
;mute-replay-warnings

# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
ca ca.crt
cert client.crt
key client.key

# Verify server certificate by checking
# that the certicate has the nsCertType
# field set to "server".  This is an
# important precaution to protect against
# a potential attack discussed here:
#  http://openvpn.net/howto.html#mitm
#
# To use this feature, you will need to generate
# your server certificates with the nsCertType
# field set to "server".  The build-key-server
# script in the easy-rsa folder will do this.
```

Una vez logrado esto nos conectamos a la VPN.



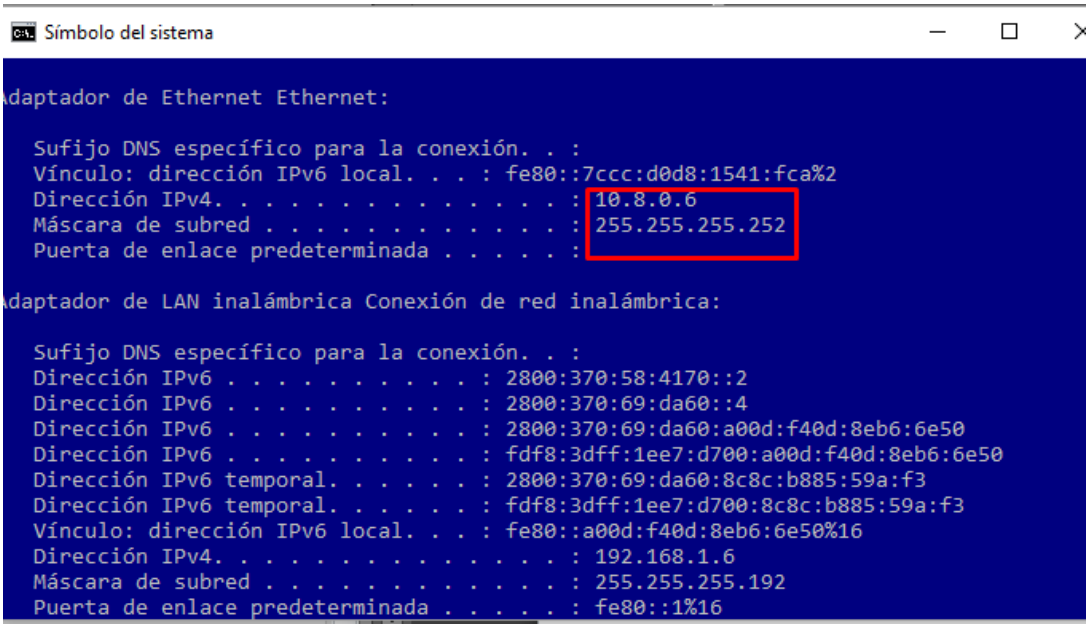
Esperamos que se conecte con el servidor VPN



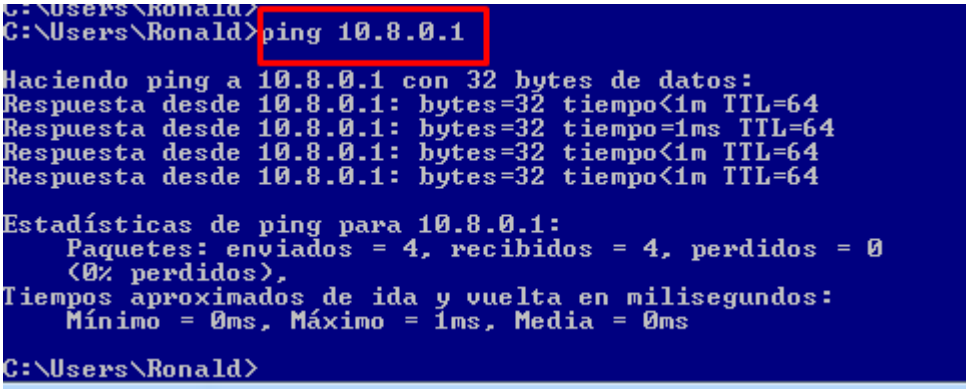
No s aparecerá un icono de color verde dando lugar a que la conexión con el servidor OpenVPN se ha realizado correctamente,

Podemos verificar que estamos dentro del túnel VPN a través de su dirección pública, de la siguiente manera.

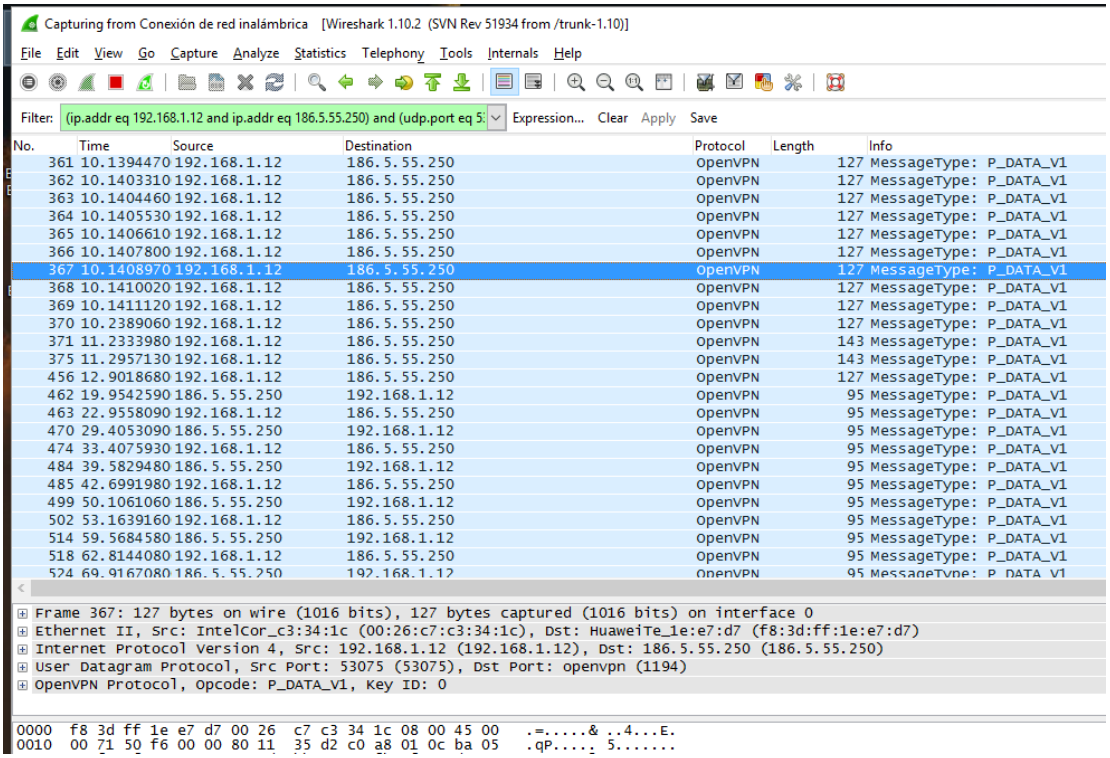
En el símbolo del sistema o cmd de Windows ejecutamos un ipconfig para ver la ip de nuestro servidor VPN.



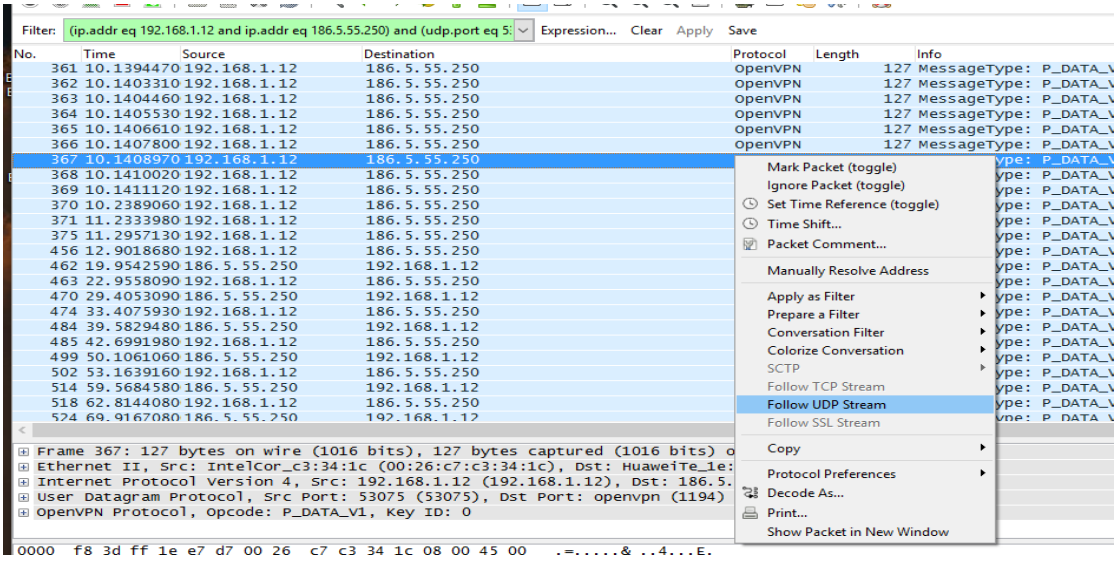
Prueba de funcionamiento del cliente realizando un ping al servidor OpenVPN.



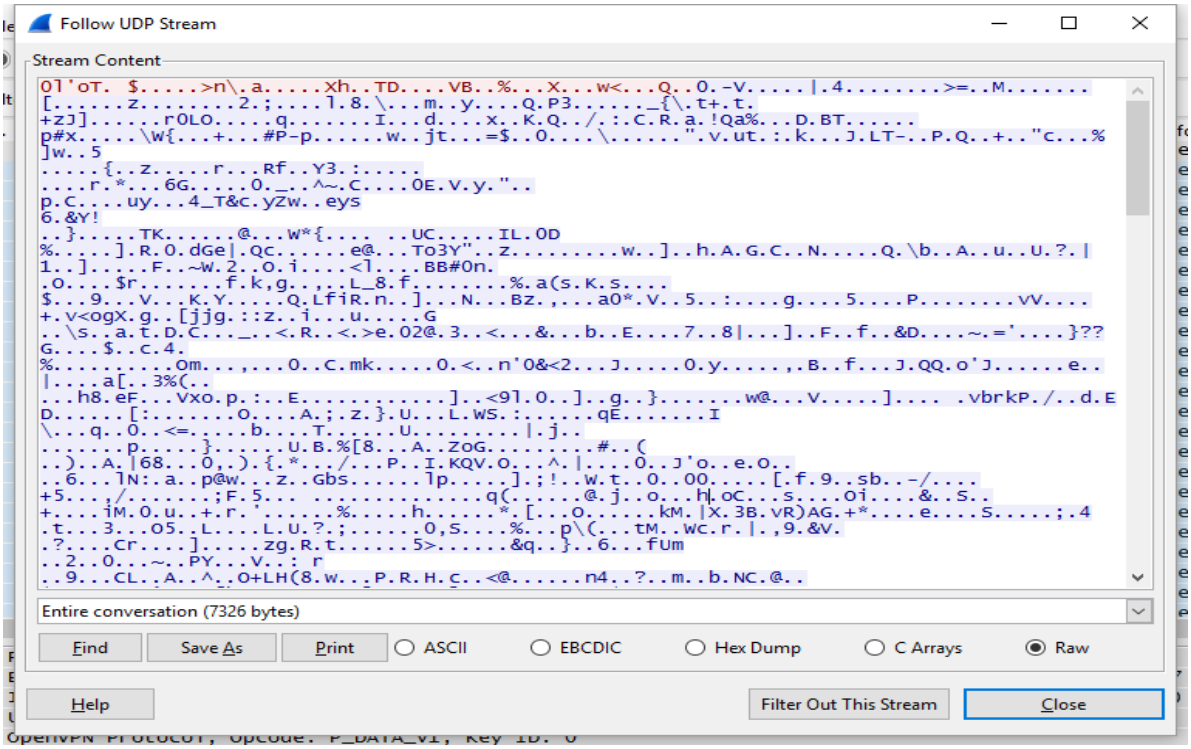
Las pruebas se las realizo con el analizador de tráfico de red Wireshark



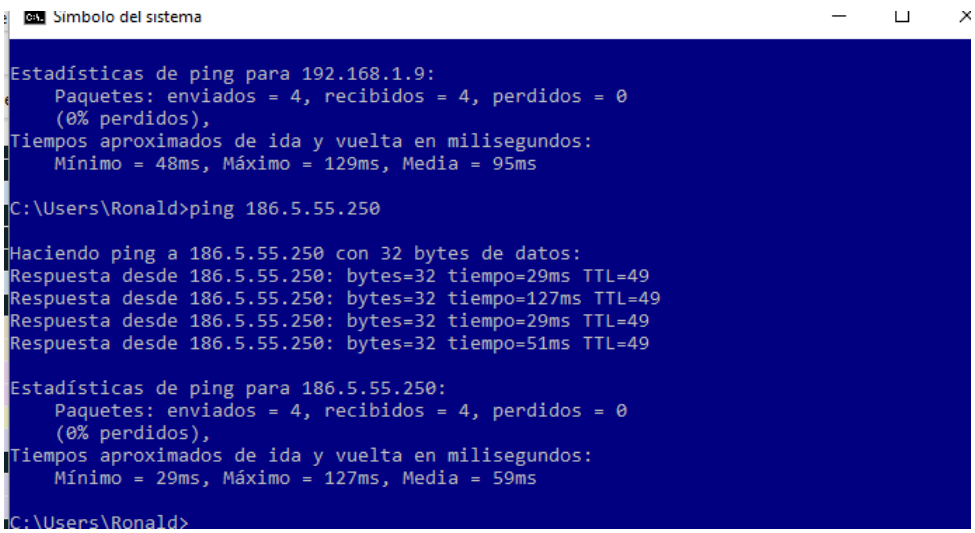
Podemos observar que el servidor VPN está funcionando correctamente y analizar el flujo UDP de la red a la que se encuentra conectado el cliente VPN.



Y observamos que se encuentra encriptado el tráfico a través del túnel VPN

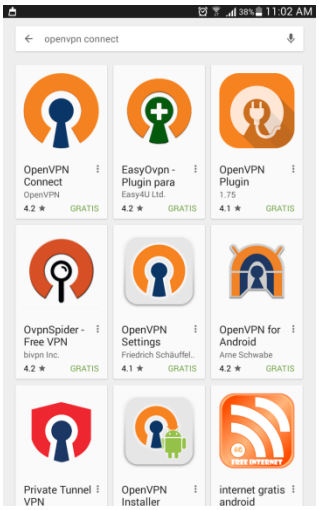


Realizamos un ping desde el cliente al servidor VPN .



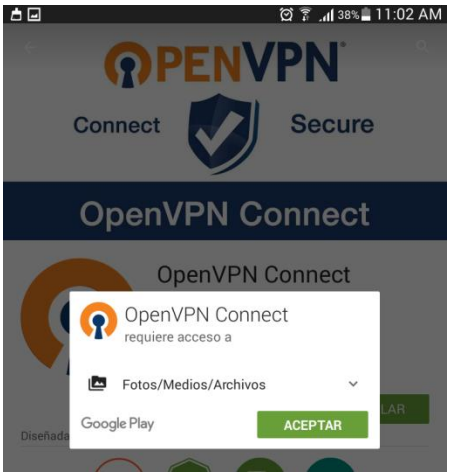
Anexo N. Configuración de un Cliente en un Sistema Android

Para configurar un cliente vpn, vamos a realizar los siguientes pasos. Buscamos en nuestro Smartphone Android la aplicación Openvpn connect



Instalamos la aplicación Openvpn connect en nuestro Smartphone

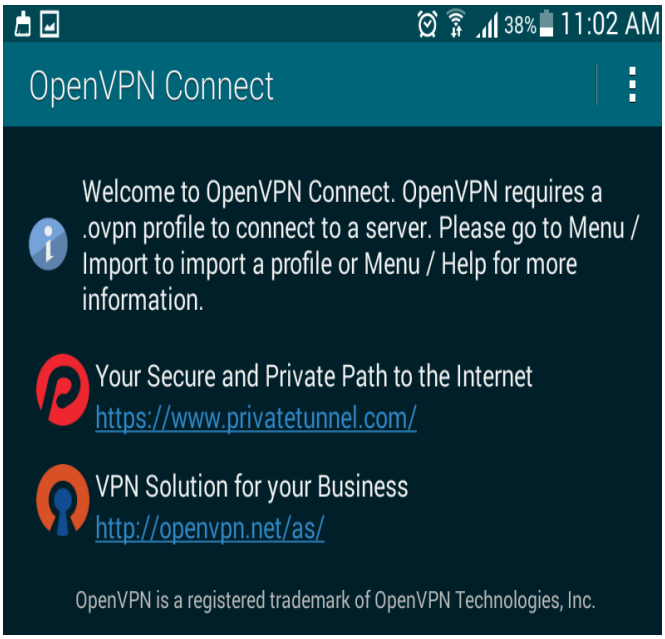
Aceptamos para acceder a los archivos de configuración de los certificados del cliente y poder seleccionar el archivo .ovpn



Se procederá a instalar la aplicación Openvpn connect en el Smartphone

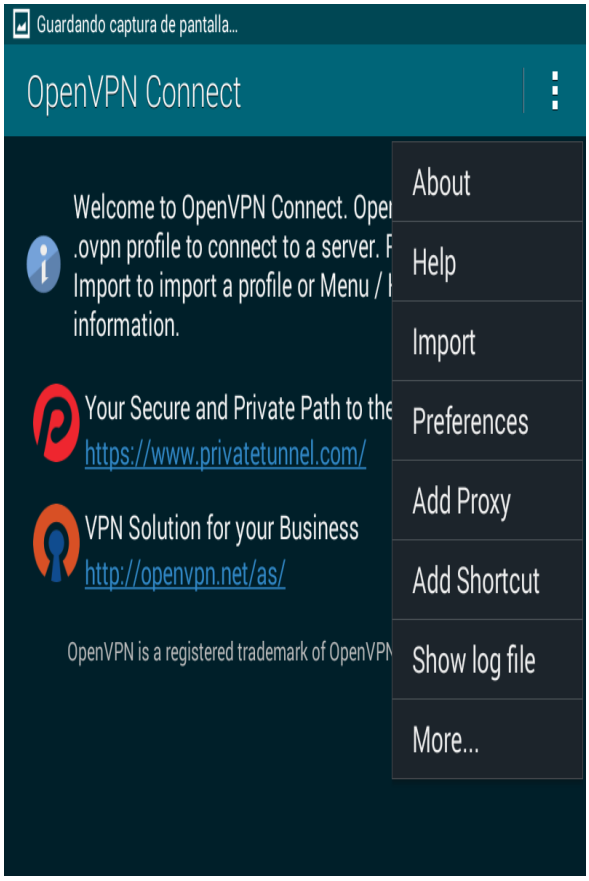
Procedemos a abrir la Aplicación Openvpn connect

Esta es la aplicación una vez instalada

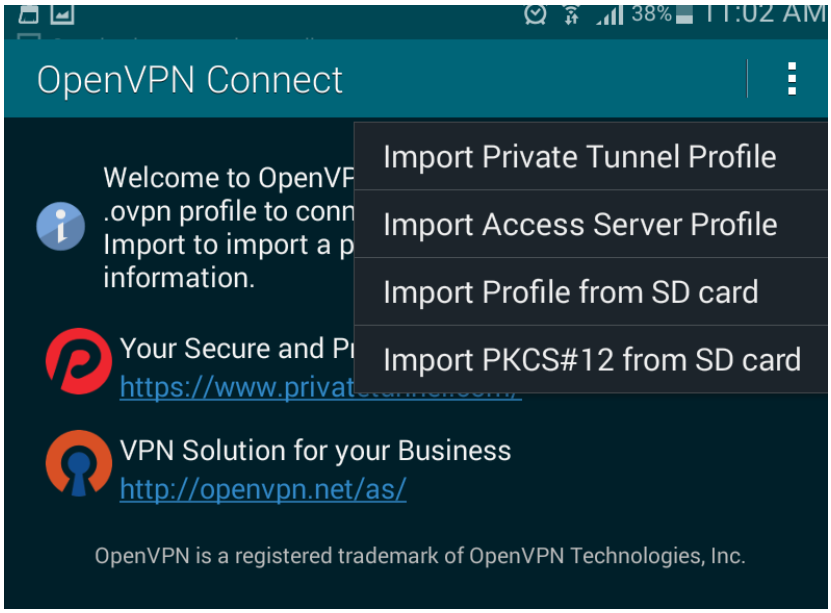


Procedemos a configurar el cliente openvpn en el sistema operativo Android

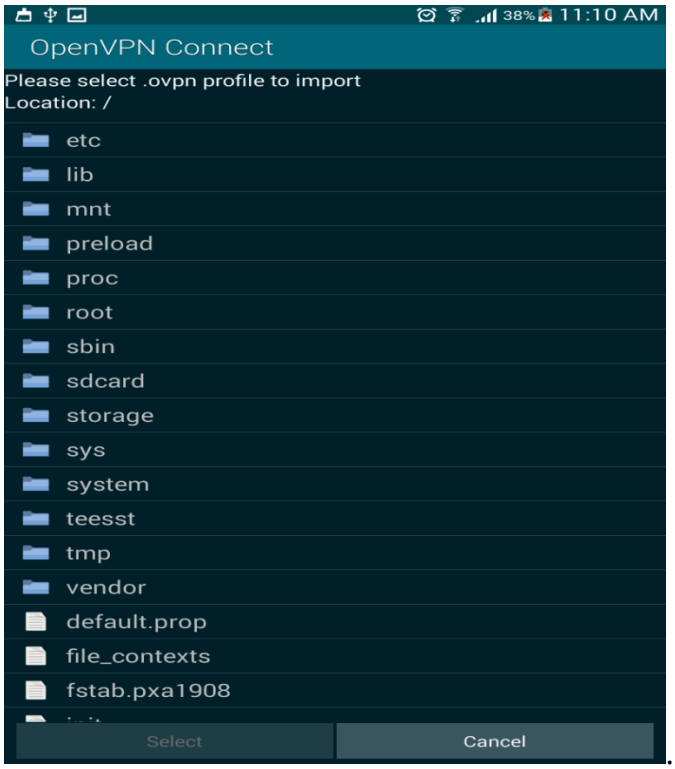
- Nos dirigimos a menú y escogemos la opción Import, para importar los certificados



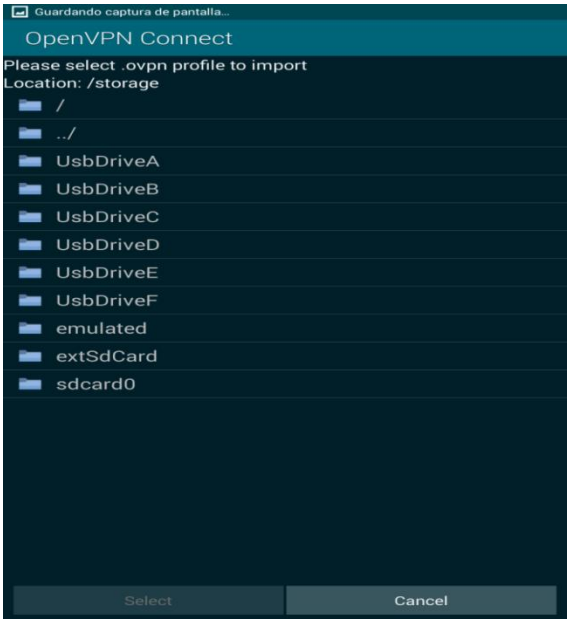
Seleccionas en donde se encuentran los certificados generados por el servidor OpenVPN.



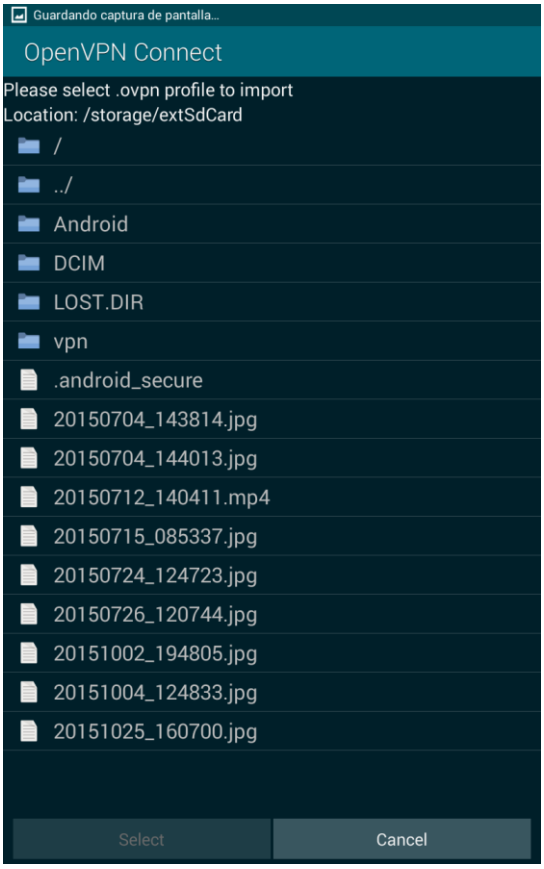
Nos vamos a la memoria de Smartphone en este caso sdcard



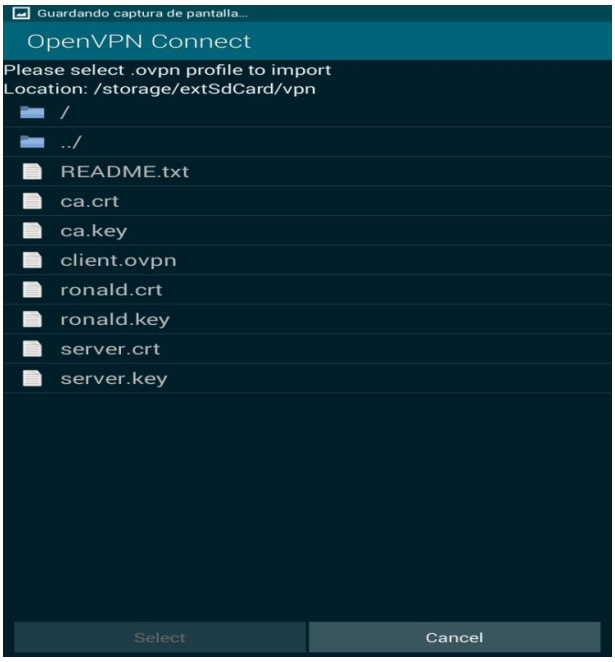
Nos dirigimos a la memoria donde guardamos y copiamos los certificados, en este caso es en la extSdcard



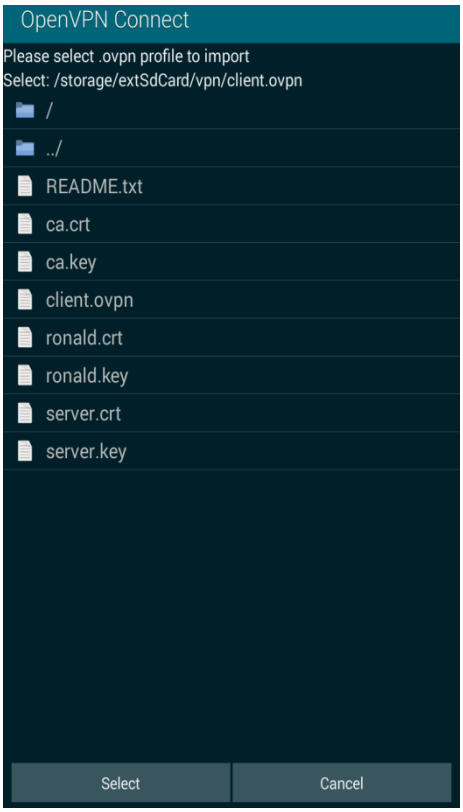
Y en esta ventana debemos encontrar los certificados realizados previamente por el Servidor OpenVPN, en este caso nos dirigimos a la carpeta vpn donde están



En esta ventana se encuentran los certificados creados por el servidor OpenVPN

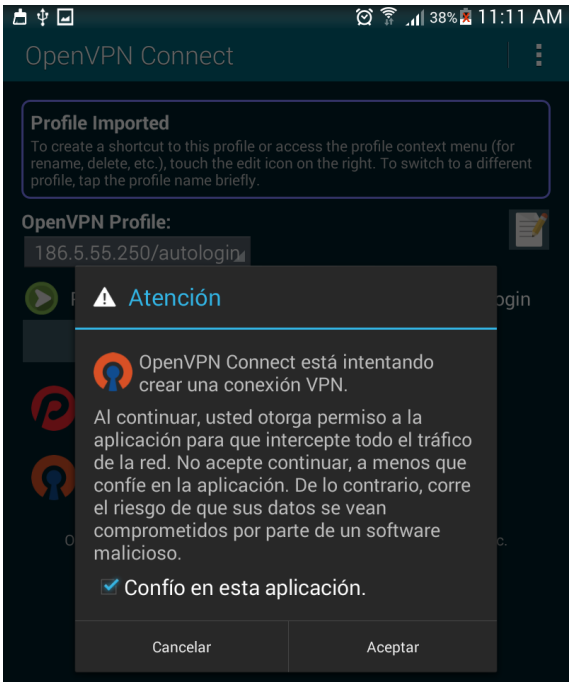


Escogemos el certificado del cliente .ovpn en este caso client.ovpn

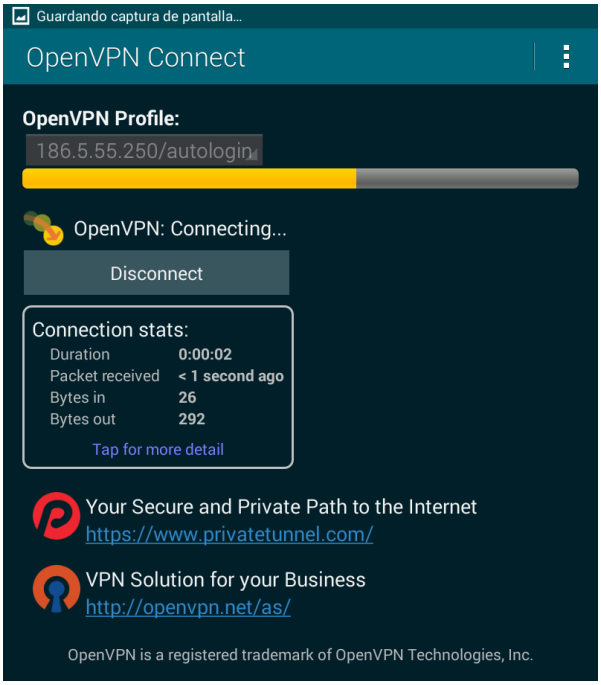


Una vez importado el certificado del cliente nos saldrá para podernos conectar con el servidor VPN.

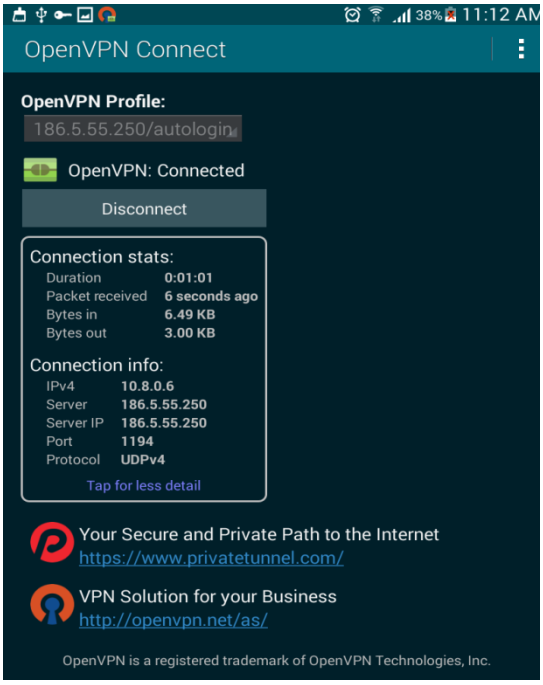
Aceptamos el acuerdo para que la aplicación interprete todo el tráfico de la red.



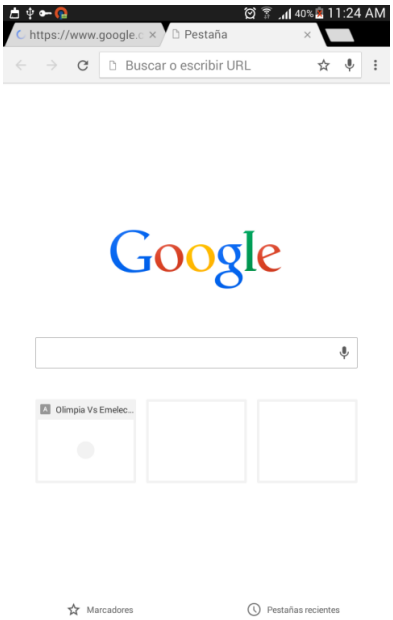
Esperamos a que se autentique con el servidor VPN .



Una vez que se autentique con el servidor se podrá conectar al servidor VPN



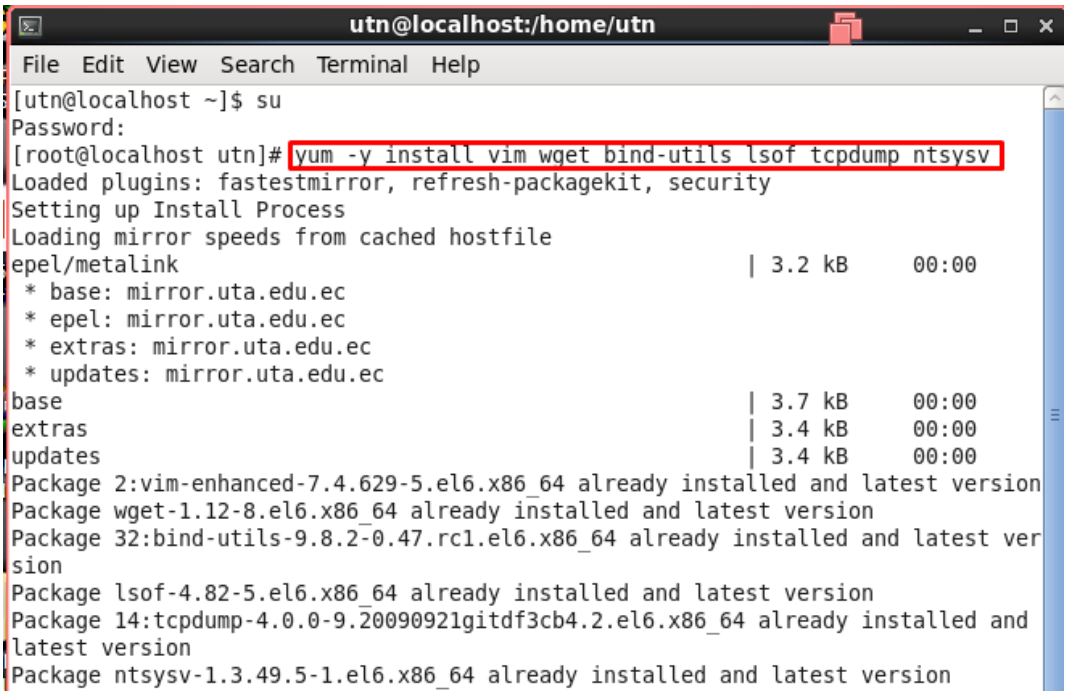
Ahora podemos ingresar y navegar de una manera segura a través de una Red Privada Virtual.



Anexo O. Configuración en CentOS - OpenSwan IPSec VPN

Instalamos el tcpdump ntsysv que posteriormente nos ayudara para ver el encriptado y la autenticación con el siguiente comando.

- yum -y install vim wget bind-utils lsof tcpdump ntsysv



Procedemos a actualizar los paquetes provenientes de nuestro servidor para que no exista ningún problema de compilación del kernel.

- yum -y update

```
[root@localhost utn]# yum -y update
Loaded plugins: fastestmirror, refresh-packagekit, security
Setting up Update Process
Loading mirror speeds from cached hostfile
 * base: mirror.uta.edu.ec
 * epel: mirror.uta.edu.ec
 * extras: mirror.uta.edu.ec
 * updates: mirror.uta.edu.ec
No Packages marked for Update
[root@localhost utn]#
```

Ahora se procede a reiniciar el sistema con el comando.

- reboot

Instalamos el repositorio Webmin repo que nos ayudara para la instalación de ipsec, para ello aplicaremos el siguiente comando.

- wget http://ftp.riken.jp/Linux/fedora/epel/6 … noarch.rpm

```
[root@localhost 90]# wget http://ftp.riken.jp/Linux/fedora/epel/6 ... noarch.rpm
--2016-07-04 12:58:54--  http://ftp.riken.jp/Linux/fedora/epel/6
Resolving ftp.riken.jp... 134.160.38.1
Connecting to ftp.riken.jp|134.160.38.1|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://ftp.riken.jp/Linux/fedora/epel/6/ [following]
--2016-07-04 12:58:55--  http://ftp.riken.jp/Linux/fedora/epel/6/
Reusing existing connection to ftp.riken.jp:80.
HTTP request sent, awaiting response... 200 OK
Length: 978 [text/html]
Saving to: “6”

100%[======================================>] 978         --.-K/s   in 0s

2016-07-04 12:58:56 (57.1 MB/s) - “6” saved [978/978]

--2016-07-04 12:58:56--  http://.../
Resolving ..... failed: Name or service not known.
wget: unable to resolve host address “...”
--2016-07-04 12:58:56--  http://noarch.rpm/
Resolving noarch.rpm... failed: Name or service not known.
```

Seguido del repositorio epel-release

- yum -y install ./epel-release-6-8.noarch.rpm

```
[root@localhost 90]# yum -y install ./epel-release-6-8.noarch.rpm
Loaded plugins: fastestmirror, refresh-packagekit, security
Setting up Install Process
Examining ./epel-release-6-8.noarch.rpm: epel-release-6-8.noarch
./epel-release-6-8.noarch.rpm: does not update installed package.
```

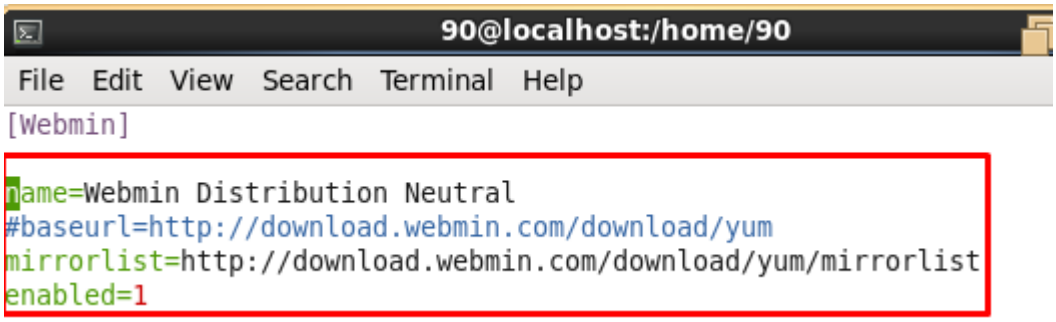
Para instalar ipsec debemos de configurar webmin para que no existan problemas al momento de proceder a instalar ipsec

- vim /etc/yum.repos.d/webmin.repo

```
[root@localhost 90]# vim /etc/yum.repos.d/webmin.repo
```

Dentro de este archivo colocamos lo siguiente:

- [Webmin]
- name =Webmin Distribution Neutral
- #baseurl=http://download.webmin.com/download/yum
- mirrorlist=http://download.webmin.com/download/yum/mirrorlist
- enabled =1

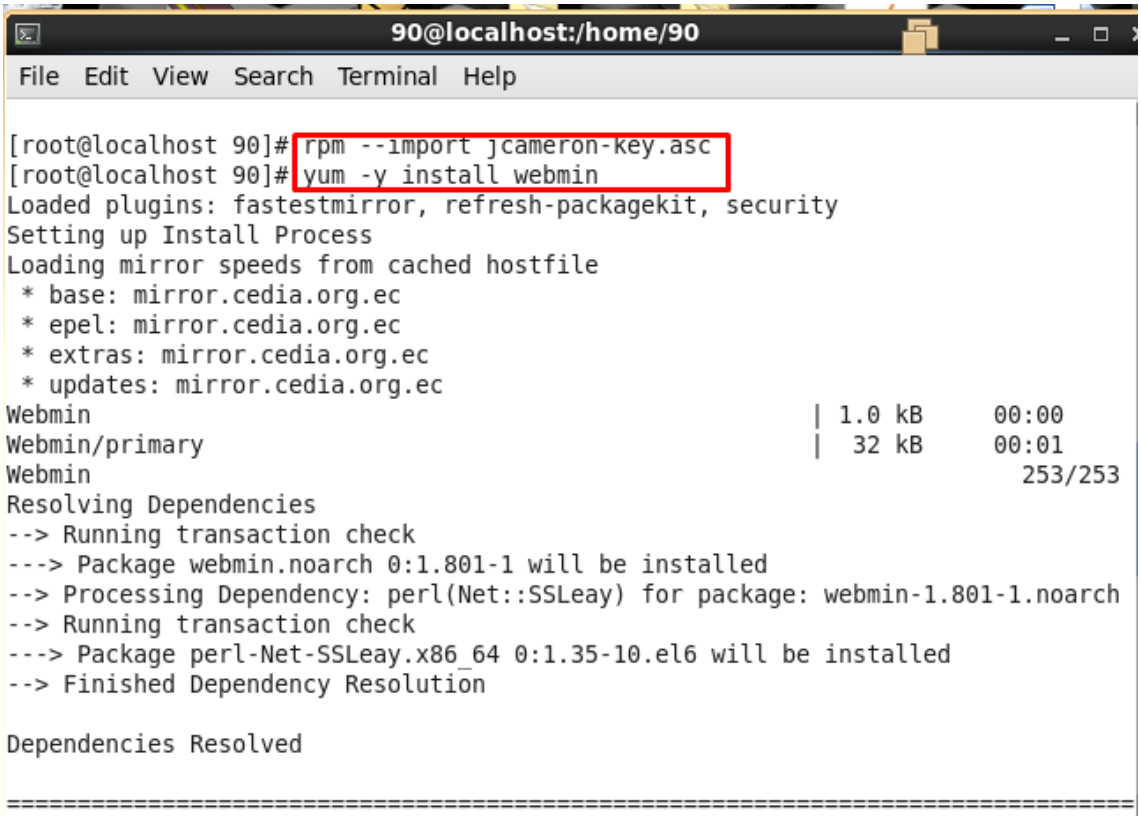


Instalamos el paquete que contiene webmin con la siguiente instrucción.

- wget http://www.webmin.com/jcameron-key.asc

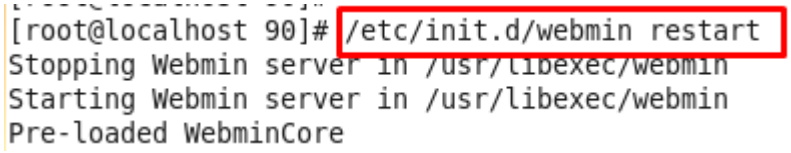
EL repositorio necesario para importar webmin y proceder a instalarlo con los siguientes comandos.

- rpm --import jcameron-key.asc
- yum -y install webmin



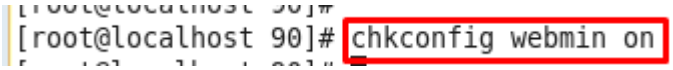
Una vez instalado se procede a restaurar el servicio de webmin.

- /etc/init.d/webmin restart



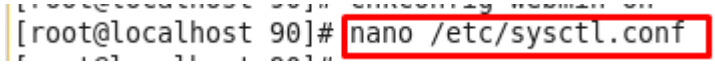
Procedemos a escribir el siguiente comando para que inicialice desde el arranque del sistema.

- chkconfig webmin on



Configurar el kernel para IPSEC de la siguiente manera.

- nano /etc/sysctl.conf



Saldrá la siguiente ventana de configuración y procedemos a colocar lo siguiente, donde existe el Control de paquetes IP llamado “forwarding”.

- net.ipv4.ip_forward = 1
- net.ipv4.conf.all.accept_redirects = 0
- net.ipv4.conf.all.send_redirects = 0

```
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.
#
# Use '/sbin/sysctl -a' to list all possible parameters.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 0

                              [ Read 37 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Verificamos los cambios con el comando.

- sysctl -p

```
[root@localhost 90]# sysctl -p
net.ipv4.ip_forward = 0
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
kernel.sysrq = 0
kernel.core_uses_pid = 1
net.ipv4.tcp_syncookies = 1
kernel.msgmnb = 65536
kernel.msgmax = 65536
kernel.shmmax = 68719476736
kernel.shmall = 4294967296
```

Configuramos selinux para que no existan inconvenientes a la hora de arrancar IPSEC.

- echo 0 > /selinux/enforce
- vim /etc/sysconfig/selinux

```
[root@localhost 90]# echo 0 > /selinux/enforce
[root@localhost 90]# nano /etc/sysconfig/selinux
```

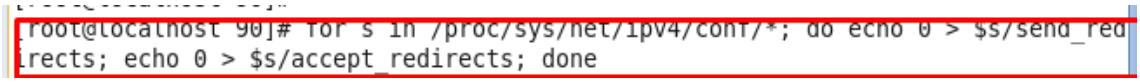
Nos indicara un archivo de texto, donde aplicaremos enforcing por:

- SELINUX=disabled

```
  GNU nano 2.0.9          File: /etc/sysconfig/selinux                Modified


# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted




                          [ Read 13 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

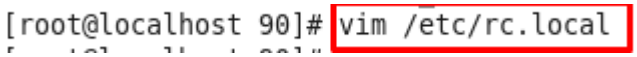
Una vez configurado el Selinux se procede a escribir el comando que realizara un acuerdo de la configuración de envió y reenvió de paquetes ip por el túnel.

- for s in /proc/sys/net/ipv4/conf/*; do echo 0 > $s/send_redirects; echo 0 > $s/accept_redirects; done
-



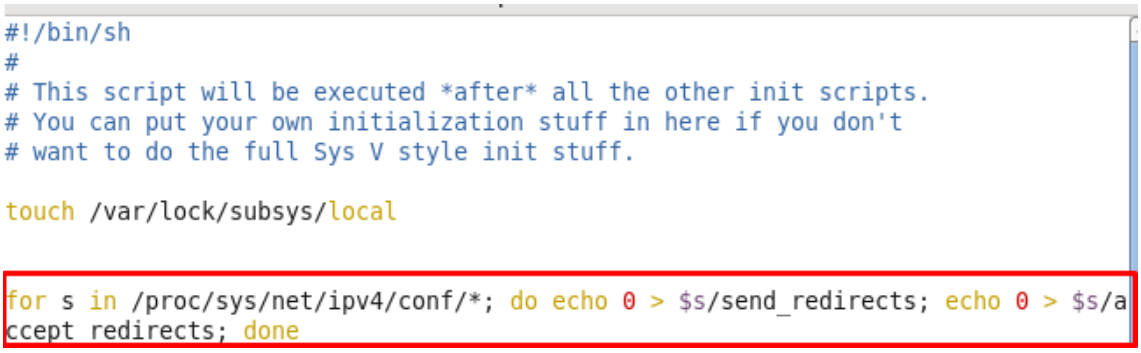
Se procede a configurar el rc de manera local.

- vim /etc/rc.local



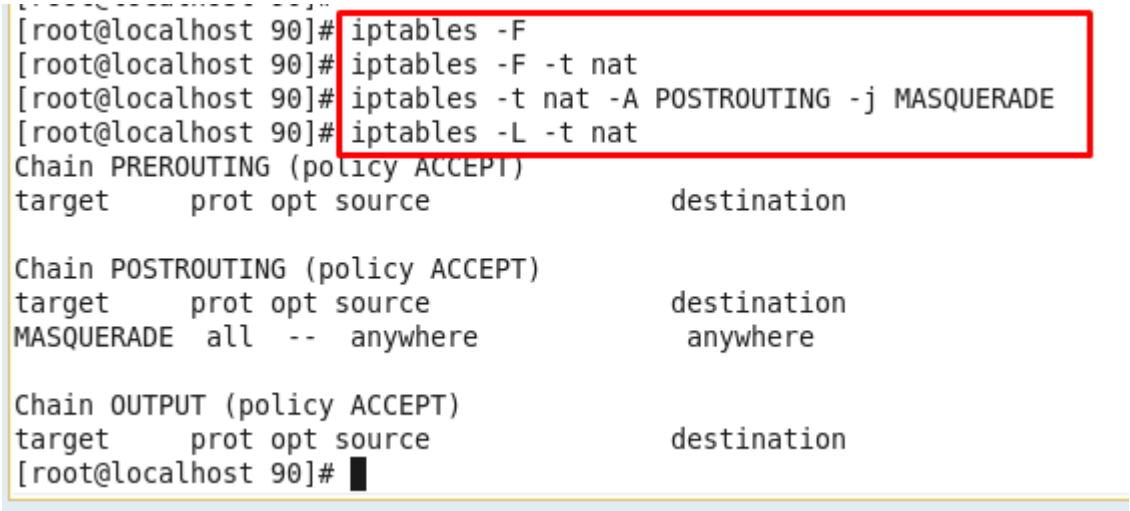
Donde se procede a escribir el mismo comando.

- for s in /proc/sys/net/ipv4/conf/*; do echo 0 > $s/send_redirects; echo 0 > $s/accept_redirects; done



Ahora se procede a configurar el Firewall de nuestro servidor para IPSEC.

- iptables -F -t nat
- iptables -t nat -A POSTROUTING -j MASQUERADE
- iptables -L -t nat



Con estas reglas configuramos los puertos de seguridad en los servidores.

- nano /etc/sysconfig/iptables
- A INPUT -p udp -m state --state NEW --dport 53 -j ACCEPT
- A INPUT -p tcp -m state --state NEW --dport 53 -j ACCEPT
- A INPUT -p udp --sport 500 --dport 500 -j ACCEPT
- A INPUT -p udp --sport 4500 --dport 4500 -j ACCEPT
- A OUTPUT -p udp --sport 500 --dport 500 -j ACCEPT
- A OUTPUT -p udp --sport 4500 --dport 4500 -j ACCEPT
- A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
- A INPUT -p icmp -j ACCEPT
- A INPUT -i lo -j ACCEPT
- A INPUT -m state --state NEW -m tcp -p --dport 22 -j ACCEPT
- A INPUT -j REJECT --reject-with icmp-host-prohibited
- A FORWARD -j REJECT --reject-with icmp-host-prohibited

```
:INPUT ACCEPT [7:2440]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [1:328]


INPUT -p udp -m state --state NEW --dport 53 -j ACCEPT
INPUT -p tcp -m state --state NEW --dport 53 -j ACCEPT
INPUT -p udp --sport 500 --dport 500 -j ACCEPT
INPUT -p udp --sport 4500 --dport 4500 -j ACCEPT
OUTPUT -p udp --sport 500 --dport 500 -j ACCEPT
OUTPUT -p udp --sport 4500 --dport 4500 -j ACCEPT
INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
INPUT -p icmp -j ACCEPT
INPUT -i lo -j ACCEPT
INPUT -m state --state NEW -m tcp -p --dport 22 -j ACCEPT
INPUT -j REJECT --reject-with icmp-host-prohibited
FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Mon Jul  4 13:24:56 2016
```

Se guarda y se reinicia las iptables.

- /etc/init.d/iptables save
- /etc/init.d/iptables restart

```
[root@localhost 90]# /etc/init.d/iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[  OK  ]
[root@localhost 90]# /etc/init.d/iptables restart
iptables: Setting chains to policy ACCEPT: nat filter      [  OK  ]
iptables: Flushing firewall rules:                         [  OK  ]
iptables: Unloading modules:                               [  OK  ]
iptables: Applying firewall rules:                         [  OK  ]
```

Instalar OpenSwan VPN con el comando.

- yum -y install openswan

```
[root@localhost 90]# yum -y install openswan
Loaded plugins: fastestmirror, refresh-packagekit, security
Setting up Install Process
Loading mirror speeds from cached hostfile
 * base: mirror.cedia.org.ec
 * epel: mirror.cedia.org.ec
 * extras: mirror.cedia.org.ec
 * updates: mirror.cedia.org.ec
Package openswan is obsoleted by libreswan, trying to install libreswan-3.15-5.3
.el6.x86_64 instead
Resolving Dependencies
--> Running transaction check
---> Package libreswan.x86_64 0:3.15-5.3.el6 will be installed
--> Processing Dependency: libunbound.so.2()(64bit) for package: libreswan-3.15-
5.3.el6.x86_64
--> Processing Dependency: libevent_pthreads-2.0.so.5()(64bit) for package: libr
eswan-3.15-5.3.el6.x86_64
--> Processing Dependency: libevent-2.0.so.5()(64bit) for package: libreswan-3.1
5-5.3.el6.x86_64
--> Running transaction check
---> Package libevent2.x86_64 0:2.0.21-2.el6 will be installed
---> Package unbound-libs.x86_64 0:1.4.20-23.el6.3 will be installed
```

Se procede a que este servicio comience desde el arranque del sistema con:

- chkconfig ipsec on

```
[root@localhost 90]# chkconfig ipsec on
```

Se reinicia el servicio ipsec.

- /etc/init.d/ipsec restart

```
[root@localhost 90]# /etc/init.d/ipsec restart
Missing control file /var/run/pluto/pluto.ctl - is pluto running?
Starting pluto IKE daemon for IPsec: Initializing NSS database
See 'man pluto' if you want to protect the NSS database with a password

.                                                          [  OK  ]
```

Copiamos el archivo de configuración de ipsec y ipsec secrets a la ruta etc/ipsec.conf_org y etc/ipsec.secrets_org respectivamente.

- cp /etc/ipsec.conf /etc/ipsec.conf_org

```
[root@localhost 90]# cp /etc/ipsec.conf /etc/ipsec.conf_org
[root@localhost 90]# cp /etc/ipsec.secrets /etc/ipsec.secrets_org
```

Se escribirán las siguientes sentencias para

```
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/default/send_redirects
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/eth0/send_redirects
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/eth1/send_redirects
bash: /proc/sys/net/ipv4/conf/eth1/send_redirects: No such file or directory
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/lo/send_redirects
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/default/accept_redirects
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/eth0/accept_redirects
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/eth1/accept_redirects
bash: /proc/sys/net/ipv4/conf/eth1/accept_redirects: No such file or directory
[root@localhost 90]# echo 0 > /proc/sys/net/ipv4/conf/lo/accept_redirects
```

En este proceso vamos a crear el proceso de las claves para la autenticación entre los dos servidores.

- ipsec newhostkey -- configdir /etc/ipsec.d --output /etc/ipsec.secrets --bits 4096

```
[root@localhost 90]# ipsec newhostkey -- configdir /etc/ipsec.d --output /etc/ip
sec.secrets --bits 4096
/usr/libexec/ipsec/newhostkey: WARNING: file "/etc/ipsec.secrets" exists, append
ing to it
Generated RSA key pair using the NSS database
```

Ahora vamos a configurar el nombre de cada servidor que tendrá la VPN para colocar sus respectivas claves de autenticación y de encriptación a nivel ip.

Este comando es para el servidor 1:

- ipsec showhostkey --left

```
[root@localhost 90]# ipsec showhostkey --left
ipsec showhostkey loading secrets from "/etc/ipsec.secrets"
ipsec showhostkey no secrets filename matched "/etc/ipsec.d/*.secrets"
ipsec showhostkey loaded private key for keyid: PPK_RSA:AQPhAjzxp
        # rsakey AQPhAjzxp
        leftrsasigkey=0sAQPhAjzxpdP7uHzK6jNmCZiijYBm4/Uj52JEcuWwgf3+Vn5pvakSnhAM
37u/CnJCLjRQTWw6ixTXLCXvzfBFD4YA5ibUU6maVtIYmnuIgVCObJXlu161KzgC49cH8HK1ive/axQ6
7iI6YH8yuH1I4cYJu9YZePBdudj4n/ZR07HV0T9XLkb884jzl8OMRkOEhTmMtCc7lrcmZSmSmKO+s8h4
PEMFkJIMtJV/Ou5JgKsMlQQCEHCo+O4pXwjS67WF2kekE5xS6FI67WeY69jbcVfte5ZyBjihaWhgS1Rg
6LgV7G8nKUSyoOoSKmX1Ej5ZgTZoUrba/zC9zxpRELbUynJNedCZiu0Gr0a0WqWOwhErM+qX21qOY/pQ
2QHL9RcOnWCszRcggud1pPiybqB4aeb73z6ThUCUS+pLaCJ+JBEguIJZ8fDegmIJLDgRPoY2zNkMJPAa
nXoikSy2kXboOJJwapV4DT8OMpcyaDIanpFar9l5QZjQMPwhm3gvMoLt6AwN8ZtJGEwnp9gAP/B/EGP+
i2150B4yoGiTOHPU5aoBHGI48ELway421WYQgljFG/EuuPNaTWFBLCMnd+NtXhIi1MJDCJED0zz3Jldx
E8XZOSgQa2tRiQ==
```

Este comando es para el servidor 2

- ipsec showhostkey –right

```
[root@localhost utn]# ipsec showhostkey --right
ipsec showhostkey loading secrets from "/etc/ipsec.secrets"
ipsec showhostkey no secrets filename matched "/etc/ipsec.d/*.secrets"
ipsec showhostkey loaded private key for keyid: PPK_RSA:AQPGwJh9N
ipsec showhostkey loaded private key for keyid: PPK_RSA:AQPRf2UG8
        # rsakey AQPRf2UG8
        rightrsasigkey=0sAQPRf2UG8Cd/Rd6NsgFWvauu+zOnKuS7/RU07QMLdFFz+nTDw+RmG4
bpZPi3KqO8COQ3VvSJDiGIymbztZikky2JHwd50vIoTblQ5HpGpYKGQKgkS7yYyUStXcGk1KcObyhXY
6TrYFfOtBy952Cp01F0gX5HcG0K2OWaqelYc607NNo1gI5MTvx+2/yumqRIZ167PlG0D2PiC6kU0Ad3
0QJbAfwzXN62ZYJASx7JPkUFEucCFSBUptcgXUxWGNgr2sCuj/oC06/4EElNtjRdMASKH8E+en9YBPa
JUs4/KhWXSJDaLsf9ua1F3/E8k5PK3tiur5FjNOizJyUx5bORGoEUUUFKGjiImxXsmO/PJvYW4Dxt0C
mQsd312ZO7spJRLG1VgiXttl8+NN5FrLhODkD8QbdZzi3Kzd9+JRXAYOvyKt88aPXLI0HDTgqUDDXrl
Fpiz0p58BAF1SH1LxClScEws2Z15Eqm9zeZrhXW2TuYlyESeNDfVyIUn1vxPmZ7jfZ2cHO87v7h+Adv
2+3VvXBCkmNKcLUOWqA861V4r
```

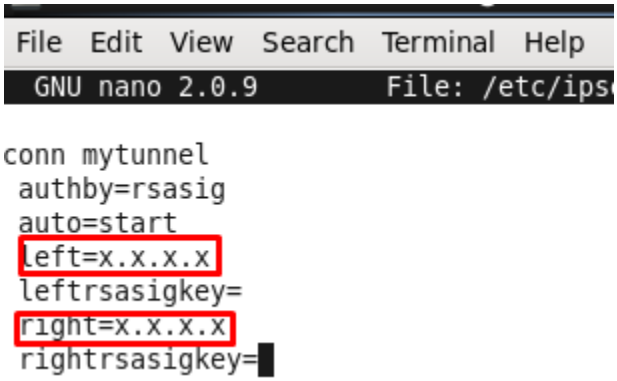
El siguiente comando para dar inicialización del túnel en mi caso es mytunnel,conf de configuración de ips y de claves de cifrado con el uso de los protocolos de la misma.

- touch /etc/ipsec.d/mytunnel.conf

```
E8XZOSgQa2tRiQ==
[root@localhost 90]# touch /etc/ipsec.d/mytunnel.conf
[root@localhost 90]#
```

Se procede a configurar el túnel con el siguiente comando, para los dos servidores.

- nano /etc/ipsec.d/mytunnel.conf



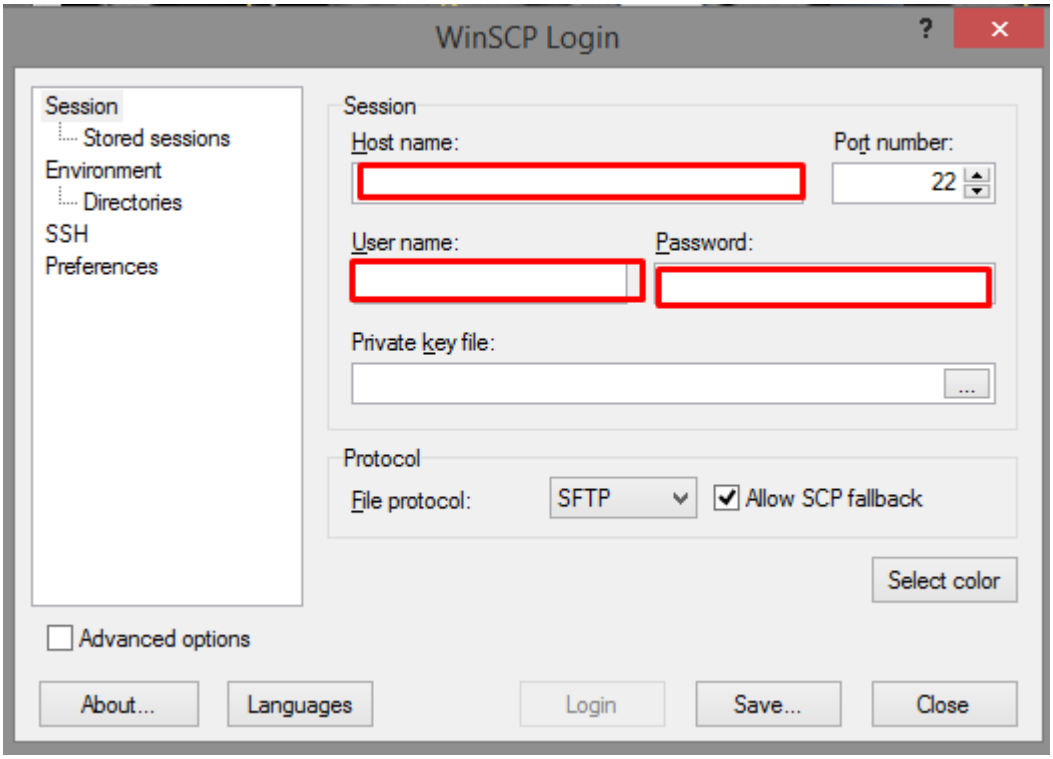
Dentro de este archivo de configuración precederemos a colocar lo siguiente:

- conn mytunnel
- authby=rsasig
- auto=start
- left=x.x.x.x
- leftrsasigkey=
- right=x.x.x.x
- rightrsasigkey=

En el parámetro left se coloca la ip publica del primer servidor, asi como en right se coloca la ip pero del segundo servidor 2.

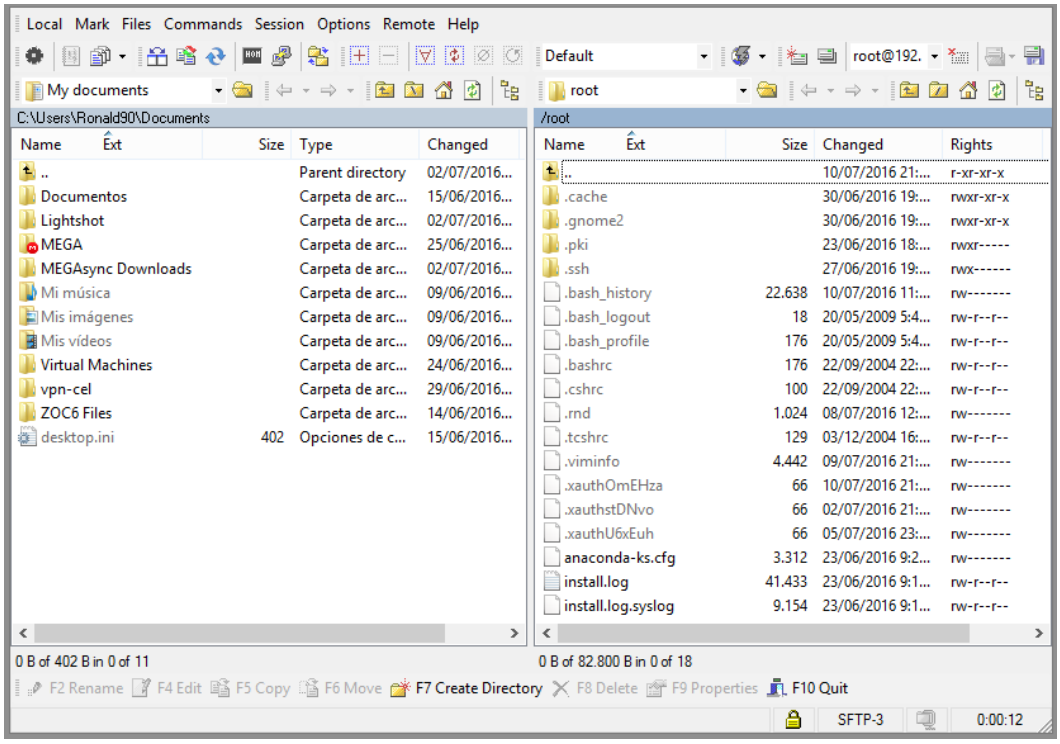
En la parte de leftrsasigkey y rightrsasigkey se coloca las llaves de autenticación elaboradas por los protocolos de autenticación.

Se debe instalar **WinSCP** en los servidores, para de esta manera poder copiar las claves de seguridad.

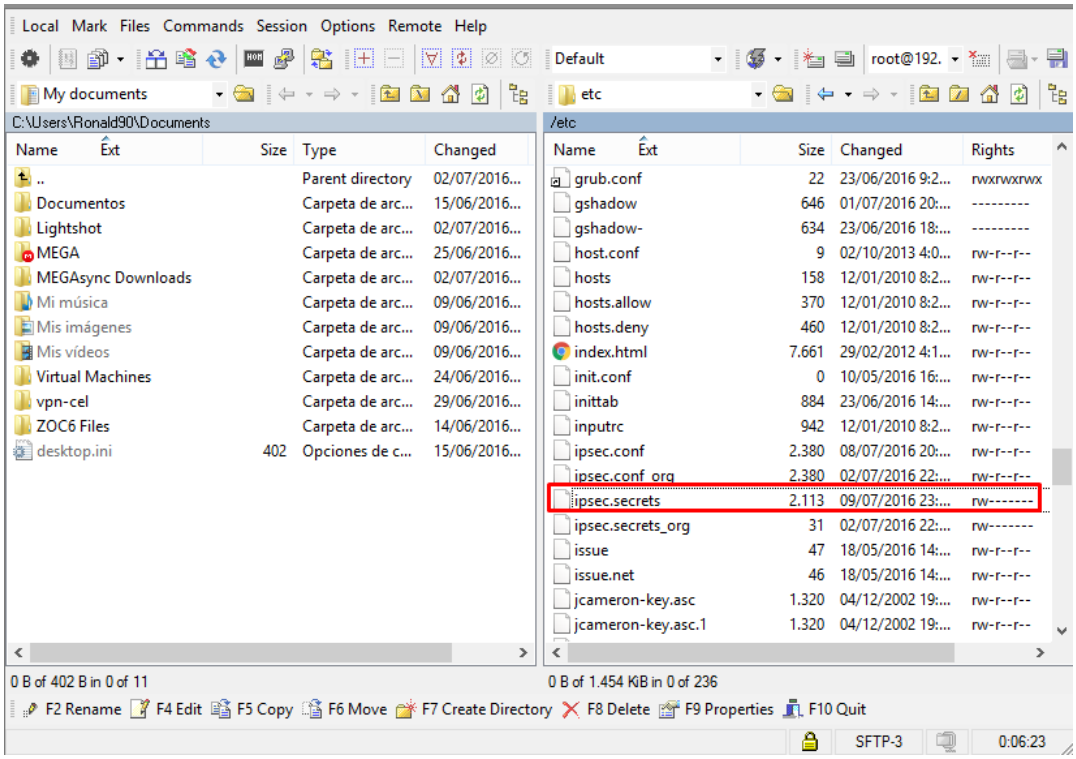


Una vez ingresado correctamente al servidor1 debemos dirigirnos a:

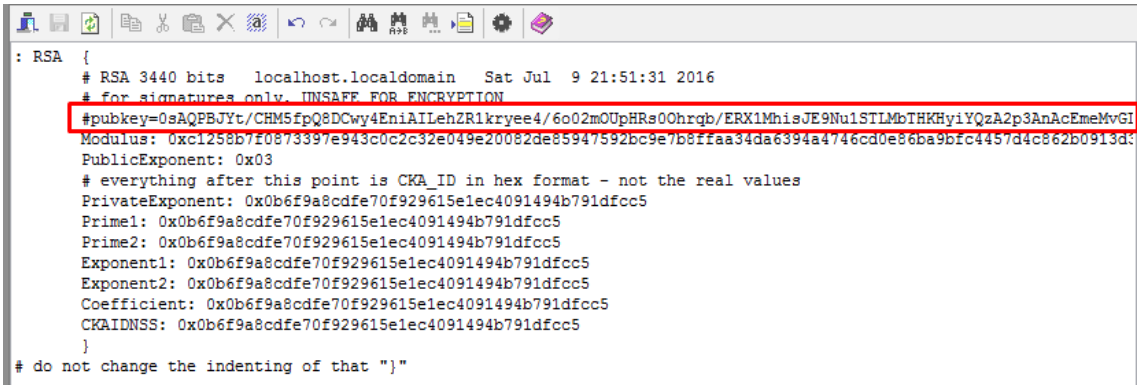
- A la carpeta etc



- Buscamos donde se encuentra ipsec.secrets

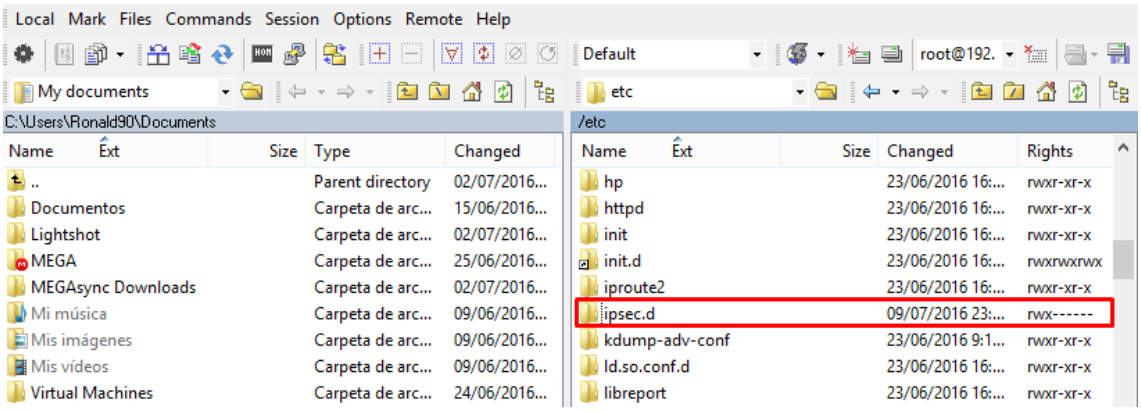


- Abrimos el editor de texto y procedemos a copiar el pubkey generado por parte del primer servidor 1.

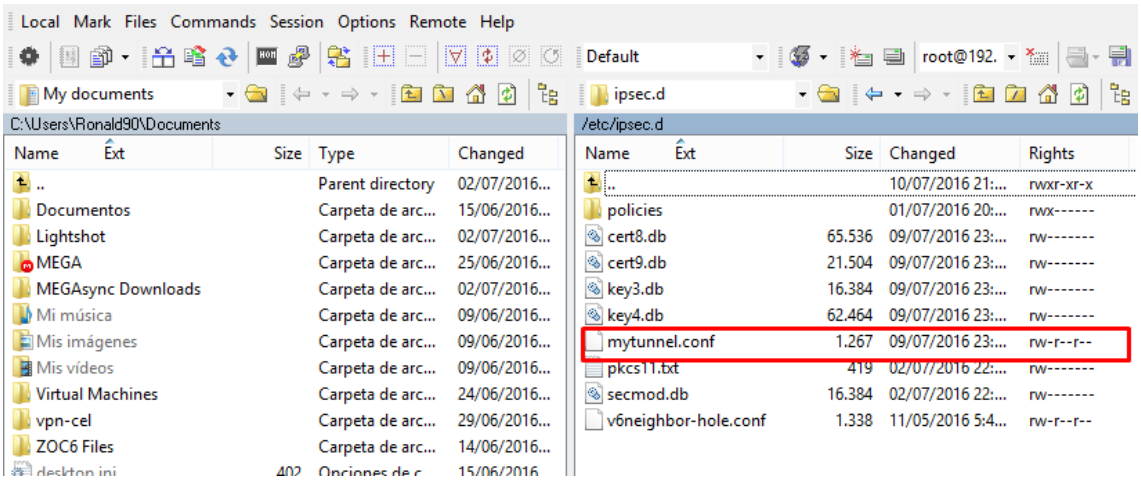


Una vez copiado la clave del servidor 1 se procede a realizar lo siguiente:

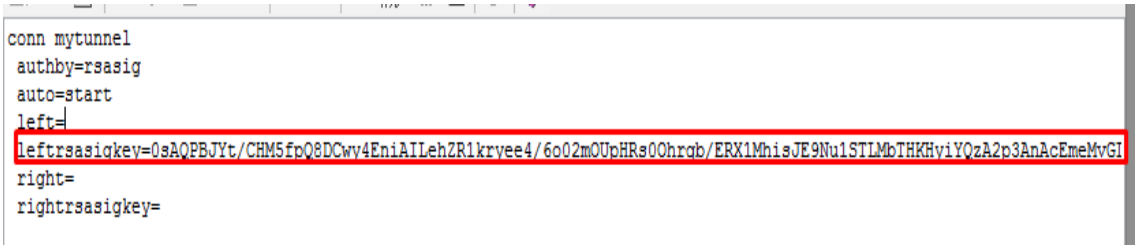
- Dentro de la carpeta etc, buscamos ipsec.d



- Una vez dentro de esta carpeta, abrimos mytunnel.conf



- Es aquí en este archivo de texto donde se copiara la clave de seguridad

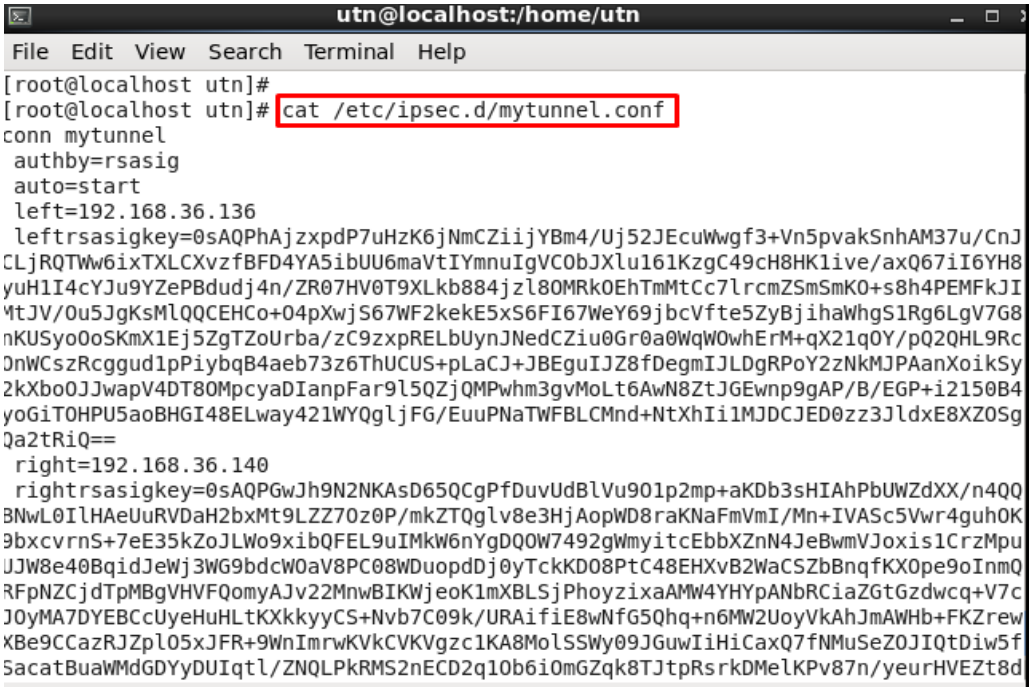


De la misma manera nos dirigimos al servidor 2 para poder copiar la clave para la autenticación de cifrado y para que IPSec esté en funcionamiento.

Con el siguiente comando vemos las laves generadas por cada uno de los servidores

- cat /etc/ipsec.d/mytunnel.conf cat /etc/ipsec.d/mytunnel.conf

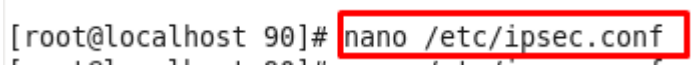
Servidor 1:


utn@localhost:/home/utn
File Edit View Search Terminal Help
[root@localhost utn]#
[root@localhost utn]# cat /etc/ipsec.d/mytunnel.conf
conn mytunnel
authby=rsasig
auto=start
left=192.168.36.136
leftrsasigkey=0sAQPhAjzxpdP7uHzK6jNmCZiijYBm4/Uj52JEcuWwgf3+Vn5pvakSnhAM37u/CnJ
CLjRQTWw6ixTXLCXvzfBFD4YA5ibUU6maVtIYmnuIgVCObJXlu161KzgC49cH8HK1ive/axQ67iI6YH8
yuH1I4cYJu9YZePBdudj4n/ZR07HV0T9XLkb884jzl8OMRkOEhTmMtCc7lrcmZSmSmKO+s8h4PEMFkJI
MtJV/Ou5JgKsMlQQCEHCo+O4pXwjS67WF2kekE5xS6FI67WeY69jbcVfte5ZyBjihaWhgS1Rg6LgV7G8
nKUSyoOoSKmX1Ej5ZgTZoUrba/zC9zxpRELbUynJNedCZiu0Gr0a0WqWOwhErM+qX21qOY/pQ2QHL9Rc
OnWCszRcggud1pPiybqB4aeb73z6ThUCUS+pLaCJ+JBEguIJZ8fDegmIJLDgRPoY2zNkMJPAanXoikSy
2kXboOJJwapV4DT8OMpcyaDIanpFar9l5QZjQMPwhm3gvMoLt6AwN8ZtJGEwnp9gAP/B/EGP+i2150B4
yoGiTOHPU5aoBHGI48ELway421WYQgljFG/EuuPNaTWFBLCMnd+NtXhIi1MJDCJED0zz3JldxE8XZOSg
Qa2tRiQ==
right=192.168.36.140
rightrsasigkey=0sAQPGwJh9N2NKAsD65QCgPfDuvUdBlVu9O1p2mp+aKDb3sHIAhPbUWZdXX/n4QQ
BNwL0IlHAeUuRVDaH2bxMt9LZZ7Oz0P/mkZTQglv8e3HjAopWD8raKNaFmVmI/Mn+IVASc5Vwr4guhOK
9bxcvrnS+7eE35kZoJLWo9xibQFEL9uIMkW6nYgDQOW7492gWmyitcEbbXZnN4JeBwmVJoxis1CrzMpu
UJW8e40BqidJeWj3WG9bdcWOaV8PC08WDuopdDj0yTckKDO8PtC48EHXvB2WaCSZbBnqfKXOpe9oInmQ
RFpNZCjdTpMBgVHVFQomyAJv22MnwBIKWjeoK1mXBLSjPhoyzixaAMW4YHYpANbRCiaZGtGzdwcq+V7c
JOyMA7DYEBCcUyeHuHLtKXkkyyCS+Nvb7C09k/URAifiE8wNfG5Qhq+n6MW2UoyVkAhJmAWHb+FKZrew
XBe9CCazRJZplO5xJFR+9WnImrwKVkCVKVgzc1KA8MolSSWy09JGuwIiHiCaxQ7fNMuSeZOJIQtDiw5f
SacatBuaWMdGDYyDUIqtl/ZNQLPkRMS2nECD2q1Ob6iOmGZqk8TJtpRsrkDMelKPv87n/yeurHVEZt8d

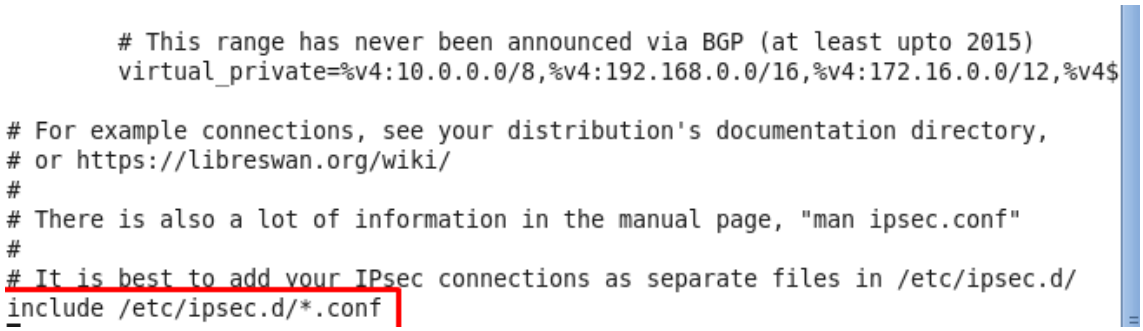
Servidor 2:


[root@localhost 90]# cat /etc/ipsec.d/mytunnel.conf
conn mytunnel
authby=rsasig
auto=start
left=192.168.36.136
leftrsasigkey=0sAQPhAjzxpdP7uHzK6jNmCZiijYBm4/Uj52JEcuWwgf3+Vn5pvakSnhAM37u/CnJ
CLjRQTWw6ixTXLCXvzfBFD4YA5ibUU6maVtIYmnuIgVCObJXlu161KzgC49cH8HK1ive/axQ67iI6YH8
yuH1I4cYJu9YZePBdudj4n/ZR07HV0T9XLkb884jzl8OMRkOEhTmMtCc7lrcmZSmSmKO+s8h4PEMFkJI
MtJV/Ou5JgKsMlQQCEHCo+O4pXwjS67WF2kekE5xS6FI67WeY69jbcVfte5ZyBjihaWhgS1Rg6LgV7G8
nKUSyoOoSKmX1Ej5ZgTZoUrba/zC9zxpRELbUynJNedCZiu0Gr0a0WqWOwhErM+qX21qOY/pQ2QHL9Rc
OnWCszRcggud1pPiybqB4aeb73z6ThUCUS+pLaCJ+JBEguIJZ8fDegmIJLDgRPoY2zNkMJPAanXoikSy
2kXboOJJwapV4DT8OMpcyaDIanpFar9l5QZjQMPwhm3gvMoLt6AwN8ZtJGEwnp9gAP/B/EGP+i2150B4
yoGiTOHPU5aoBHGI48ELway421WYQgljFG/EuuPNaTWFBLCMnd+NtXhIi1MJDCJED0zz3JldxE8XZOSg
Qa2tRiQ==
right=192.168.36.140
rightrsasigkey=0sAQPGwJh9N2NKAsD65QCgPfDuvUdBlVu9O1p2mp+aKDb3sHIAhPbUWZdXX/n4QQ
BNwL0IlHAeUuRVDaH2bxMt9LZZ7Oz0P/mkZTQglv8e3HjAopWD8raKNaFmVmI/Mn+IVASc5Vwr4guhOK
9bxcvrnS+7eE35kZoJLWo9xibQFEL9uIMkW6nYgDQOW7492gWmyitcEbbXZnN4JeBwmVJoxis1CrzMpu
UJW8e40BqidJeWj3WG9bdcWOaV8PC08WDuopdDj0yTckKDO8PtC48EHXvB2WaCSZbBnqfKXOpe9oInmQ
RFpNZCjdTpMBgVHVFQomyAJv22MnwBIKWjeoK1mXBLSjPhoyzixaAMW4YHYpANbRCiaZGtGzdwcq+V7c
JOyMA7DYEBCcUyeHuHLtKXkkyyCS+Nvb7C09k/URAifiE8wNfG5Qhq+n6MW2UoyVkAhJmAWHb+FKZrew
XBe9CCazRJZplO5xJFR+9WnImrwKVkCVKVgzc1KA8MolSSWy09JGuwIiHiCaxQ7fNMuSeZOJIQtDiw5f
SacatBuaWMdGDYyDUIqtl/ZNQLPkRMS2nECD2q1Ob6iOmGZqk8TJtpRsrkDMelKPv87n/yeurHVEZt8d

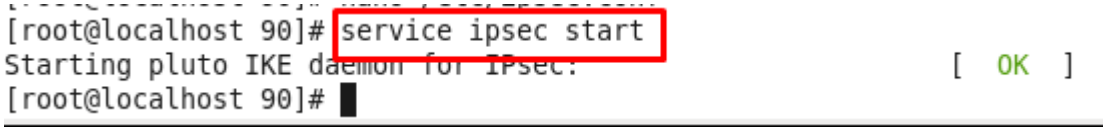
Se procede a ver el estado de configuración de ipsec, con el siguiente comando.



Dentro de esta configuración observamos que no se encuentre comentado lo siguiente:



Procedemos a reiniciar el servicio ipsec en el servidor, con el comando.



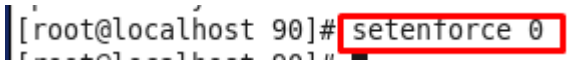
Una vez iniciado verificamos ipsec si se encuentra con algún error ya sea de Kernel o de algún otro parámetro de configuración.

- ipsec verify

Con los siguientes comandos ponemos a ipsec en estado de habilitado y que se encuentra habilitado el protocolo de seguridad dentro del túnel VPN.

- setenforce 0

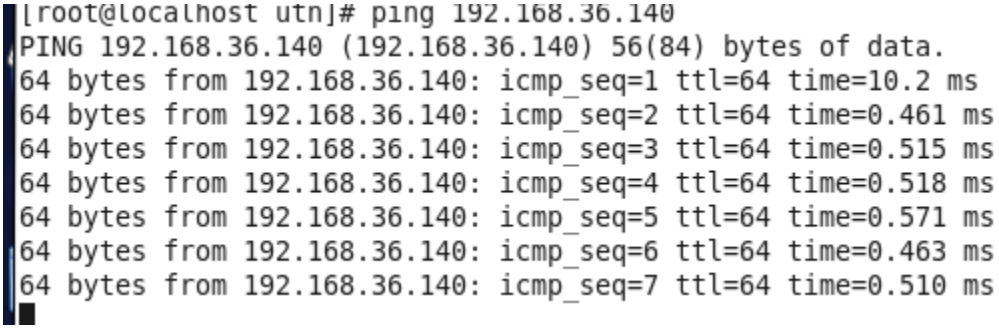
- ipsec auto --add mytunnel
- ipsec auto --up mytunnel



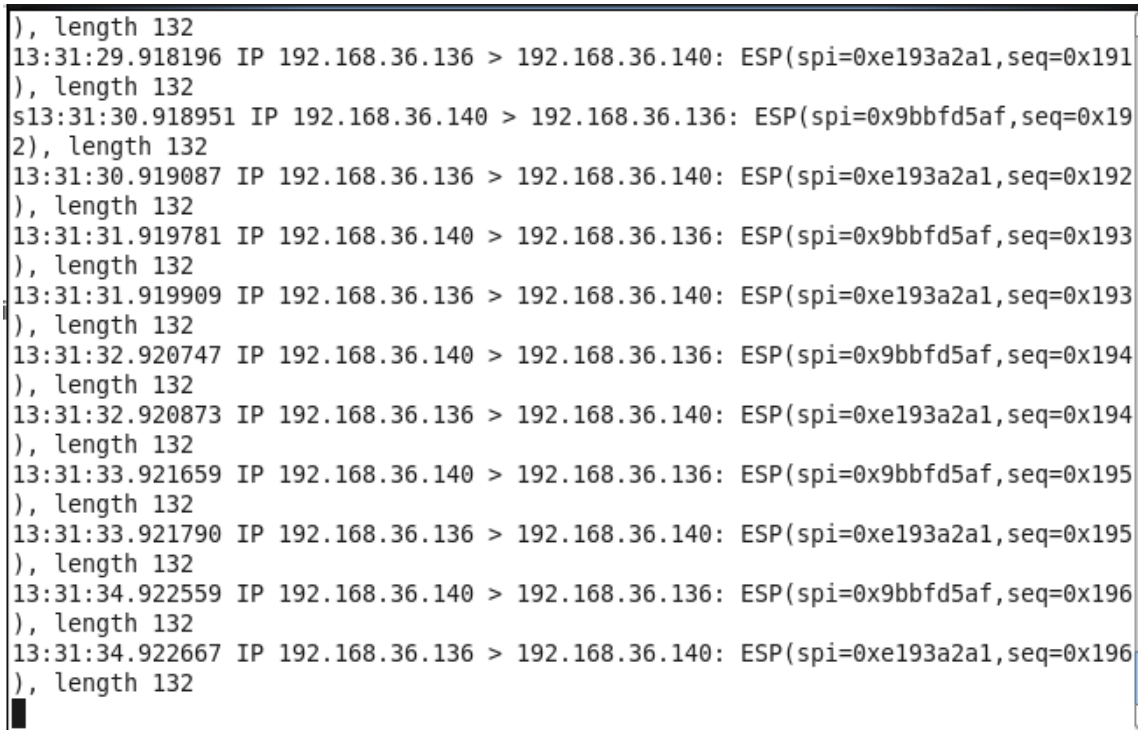
Una vez terminado de configurar IPSec se procede a verificar con un ping desde un servidor a otro servidor de la siguiente manera.

- El primer servidor 1 ejecutamos el icmp con el comando ping.
- Al segundo servidor 2 ejecutamos el tcpdump -n -i eth0 esp para de esta manera observar el funcionamiento de los protocolos de seguridad, IPSec.

Servidor 1:



Servidor 2:


), length 132
13:31:29.918196 IP 192.168.36.136 > 192.168.36.140: ESP(spi=0xe193a2a1,seq=0x191
), length 132
s13:31:30.918951 IP 192.168.36.140 > 192.168.36.136: ESP(spi=0x9bbfd5af,seq=0x19
2), length 132
13:31:30.919087 IP 192.168.36.136 > 192.168.36.140: ESP(spi=0xe193a2a1,seq=0x192
), length 132
13:31:31.919781 IP 192.168.36.140 > 192.168.36.136: ESP(spi=0x9bbfd5af,seq=0x193
), length 132
13:31:31.919909 IP 192.168.36.136 > 192.168.36.140: ESP(spi=0xe193a2a1,seq=0x193
), length 132
13:31:32.920747 IP 192.168.36.140 > 192.168.36.136: ESP(spi=0x9bbfd5af,seq=0x194
), length 132
13:31:32.920873 IP 192.168.36.136 > 192.168.36.140: ESP(spi=0xe193a2a1,seq=0x194
), length 132
13:31:33.921659 IP 192.168.36.140 > 192.168.36.136: ESP(spi=0x9bbfd5af,seq=0x195
), length 132
13:31:33.921790 IP 192.168.36.136 > 192.168.36.140: ESP(spi=0xe193a2a1,seq=0x195
), length 132
13:31:34.922559 IP 192.168.36.140 > 192.168.36.136: ESP(spi=0x9bbfd5af,seq=0x196
), length 132
13:31:34.922667 IP 192.168.36.136 > 192.168.36.140: ESP(spi=0xe193a2a1,seq=0x196
), length 132

Anexo P. Firewall de los servidores de la Red Privada Virtual VPN

Con el comando iptables –L podemos ver el firewall de nuestro servidor VPN.

```
[root@UTNVPN-Ronald Ronald]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:https

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Se procede a configurar el firewall de nuestro servidor, con las siguientes reglas de firewall.

Esta regla para que acepte el tráfico que proviene del túnel a la dirección IP de nuestro servidor.

- iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -j SNAT --to-source 186.5.55.250

Escribimos las siguientes reglas:

- iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
- iptables -A FORWARD -s 10.8.0.0/24 -j ACCEPT
- iptables -A FORWARD –j REJECT

```
Contraseña:
[root@UTNVPN-Ronald Ronald]# iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -j SNAT --to-source 186.5.55.250
[root@UTNVPN-Ronald Ronald]# service iptables save
iptables: Guardando las reglas del cortafuegos en /etc/sysc[  OK  ]tables:
[root@UTNVPN-Ronald Ronald]#
[root@UTNVPN-Ronald Ronald]# iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A FORWARD -s 10.8.0.0/24 -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A FORWARD -j REJECT
```

Escribimos las siguientes reglas para aceptar el tráfico de la VPN a través del túnel que se encuentra encriptado por los protocolos de encriptación.

iptables -A INPUT -i tun+ -j ACCEPT
iptables -A FORWARD -i tun+ -j ACCEPT
iptables -A OUTPUT -o tun+ -j ACCEPT
iptables -A FORWARD -o tun+ -j ACCEPT

iptables -A INPUT -i tap+ -j ACCEPT
iptables -A FORWARD -i tap+ -j ACCEPT
iptables -A OUTPUT -o tap+ -j ACCEPT
iptables -A FORWARD -o tap+ -j ACCEPT

```
[root@UTNVPN-Ronald Ronald]# iptables -A INPUT -i tun+ -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A FORWARD -i tun+ -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A OUTPUT -o tun+ -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A FORWARD -o tun+ -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A INPUT -i tap+ -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A FORWARD -i tap+ -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A OUTPUT -o tap+ -j ACCEPT
[root@UTNVPN-Ronald Ronald]# iptables -A FORWARD -o tap+ -j ACCEPT
```

Habilitamos el puerto 80.

- Iptables –A IMPUT –m state –state NEW –m tcp –p tcp –dport 80 –j ACCEPT

```
[root@UTNVPN-Ronald Ronald]# iptables -A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
[root@UTNVPN-Ronald Ronald]# service iptables save
iptables: Guardando las reglas del cortafuegos en /etc/sysc[  OK  ]tables:
[root@UTNVPN-Ronald Ronald]# service iptables restart
iptables: Poniendo las cadenas de la polÃ-tica ACCEPT: nat [  OK  ]
iptables: Guardando las reglas del cortafuegos:           [  OK  ]
iptables: Descargando mÃ³dulos:                            [  OK  ]
iptables: Aplicando reglas del cortafuegos:               [  OK  ]
iptables: Cargando mÃ³dulos adicionales:nf_conntrack_ftp   [  OK  ]
```

Se guarda y se reinician las iptables.

- service iptables save
- service iptables restart

```
[root@UTNVPN-Ronald Ronald]# service iptables save
iptables: Guardando las reglas del cortafuegos en /etc/sysc[  OK  ]tables:
[root@UTNVPN-Ronald Ronald]# service iptables restart
iptables: Poniendo las cadenas de la polÃ-tica ACCEPT: nat [  OK  ]
iptables: Guardando las reglas del cortafuegos:            [  OK  ]
iptables: Descargando mÃ³dulos:                             [  OK  ]
iptables: Aplicando reglas del cortafuegos:                [  OK  ]
iptables: Cargando mÃ³dulos adicionales:nf_conntrack_ftp    [  OK  ]
[root@UTNVPN-Ronald Ronald]#
```