ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

"ANÁLISIS Y COMPARACIÓN DE LA SEGURIDAD EN DOS ESQUEMAS DE RED, UTILIZANDO LOS PROTOCOLOS IPV4 E IPV6"

INFORME DE PROYECTO DE GRADUACIÓN

Previo a la obtención del Título de:

INGENIERO EN TELEMÁTICA

Presentado por:

DANIEL ANDRÉS PÁEZ SÁNCHEZ

ISRAEL EMANUEL MALDONADO BELTRÁN

GUAYAQUIL – ECUADOR

Año 2015

A G R A D E C I M I E N T O

Agradecemos primeramente a DIOS por permitirnos culminar esta etapa de nuestra vida. Agradecemos también a todos y cada uno de nuestros mentores que compartieron con nosotros su experiencia y enseñanzas en nuestro paso por la ESPOL. Y no podemos dejar de agradecer a nuestros amigos y familiares que formaron parte del día a día en nuestra carrera universitaria.

D E D I C A T O R I A

Dedico este trabajo a mi abuela Ana Susana Eguez (+), a mis padres Daniel Gerardo Páez y Tannia Sánchez por su motivación y apoyo incondicional. Gracias a ellos he podido culminar mis objetivos

Daniel Páez

Dedico este trabajo a DIOS, a mis Padres César Maldonado y Elisa B. de Maldonado, a mi hermana Mariela Maldonado y a todos aquellos que estuvieron siempre cerca ayudándome a cumplir esta meta.

Israel Maldonado

TRIBUNAL DE SUSTENTACIÓN

Ph.D. Sixto García Aguilar

SUBDECANO SUBROGANTE DE LA FIEC

Mg. José Patiño Sánchez

DIRECTOR DEL PROYECTO DE GRADUACIÓN

Mg. Albert Espinal Santana

MIEMBRO DEL TRIBUNAL

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de este Informe, nos corresponde exclusivamente; y el patrimonio intelectual del mismo, a la **ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**"

(Reglamento de Graduación de la ESPOL)

Daniel Andrés Páez Sánchez

Israel Emanuel Maldonado Beltrán

RESUMEN

No es una novedad el hecho de que la tecnología está en constante evolución, las redes informáticas se han vuelto indispensables en nuestro diario vivir. El crecimiento de éstas es abrumador, por ende la capacitación en cuanto a la administración de las mismas debe ser prioritaria.

Como bien se sabe, el protocolo IP que ha sido altamente difundido, nos permite establecer comunicaciones a través de una red. La versión IPv4 ha permitido dar grandes pasos en el desarrollo de las tecnologías de la información, por lo que su uso ha sido muy extendido, sin embargo, debido al crecimiento exponencial de las redes alrededor del mundo, se presentó el agotamiento de direcciones IPv4, y se decidió implementar un nuevo protocolo: IPv6. Este protocolo a diferencia de IPv4 no ha sido altamente difundido, todavía.

La seguridad informática es un aspecto que siempre debe ser tomado en cuenta al momento de realizar algún cambio o actualización de nuestros diversos sistemas de información. Los denominados “huecos” de seguridad siempre estarán presentes en cualquier sistema o protocolo de red presentando riesgos que toda organización debe estar lista para mitigar de una u otra forma.

ÍNDICE GENERAL

ABREVIATURAS Y SIMBOLOGÍA

ACK	Acuse de recibo
AES	Cifrado de datos avanzado
AH	Cabecera de autenticación
ARP	Protocolo de resolución de direcciones
BDR	Enrutador designado de reserva
BGP	Protocolo de encaminamiento – “Puertas afuera”
DDOS	Denegación de servicio distribuida
DES	Cifrado de datos estándar
DOS	Denegación de servicio
DR	Enrutador designado
EIGRPv3	Protocolo de Encaminamiento – “Puertas adentro mejorado”
ESP	Carga útil de seguridad de encapsulación
FTP	Protocolo de transferencia de archivos
ICMP	Protocolo de mensaje control de internet
IIS	Servicios de información de internet
IOS	Sistema operativo de redes de datos (Propiedad de CISCO SYSTEMS)
IP	Protocolo de Internet
IPSEC	Protocolo de seguridad de Internet

ISAKMP	Asociación de seguridad de Internet, Protocolo de Manejo de claves
IS-IS	Protocolo de encaminamiento – “Sistema intermedio a Sistema Intermedio”
LAN	Red de área local
MD5	Algoritmo de Resumen del Mensaje 5
MPLS	Protocolo de distribución por etiquetas
NAT	Protocolo de traducción de direcciones
OSI	Modelo de interconexión de sistemas abiertos
OSPFv3	Protocolo de encaminamiento – “El camino más corto primero”
RIPNG	Protocolo de Información de Enrutamiento
SHA	Algoritmo de seguridad hash
SSH	Intérprete de órdenes seguro
SSL	Capa de conexión segura
SYN	Mensaje de sincronización en el protocolo TCP
TCP	Protocolo de control de transmisión
TLS	Seguridad de la capa de transporte
UDP	Protocolo Datagrama de Usuario
VLSM	Máscaras de subred de tamaño variable
VPN	Red privada virtual
WAMP	Servidor de Windows, contiene APACHE, Mysql y PHP
WAN	Red de área amplia

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INTRODUCCIÓN

Internet es un canal de comunicación que unifica diferentes sistemas de información y que sirve para conectar a diversos agentes que están repartidos por todo el mundo.

En el mundo de la informática y las comunicaciones se han incluido los denominados protocolos, los cuales son reglas estrictas de cómo se dará una comunicación. Dentro de estos protocolos, tal vez el que tenga mayor trascendencia, es el protocolo IP; el cual establece los pasos por los que se va a regir una transmisión de datos en una red de información.

Desde un principio, al desarrollar el protocolo IP y otros tantos protocolos de comunicación, el primer interés fue asegurar el establecimiento de la conexión, es decir que la información llegara a su destino. El siguiente objetivo fue que la información se transmitiera en el menor tiempo posible, de tal manera que se desarrollaron sistemas de información sin tener muy presente la seguridad de la misma.

El hecho de plantear una solución con mucho esfuerzo para luego obtener resultados ineficientes y el hecho de considerar un riesgo mínimo el desprecio de la seguridad fueron unas de las razones por las que la seguridad de la

información no fue implementada como una prioridad en los protocolos de comunicación que hoy en día manejamos.

Con el tiempo nos hemos percatado que a pesar de que hemos tenido un alto desarrollo en cuanto a conectividad y velocidad de las comunicaciones, los riesgos de seguridad siempre estarán vigentes existiendo amenazas desde hace varias décadas atrás hasta el día de hoy.

A través de los años se han podido solucionar varios problemas de seguridad con la implementación de nuevos protocolos y medidas de precaución, sin embargo cada vez que se soluciona un problema aparece uno nuevo, por lo que en la actualidad es indispensable la concientización de la seguridad en nuestros entornos de comunicación.

CAPÍTULO 1

GENERALIDADES

En el presente capítulo se describen las consideraciones que se tomaron para plantear nuestro objeto de estudio, se justifica el "¿por qué?" del mismo y se propone una solución planteando los respectivos objetivos. Se define la forma en la que se desarrollará nuestro proyecto y se describen los alcances y limitaciones de éste.

1.1 DESCRIPCIÓN DEL PROBLEMA

La versión 4 del protocolo IP, asigna una dirección única a un host o sitio web a nivel mundial, es decir, no existe esta misma dirección ip en ninguna otra parte del mundo. Éstas son las denominadas IP's públicas. Debido al agotamiento de éstas direcciones, se han implementado diferentes soluciones temporales, tales como NAT o VLSM, la primera

por ejemplo, permite asignar IP’s privadas dentro de una red, de tal manera que tales direcciones no sean visibles fuera de esta red, por lo que estas pueden estar repetidas dentro de otras redes.

IPv6 apareció como posible solución a lo que es deficiente en IPv4. Es indudable que al implementar IPv6 no vamos a tener problema en cuanto a la asignación de IPs únicas, pues el número aumenta notablemente, sin embargo el tema de seguridad no ha cambiado en muchos aspectos.

Muchos ataques realizados en IPv4, aprovechando las vulnerabilidades de una red, también son factibles en IPv6. Un ejemplo de esto lo podemos ver en la realización de un ARP spoofing en una red IPv4, el cual también puede ser llevado a cabo con un ND (Neigboor Discovery) Spoofing en IPv6.

En los próximos años el protocolo IPv6 reemplazará ineludiblemente al protocolo IPv4. A pesar de esto, la conciencia sobre detalles técnicos y aspectos de seguridad de este "nuevo" protocolo sigue siendo escasa en los administradores de red de distintas organizaciones en Ecuador. Se tiene escaso conocimiento sobre las ventajas que trae el uso de IPv6 en cuanto a temas de seguridad respecta, además, la respuesta de una

red con protocolos que implementen IPv6 no ha sido profundizada tomando en cuenta parámetros de seguridad.

1.2 JUSTIFICACIÓN

A día de hoy para toda persona involucrada en el campo tecnológico es una necesidad estar actualizada con los últimos avances y cambios propios de éste campo. No es la excepción para el caso de los Ingenieros de Redes quienes se desarrollan en un área que cada día exige más velocidad, mayor número de usuarios, mayor capacidad de integración y seguridad. A su vez ésta área presenta nuevos protocolos, nuevas tecnologías y nuevas tendencias. Siendo así, el protocolo de IP versión 6, el cambio inminente en las redes de datos.

Los Ingenieros de redes deben estar preparados para éste tipo de cambios, anticipándose a los mismos, manejando información precisa y comprobada. Por ello es una necesidad conocer los detalles técnicos y los aspectos de seguridad que nos ofrece el protocolo IPv6. La seguridad es un factor crítico en los sistemas informáticos recayendo gran parte de ésta responsabilidad en la red en sí misma.

Debemos contar con datos precisos sobre la respuesta de una red que se encuentre configurada con el protocolo IPv6 y que esté bajo los

efectos de ataques informáticos que pudieran vulnerar la seguridad de la misma.

1.3 SOLUCIÓN PROPUESTA

Evaluar la respuesta de las redes IPv6 frente a posibles ataques a la seguridad informática de las mismas. Se obtendrán datos estadísticos a partir de las pruebas realizadas, los mismos que nos darán una mejor idea sobre la seguridad en IPv6. Estos resultados serán una nueva fuente de información para administradores de red que deseen conocer más detalles sobre la seguridad en IPv6.

1.4 OBJETIVOS GENERALES

- Analizar la respuesta, en términos de seguridad, de 2 protocolos que usen IPv6 evaluando los parámetros de disponibilidad, integridad y confidencialidad de la información.

- Comparar la respuesta, en términos de seguridad, de 2 protocolos que usen IPv6 evaluando los parámetros de disponibilidad, integridad y confidencialidad de la información.

- Analizar cómo se afectan ambos protocolos después de haber realizado los ataques de red.

1.5 OBJETIVOS ESPECÍFICOS

- Cuantificar la respuesta de los protocolos OSPFv3 y RIPng considerando los parámetros de disponibilidad, integridad y confidencialidad en condiciones normales.

- Cuantificar la respuesta de los protocolos OSPFv3 y RIPng considerando los parámetros de disponibilidad, integridad y confidencialidad bajo efectos de algún ataque de red.

- Cuantificar la respuesta de los protocolos OSPFv3 y RIPng considerando los parámetros de disponibilidad, integridad y confidencialidad haciendo uso de un túnel VPN-IPSec en condiciones normales.

- Cuantificar la respuesta de los protocolos OSPFv3 y RIPng considerando los parámetros de disponibilidad, integridad y confidencialidad haciendo uso de un túnel VPN-IPSec bajo efectos de algún ataque de red.

- Declarar cuál de los dos protocolos IPv6 ofrece mayor seguridad en base a la respuesta de cada parámetro evaluado.

1.6 METODOLOGÍA

Se diseñará e implementará virtualmente dos redes de datos conformadas por enrutadores y ordenadores, la primera red será configurada con el protocolo OSPFv3 y la segunda con RIPng. Se manejará un modelo cliente-servidor, donde los clientes intercambiarán información con las bases de datos de los servidores interactuando con los aplicativos prestados por éstos. En la parte de seguridad, se implementará un túnel VPN-IPSec.

Efectuaremos distintos tipos de ataques informáticos a las redes implementadas y se medirá la respuesta de cada una tomando en consideración los siguientes parámetros: Disponibilidad, integridad y confidencialidad. Luego procesaremos estadísticamente los resultados de cada prueba y se elaborarán tablas que nos permitirán comparar la respuesta de ambos protocolos.

Los ataques que realizaremos sobre las redes serán lo más efectivos posibles, ataques tales como ICMP6 flooding o falseo de rutas, serán implementados de tal manera que podamos comprobar la seguridad que brindan los protocolos IPv6.

1.7 ALCANCES Y LIMITACIONES

Los esquemas de red a ser implementados, simularán un entorno empresarial, donde cada nodo de la red representa las distintas sucursales y matrices que una empresa puede llegar a tener. A nivel lógico se representará las diferentes redes internas que una compañía maneja poniendo especial enfoque en la red de servidores y en la red de acceso para los empleados desde la cual se puede llevar a cabo algún tipo de ataque informático.

Los esquemas de red que implementaremos no representan, por ejemplo, la red de servicios de un ISP, dada la magnitud de estas redes y los protocolos y arquitecturas que conllevan. Otra limitante es el uso de protocolos propietarios ya que por estar definidos de ésta manera, no pueden ser usados por todos los fabricantes de equipos de comunicaciones, por ésta razón decidimos usar OSPFv3 y RIPng los cuales son de uso abierto.

CAPÍTULO 2

MARCO TEÓRICO

En este capítulo se desarrolla la parte conceptual de nuestro proyecto, se describen los fundamentos teóricos de las redes de datos, protocolos que pueden ser implementados en las mismas además de sistemas de seguridad y posibles formas de ejecutar un ataque de red.

2.1 MODELO OSI

El modelo OSI es un modelo de referencia de Interconexión de Sistemas Abiertos y se encuentra formado por 7 capas que son:

Aplicación: Proporciona servicios al usuario final, además de proveer comunicación entre dos procesos de aplicación, como pueden ser: programas de aplicación, aplicaciones de red, etc.

Presentación: Determina la forma de presentación de los datos sin preocuparse de su significado.

Sesión: Establece el inicio y termino de la sesión, además de establecer el orden en que los mensajes deben fluir entre usuarios finales.

Transporte: Actúa como un puente entre los tres niveles inferiores totalmente orientados a las comunicaciones y los tres niveles superiores totalmente orientados a el procesamiento. Además, garantiza una entrega confiable de la información.

Red: Define el enrutamiento y el envío de paquetes entre redes.

Enlace de datos: Proporciona facilidades para la transmisión de bloques de datos entre dos estaciones de red, además de detectar errores a nivel físico.

Físico: Define el medio de comunicación utilizado para la transferencia de información.

2.2 MODELO TCP/IP

El modelo TCP/IP es una familia de protocolos para la comunicación por una red de datos. Haciendo una comparación entre estos dos protocolos, el modelo OSI nos servirá como un modelo conceptual para la arquitectura de cualquier red informática mientras que el modelo TCP/IP servirá como una referencia de una arquitectura real. A diferencia del modelo OSI, este modelo presenta 5 capas que son:

Aplicación: Es el nivel más alto, en la cual cada aplicación interactúa con uno de los protocolos del nivel de transporte para enviar o recibir datos.

Transporte: Proporciona la comunicación entre un programa de aplicación y otro. Regula el flujo de información asegurando que los datos lleguen sin errores.

Internet: Maneja la comunicación de una máquina a otra. Maneja la entrada de datagramas y verifica su validez.

Acceso al medio: (También llamada **Interfaz de red**) Capa de nivel inferior, responsable de aceptar los datagramas IP y transmitirlos hacia una red específica.

2.3 INTERNET PROTOCOL (IP)

Es un protocolo de comunicación, clasificado dentro de la capa de red del modelo OSI. Su función principal es poder establecer la comunicación entre un origen y destino mediante un protocolo no orientado a conexión (es decir, no se requiere un acuerdo previo entre estos dos dispositivos). Permite la transferencia de paquetes conmutados a través de distintas redes físicas previamente enlazadas.

Los datos basados en una red IP se enviarán mediante bloques denominados paquetes o datagramas, y no provee ninguna fiabilidad en cuanto a si el paquete alcanza el destino o no, si necesita de esta fiabilidad se apoya en el protocolo TCP de la capa de transporte.

El protocolo IP intenta proporcionar seguridad mediante las “sumas de comprobación” o checksums de sus cabeceras, que contienen las direcciones de las máquinas de origen y destino, direcciones que posteriormente serán utilizadas por los enrutadores para decidir el tramo de la red por el cual reenviarán los paquetes.

2.4 QUÉ ES IPV4

Es la cuarta versión del protocolo IP, la cual utiliza direcciones de 32 bits limitándola a tan solo 2^32= 4 294 967 296 direcciones únicas, muchas

de éstas dedicadas a redes locales (LANs). Esta limitación se hizo más notoria al ver que escaseaban las direcciones IPv4 hace ya algunos años lo cual ayudó al impulso de IPv6. Ejemplo de una dirección IPv4: *192.168.1.1*

2.5 QUÉ ES IPV6

Sexta versión del protocolo IP, diseñado para reemplazar a su antecesor IPv4, que actualmente está implementado en la mayoría de dispositivos que acceden a internet.

Posibilita un mucho mayor rango de direcciones: 2^128 o 340 sextillones de direcciones, de tal manera que pudiera proporcionar direcciones propias y permanentes a una infinidad de dispositivos.

Algunos de los cambios más relevantes que podemos nombrar en cuanto a IPv6 con respecto a IPv4, son los siguientes:

- Capacidad extendida de direccionamiento
- Autoconfiguración de direcciones libres de estado
- Multicast (no utiliza broadcast)
- Seguridad de nivel de red, obligatoria (IPsec)
- Proceso simplificado en los enrutadores

Otro cambio que se puede apreciar entre este protocolo y su predecesor es la notación con la que se representan las direcciones. Mientras que en IPv4 la notación más utilizada es la decimal, en IPv6 la hexadecimal es la que más se utiliza. Ejemplo de una dirección IPv6: *2001:0DB8:02de::0e13*

2.6 PROTOCOLOS DE ENRUTAMIENTO

Los protocolos de enrutamiento son el conjunto de reglas que utiliza un enrutador cuando se comunica con otro, con el fin de compartir información de enrutamiento.

El enrutamiento por parte de estos dispositivos se refiere a buscar el mejor camino o ruta posible en una red de datos por la cual retransmitir la información, para esto se toman en cuenta parámetros como por ejemplo: la métrica de la red, la cual se calcula en base a diferentes parámetros de las rutas como por ejemplo el número de saltos para ir de un nodo a otro o el ancho de banda de los enlaces.

Clasificación de los métodos de enrutamiento

- Estáticos
- Dinámicos

Clasificación de los protocolos de enrutamiento dinámico

- Por vector distancia
- Por estado de enlace

2.6.1 OSPFV3

La versión 3 del protocolo OSPF (Open Shortest Path First o su traducción al español, “primero la ruta más corta y despejada”) fue diseñada para manejar redes configuradas con el protocolo IPv6. Está definido en la RFC 5340 y continúa siendo un protocolo de estado de enlace. Entre sus principales características tenemos:

- Está basado en OSPFv2 incluyendo mejorías.
- Distribuye prefijos IPv6.
- Se ejecuta directamente sobre IPv6.
- El método de autenticación ahora hace uso del protocolo IPsec.
- OSPFv3 se ejecuta sobre un enlace en vez de en una subred.

OSPFv3 usa las direcciones IPv6 de enlace local para identificar a los vecinos adyacentes de un determinado enrutador. Por tanto cuando se haga uso del comando “ipv6 ospf neighbor ipv6_address”, la dirección configurada debe ser la dirección de enlace local del vecino.

OSPFv3 ha mantenido de OSPFv2 las siguientes características:

- 5 tipos de paquetes (Hello, DBD, LSR, LSU, LSA)
- Implementación de mecanismos para descubrimiento de vecinos y formación de adyacencias.
- Cálculos de la mejor ruta se basan en el algoritmo SPF, también conocido como algoritmo de Dijkstra.
- Realiza un procedimiento para elegir al “enrutador designado” en topologías multiacceso.
- Soporta múltiples áreas (incluyendo NSSA) y múltiples topologías (NBMA, point-to-multipoint, point-to-point, y broadcast).
- El ID de enrutador, ID de área y el ID de enlace local continúa siendo una dirección de 32 bits.
- Los DR y BDR son identificados por su ID del enrutador y no por su dirección ip.
- Diferencias de OSPFv3 en relación a su versión 2:
- OSPFv3 es ejecutado sobre un enlace y configurado individualmente en una interfaz.
- Direcciones de enlace local con formato IPv6 son requeridas.
- Ahora se manejan 3 alcances para difusión masiva de mensajes tipo LSA: enlace local, área y sistema autónomo.

- Cada interfaz puede soportar múltiples instancias del protocolo OSPFv3.
- La sentencia “network” usada en el submodo de configuración del protocolo OSPFv2 es reemplazada por la sentencia de interfaz “ipv6 ospf process-id area area-id”.
- La dirección FF02::5 es el equivalente de la dirección 224.0.0.5 utilizada en la versión 2. Esta dirección IPv6 representa a todos los enrutadores con alcance de enlace local.
- La dirección FF02::6 es el equivalente de la dirección 224.0.0.6 utilizada en la versión 2. Esta dirección IPv6 representa a todos los enrutadores designados en alcance de enlace local.

OSPFv3 presenta un nuevo campo llamado “Instance ID” el cual hace posible configurar múltiples instancias OSPFv3 en una sola interfaz. Para que 2 instancias del proceso OSPFv3 establezcan comunicación, es necesario que tengan el mismo valor de “Instance ID”, el cual predeterminadamente es 0.

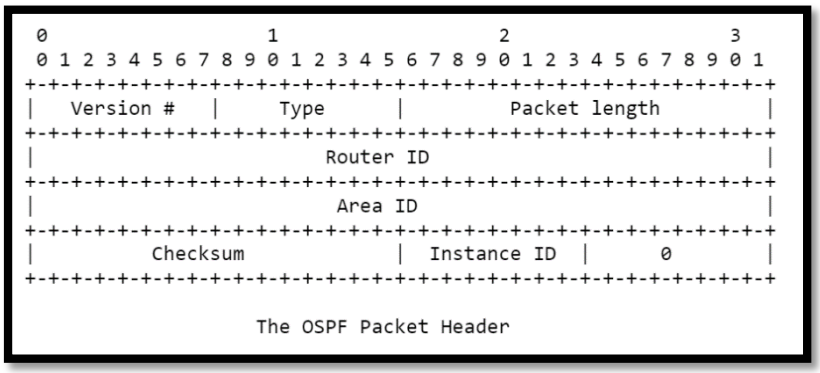


Figura 2.1 Formato del encabezado de un paquete OSPFv3 [1]

OSPFv3 hace uso de las extensiones de cabecera IPsec AH y ESP del protocolo IPv6 para el manejo de su seguridad. Dejando de lado el mecanismo de autenticación que se manejaba en la versión 2 pues ahora es trabajo de IPv6 y IPsec asegurar el correcto nivel de autenticación a usar.

Comandos de configuración del protocolo en plataformas CISCO:

Configurar el proceso de enrutamiento OSPF

R1(config)# **ipv6 router ospf** process-id

Configurar el router-id

R1(config-rtr)# **router-id** router-id

El proceso de selección del router-id es el mismo que para OSPFv2:

1. Se elige el router-id explícitamente configurado con el comando “router-id”.
2. En su defecto, se escogerá la dirección de loopback más alta que se encuentre en uso.
3. En su defecto, se escogerá la dirección IPv4 más alta y activa.
4. En su defecto, el router-id debe ser explícitamente configurado.

Habilitar una instancia OSPF en una interfaz

R1(config-if)#**ipv6 ospf** process-id **area** area-id [**instance** instance-id]

Process-id - Es un identificador local del proceso OSPF y puede ser cualquier entero positivo.

Area-id - Especifica el área que debe asociarse a la interfaz OSPF.

Instance-id (Opcional) - Controla la selección de otros enrutadores como “enrutadores vecinos”.

Especificar el costo de envío de un paquete en una interfaz

R1(config-if)#**ipv6 ospf cost** interface-cost

El rango del costo de interfaz es un número de 1 a 65535.

Cambiar la prioridad usada en la elección del enrutador designado

R1(config-if)#**ipv6 ospf priority** number-value

El rango de valor de prioridad es de 0 a 255 siendo 1 el valor predeterminado.

Definir un área como “stub” o “totally stub”

R1(config-rtr)#**area** area-id **stub** [**no-summary**]

El parámetro no-summary es configurado únicamente en el enrutador ABR e indica que el área es un área totally stub.

Verificación y solución de problemas

R1#**show ipv6 ospf neighbor**

R1#**show ipv6 ospf interface**

2.6.2 RIPNG

Definido en la RFC 2080, el protocolo de enrutamiento de información, próxima generación, por sus siglas en inglés (RIPng),

conserva ciertas características de su predecesor, el ya conocido protocolo RIPv2:

- Se mantiene como un protocolo de enrutamiento de tipo “Vector Distancia”, es decir está basado en el algoritmo Bellman-Ford.
- Fue diseñado como un protocolo “puerta adentro” y para redes de baja escalabilidad.
- El número máximo de saltos para alcanzar un destino sigue siendo 15.
- La distancia administrativa del protocolo continúa siendo 120.
- Continúa usando las técnicas de “horizonte dividido” y “envenenamiento reverso” para la prevención de lazos de enrutamiento.

A continuación hemos enlistado las diferencias con RIPv2:

- RIPng ha sido diseñado para enrutar redes y prefijos del protocolo IPv6.
- RIPng hace uso del puerto 521 UDP a diferencia del puerto 520 UDP usado por RIPv2.

- El grupo de direcciones multicast del protocolo es FF02::9 en lugar del grupo 224.0.0.9 usado en la versión para IPv4 del protocolo.

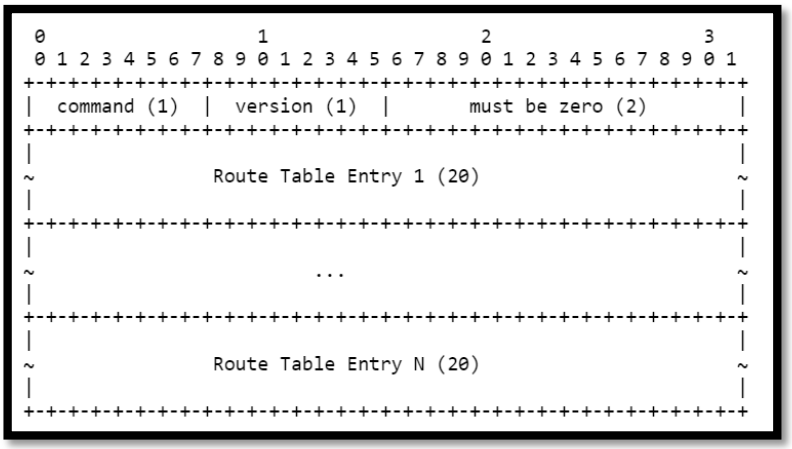


Figura 2.2 Formato del encabezado de un paquete RIPNG [2]

El algoritmo de horizonte dividido evita que un enrutador incluya rutas aprendidas desde un vecino, en actualizaciones que sean enviadas hacia ese mismo vecino. En el caso de una red broadcast, cualquier ruta aprendida desde cualquier vecino, no será incluida en actualizaciones enviadas hacia esa red.

La técnica de envenenamiento reverso sí incluye dichas rutas en las actualizaciones pero les establece una métrica infinita de tal modo que sean inalcanzables y no elegibles para la lógica del enrutador.

Ventajas de RIPng

- Requiere de una configuración simplificada comparado con otros protocolos.
- Su algoritmo de selección es más simple que el de otros protocolos, por lo que calcular la “mejor” ruta es más rápido en comparación con enrutadores de similares prestaciones.
- Tiene soporte en muchos fabricantes.

Desventajas de RIPng

- La principal desventaja es el hecho de tomar en cuenta un único criterio para la elección de la mejor ruta, el número de saltos, sin considerar otros importantes factores como el retardo o el ancho de banda del canal.
- El máximo número de saltos es 15, lo que limita su uso a redes de tamaño pequeño o mediano.

Comandos de configuración del protocolo en plataformas CISCO:

Habilitando RIP globalmente

R1(config)# **ipv6 router rip** name

Habilitando RIP en una interfaz

R1(config-if)# **ipv6 rip** name **enable**

Deshabilitar la técnica de horizonte dividido

R1(config-rtr)# **no split-horizon**

Verificación y solución de problemas

R1# **show ipv6 protocols**

R1(config)# **debug ipv6 rip** [interface-type interface-number]

2.7 SEGURIDAD EN LAS REDES INFORMÁTICAS

A continuación se describen los puntos más importantes a tener en cuenta cuando hablamos de seguridad informática en un proceso de comunicación entre dos entidades:

La autenticación de entidades.- Es el proceso de verificación de la identidad digital del remitente de una comunicación como una petición para conectarse. El remitente siendo autenticado puede ser una persona que usa un ordenador, un ordenador por sí mismo o un programa del ordenador.

La confidencialidad de los datos.- Es la propiedad de la información, por la que se garantiza que esté accesible únicamente a personal autorizado.

La integridad de los datos.- Es el estado de protección, corrección y completitud en la estructura de los datos en una aplicación.

El control de acceso.- Es el conjunto de mecanismos y protocolos, a través de los cuales varios dispositivos se ponen de acuerdo para compartir un medio de transmisión común.

El no repudio.- Puede referirse al servicio que proporciona pruebas en la integridad y origen de los datos, o a la autentificación que con un alto aseguramiento puede ser afirmado como genuino.

La disponibilidad.- Es un factor que nos indica cuanto tiempo la información se encuentra a disposición de quienes pueden acceder a ella.

El anonimato.- Es el estado anónimo de una entidad, es decir, que la identidad de dicha entidad es desconocida.

En el presente proyecto se tomará en cuenta tres de estos servicios de seguridad, que son la confidencialidad, integridad y disponibilidad de la información. Estos parámetros serán puestos a prueba para comparar la eficiencia de seguridad de los protocolos OSPFv3 y RIPng.

2.7.1 IPSEC

Es un conjunto de protocolos que proporcionan servicios de seguridad en la capa IP, permitiendo a un sistema seleccionar:

- Los protocolos de seguridad
- Determinar los algoritmos a utilizar para los servicios
- Implementar cualquier algoritmo criptográfico requerido para proporcionar los servicios solicitados

Los protocolos de IPSec actúan en la capa de red, la capa 3 del modelo OSI. Otros protocolos de seguridad para Internet cuyo uso es muy extendido son SSL, TLS y SSH, los cuales operan en las capas de transporte hacia arriba (capas OSI 4 a 7). Esto hace que IPSec sea más flexible ya que puede ser utilizado para proteger protocolos de la capa 4, incluyendo TCP y UDP, los protocolos de capa de transporte más usados.

IPv6 incluye explícitamente la posibilidad de utilizar el modelo de seguridad IPSec (Internet Protocol Security) que proporciona autenticidad, integridad y confidencialidad a las comunicaciones de extremo a extremo.

2.7.2 PASOS PARA LA IMPLEMENTACIÓN DE IPSEC

IPSec proporciona la estructura y el administrador elige los algoritmos utilizados para implementar los servicios de seguridad dentro de esa estructura. Existen cuatro apartados de estructura IPSec que deben completarse.

1. Al configurar un gateway de IPSec para proporcionar servicios de seguridad, primero **debe elegirse un protocolo IPsec**. Las opciones son ESP o ESP con AH.

2. El segundo apartado es un **algoritmo de cifrado** siempre que IPsec se implemente con ESP. Se debe seleccionar el algoritmo de cifrado adecuado para el nivel de seguridad deseado:

   - DES (cifra y descifra los datos del paquete)
   - 3DES (cifrado superior al DES de 56 bits)

- AES (cifrado más fuerte según la longitud de la clave utilizada.)

3. El tercer apartado es la autenticación. Seleccione un **algoritmo de autenticación** para proporcionar la integridad de los datos:

   - MD5 (autentica datos de paquetes con una clave secreta compartida de 128 bits) o
   - SHA (autentica datos de paquetes con una clave secreta compartida de 160 bits)

4. El último apartado es el **grupo de algoritmos Diffie-Hellman (DH).** Establece que los pares compartan la información de clave. Seleccione el grupo que desea utilizar: DH1 o DH2.

   - DH: permite que dos partes establezcan una clave secreta compartida mediante el cifrado y los algoritmos de hash, como DES y MD5, sobre un canal de comunicaciones no seguro.

2.7.3 PROTOCOLOS DE SEGURIDAD DE IPSEC

Los protocolos de seguridad IPSEC se pueden clasificar en dos: AH y ESP.

Cabecera de autenticación (AH)

- Integridad sin conexión,
- Autentificación del origen de datos, y
- Un servicio opcional de protección anti replay.

Proporciona un medio al receptor de los paquetes IP para autenticar el origen de los datos y para verificar que dichos datos no han sido alterados en tránsito. Sin embargo no proporciona ninguna garantía de confidencialidad, es decir, los datos transmitidos pueden ser vistos por terceros.

Tal como indica su nombre, AH es una cabecera de autenticación que se inserta entre la cabecera IP estándar (tanto IPv4 como IPv6) y los datos transportados, que pueden ser un mensaje TCP, UDP o ICMP, o incluso un datagrama IP completo.

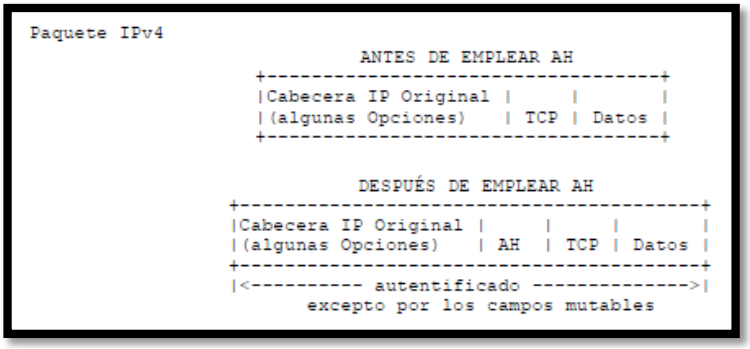


Figura 2.3 Paquete IPv4 antes y después de emplear AH [3]

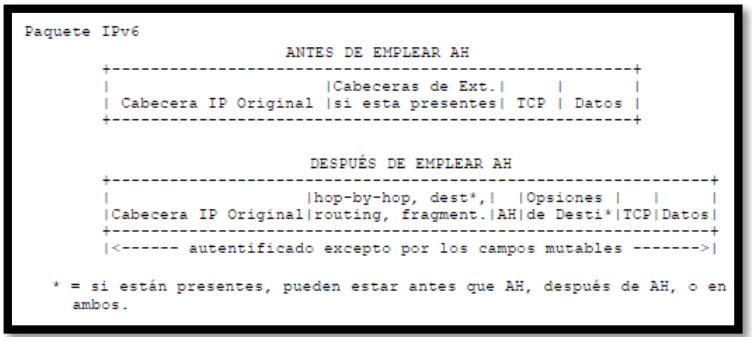


Figura 2.4 Paquete IPv6 antes y después de emplear AH [3]

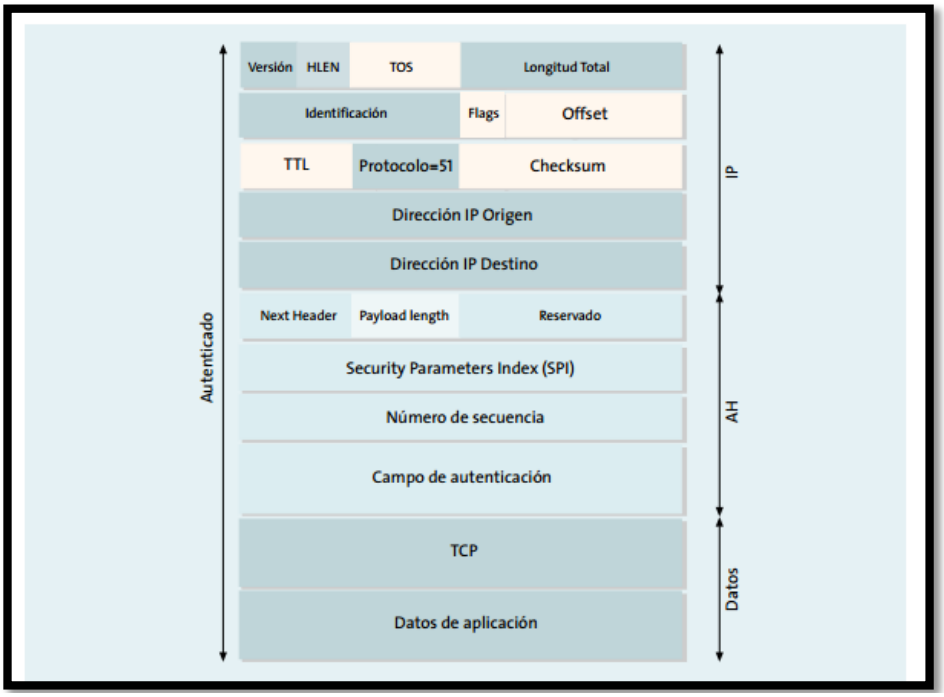


Figura 2.5 Estructura de un diagrama AH [4]

La figura 2.6 muestra el modo en que funciona el protocolo AH. El emisor calcula un extracto del mensaje original el cual se copia en uno de los campos de la cabecera AH. El paquete así construido se

envía a través de la red, repitiéndose en el extremo receptor el cálculo del extracto y comparándolo con el recibido en el paquete.

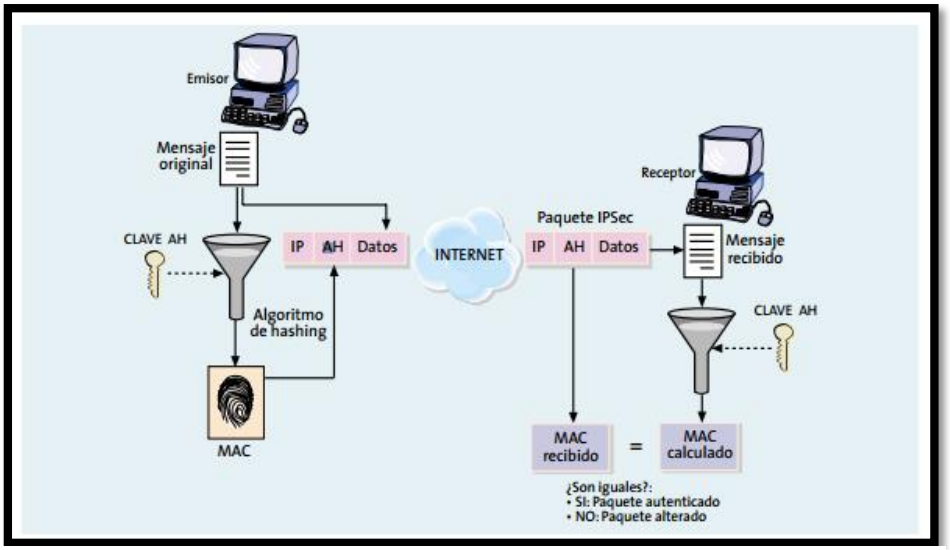


Figura 2.6 Funcionamiento del protocolo AH [4]

Si son iguales, el receptor tiene la seguridad de que el paquete IP no ha sido modificado en tránsito y que procede efectivamente del origen esperado.

Carga de seguridad encapsulada (ESP)

Puede proporcionar confidencialidad (**cifrado**) y confidencialidad limitada de flujo de tráfico. También puede proporcionar:

- Integridad sin conexión,
- Autentificación del origen de datos, y

- Un servicio de protección anti replay.

Dado que ESP proporciona más funciones que AH, el formato de la cabecera es más complejo: este formato consta de una cabecera y una cola que rodean los datos transportados. Dichos datos pueden ser cualquier protocolo IP (por ejemplo TCP, UDP, ICMP o incluso un paquete IP completo).

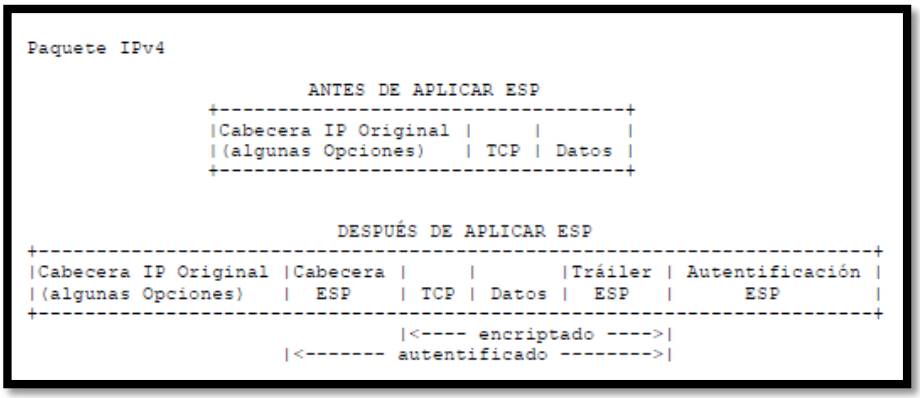


Figura 2.7 Paquete IPv4 antes y después de emplear ESP [3]

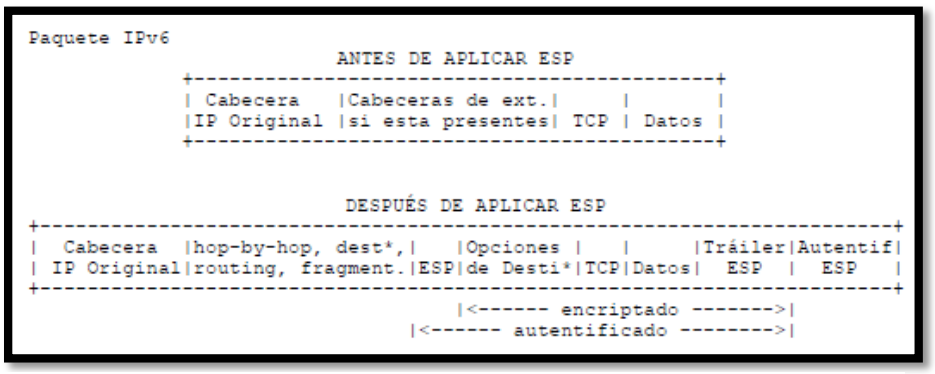


Figura 2.8 Paquete IPv6 antes y después de emplear ESP [3]

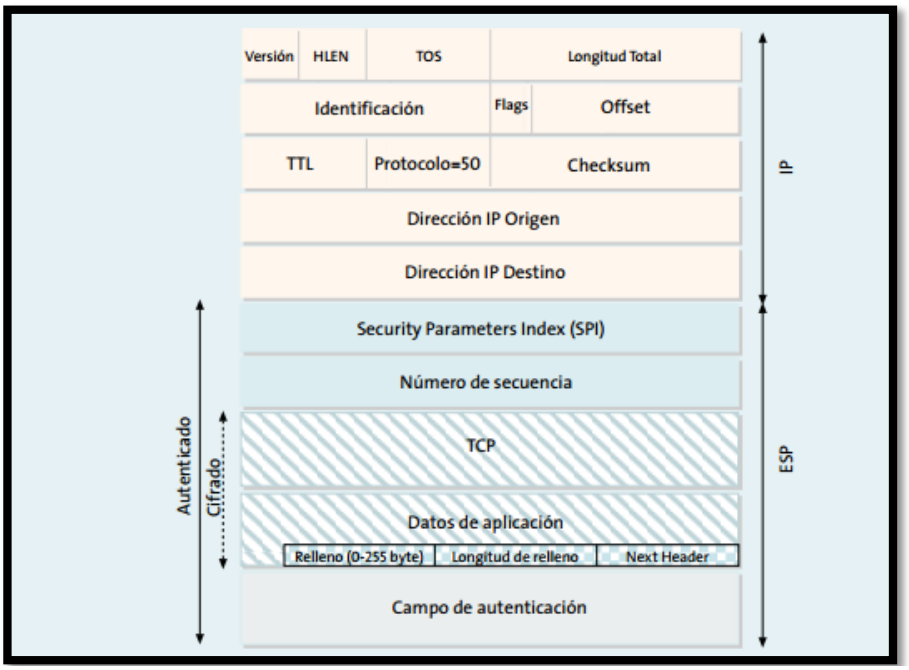


Figura 2.9 Estructura de un diagrama ESP [4]

En la figura 2.10 se muestra como el protocolo ESP permite enviar datos de forma confidencial. El emisor toma el mensaje original, lo cifra, utilizando una clave determinada, y lo incluye en un paquete IP, a continuación de la cabecera ESP. Durante el tránsito hasta su destino, si el paquete es interceptado por un tercero solo obtendrá un conjunto de bits ininteligibles.

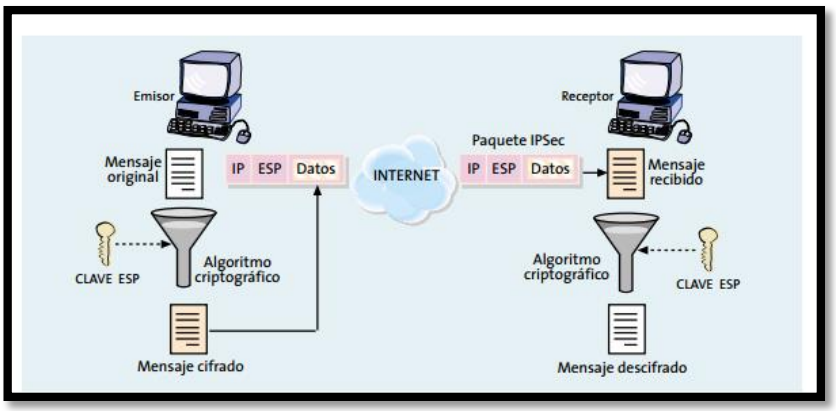


Figura 2.10 Funcionamiento del protocolo ESP [4]

En el destino, el receptor aplica de nuevo el algoritmo de cifrado con la misma clave, recuperando los datos originales.

Está claro que la seguridad de este protocolo reside en la robustez del algoritmo de cifrado, es decir, que un atacante no puede descifrar los datos sin conocer la clave. Así también en que la clave ESP únicamente la conocen el emisor y el receptor.

2.7.4 MODOS DE USO

Estos protocolos pueden aplicarse solos o en conjunto con otros para proporcionar un conjunto de servicios de seguridad en IPV4 e IPV6. Cada protocolo soporta dos modos de uso:

Modo transporte

Los protocolos proporcionan protección sobre todo a los protocolos de capa superior.

Es extremo a extremo, es decir los ordenadores de los extremos finales realizan el procesado de seguridad.

En este modo el contenido transportado dentro del datagrama AH o ESP son datos de la capa de transporte (por ejemplo, datos TCP o UDP). Por tanto, la cabecera IPsec se inserta inmediatamente a continuación de la cabecera IP y antes de los datos de los niveles superiores que se desean proteger.

El modo transporte tiene la ventaja de que asegura la comunicación, pero requiere de que ambos extremos entiendan el protocolo IPsec.

Modo túnel

Los protocolos son aplicados a paquetes (a los que se les hizo un túnel a través de IP).

Es puerta a puerta, en el que la seguridad del tráfico de paquetes es proporcionada a varias máquinas (incluso a toda la red local) por un único nodo.

En este el contenido del datagrama AH o ESP es un datagrama IP completo, incluida la cabecera IP original. Así se toma un datagrama IP al cual se añade inicialmente una cabecera AH o ESP, posteriormente se añade una nueva cabecera IP que es la que se utiliza para encaminar los paquetes a través de la red.

El modo túnel se usa normalmente cuando el destino final de los datos no coincide con el dispositivo que realiza las funciones IPsec.

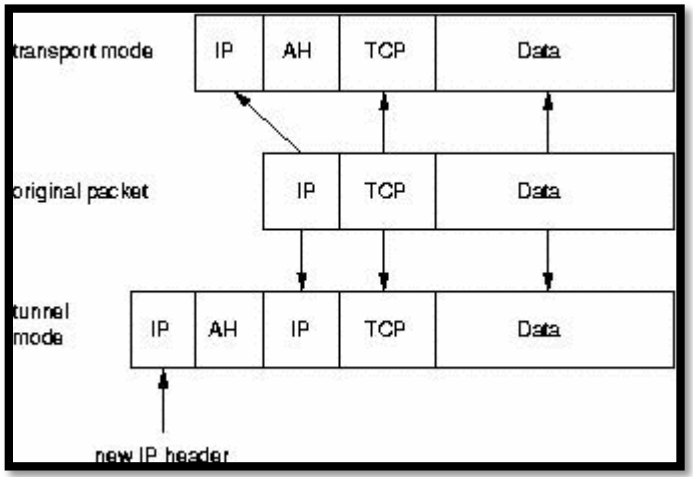


Figura 2.11 Comparación entre el modo transporte y el modo túnel

El modo túnel es útil cuando se utiliza junto a ESP para ocultar la identidad de los nodos que se están comunicando, otra aplicación del modo túnel, tanto con ESP como con AH, es poder establecer VPN's a través de redes públicas.

En la primera parte de la figura 2.12 se presentan dos hosts que entienden IPsec y que se comunican de forma segura. Esta comunicación se realiza en modo transporte, por tanto la información que se protege es únicamente el protocolo TCP o UDP así como los datos de la aplicación.

En la segunda parte de la figura 2.12 se muestran dos redes que utilizan dos gateways IPsec para conectarse, y por tanto emplean una implementación modo túnel. Se puede ver que la comunicación se realiza a través de una red de datos pública, entre un PC situado en una red local con otro PC situado en una red local remota, de modo que entre los Gateway IPSec se establece un túnel a través del cual viajan protegidas las comunicaciones entre ambas redes locales.

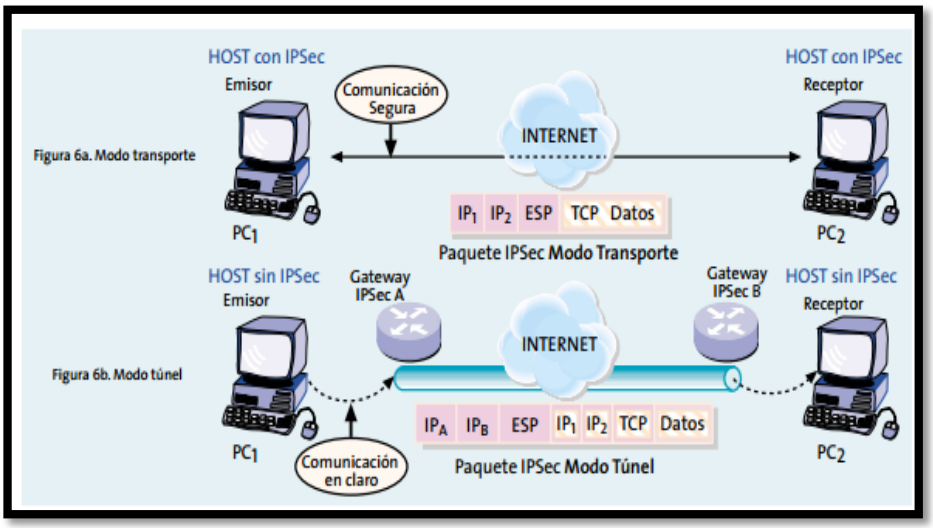


Figura 2.12 Los modos de funcionamiento transporte y túnel de IPsec [4]

2.8 ATAQUES A LOS SERVICIOS DE SEGURIDAD

Existen múltiples ataques a los servicios de seguridad los cuales deben ser minimizados, a continuación ejemplos clásicos de ataques a redes de datos:

- Ataques sobre la identidad de las entidades (interceptación y suplantación)
- Ataques sobre la información (revelación, reenvío, manipulación y repudio de datos)
- Ataques sobre los servicios (negación del servicio y apropiación).

Tabla 1 Servicios vs. Ataques

	Ataques pasivos		Ataques activos			
	Obtención del Contenido	Análisis de Tráfico	Suplantación	Repetición	Modificación	Interrupción
Autenticación			✓			
Control de Acceso			✓			
Confidencialidad	✓	✓				
Integridad				✓	✓	
No Repudio						
Disponibilidad						✓

2.8.1 DOS ATTACK

Es un ataque a una red de computadores donde el objetivo es volver inaccesible un servicio o recurso de red a sus usuarios legítimos. Una forma de llevar a cabo éste ataque es saturar el ancho de banda de una red o recurso de tal modo que se provoque la pérdida de conectividad de dicho recurso. Una ampliación de este ataque es el DDoS, el cual consiste en generar un gran flujo de información desde diferentes puntos de conexión de una red, la forma más común de llevar a cabo estos ataques es una botnet.

El ancho de banda de un servicio no es lo único que se puede saturar, existen otros recursos como el tiempo de procesamiento

del CPU o el espacio de disco de un servidor los cuales pueden ser llevados al límite causando irregularidades en el servicio que prestan.

También se podría alterar la información de las tablas de enrutamiento con el fin de volver inalcanzable un destino prestador de algún servicio crítico.

Se podría también obstruir los medios físicos de comunicación entre los clientes y los servidores con objeto de cortar la comunicación.

Inundación SYN

Cuando un cliente desea establecer conexión con un servidor, envía un paquete TCP/SYN, si el servidor acepta la solicitud responderá con un TCP/SYN-ACK y finalmente el cliente responderá con un TCP/ACK (Establecimiento de una conexión TCP de 3 vías).

Dada la naturaleza de este proceso, un atacante podría enviar múltiples paquetes TCP/SYN, generalmente con la dirección ip origen falsificada, provocando que el servidor intente iniciar una

comunicación respondiendo con un paquete TCP/SYN-ACK a cada solicitud recibida y dado que las direcciones ip origen no existen o bien nunca solicitaron la conexión, la respuesta TCP/ACK jamás será recibida por el servidor.

Estos intentos de iniciar una comunicación consumen recursos del servidor y alcanzan el límite de conexiones que se puede establecer disminuyendo así la capacidad del servidor para responder solicitudes de conexión legítimas.

Inundación ICMP

Esta técnica tiene como objetivo saturar el ancho de banda del sistema víctima enviando de forma continua una gran cantidad de paquetes de tipo ICMP, echo request, los cuales serán respondidos con una gran cantidad de paquetes echo reply.

Si el atacante cuenta con un ancho de banda mayor que la víctima, podría fácilmente generar una cantidad de tráfico mayor de lo que la víctima puede manejar.

Una variante del ataque ICMP flood es el ataque SMURF, el cual amplifica considerablemente los efectos del ataque ICMP flood. En

éste ataque, se envían paquetes ICMP, echo request a una dirección de broadcast usando como origen la dirección ip de la víctima. De éste modo todos los equipos conectados en la red responderán con paquetes echo reply al sistema víctima pudiendo llegar a saturar su ancho de banda. Éste tipo de ataque puede llegar a afectar también a los sistemas intermediarios, es decir a los sistemas que responderán con un echo reply al sistema víctima**.**

2.8.2 FALSEO DE RUTAS

Este tipo de ataque consiste en el envío de falsas actualizaciones de enrutamiento de tal modo que la tabla de encaminamiento de los enrutadores se vea modificada, dirigiendo el tráfico hacia un destino erróneo, por lo general este destino es el atacante quien desea ésta información en su poder con fines maliciosos.

Existen varias herramientas que nos permiten llevar a cabo éste ataque, una de ellas es el software Quagga.

Quagga es un suite de software libre para poder usar la familia de sistemas operativos Unix como enrutadores. Actúa como conmutador del GNU Zebra, el cual a su vez es un demonio que se encarga de manejar las tablas de enrutamiento del núcleo (un

demonio o DAEMON es un tipo especial de proceso informático no interactivo, es decir, que se ejecuta en segundo plano en vez de ser controlado directamente por el usuario).

Quagga provee implementaciones para los siguientes protocolos:

- RIPv1/RIPv2 para IPv4 y RIPng para IPv6
- OSPFv2 y OSPFv3
- BGPv4+
- IS-IS para IPv4 e IPv6
- BABEL, enrutamiento de malla inalámbrica (IPv4 e IPv6)
- Protocolo de distribución de etiquetas MPLS

Un atacante astuto podría determinar cuál es el segmento de red donde se aloja un servidor en particular haciendo uso de algún sniffer o alguna otra herramienta de detección de hosts en una red.

Una vez determinado este segmento de red, se puede hacer uso de Quagga para que envíe actualizaciones con una métrica menor hacia el segmento de red que queremos suplantar, de este modo se modificarían las tablas de encaminamiento de los dispositivos de

capa 3 haciendo que éstos envíen el tráfico hacia el segmento de red del atacante y su dirección ip en particular.

CAPÍTULO 3

DISEÑO E IMPLEMENTACIÓN DE LOS ESCENARIOS DE PRUEBAS

Para realizar la comparación entre dos protocolos de enrutamiento en un ambiente IPv6, fue necesario realizar la debida implementación, la cual consistió en armar virtualmente dos redes, una por cada protocolo. Detallamos tanto el hardware como el software que fueron usados en la implementación virtual, anotamos también la descripción de los parámetros que evaluamos, las configuraciones de los dispositivos de red y las pruebas que se realizaron a cada protocolo.

3.1 CONSIDERACIONES TÉCNICAS

Para la implementación de nuestra red instalamos varias máquinas virtuales haciendo uso de la herramienta Virtualbox que a su vez puede integrarse con GNS3, herramienta de virtualización de enrutadores. En conjunto ambas herramientas nos permitieron efectuar los ataques de red sobre los protocolos OSPFv3 y RIPng y evaluar la respuesta de ambos. Cabe mencionar que las pruebas realizadas en nuestro entorno virtual son 100% reproducibles en un entorno real usando enrutadores y ordenadores personales.

3.1.1 HARDWARE DEL DISPOSITIVO VIRTUALIZADOR

Tabla 2 Especificaciones del dispositivo virtualizador

Características	Descripción
CPU	Intel Core i5-3317U
Disco Duro	500 GB
Memoria RAM	12,00 GB
Sistema Operativo	Windows 7 (64 bits)
Velocidad de Reloj	1.70GHz

3.1.2 SOFTWARE DE LOS DISPOSITIVOS DE RED VIRTUALES

A continuación se detallan los recursos físicos asignados a cada máquina virtual y el software instalado en cada una de ellas.

Tabla 3 Especificaciones del cliente y servidor

Características	Descripción
Espacio de disco asignado	15 GB
Memoria RAM asignada	1.5 GB
Sistema Operativo	Windows 7 (32 bits)

Software del servidor

WAMP Server versión 5.6 (Ver Anexo A para una breve descripción de WAMP). En la carpeta C: \wamp\www colocamos los archivos contacto.html y contacto.php (Ver Anexo B para el detalle de estos archivos.)

Tabla 4 Especificaciones del atacante

Características	Descripción
Espacio de disco asignado	7.3 GB
Memoria RAM asignada	2.5 GB
Sistema Operativo	Ubuntu 14 (32 bits)

Software del atacante

LAMP (Linux, Apache, Mysql y PHP)

Se creó la misma base de datos que tenemos en el servidor Windows con el objeto de suplantarlo y engañar al cliente:

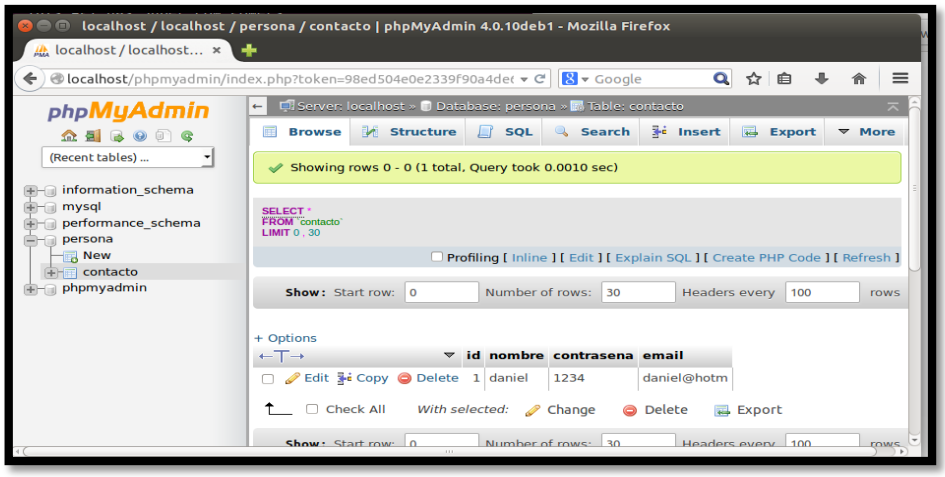


Figura 3.1 Tabla contacto en la base de datos del atacante

También se instaló Quagga para poder enviar actualizaciones de enrutamiento hacia los enrutadores originales y de este modo modificar sus tablas de enrutamiento, así pudimos recibir información que no estaba destinada a nosotros inicialmente. Ver Anexo C (Instalación del software Quagga)

Tabla 5 Especificaciones de los enrutadores

Características	Descripción
Modelo virtualizado	C3725
Memoria RAM asignada	192 MB
Memoria NVRAM asignada	256 KB
IOS	c3725-adventerprisek9-mz.124-15.T14
Adaptadores virtuales	2 puertos fast-ethernet (GT96100-FE) 2 puertos WIC (WAN-Interface-Card) (WIC-2T)

3.2 DESCRIPCIÓN DE LOS PARÁMETROS A EVALUAR

Para la prueba de disponibilidad, el factor cuantificable fue el tiempo. Medimos el tiempo que tomó transferir un archivo desde el servidor hacia el cliente en condiciones normales y bajo los efectos de algún ataque de red. Adicionalmente se midió el tiempo de convergencia de ambos protocolos.

Para la prueba de confidencialidad, el parámetro a cuantificar fue la cantidad de información que el atacante pudo obtener sin ser el destinatario legítimo de un proceso de comunicación.

Para poder cuantificar la prueba de integridad, se midió el tiempo que le tomó al atacante pasar de engañar al cliente a engañar al servidor pues éste proceso fue necesario para poder enviar información alterada hacia el servidor desde el atacante como si fuese el cliente legítimo.

3.3 PRUEBAS PARA OSPFV3

La prueba de disponibilidad consistió en medir el tiempo que le tomó a un archivo estándar de 1MB transmitirse desde el servidor hacia el cliente haciendo uso del protocolo FTP, adicionalmente se calculó la tasa de transferencia de datos.

En el servidor hicimos uso de FTP montado en IIS. En el atacante usamos la herramienta Flood_router6 que se encuentra dentro de la Suite de herramientas thc-ipv6. (Ver Anexo D para una breve descripción de las herramientas de la suite thc-ipv6)

Primero medimos la respuesta del protocolo en condiciones normales, es decir sin ningún ataque de red ejecutándose.

Luego medimos la respuesta del protocolo bajo el efecto del ataque router-advertisements flood el cual se llevó a cabo con la herramienta antes mencionada.

Como última prueba de disponibilidad medimos el tiempo de convergencia de la VPN luego de recuperarse de una baja de sus interfaces.

El número de muestras fue de 47 (este número fue respaldado de acuerdo a lo explicado en el Anexo E). En Anexo J se encuentran las 47 muestras de cada prueba realizada.

Para las pruebas de confidencialidad e integridad hicimos uso del servicio web proporcionado por APACHE, que se encuentra dentro de WAMP en la máquina del servidor.

En el atacante usamos Quagga para habilitar el enrutamiento OSPFv3. Configuramos Quagga de tal modo que se enviaron actualizaciones de enrutamiento hacia la puerta de enlace del atacante indicándole que posee una mejor ruta (menor métrica) hacia la red del servidor. De éste modo el atacante pudo recibir información transmitida desde el cliente cuyo destino legítimo era el servidor violando así la confidencialidad de la información.

Luego realizamos los mismos pasos pero ésta vez teniendo habilitado un túnel VPN-IPSec entre la puerta de enlace del servidor y la del cliente. (Ver Anexo F para la configuración de los archivos de Quagga)

Una vez obtenida la información del cliente (SIN VPN SOLAMENTE) configuramos Quagga para que le indique a la puerta de enlace que tiene una mejor ruta hacia la red del cliente y así poder enviar información alterada desde el atacante hacia el servidor como si fuese el cliente legítimo, de éste modo se llevó a cabo la prueba de integridad de la información. Para poder cuantificar ésta prueba, se midió el tiempo que

nos lleva pasar de engañar al cliente a engañar al servidor pues debe ser un tiempo muy corto para que el usuario no perciba la transición y se dé cuenta que “algo no anda bien”.

En primera instancia, se consideró la configuración manual de los archivos de Quagga así como el cambio de IP dentro del host atacante para poder pasar de engañar al cliente a engañar al servidor pero esto demandaba de varias acciones que en conjunto requerían de 2 a 4 minutos aproximadamente lo que resultó ineficiente para un ataque de este tipo. Para solucionar este inconveniente, se automatizó este proceso haciendo uso de scripts y así poder efectuar el ataque en un menor tiempo (aproximadamente 40 segundos).

Para una mayor precisión en cuanto a la medición de este tiempo, se tomó en cuenta los mensajes debug de los enrutadores.

Ver Anexo G para el detalle técnico de cada prueba realizada.

3.3.1 CONFIGURACIONES DE LOS DISPOSITIVOS DE RED

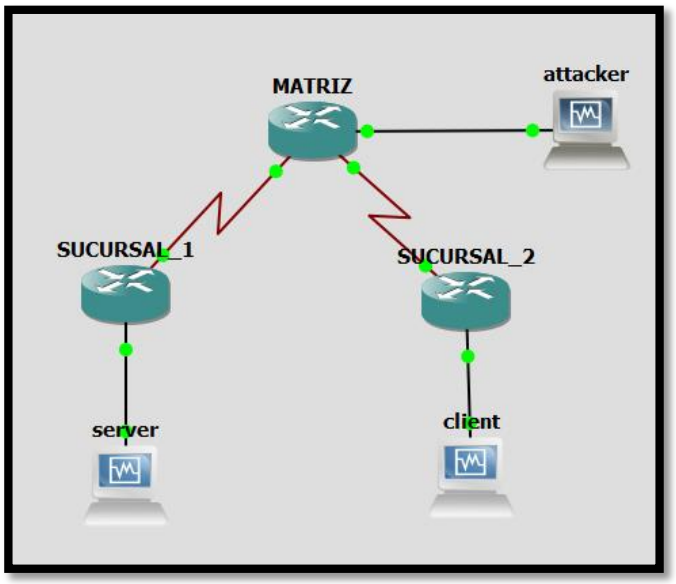


Figura 3.2 Esquema de red utilizado con el protocolo OSPFv3

Tabla 6 Direcciones de red para los dispositivos de red

SUCURSAL_1	
S0/0	2001:db8:100: :1 /126
S0/1	2001:db8:100: :9 /126
F0/0	2001:db8:100:1: :1 /64
MATRIZ	
S0/0	2001:db8:100: :5 /126
S0/1	2001:db8:100: :2 /126
F0/0 (al atacante)	2001:db8:100:4: :1 /64
SUCURSAL_2	
S0/0	2001:db8:100: :6 /126
S0/1	2001:db8:100: :A /126
F0/0	2001:db8:100:3: :1 /64
Servidor	
Eth1	2001:db8:100:1: :2 /64
Cliente	
Eth1	2001:db8:100:3 : :2 /64
Atacante	
Eth1	2001:db8:100:4: :2 /64
Eth2	2001:db8:100:1: :2 /64

En el Anexo H se encuentran los comandos de configuración de cada enrutador y en el Anexo I la implementación de la VPN para la prueba de confidencialidad.

3.3.2 PRUEBA DE DISPONIBILIDAD

Condiciones normales (sin ataque)

Tabla 7 Tiempo de transmisión – condiciones normales –OSPF

VARIABLES ESTADISTICAS	DESCRIPCION
Media	7,925 s
Desviación estándar	0,179
Varianza	0,0323
Mínimo	7,675 s
Máximo	8,412 s

Tabla 8 Velocidad de transmisión – condiciones normales – OSPF

VARIABLES ESTADISTICAS	DESCRIPCION
Media	123,20 Kb/s
Desviación estándar	2,75
Varianza	7,59
Mínimo	116,02 Kb/s
Máximo	127,17 Kb/s

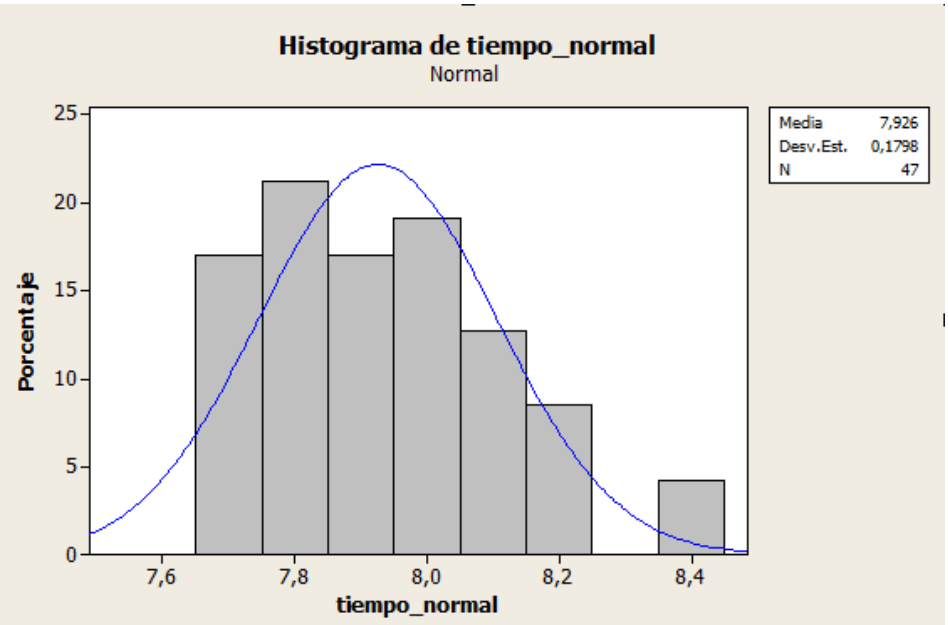


Figura 3.3 Histograma de tiempo de transmisión – condiciones normales- OSPFv3

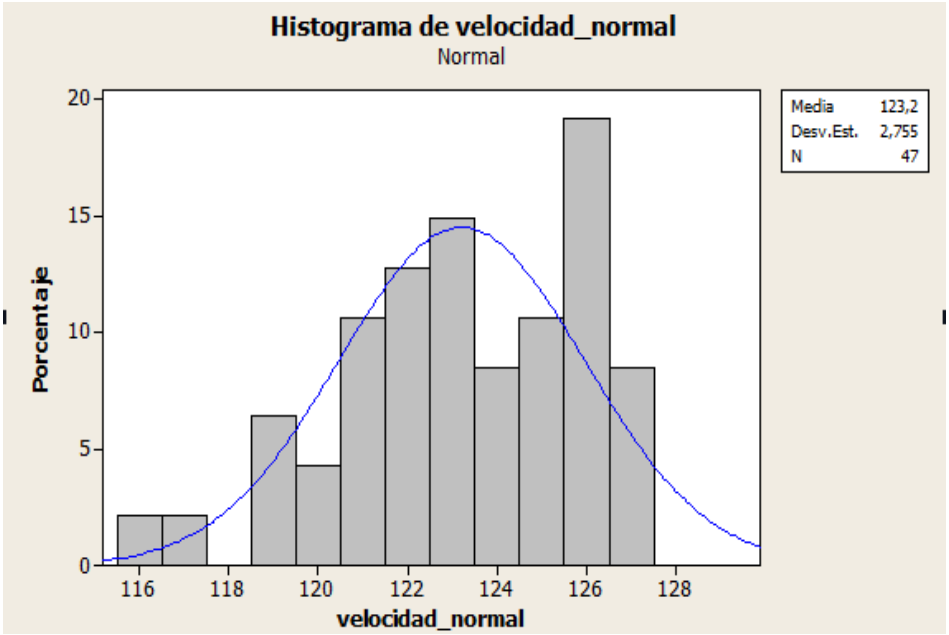


Figura 3.4 Histograma de velocidad de transmisión - condiciones normales- OSPFv3

Como podemos observar en el histograma de tiempo existe un dato aberrante de 8,4 segundos; el tiempo de transmisión se mantuvo entre los 7,7 segundos y los 8,2 segundos. En el histograma de velocidad de transmisión se puede notar que siempre se mantuvo entre los 119 y los 127 Kb/seg.

Bajo ataque flood

El segundo escenario en las pruebas de disponibilidad fue un ambiente de ataque DOS, generado por la herramienta flood_router6 de la suite de herramientas thc-ipv6, el cual simula un ataque DOS parcial. Se obtuvieron los siguientes resultados:

Tabla 9 Tiempo de transmisión – bajo ataque DOS – OSPF

VARIABLES ESTADISTICAS	DESCRIPCION
Media	107,28 s
Desviación estándar	35,780
Varianza	1280,220
Mínimo	44,072 s
Máximo	177,41 s

Como se puede apreciar, la media de los resultados fue de 107,28 segundos, es decir 13,5 veces más tiempo del que tomo al hacerlo en un ambiente normal.

Tabla 10 Velocidad de transmisión – bajo ataque DOS – OSPF

VARIABLES ESTADISTICAS	DESCRIPCION
Media	10,251 Kb/s
Desviación estándar	3,733
Varianza	13,936
Mínimo	5,501 Kb/s
Máximo	22,146 Kb/s

Así también la media en velocidad de transmisión fue de 10,251 Kb/s es decir la velocidad es 12 veces menor a la velocidad en condiciones normales.

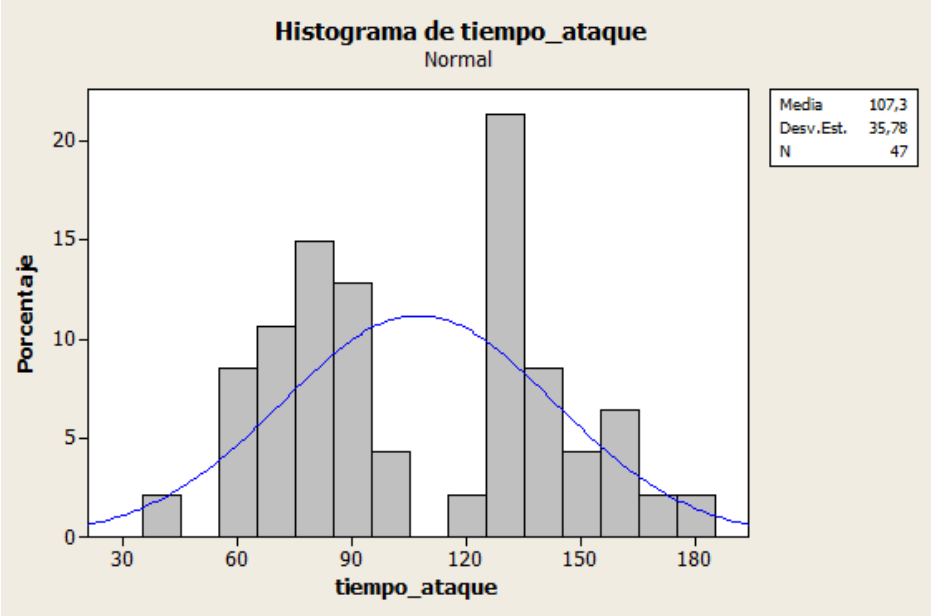


Figura 3.5 Histograma de tiempo de transmisión – bajo ataque DOS – OSPFv3

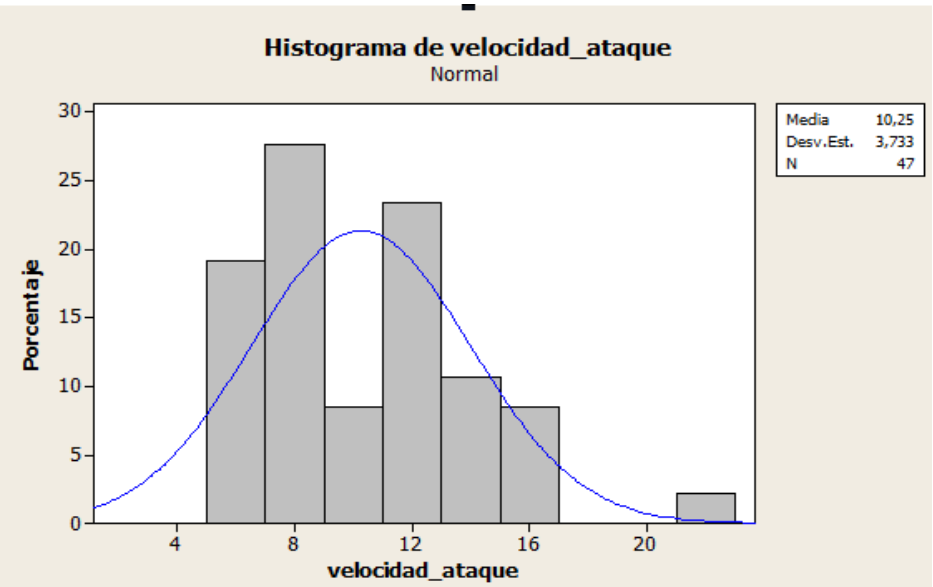


Figura 3.6 Histograma de velocidad de transmisión – bajo ataque DOS

Se puede apreciar que el rango de tiempo de trasmisión se mantuvo entre 60 y 180 segundos, y el rango en la velocidad de transmisión entre 6 y 16 Kb/s, teniendo un dato aberrante de 22 Kb/s.

Convergencia al subir VPN

Como tercer escenario de disponibilidad consideramos un ambiente de convergencia de la red, al bajar y subir la interfaz túnel entre el enrutador 1 y el enrutador 3. Medimos el tiempo que le tomó a la VPN volver a estar operativa y hábil para la transmisión de paquetes. Lo que se hizo fue bajar y subir la interfaz serial de R1.

Los datos tomados fueron los siguientes:

Tabla 11 Tiempo de convergencia –VPN – OSPF

VARIABLES ESTADISTICAS	DESCRIPCION
Media	24,865 s
Desviación estándar	6,546
Varianza	42,851
Mínimo	14,480 s
Máximo	44,200 s

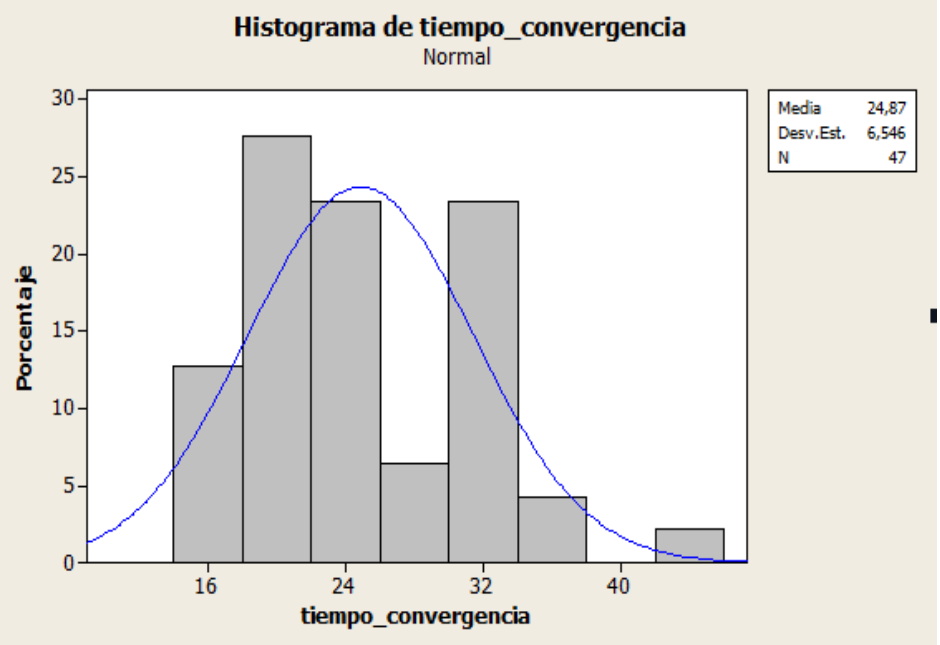


Figura 3.7 Histograma de velocidad de convergencia – VPN – OSPFv3

Podemos observar un tiempo de 24,86 segundos como media de convergencia y un rango aproximadamente de 16 a 36 segundos, teniendo como un dato aberrante 44 segundos.

3.3.3 PRUEBA DE CONFIDENCIALIDAD

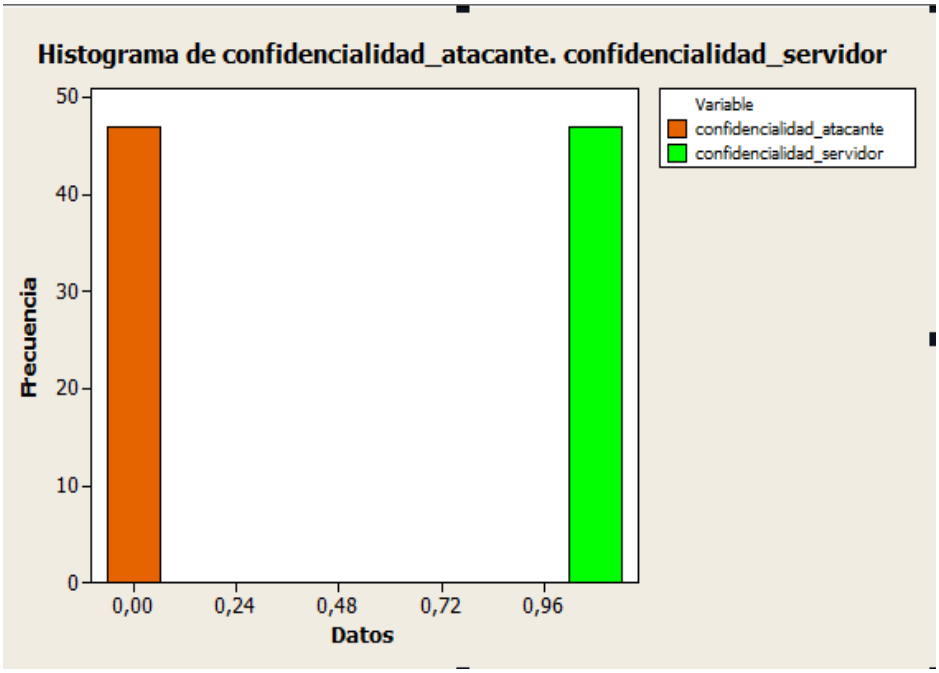


Figura 3.8 Histograma de confidencialidad– sin ataque – OSPFv3

El histograma de la figura 4-11 refleja un escenario de seguridad con VPN, a pesar de que el atacante ha alterado la tabla de enrutamiento del router MATRIZ modificando la ruta hacia la red del servidor, no podrá recibir información alguna por parte del cliente debido a que en nuestro caso existe una ruta estática a través de la VPN. Se observa que todo el tráfico fue dirigido desde el cliente hasta el servidor (100%), mientras que el atacante capturó el 0% del tráfico enviado.

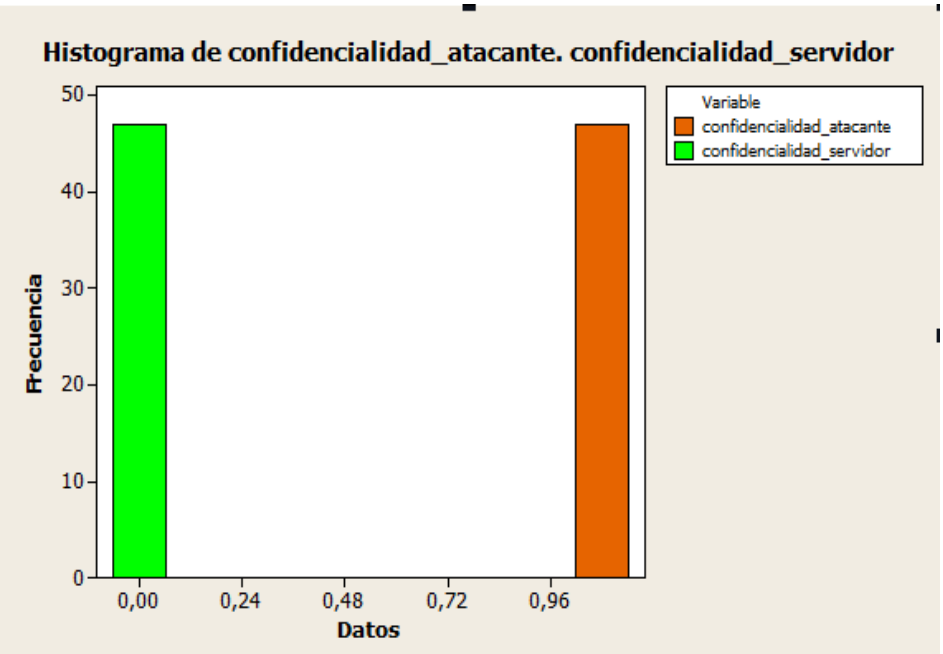


Figura 3.9 Histograma de confidencialidad– con ataque– OSPFv3

El segundo caso hace referencia al escenario en que la red se encuentra sin algún tipo de seguridad que pueda resguardarla (como la VPN), ya sea porque se encuentra desactivada o mal configurada. El atacante ejecuta la aplicación Quagga e inserta una nueva ruta suplantando al servidor, de esta manera se redirige todo el tráfico hacia el atacante (como se aprecia en el histograma) obteniendo el 100% del tráfico, mientras que, el servidor quedaría “aislado”, desconectado del cliente y recibiendo el 0% de información por parte del mismo.

Debemos enfatizar que en ambos casos los protagonistas de los escenarios son tanto el servidor como el atacante, mientras que el cliente muy probablemente ignora el ataque, teniendo siempre acceso a la información, ya sea la verídica o la suplantada.

3.3.4 PRUEBA DE INTEGRIDAD

Tiempo de convergencia al cambiar ruta

A continuación se muestran los resultados del tiempo (en tabla e histograma) que tomó hacer el cambio de ruta de la red del servidor a la red del cliente sobre el protocolo OSPF enviando actualizaciones falsas de enrutamiento mediante el uso de Quagga. Se suplanta tanto al servidor como al cliente:

Tabla 12 Tiempo de ataque a integridad de información – OSPF v3

VARIABLES ESTADISTICAS	DESCRIPCION
Media	10,042 s
Desviación estándar	0,0408
Varianza	0,001
Mínimo	10 s
Máximo	10,187 s

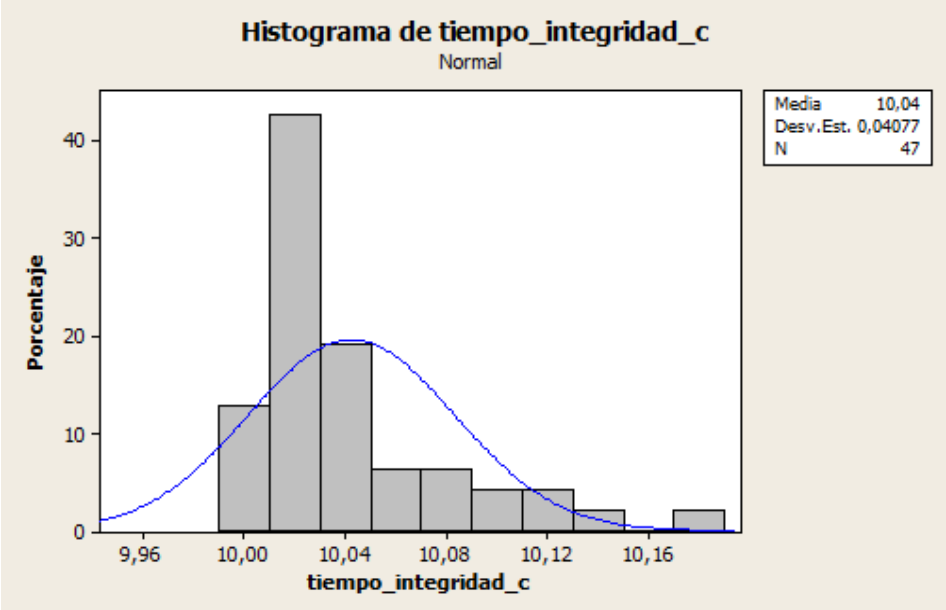


Figura 3.10 Histograma de tiempo – integridad – OSPFv3

Como se puede observar se obtuvo un tiempo de 10 segundos como media, con varianza casi nula, es decir se mantuvo uniforme en las pruebas que hicimos. Este tiempo es razonable debido a que las pruebas se estuvieron realizando en una red con pocos dispositivos de enrutamiento.

3.4 PRUEBAS PARA RIPNG

Las pruebas realizadas al protocolo RIPng son exactamente las mismas pruebas que fueron descritas para el protocolo OSPFv3.

3.4.1 CONFIGURACIONES DE LOS DISPOSITIVOS DE RED

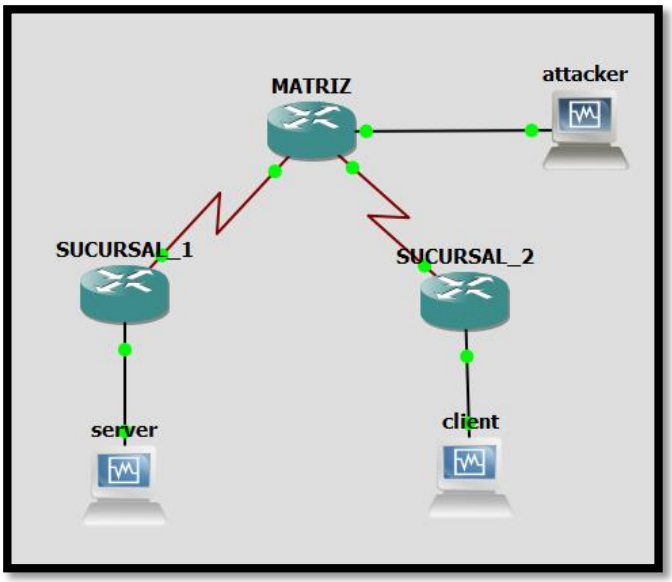


Figura 3.11 Esquema de red utilizado con el protocolo RIPng

Tabla 13 Direcciones de red para los dispositivos de red

SUCURSAL_1	
S0/0	2001:db8:100: :1 /126
S0/1	2001:db8:100: :9 /126
F0/0	2001:db8:100:1: :1 /64
MATRIZ	
S0/0	2001:db8:100: :5 /126
S0/1	2001:db8:100: :2 /126
F0/0 (al atacante)	2001:db8:100:4: :1 /64
SUCURSAL_2	
S0/0	2001:db8:100: :6 /126
S0/1	2001:db8:100: :A /126
F0/0	2001:db8:100:3: :1 /64
Servidor	
Eth1	2001:db8:100:1: :2 /64
Cliente	
Eth1	2001:db8:100:3 : :2 /64
Atacante	
Eth1	2001:db8:100:4: :2 /64
Eth2	2001:db8:100:1: :2 /64

En el Anexo H se encuentran los comandos de configuración de cada enrutador y en el Anexo I la implementación de la VPN para las pruebas de seguridad.

3.4.2 PRUEBA DE DISPONIBILIDAD

Condiciones normales (sin ataque)

Tabla 14 Tiempo de transmisión – condiciones normales - RIPNG

VARIABLES ESTADISTICAS	DESCRIPCION
Media	8,025 s
Desviación estándar	0,291
Varianza	0,0847
Mínimo	7,608 s
Máximo	9,205 s

Tabla 15 Velocidad de transmisión – condiciones normales – RIPNG

VARIABLES ESTADISTICAS	DESCRIPCION
Media	121,762 Kb/s
Desviación estándar	4,22
Varianza	17,81
Mínimo	106,03 Kb/s
Máximo	128,29 Kb/s

Como se puede apreciar en la tabla VIII, la media en tiempo de transmisión fue muy parecida a la del tiempo en OSPF, así también la velocidad de transmisión fue muy cercana a la velocidad de transmisión que se presentó en OSPF en tiempos normales.

A continuación se presentan los histogramas en base a los datos medidos en las pruebas:

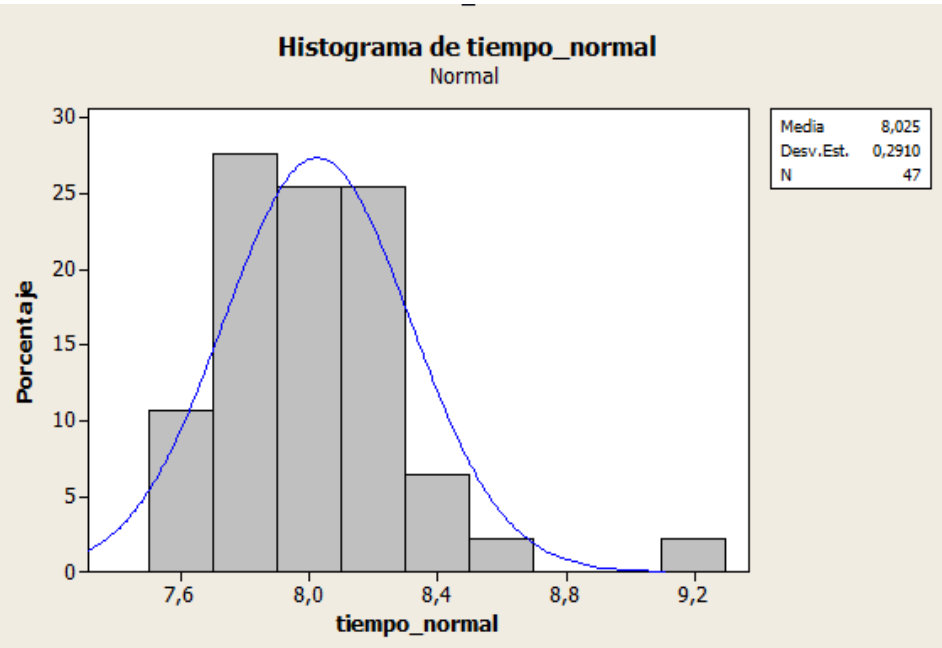


Figura 3.12 Histograma de tiempo de transmisión – condiciones normales- RIPNG

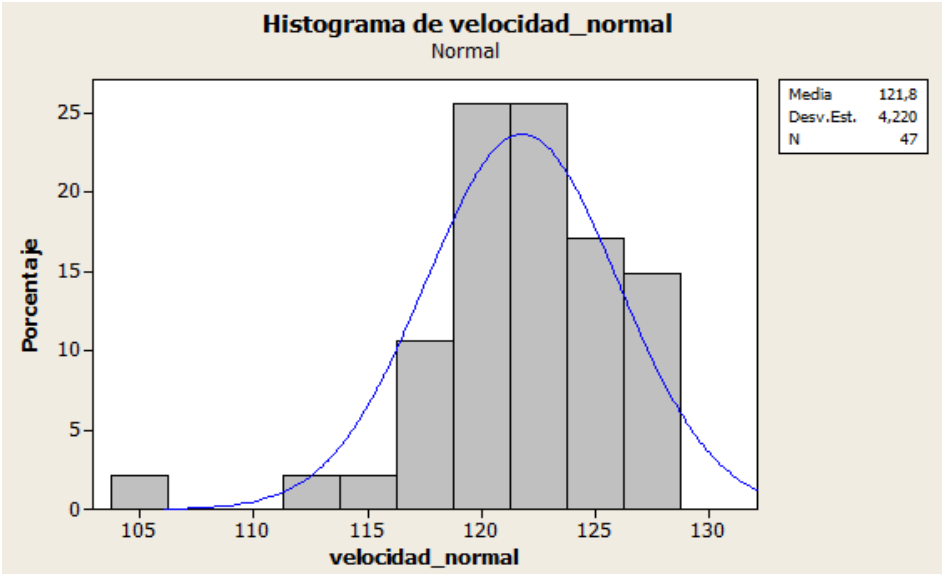


Figura 3.13 Histograma de velocidad de transmisión – condiciones normales- RIPNG

Como se puede notar en los histogramas, el rango de tiempo para RIPng fue un poco mayor al de OSPF, llegando a estar entre 7,6 y 8,6 segundos. Mientras que la velocidad de transmisión presenta un rango menor al que se presentó en el histograma de velocidad de transmisión en OSPF, estando entre 115 Kb/s y 128 Kb/s.

Bajo ataque flood

La misma prueba de inundación (flood) para simular un ataque DOS, la utilizamos para probar tiempo y velocidad de transmisión en RIPng, obteniendo los resultados que se presentan en las siguientes tablas:

Tabla 16 Tiempo de transmisión – bajo ataque DOS – RIPNG

VARIABLES ESTADISTICAS	DESCRIPCION
Media	133,20 s
Desviación estándar	29,51
Varianza	870,69
Mínimo	72,26 s
Máximo	195,39 s

Tabla 17 Velocidad de transmisión – bajo ataque DOS – RIPNG

VARIABLES ESTADISTICAS	DESCRIPCION
Media	7,728 Kb/s
Desviación estándar	2,093
Varianza	4,382
Mínimo	4,995 Kb/s
Máximo	13,506 Kb/s

Como podemos apreciar, el ataque DOS afectó un poco más a esta red que a la configurada con OSPF, pues su tiempo de transmisión fue relativamente mayor, con una media de 26 segundos más (133s versus 107s en OSPF). Además se puede apreciar que la varianza obtenida fue mucho menor a la varianza en OSPF debido a que en este escenario los valores tuvieron un rango más ajustado.

Así también la media de velocidad de transmisión disminuyó con respecto al protocolo anterior (7,7 Kb/s versus 10,25 Kb/s en OSPF).

Con estos datos se obtuvo los siguientes histogramas:

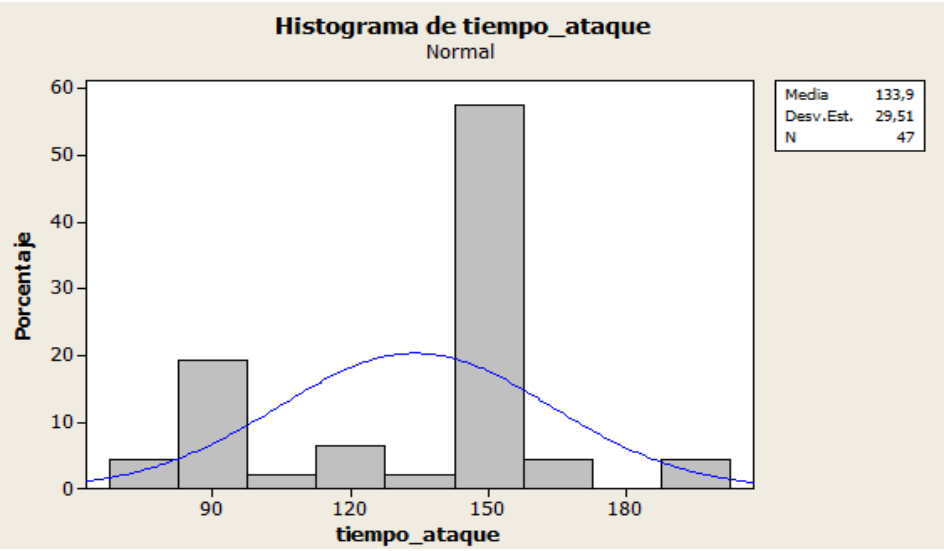


Figura 3.14 Histograma de tiempo de transmisión – bajo ataque DOS – RIPNG

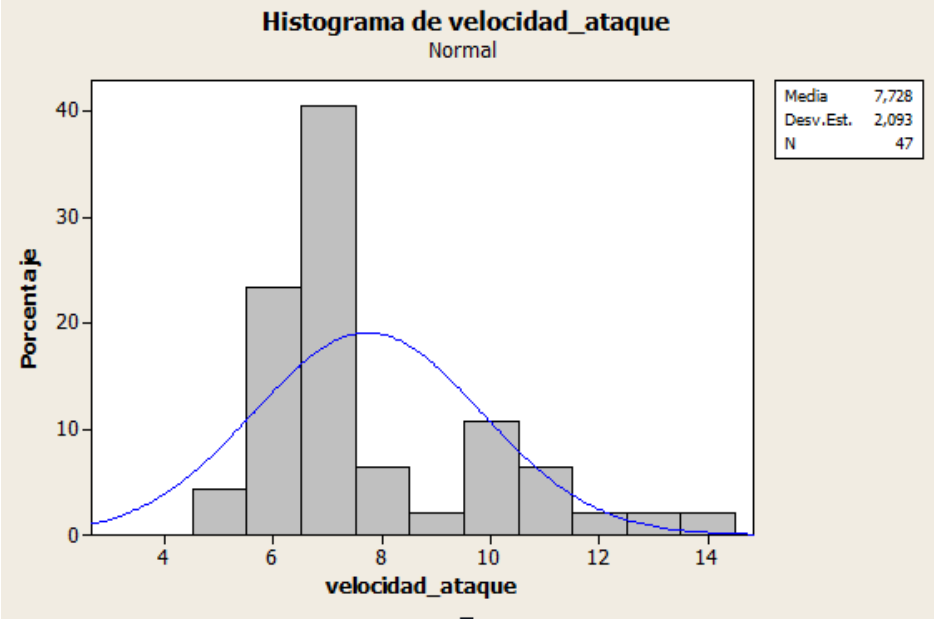


Figura 3.15 Histograma de velocidad de transmisión – bajo ataque DOS – RIPNG

Se puede notar en el histograma de tiempo de transmisión que el rango en este caso abarca menos valores, teniendo valores desde 80 a 170 segundos, es decir que los valores fueron más cercanos entre sí o tuvieron menos dispersión que en OSPF. Así mismo el rango en velocidad de transmisión fue un poco menor, cayendo en valores entre 5 y 14 Kb/s.

Convergencia al subir VPN en RIPng

A continuación se muestran los datos (tabla e histograma) de la prueba desarrollada anteriormente pero ahora sobre el protocolo RIPng:

Tabla 18 Tiempo de convergencia –VPN – RIPNG

VARIABLES ESTADISTICAS	DESCRIPCION
Media	17,817 s
Desviación estándar	4,468
Varianza	19,967
Mínimo	10,192 s
Máximo	31,824 s

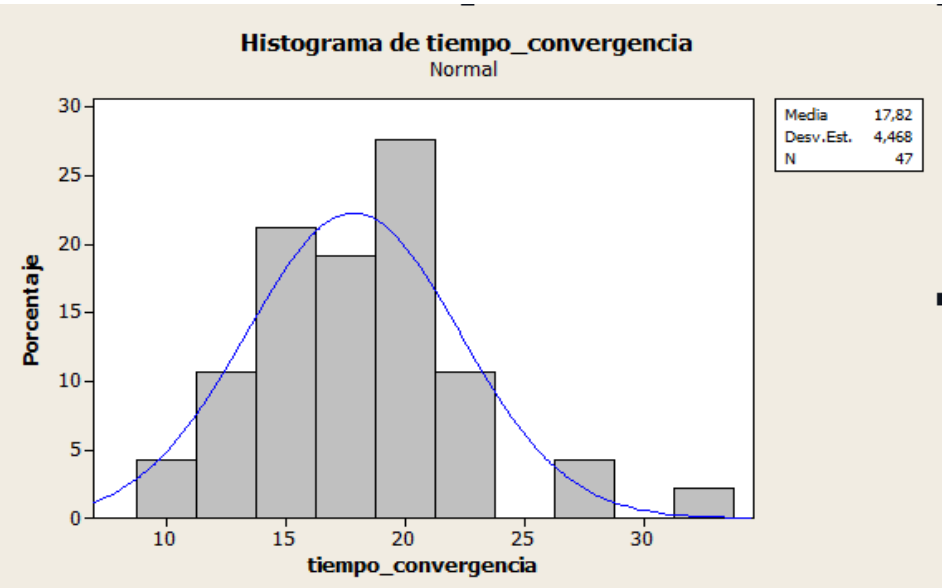


Figura 3.16 Histograma de tiempo de convergencia – VPN – RIPNG

Como nos podemos dar cuenta la media del tiempo de convergencia en RIPng fue relativamente menor. La varianza en RIPng también resulto ser menor que en OSPF, lo que quiere decir que los datos tomados en RIPng fueron más uniformes, teniendo un rango de valores entre 10s y 23s y otros datos aberrantes de 27 y 31s.

3.4.3 PRUEBA DE CONFIDENCIALIDAD

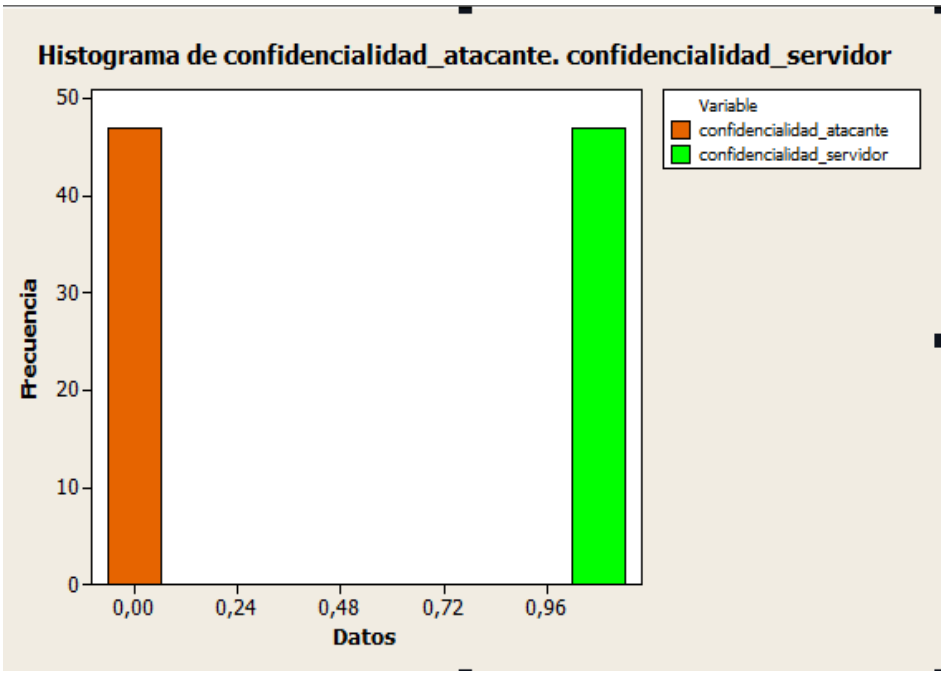


Figura 3.17 Histograma de confidencialidad– sin ataque- RIPNG

Reproduciendo la misma prueba de confidencialidad efectuada al protocolo OSPFv3 pero ahora sobre el protocolo RIPng, se obtienen los mismos resultados al tener habilitada una VPN entre la red del servidor y la red del cliente. El atacante no percibe ninguna información por parte del cliente mientras que el servidor recibe el 100% de información enviada desde el cliente.

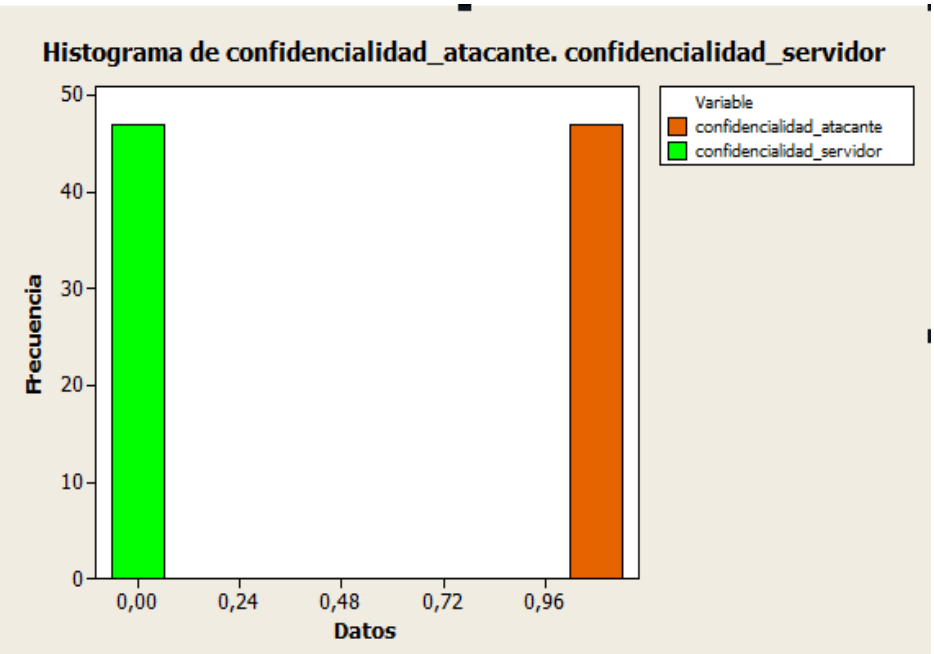


Figura 3.18 Histograma de confidencialidad – con ataque- RIPNG

Al igual que en OSPFv3 sino hubiera una VPN configurada, el atacante podrá percibir el 100% del tráfico enviado desde el cliente hacia el servidor haciendo uso del falseo de rutas que permite ejecutar el software Quagga. De igual forma el servidor quedará incomunicado del cliente.

3.4.4 PRUEBA DE INTEGRIDAD

Tiempo de convergencia al cambiar ruta

A continuación se muestran los resultados (tabla e histograma) con las pruebas en la red de RIPng:

Tabla 19 Tiempo de ataque a integridad de información – RIPNG

VARIABLES ESTADISTICAS	DESCRIPCION
Media	11,717 s
Desviación estándar	2,656
Varianza	7,052
Mínimo	16,245 s
Máximo	11,287 s

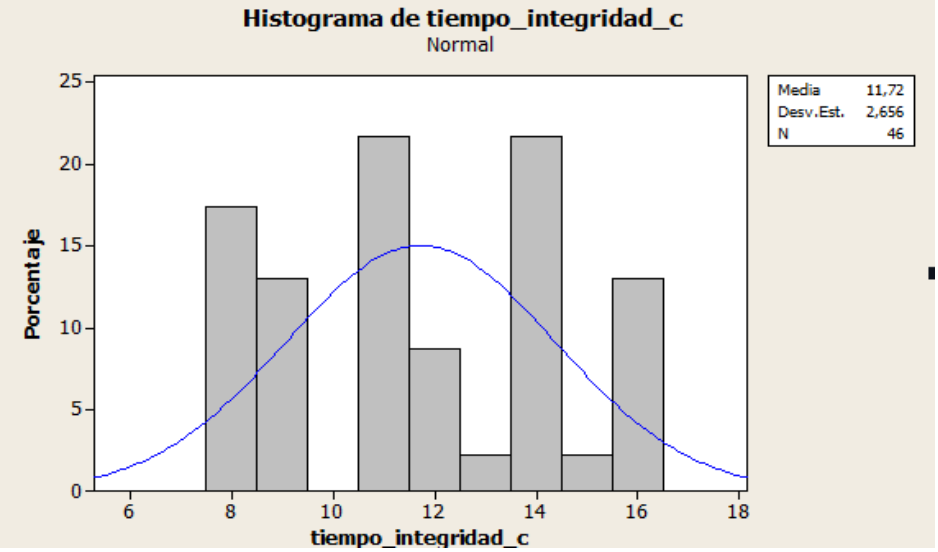


Figura 3.19 Histograma de tiempo – integridad – RIPNG

Como se puede observar en los resultados con RIPng, estos no fueron tan uniformes como en el caso anterior, teniendo una varianza mayor. El valor máximo fue de 16 segundos indicando que en el peor de los casos un atacante demoraría un poco más en alterar la información enviada. La convergencia de la red por modificación de rutas fue de mayor tiempo que en el caso con OSPFv3.

CAPÍTULO 4

RESULTADOS Y ANÁLISIS

En este capítulo se compara la respuesta de ambos protocolos considerando los resultados obtenidos en cada una de las pruebas anteriores (Disponibilidad, Confidencialidad e Integridad). Al final se hace una comparativa general que incluye cada parámetro evaluado.

4.1 PRESENTACIÓN Y ANÁLISIS DE LA TABLA DE RESULTADOS DE DISPONIBILIDAD DE OSPFV3 VS RIPNG

Tabla 20 TIEMPO Y VELOCIDAD DE TRANSMISIÓN OSPFV3 VS RIPNG (SIN ATAQUE)

VARIABLES ESTADISTICAS	OSPFv3	RIPNG
TIEMPO DE TRANSMISION		
Media	7,925 s	8,05 s
Desviación estándar	0,179	0,291
Varianza	0,0323	0,0847

Mínimo	7,675 s	7,608 s
Máximo	8,412 s	9,205 s
VELOCIDAD DE TRANSMISION		
Media	123,20 Kb/s	121,762 Kb/s
Desviación estándar	2,75	4,22
Varianza	7,59	17,81
Mínimo	116,02 Kb/s	106,03 Kb/s
Máximo	127,17 Kb/s	128,29 Kb/s

El tiempo promedio de transmisión de un archivo estándar de 1MB sobre el protocolo OSPFv3 fue menor que el tiempo promedio registrado para RIPng de igual modo el promedio para la velocidad de transmisión fue más conveniente en OSPFv3 que en RIPng pues presentó un mayor valor.

Tabla 21 TIEMPO Y VELOCIDAD DE TRANSMISIÓN OSPFV3 VS RIPNG (CON ATAQUE FLOOD)

VARIABLES ESTADISTICAS	**OSPFv3**	**RIPNG**
TIEMPO DE TRANSMISION		
Media	107,28 s	133,20 s
Desviación estándar	35,780	29,51
Varianza	1280,220	870,69
Mínimo	44,072 s	72,26 s
Máximo	177,41 s	195,39 s
VELOCIDAD DE TRANSMISION		
Media	10,251 Kb/s	7,728 Kb/s
Desviación estándar	3,733	2,093
Varianza	13,936	4,382
Mínimo	5,501 Kb/s	4,995 Kb/s
Máximo	22,146 Kb/s	13,506 Kb/s

El ataque de inundación de mensajes router-advertisements ejecutado con la herramienta flood_router6 afectó considerablemente el tiempo de

transmisión del archivo en ambos protocolos, sin embargo la mejor respuesta la continúa presentando el protocolo OSPFv3 con una media para tiempo y velocidad de transmisión menor a los resultados obtenidos en RIPng.

Tabla 22 TIEMPO DE CONVERGENCIA DE LA VPN OSPFv3 vs RIPNG

VARIABLES ESTADISTICAS	OSPFv3	RIPNG
TIEMPO DE CONVERGENCIA		
Media	24,865 s	17,817 s
Desviación estándar	6,546	4,468
Varianza	42,851	19,967
Mínimo	14,480 s	10,192 s
Máximo	44,200 s	31,824 s

El tiempo promedio de convergencia de la VPN sobre RIPng luego de una falla en la red es menor al tiempo presentado en OSPFv3 de igual modo sus tiempos mínimos y máximos son menores a los de OSPFv3.

4.2 PRESENTACIÓN Y ANÁLISIS DE LA TABLA DE RESULTADOS DE CONFIDENCIALIDAD DE OSPFV3 VS RIPNG

Tabla 23 PORCENTAJE DE INFORMACIÓN OSPFV3 VS RIPNG

VARIABLES	OSPFv3	RIPNG
CON VPN		
Porcentaje de Información recibida por el ATACANTE	0%	0%
Porcentaje de Información recibida por el SERVIDOR	100%	100%
SIN VPN		

Porcentaje de Información recibida por el ATACANTE	100%	100%
Porcentaje de Información recibida por el SERVIDOR	0%	0%

Como se registra en la tabla, ambos protocolos presentaron la misma respuesta para la prueba de confidencialidad realizada.

4.3 PRESENTACIÓN Y ANÁLISIS DE LA TABLA DE RESULTADOS DE INTEGRIDAD DE OSPFV3 VS RIPNG

Tabla 24 TIEMPO REQUERIDO PARA EFECTUAR UN ATAQUE DE INTEGRIDAD Y PORCENTAJE DE INFORMACIÓN MODIFICADA OSPFv3 vs RIPNG

VARIABLES ESTADISTICAS	**OSPFv3**	**RIPNG**
TIEMPO		
Media	10,042 s	11,717 s
Desviación estándar	0,0408	2,656
Varianza	0,001	7,052
Mínimo	10 s	16,245 s
Máximo	10,187 s	11,287 s
PORCENTAJE DE INFORMACIÓN MODIFICADA	100%	100%

Como se puede apreciar, el tiempo requerido para llevar a cabo un ataque de integridad sobre el protocolo OSPFv3 es menor al tiempo requerido para llevar a cabo el mismo ataque sobre el protocolo RIPng.

4.4 TABLA DE RESUMEN DE RESULTADOS OSPFV3 VS RIPNG

A continuación se presenta un resumen de las tablas anteriormente expuestas:

Tabla 25 DISPONIBILIDAD

VARIABLES ESTADISTICAS	OSPFv3	RIPNG
Tiempo promedio en condiciones normales	7,925 s	8,025 s
Tiempo promedio en ambiente de ataque DOS	107,28 s	133,20 s
Tiempo promedio de convergencia VPN	24,865 s	17,817 s
Velocidad promedio en condiciones normales	123,20 Kb/s	121,76 Kb/s
Velocidad promedio en ambiente de ataque DOS	10,251 Kb/s	7,728 Kb/s

De acuerdo a lo expuesto, el protocolo OSPFv3 presenta una mejor respuesta frente a ataques de red en términos de disponibilidad.

Tabla 26 CONFIDENCIALIDAD

VARIABLES ESTADISTICAS	OSPFv3	RIPNG
Porcentaje tráfico hacia el servidor condiciones normales	100%	100%
Porcentaje tráfico hacia el atacante condiciones normales	0%	0%
Porcentaje tráfico hacia el servidor bajo redirección de trafico	0%	0%
Porcentaje tráfico hacia el atacante bajo redirección de trafico	100%	100%

En términos de confidencialidad, ambos protocolos reflejan los mismos resultados.

Tabla 27 INTEGRIDAD

VARIABLES ESTADISTICAS	OSPFv3	RIPNG
Tiempo promedio de ataque en la integridad de la información	10,042 s	11,717
Porcentaje que se modificó a la información	100%	100%

En términos de integridad, RIPng presenta una mayor seguridad, pues, sobre este protocolo toma más tiempo efectuar un ataque de este tipo.

CONCLUSIONES Y RECOMENDACIONES

Culminando el presente análisis comparativo de seguridad entre dos protocolos IPv6, entre los cuales escogimos OSPFv3 y RIPng, se determinaron las siguientes conclusiones:

1. En cuanto a la prueba de disponibilidad pudimos observar que el protocolo RIPng se vio más afectado en cuanto al ataque DOS, teniendo un tiempo 24,16% mayor al que obtuvimos con el tiempo de transmisión con OSPFv3, así mismo la velocidad de transmisión disminuyó en el mismo porcentaje. En consecuencia, podemos concluir que OSPFv3 resulta ser más eficiente a los ataques DOS.

2. En cuanto a la prueba de confidencialidad, el empate entre los dos protocolos es evidente y también lo es la manera en que la tecnología de seguridad implementada, en este caso la VPN, es efectiva para contrarrestar este tipo de ataque en ambos protocolos.

3. En cuanto a la prueba de integridad según lo que se puede observar en el cuadro de resumen, en ambos casos se pudo modificar y hacer llegar la misma cantidad de información al servidor. Para el tiempo de convergencia se obtuvieron resultados similares en ambos protocolos una vez insertadas las falsas actualizaciones de enrutamiento. Los resultados de RIPng variaron un poco más que los de OSPFv3 y en el peor de los casos tuvieron un tiempo de convergencia un poco mayor siendo esto un beneficio para la seguridad pues un posible atacante tardaría más en llevar a cabo un ataque de este tipo.

4. Finalmente podemos declarar como ganador al protocolo OSPFv3 pero con ventaja mínima pues en las dos últimas pruebas hubo un empate en cuanto a los resultados, mientras que, en las pruebas de disponibilidad, se pudo notar que OSPFv3 ofreció una mejor respuesta frente a ataques de este tipo.

A continuación presentamos las respectivas recomendaciones:

1. Realizar un análisis exhaustivo de la configuración de la red con direccionamiento IPv6, por ejemplo sus tablas de enrutamiento pues IPv6 es un protocolo relativamente nuevo, y aún no muy difundido o implementado a nivel nacional.

2. Tener siempre presente la implementación y configuración de un sistema de seguridad en nuestra red, el cual prevenga y evite cualquier intrusión potencial. Este sistema o tecnología de seguridad debe contar por ejemplo con firewall, VPN, IPS/IDS, etc.

3. Verificar siempre que sea posible el correcto funcionamiento de los sistemas de seguridad ya implementados dentro de la red.

4. En cuanto sea posible se debe descargar actualizaciones para los sistemas operativos con el fin de mitigar nuevas amenazas.

5. En caso de algún ataque en algún equipo o LAN de la red, se recomienda aislar a este equipo o LAN, para evitar que el ataque se propague al resto de la red.

6. Realizar futuros análisis comparativos con otros protocolos de enrutamiento IPv6 como son IS-IS y EIGRPv3, con el fin de mostrar

ventajas o desventajas que puedan presentar estos protocolos en términos de seguridad.

GLOSARIO

Algoritmo.- Conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad.

Cifrado.- Método que permite aumentar la seguridad de un mensaje o de un archivo mediante la codificación del contenido, de manera que solo pueda leerlo la persona que disponga de la clave de cifrado adecuada para descodificarlo.

Conmutación.- Conexión que realizan los diferentes nodos que existen en distintos lugares para lograr conectar a dos usuarios de una red de telecomunicaciones.

Datagrama.- Un datagrama es un fragmento de paquete que es enviado con la suficiente información para que la red pueda encaminar el fragmento hacia el equipo terminal de datos receptor.

Denegación de Servicio.- Ataque a un sistema de computadoras o red que causa que un servicio o recurso sea inaccesible a los usuarios legítimos.

Enrutamiento.- Función de buscar un camino entre todos los posibles en una red de paquetes cuyas topologías poseen una gran conectividad.

Protocolo.- Conjunto de reglas y estándares que controlan la secuencia de mensajes que ocurren durante una comunicación entre entidades que forman una red.

BIBLIOGRAFÍA

[1] Rob Coltun, Dennis Ferguson, John Moy, Acee Lindem (2008, Julio) OSPF for IPV6, https://tools.ietf.org/html/rfc5340, fecha de consulta septiembre del 2014.

[2] Gary Scott Malkin, Robert E. Minnear (1997, Enero), RIPng for IPv6, https://www.ietf.org/rfc/rfc2080.txt, fecha de consulta noviembre del 2014.

[3] Hugo Adrian Francisconi (2005, Agosto), IPsec en Ambientes IPv4 e IPv6, fecha de consulta septiembre del 2014.

[4] Santiago Pérez Iglesias (2001, Noviembre), Análisis del protocolo IPsec: el estándar de seguridad en IP, http://www.frlp.utn.edu.ar/materias/internetworking/apuntes/IPSec/ipsec.pdf, fecha de consulta: octubre del 2014.

[5] Juan Luis García Rambla (2012), Ataques en redes de datos IPv4 e IPv6, fecha de consulta noviembre del 2014.

[6] Castro Gil Manuel Alonso, Díaz Urueta Gabriel, Alzórriz Armendáriz Ignacio, Sancristóbal Ruiz Elio (2013), Procesos y herramientas para la seguridad de redes, fecha de consulta noviembre del 2014.

[7] Purificación Aguilera López (2010, Junio), Seguridad informática, fecha de consulta Diciembre del 2014.

[8] Güimi (2009, Abril), Redes de Comunicaciones, http://guimi.net/index.php?pag_id=cmsxp05_documentacion.html, fecha de consulta octubre del 2014.

[9] Monografías.com, Modelo OSI, http://www.monografias.com/trabajos13/modosi/modosi.shtml, fecha de consulta octubre del 2014.

[10] CiscoPress.com, RIP and RIPng Routing, http://www.ciscopress.com/articles/article.asp?p=2180210&seqNum=10, fecha de consulta octubre del 2014.

[11] Wikipedia.com, Ataque de Denegación de Servicio, http://es.wikipedia.org/wiki/Ataque_de_denegaci%C3%B3n_de_servicio, fecha de consulta noviembre del 2104.

ANEXO A

WAMP SERVER 5.6

Plataforma de desarrollo web para Windows que permite servir páginas HTML a internet, además de poder gestionar datos en ellas. Al mismo tiempo, WAMP proporciona lenguajes de programación para desarrollar aplicaciones web.

Su sistema de infraestructura de internet usa las siguientes herramientas:

- Windows, como sistema operativo.
- Apache, como servidor web.
- MySQL, como gestor de bases de datos.
- PHP (generalmente), Perl, o Python, como lenguajes de programación.

ANEXO B

ARCHIVOS DE CONFIGURACIÓN EN EL SERVIDOR

Ambos archivos fueron colocados en la ruta /wamp/www

Contacto.html

```
<!DOCTYPE html>
<html>
 <head>
  <title>Formulario de contacto</title>
 </head>
 <body>
            <form action="contacto.php" method="post">
            <label for="nombre">Nombre:</label>
             <input id="nombre" type="text" name="nombre"
placeholder="Nombre y Apellido" required="" />
             <label for="email">Email:</label>
             <input id="email" type="email" name="email"
placeholder="ejemplo@correo.com" required="" />
             <label for="contrasena">Contrasena:</label>
             <input id="contrasena" type="password"
name="contrasena" placeholder="Elija una buena contrasena"
required="" />
             <input id="submit" type="submit" name="submit"
value="Enviar" />
            </form>
 </body>
</html>
```

Contacto.php

```
<?php
        $nombre = $_POST['nombre'];
        $email = $_POST['email'];
        $contrasena = $_POST['contrasena'];

        $conexion =
mysqli_connect("127.0.0.1","root","","personas") or die("Error ".
mysqli_error($conexion));
        $query = "INSERT INTO personas
(id,nombre,email,contrasena) VALUES (id, '$nombre', '$email',
'$contrasena')";

        mysqli_query($conexion,$query);
        echo mysqli_error($conexion);

        mysqli_close($conexion);
?>
```

ANEXO C

INSTALACIÓN DEL SOFTWARE QUAGGA EN EL ATACANTE

Se lo instala mediante el comando:

apt-get install quagga

Luego se copian los archivos de ejemplo que se encuentran en la siguiente ruta:

/usr/share/doc/quagga/examples/

Consideraremos los siguientes archivos en /etc/quagga:

daemons, ripngd.conf, ospf6d.conf, zebra.conf

En el archivo daemons se habilita el enrutamiento con el que vamos a trabajar, por ejemplo si deseamos habilitar ospf tendríamos que tener lo siguiente en el archivo:

zebra=yes
bgpd=no
ospfd=yes
ospf6d=no
ripd=no
ripngd=no
isisd=no

En el archivo **zebra.conf** agregamos las siguientes líneas, de acuerdo al requerimiento de nuestra red:

```
Interface eth1
ipv6 address 2001:db8:100:4::2 /64
no shutdown
!
Interface eth2
ipv6 address 2001:db8:100:1::2 /64
no shutdown
```

Para iniciar el servicio de quagga:

sudo /etc/init.d/quagga start

En R2 podremos ver los mensajes por consola al levantar y apagar el servicio de quagga, este es el mensaje que se muestra cuando se inicia el servicio quagga:

**Mar 1 02:37:12.883: %OSPFv3-5-ADJCHG: Process 1, Nbr 255.1.1.1 on FastEthernet0/1 from LOADING to FULL, Loading Done*

Al apagar Quagga observaremos:

**Mar 1 01:21:29.967: %OSPFv3-5-ADJCHG: Process 1, Nbr 255.1.1.1 on FastEthernet0/1 from FULL to DOWN, Neighbor Down: Dead timer expired*

30 segundos es el tiempo aproximado en el cual se actualiza la tabla de enrutamiento.

ANEXO D

LISTA DE HERRAMIENTAS QUE PERTENECEN AL SUITE THC-IPV6

parasite6

Petición/anuncio spoofer de un icmp vecino, coloca al atacante en una posición hombre en el medio, parecido a ARP MITM (y parasite).

alive6

Un scan en vivo muy efectivo que detectara todos los sistemas de escucha de esta dirección.

dnsdict6

Diccionario dns en ipv6 de fuerza bruta.

fake_router6

Permite al atacante anunciarse como un enrutador en la red con la más alta prioridad.

redir6

Redirige el trafico al atacante de forma inteligente (hombre-en-el-medio) con un spoofer de redireccionamiento icmpv6.

toobig6

Decrementador mtu con la misma inteligencia que el redir6.

detect-new-ip6

Detecta nuevos dispositivos IPV6 que se unan a la red, el atacante puede ejecutar un script para escanear automáticamente estos sistemas.

dos-new-ip6

Detectar nuevos dispositivos IPv6 y decirles que su IP elegida colisiona en la red (DOS).

trace6

traceroute6 veloz con soporte ICMP6 echo request y TCP-SYN.

flood_router6

Inunda un objetivo con router-advertisements (anuncios de enrutador) al azar.

flood_advertise6

Inunda un objetivo con neighbor advertisements (anuncios de vecino) al azar.

fuzz_ip6

Fuzzer para ipv6.

implementation6

Realiza varias comprobaciones de implementación en ipv6.

implementation6d

Escucha un demonio para implementation6 para que compruebe detrás de un FW.

fake_mld6

Permite anunciarse al atacante en un grupo multicast de su elección en la red.

fake_mld26

Misma funcionalidad que la anterior pero para MLDv2.

fake_mldrouter6

Falsea mensajes MLD de enrutador.

fake_mipv6

Permite el robo de un móvil IP si IPSEC no es necesario para la autenticación.

fake_advertiser6

Anuncia al atacante en la red.

smurf6

Smurfer local.

rsmurf6

Smurfer remoto, se sabe que funciona sólo en contra de linux en el momento.

exploit6

Conoce vulnerabilidades IPv6 para ponerlos a prueba frente a un objetivo.

denial6

Consiste en una colección de ensayos de denegación de servicio en contra de un objetivo.

thcping6

Permite el envío de un paquete ping6 hecho a mano.

sendpees6

Herramienta que genera una petición neighbour-solicitation con CGAs (relacionado a cifrado) para mantener la CPU ocupada.

ANEXO E

PRUEBAS ESTADÍSTICAS

La estadística es una ciencia que nos permite interpretar información de manera clara y precisa, por eso es indispensable tabular los datos que obtenemos en cada una de las pruebas de convergencia y confiabilidad para poder comparar los IGP's y obtener resultados que sustenten por qué un protocolo de enrutamiento prevalece sobre otro. La ecuación 2 nos permite calcular el número mínimo de observaciones que deben efectuarse; dónde n es el número mínimo de observaciones, W el rendimiento mínimo esperado, Z_β poder estadístico y Z_α nivel de confianza asignado**.**

$$n = \frac{(W - W^2)\,[Z_\beta + 1.4\,(Z_\alpha)]^2}{W^2}$$

Para nuestro proyecto hemos decidido tomar los siguientes valores:
Nivel de confianza (Z_α) 99%, Diferencia mínimo observable (W) 30% y Poder estadístico (Z_β) 80%.

Usando estos valores, la ecuación queda de la siguiente manera:

$\mathbf{Z_\alpha}$ **=** 2,576 $\mathbf{Z_\beta}$ **=** 0,842 **W =** 0,30

$$n = \frac{(0{,}30 - 0{,}30^2)\,[0{,}842 + 1.4\,(2{,}576)]^2}{0{,}30^2} = 47$$

Tenemos como resultado que el número mínimo de observaciones debe ser 47 por tanto este será aplicado para cada una de las pruebas y posterior tabulación de datos. Una vez que hemos tomado el número de observaciones suficientes procedemos a la aplicación de estadística descriptiva.

ANEXO F

CONFIGURACIÓN DE LOS ARCHIVOS DENTRO DE QUAGGA

OSPFV3

En el archivo ospf6d.conf tenemos predeterminadas las siguientes configuraciones para cada interfaz:

```
!
! Zebra configuration saved
from vty
!   2003/11/28 00:49:49
10
@plant
password zebra
log stdout
service advanced-vty
!
debug ospf6 neighbor state
!
```

Las siguientes líneas fueron añadidas al final del archivo:

```
interface eth1
ipv6 ospf6 instance-id 0
!
router ospf6
router-id 1.1.1.1
area 0.0.0.0 range
2001:db8:100:4::/64
interface eth1 area 0.0.0.0

area 0.0.0.0 range
2001:db8:100:1::/64
interface eth2 area 0.0.0.0
```

Se puede establecer una sesión de telnet hacia nuestro router virtual quagga con el siguiente comando: *telnet 127.0.0.1 2608.* El puerto 2608 es el puerto utilizado para las configuraciones de OSPFv3.

Para comunicarse por telnet tomar en cuentas lo puertos de acuerdo al protocolo de enrutamiento que se esté utilizando:

zebra: 2601

ripd: 2602

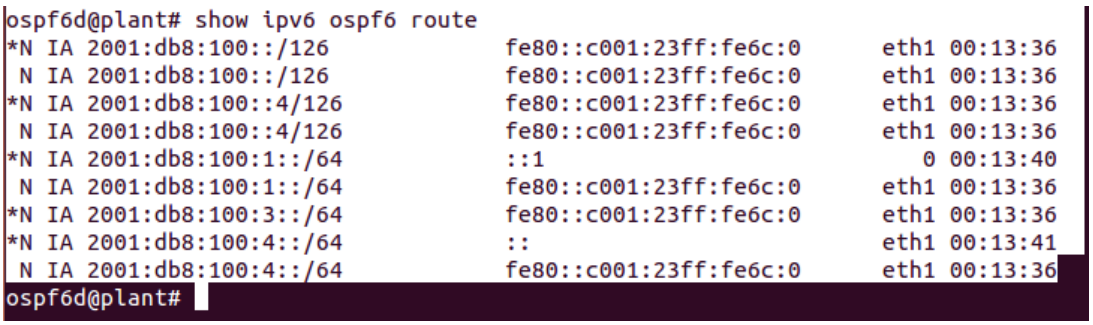
ripng: 2603

ospfd: 2604

bgpd: 2605

ospf6d: 2606

Es posible correr algunos comandos en la consola de OSPFv3:



```
ospf6d@plant# show ipv6 ospf6 interface eth1
eth1 is up, type BROADCAST
  Interface ID: 3
  Internet Address:
    inet : 192.168.1.3/24
    inet6: 2001:db8:100:4::2/64
    inet6: 2001:db8:100:4:b575:6868:1fd9:1ae4/64
    inet6: 2001:db8:100:4:a00:27ff:fede:fe81/64
    inet6: fe80::a00:27ff:fede:fe81/64
  Instance ID 0, Interface MTU 1500 (autodetect: 1500)
  MTU mismatch detection: enabled
  Area ID 0.0.0.0, Cost 1
  State BDR, Transmit Delay 1 sec, Priority 1
  Timer intervals configured:
   Hello 10, Dead 40, Retransmit 5
  DR: 2.2.2.2 BDR: 255.1.1.1
  Number of I/F scoped LSAs is 2
    0 Pending LSAs for LSUpdate in Time 00:00:00 [thread off]
    0 Pending LSAs for LSAck in Time 00:00:00 [thread off]
ospf6d@plant#
```

```
ospf6d@plant# show ipv6 ospf6 neighbor
Neighbor ID     Pri    DeadTime  State/IfState         Duration I/F[State]
2.2.2.2           1    00:00:31    Full/DR             00:17:41 eth1[BDR]
ospf6d@plant#
```

RIPNG

En el archivo ripngd.conf tenemos predeterminadas las siguientes configuraciones para cada interfaz:

```
! -*- rip -*-
!
! RIPngd sample configuration
file
!
! $Id: ripngd.conf.sample,v 1.1
2002/12/13 20:15:30 paul Exp
$
!
hostname ripngd
password zebra
!
! debug ripng events
! debug ripng packet
!
```

Las siguientes líneas fueron añadidas al final del archivo:

```
router ripng
network 2001:db8:100:4::/64
network 2001:db8:100:1::/64
```

Establecemos una sesión telnet hacia nuestro router virtual quagga usando el siguiente comando: *telnet 127.0.0.1 2603.*

Es posible ejecutar algunos comandos en la consola de RIPng:

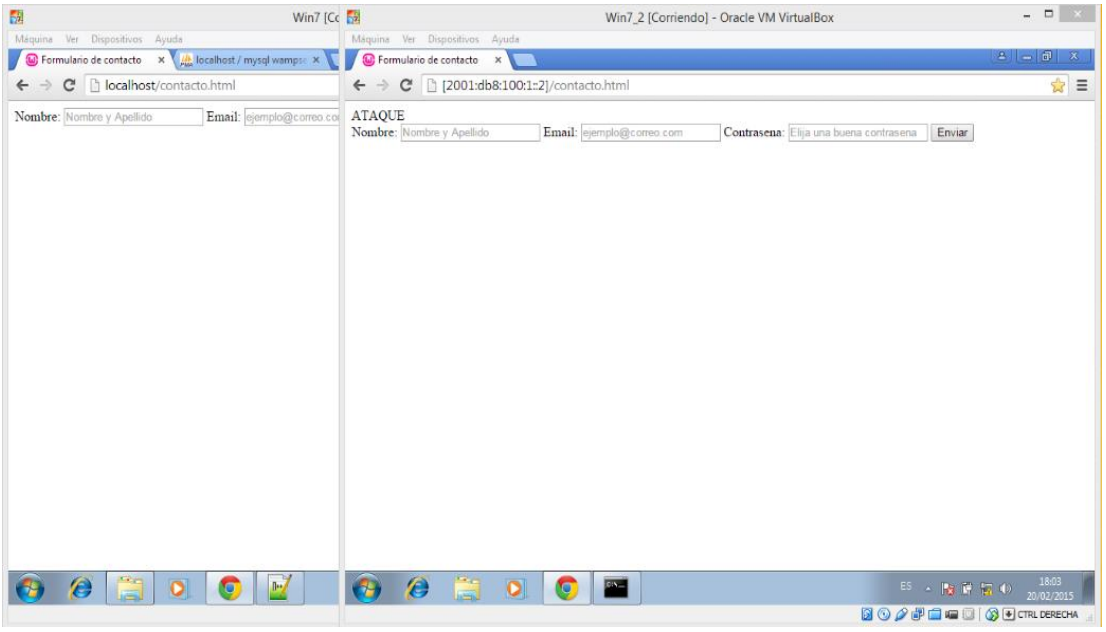
```
ripngd# show ipv6 ripng status
Routing Protocol is "RIPng"
  Sending updates every 30 seconds with +/-50%, next due in 13 seconds
  Timeout after 180 seconds, garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribution metric is 1
  Redistributing:
  Default version control: send version 1, receive version 1
    Interface        Send  Recv
    eth1             1     1
    eth2             1     1
  Routing for Networks:
    2000::/6
    2001:db8:100:4::/64
  Routing Information Sources:
    Gateway          BadPackets BadRoutes  Distance Last Update
    fe80::c002:14ff:fe2c:0
                           0          0        120      00:00:34
```

```
ripngd# show running-config

Current configuration:
!
hostname ripngd
password zebra
!
router ripng
 network 2000::/6
 network 2001:db8:100:4::/64
!
line vty
!
end
```

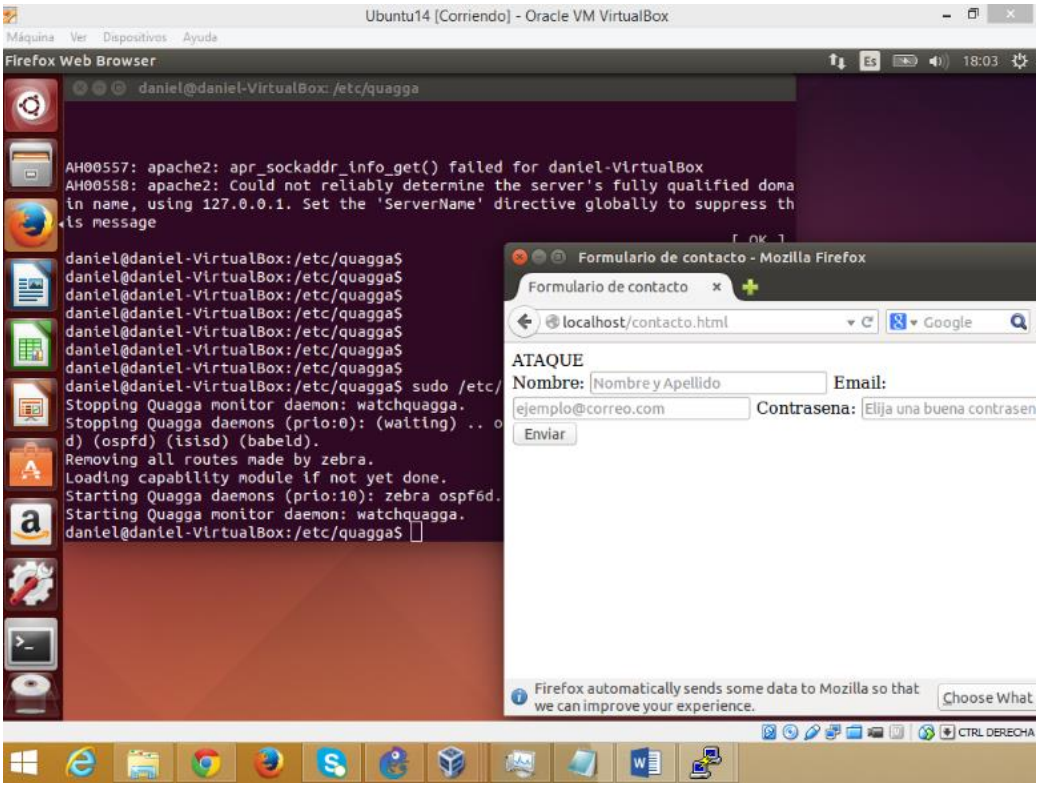
ANEXO G

DETALLE TÉCNICO DE LAS PRUEBAS REALIZADAS

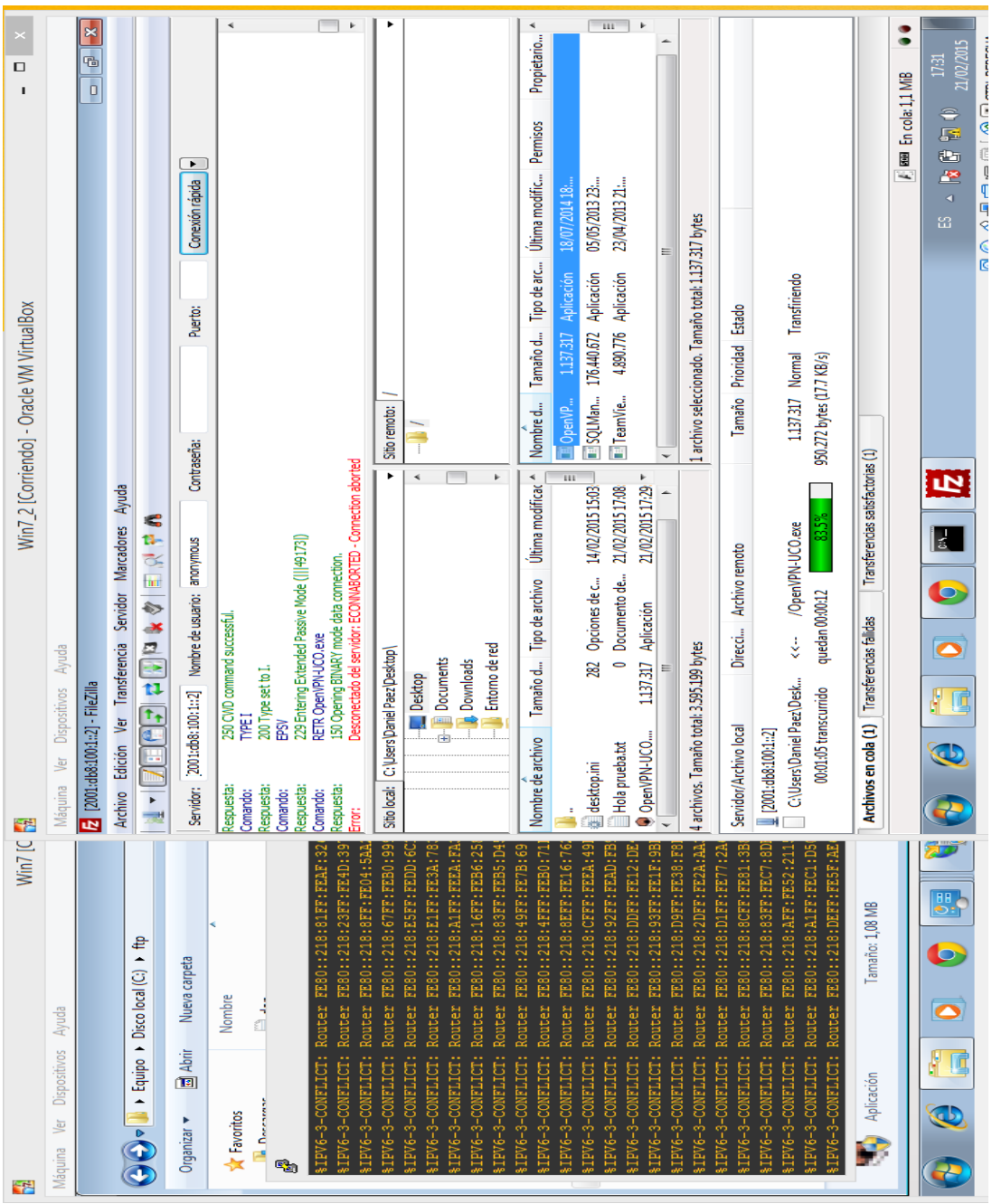
Imagen donde se aprecia el formulario web desplegado en el servidor y el formulario web suplantado por el atacante en el cliente (en la parte izquierda aparece el formulario del servidor).

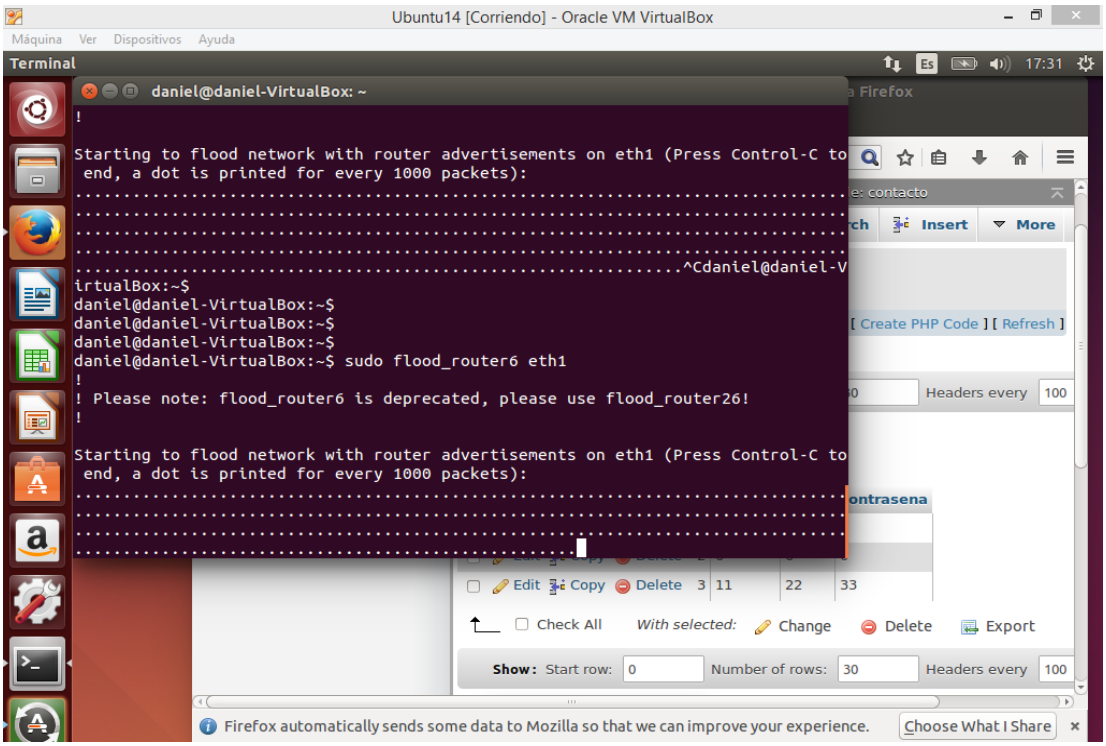


Formulario WEB en el atacante:



Transmisión de un archivo mediante el protocolo FTP desde el servidor al cliente, mientras el atacante ejecuta un ataque DOS (Denial of service) parcial:





Prueba de Disponibilidad

Se descarga el archivo, y se instalan los siguientes paquetes como prerrequisitos:

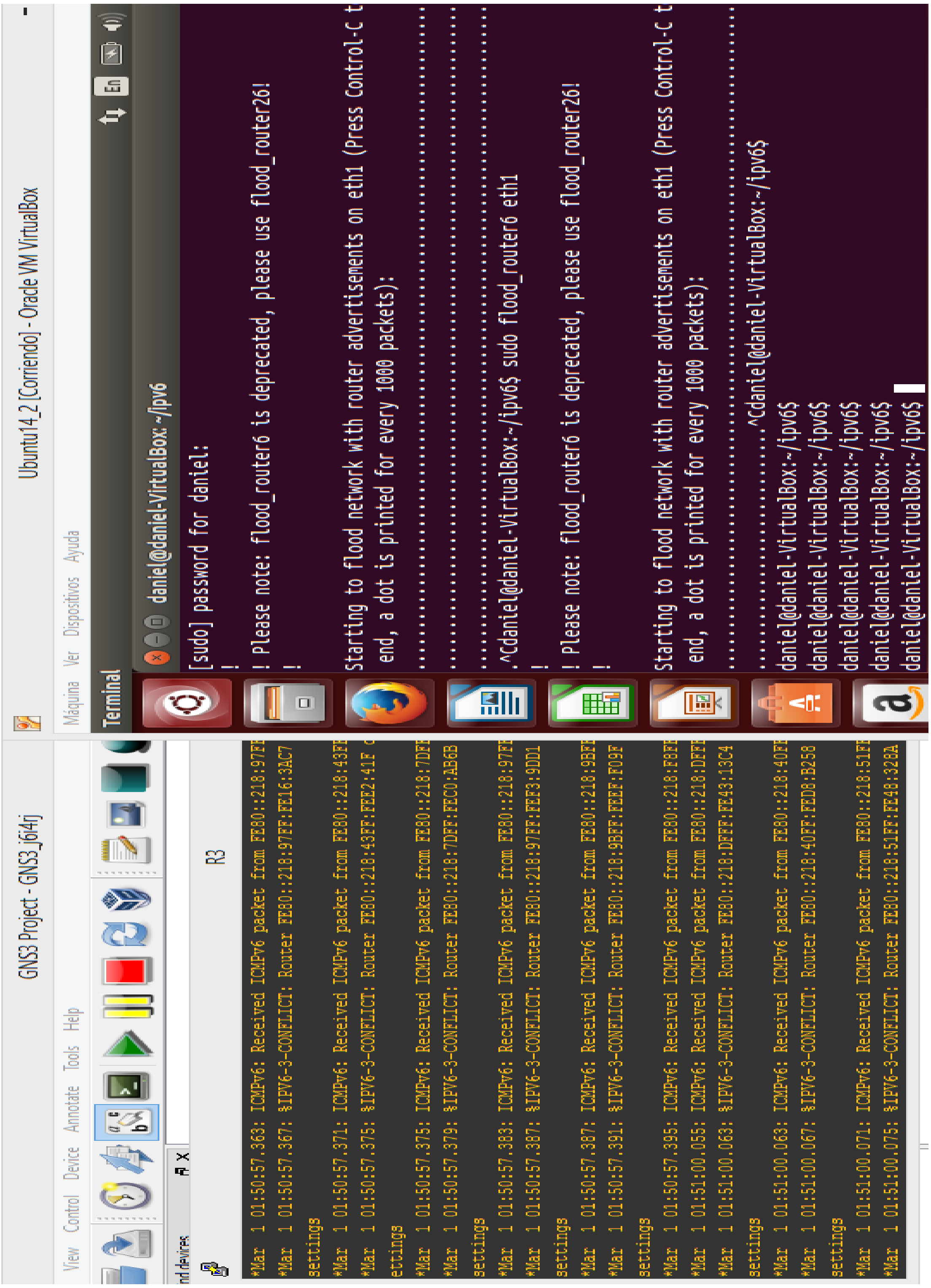
$ sudo apt-get install libpcap-dev libssl-dev libcap-dev

Para compilar: $ sudo make

Para instalar: $ sudo make install

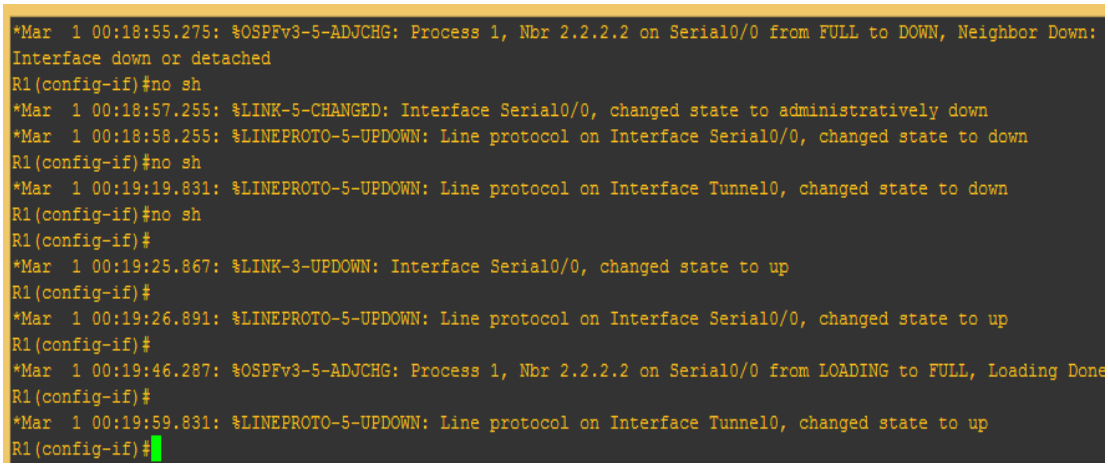
Luego ingresamos a la carpeta donde se instaló el suite de herramientas thc-ipv6 (en nuestro caso /home/daniel/ipv6) y realizamos el ataque:

$ sudo flood_router6 eth1



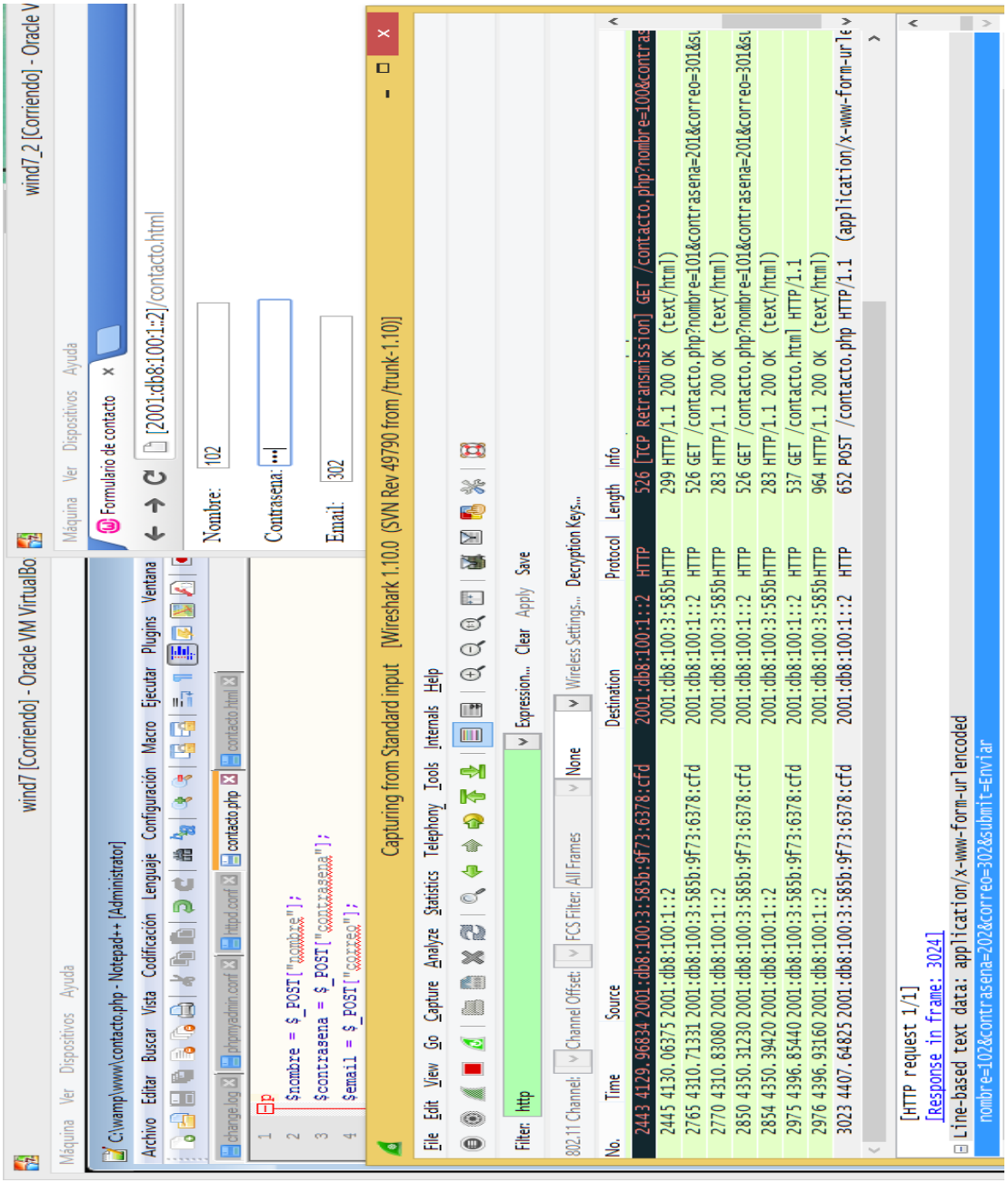
Prueba de convergencia de VPN (en disponibilidad)

Como podemos observar en el enrutador R1 al bajar subir la serial S0/0, primero se muestra el estado de la interfaz UP, luego se muestra el mensaje que se ha descubierto un nuevo vecino (Neighbour 2.2.2.2 que corresponde a R2), y por último el túnel levanta nuevamente.



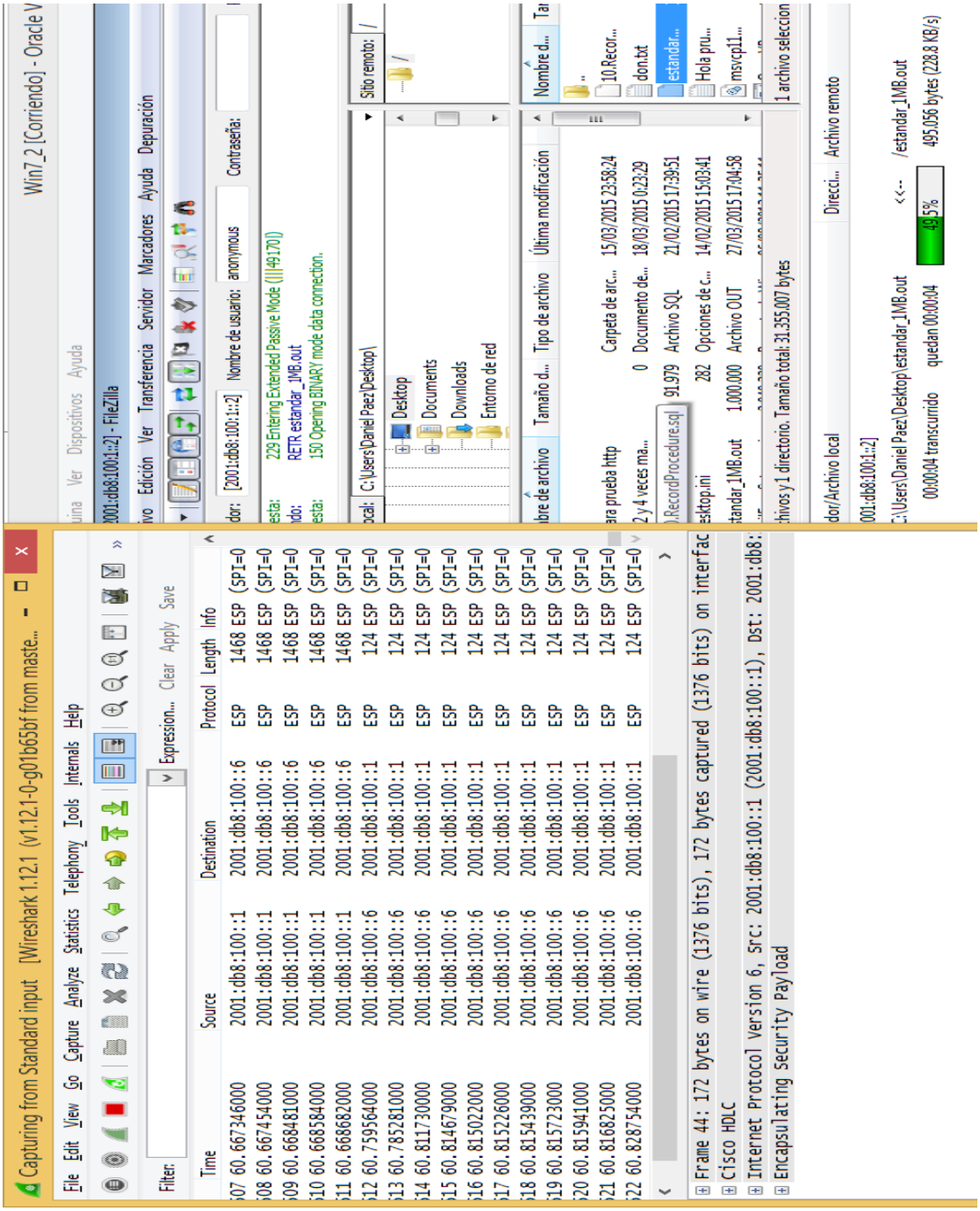
Prueba de Confidencialidad

Debido a que el atacante suplanto la dirección IP del servidor, podrá ver la información que el cliente envía, por ejemplo los datos del formulario. El atacante puede hacer uso del sniffer wireshark.



En el caso anterior el atacante pudo ver la información transmitida debido a que no había ninguna tecnología de seguridad configurada en la red. En cambio, cuando tenemos una VPN configurada y un atacante de alguna manera logra capturar los datos, solo podrá observar los paquetes con

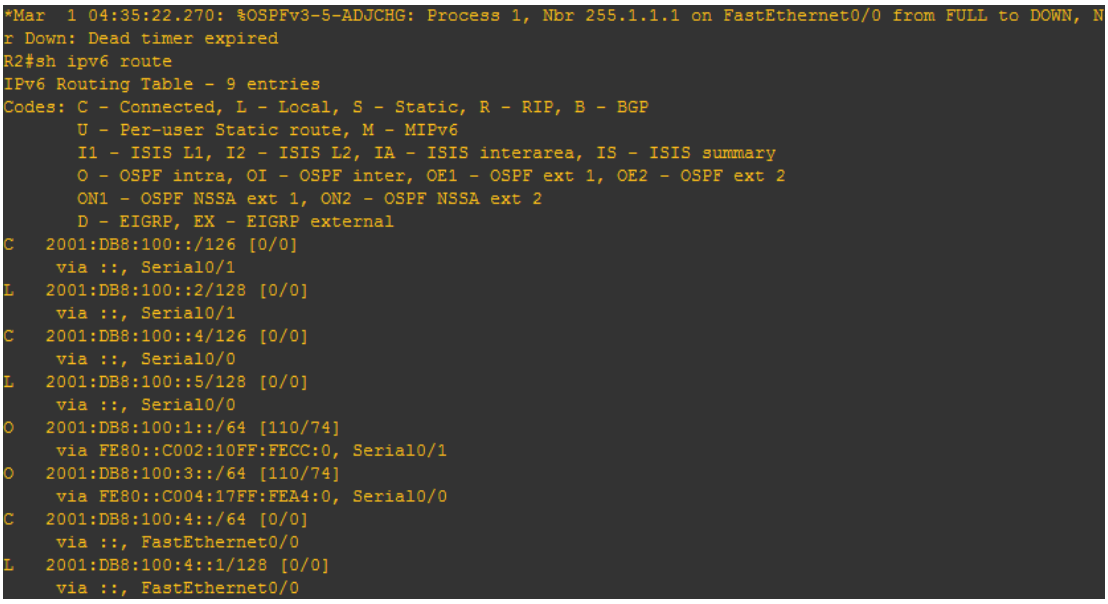
encabezados ESP, sin importar el protocolo con el que se transmita la información (en nuestro caso HTTP o FTP).



PRUEBA DE INTEGRIDAD

Al encender el servicio quagga en el atacante y suplantar la dirección del servidor, veremos el cambio en la tabla de enrutamiento de R2 cómo alcanza a la red 2001:db8:100:1:: /64 donde 2001:db8:100:1::2 /64 es el servidor.

Cuando está detenido el servicio:



Suplantando al servidor:

```
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2001:DB8:100::/126 [0/0]
     via ::, Serial0/1
L   2001:DB8:100::2/128 [0/0]
     via ::, Serial0/1
C   2001:DB8:100::4/126 [0/0]
     via ::, Serial0/0
L   2001:DB8:100::5/128 [0/0]
     via ::, Serial0/0
O   2001:DB8:100:1::/64 [110/11]
     via FE80::A00:27FF:FEDE:FE81, FastEthernet0/0
O   2001:DB8:100:3::/64 [110/74]
     via FE80::C004:17FF:FEA4:0, Serial0/0
C   2001:DB8:100:4::/64 [0/0]
     via ::, FastEthernet0/0
L   2001:DB8:100:4::1/128 [0/0]
     via ::, FastEthernet0/0
L   FF00::/8 [0/0]
     via ::, Null0
```

Suplantando al cliente:

```
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2001:DB8:100::/126 [0/0]
     via ::, Serial0/1
L   2001:DB8:100::2/128 [0/0]
     via ::, Serial0/1
C   2001:DB8:100::4/126 [0/0]
     via ::, Serial0/0
L   2001:DB8:100::5/128 [0/0]
     via ::, Serial0/0
O   2001:DB8:100:1::/64 [110/74]
     via FE80::C001:26FF:FED4:0, Serial0/1
O   2001:DB8:100:3::/64 [110/11]
     via FE80::A00:27FF:FE19:E6E4, FastEthernet0/0
C   2001:DB8:100:4::/64 [0/0]
     via ::, FastEthernet0/0
L   2001:DB8:100:4::1/128 [0/0]
     via ::, FastEthernet0/0
L   FF00::/8 [0/0]
     via ::, Null0
```

Para observar el cambio en RIPNG

debug ipv6 rip serial 0/0

debug ipv6 rip serial 0/1

debug ipv6 rip fa 0/1

```
R2#
*Mar  1 00:14:57.059: RIPng: 2001:DB8:100:1::/64, metric changed to 2
*Mar  1 00:14:57.059: RIPng: 2001:DB8:100:1::/64, added path FE80::A00:27FF:FE19:E6E4/FastEthernet0/0
*Mar  1 00:14:57.063: RIPng: Triggered update requested
R2#
*Mar  1 00:14:58.063: RIPng: generating triggered update for RIP1
*Mar  1 00:14:58.067: RIPng: Sending multicast update on Serial0/0 for RIP1
*Mar  1 00:14:58.067:        src=FE80::C002:14FF:FE2C:0
*Mar  1 00:14:58.067:        dst=FF02::9 (Serial0/0)
*Mar  1 00:14:58.067:        sport=521, dport=521, length=32
*Mar  1 00:14:58.067:        command=2, version=1, mbz=0, #rte=1
*Mar  1 00:14:58.071:        tag=0, metric=2, prefix=2001:DB8:100:1::/64
R2#
*Mar  1 00:15:04.147: RIPng: response received from FE80::C003:FFF:FE60:0 on Serial0/0 for RIP1
```

Para observar el cambio en OSPF

debug ipv6 ospf events

debug ipv6 ospf adjacency

```
*Mar  1 01:11:25.035: %OSPFv3-5-ADJCHG: Process 1, Nbr 255.1.1.1 on FastEthernet0
/0 from LOADING to FULL, Loading Done
*Mar  1 01:11:25.039: OSPFv3: Rcv LS UPD from 255.1.1.1 on FastEthernet0/0 length
 104 LSA count 2
*Mar  1 01:11:25.039:    Delete LSA 255.1.1.1/0, type 0x2001 from maxage
*Mar  1 01:11:25.099: OSPFv3: Rcv LS UPD from 1.1.1.1 on Serial0/1 length 64 LSA
count 1
*Mar  1 01:11:25.103: OSPFv3: Rcv LS UPD from 3.3.3.3 on Serial0/0 length 64 LSA
count 1
R2#
R2#
*Mar  1 01:11:34.899: OSPFv3: Neighbor change Event on interface FastEthernet0/0
*Mar  1 01:11:34.899: OSPFv3: DR/BDR election on FastEthernet0/0
*Mar  1 01:11:34.903: OSPFv3: Elect BDR 255.1.1.1
*Mar  1 01:11:34.903: OSPFv3: Elect DR 2.2.2.2
*Mar  1 01:11:34.903:         DR: 2.2.2.2 (Id)    BDR: 255.1.1.1 (Id)
R2#
```

Modelo de script para la suplantación del servidor/cliente

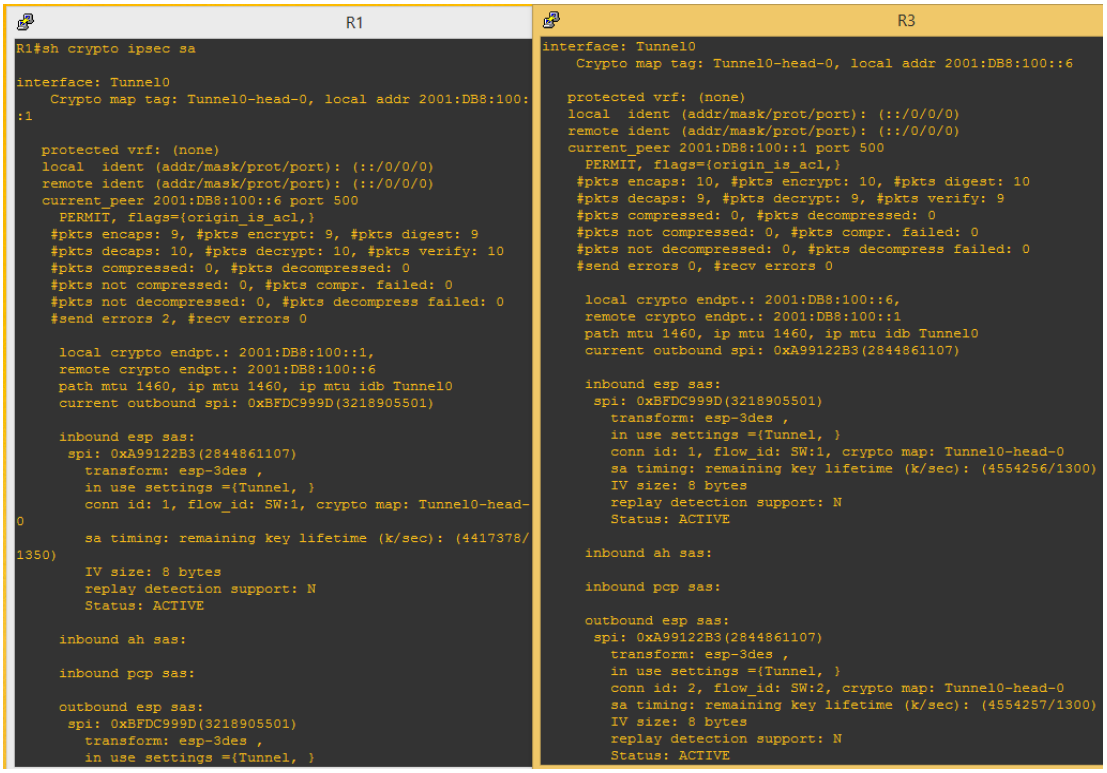
```
#!/bin/bash


sed -i "23carea 0.0.0.0 range 2001:db8:100:1::/64" /etc/quagga/ospf6d.conf

sed -i "24cipv6 address 2001:db8:100:1::2/64" /etc/quagga/zebra.conf


ifconfig eth2 inet 6 del 2001:db8:100:3::2/64

ifconfig eth2 inet 6 add 2001:db8:100:1::2/64


/etc/init.d/quagga restart
```

Script para ver el último registro capturado

```
#!/bin/sh

for i in 1 2 3 4 5 6 7 8 9

do mysql -hlocalhost -uroot -pdaniel -Dpersona -s -e "select * from contacto where id = (select max(id) from contacto)"

sleep 5

done
```

Algunos comandos ejecutados en la consola de los enrutadores para observar el estado de la vpn

show crypto ipsec sa – Muestra la configuración usada por los SAs actuales en IPV6



R1

```
R1#sh crypto ipsec sa

interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 2001:DB8:100:
:1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (::/0/0/0)
   remote ident (addr/mask/prot/port): (::/0/0/0)
   current_peer 2001:DB8:100::6 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 9, #pkts encrypt: 9, #pkts digest: 9
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify: 10
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 2, #recv errors 0

     local crypto endpt.: 2001:DB8:100::1,
     remote crypto endpt.: 2001:DB8:100::6
     path mtu 1460, ip mtu 1460, ip mtu idb Tunnel0
     current outbound spi: 0xBFDC999D(3218905501)

     inbound esp sas:
      spi: 0xA99122B3(2844861107)
        transform: esp-3des ,
        in use settings ={Tunnel, }
        conn id: 1, flow_id: SW:1, crypto map: Tunnel0-head-
0
        sa timing: remaining key lifetime (k/sec): (4417378/
1350)
        IV size: 8 bytes
        replay detection support: N
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xBFDC999D(3218905501)
        transform: esp-3des ,
        in use settings ={Tunnel, }
```

R3

```
interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 2001:DB8:100::6

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (::/0/0/0)
   remote ident (addr/mask/prot/port): (::/0/0/0)
   current_peer 2001:DB8:100::1 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 10, #pkts encrypt: 10, #pkts digest: 10
    #pkts decaps: 9, #pkts decrypt: 9, #pkts verify: 9
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 2001:DB8:100::6,
     remote crypto endpt.: 2001:DB8:100::1
     path mtu 1460, ip mtu 1460, ip mtu idb Tunnel0
     current outbound spi: 0xA99122B3(2844861107)

     inbound esp sas:
      spi: 0xBFDC999D(3218905501)
        transform: esp-3des ,
        in use settings ={Tunnel, }
        conn id: 1, flow_id: SW:1, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4554256/1300)
        IV size: 8 bytes
        replay detection support: N
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xA99122B3(2844861107)
        transform: esp-3des ,
        in use settings ={Tunnel, }
        conn id: 2, flow_id: SW:2, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4554257/1300)
        IV size: 8 bytes
        replay detection support: N
        Status: ACTIVE
```

show crypto engine connections active - Muestra un sumario de información de configuración de los motores de criptografía

```
R1#sh crypto engine connections active
Crypto Engine Connections

   ID Interface  Type  Algorithm           Encrypt  Decrypt
    1 Se0/1      IPsec 3DES                    0       10
    2 Se0/1      IPsec 3DES                    9        0
 1001 Se0/1      IKE   MD5+3DES                0        0

R1#
R1#
```

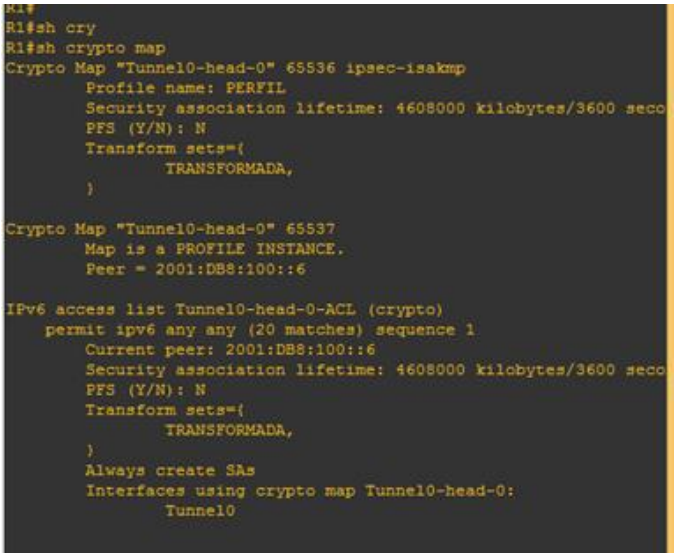
```
R3#sh crypto engine connections active
Crypto Engine Connections

   ID Interface  Type  Algorithm           Encrypt  Decrypt IP-Address
    1 Tu0        IPsec 3DES                    0        9 2001:DB8:100::6
    2 Tu0        IPsec 3DES                   10        0 2001:DB8:100::6
 1001 Tu0        IKE   MD5+3DES                0        0 2001:DB8:100::6
```

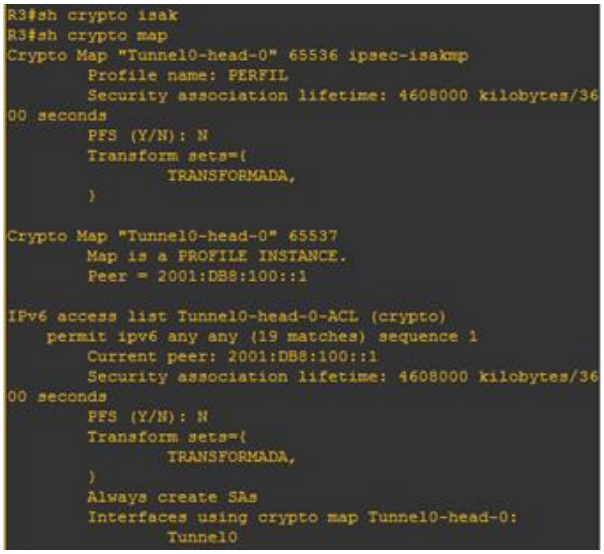
show crypto isakmp policy – Muestra los parámetros por cada política de IKE

```
Global IKE policy
Protection suite of priority 1
        encryption algorithm:   Three key triple DES
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
R1#
R1#
```

```
R3#sh crypto isakmp policy

Global IKE policy
Protection suite of priority 1
        encryption algorithm:   Three key triple DES
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
```

Show crypto map – Muestra la configuración del mapa de criptografía. Los mapas de criptografía mostrados en esta salida de comando son generados dinámicamente. El usuario no tiene que estar configurando los mapas de criptografía.





ANEXO H

COMANDOS DE CONFIGURACION EN LOS ENRUTADORES

OSPFv3

R1>enable

R1#configure terminal

R1(config)#ipv6 unicast-routing

R1(config)#interface f0/0

R1(config-if)#ipv6 address 2001:db8:100:1::1/64

R1(config-if)# ipv6 ospf 1 area 0

R1(config-if)#no shutdown

R1(config)#interface s0/0

R1(config-if)#ipv6 address 2001:db8:100::1/126

R1(config-if)# ipv6 ospf 1 area 0

R1(config-if)#no shutdown

R1(config-if)# interface s0/1

R1(config-if)#ipv6 address 2001:db8:100::9/126

R1(config-if)#ipv6 ospf 1 area 0

R1(config-if)#exit

R1(config)#ipv6 router ospf 1

R1(config-rtr)# router-id 1.1.1.1

```
R2>enable
R2#configure terminal
R2(config)#ipv6 unicast-routing
R2(config)#interface f0/0
R2(config-if)#ipv6 address 2001:db8:100:4::1/64
R2(config-if)# ipv6 ospf 1 area 0
R2(config-if)#no shutdown
R2(config)#interface s0/0
R2(config-if)#ipv6 address 2001:db8:100::5/126
R2(config-if)# ipv6 ospf 1 area 0
R2(config-if)#no shutdown
R2(config-if)# interface s0/1
R2(config-if)#ipv6 address 2001:db8:100::2/126
R2(config-if)#ipv6 ospf 1 area 0
R2(config-if)#exit
R2(config)#ipv6 router ospf 1
R2(config-rtr)# router-id 2.2.2.2

R3>enable
R3#configure terminal
R3(config)#ipv6 unicast-routing
R3(config)#interface f0/0
```

R3(config-if)#ipv6 address 2001:db8:100:3::1/64

R3(config-if)# ipv6 ospf 1 area 0

R3(config-if)#no shutdown

R3(config)#interface s0/0

R3(config-if)#ipv6 address 2001:db8:100::6/126

R3(config-if)# ipv6 ospf 1 area 0

R3(config-if)#no shutdown

R3(config-if)# interface s0/1

R3(config-if)#ipv6 address 2001:db8:100::A/126

R3(config-if)#ipv6 ospf 1 area 0

R3(config-if)#exit

R3(config)#ipv6 router ospf 1

R3(config-rtr)# router-id 3.3.3.3

RIPng

R1(config)#ipv6 unicast-routing

R1(config)#Ipv6 router rip RIP1

R1(config)#interface f0/0

R1(config-if)#ipv6 rip RIP1 enable

R1(config-if)#ipv6 address 2001:db8:100:1::1/64

R1(config-if)#no shutdown

R1(config)#int s0/0

```
R1(config-if)#ipv6 rip RIP1 enable
R1(config-if)#ipv6 address 2001:db8:100::1/126
R1(config-if)#no shutdown
R1(config)#int s0/1
R1(config-if)#ipv6 rip RIP1 enable
R1(config-if)#ipv6 address 2001:db8:100::9/126
R1(config-if)#no shutdown
R1(config-if)#exit

R2(config)#ipv6 unicast-routing
R2(config)#Ipv6 router rip RIP1
R2(config)#interface f0/0
R2(config-if)#ipv6 rip RIP1 enable
R2(config-if)#ipv6 address 2001:db8:100:4::1/64
R2(config-if)#no shutdown
R2(config)#int s0/0
R2(config-if)#ipv6 rip RIP1 enable
R2(config-if)#ipv6 address 2001:db8:100::5/126
R2(config-if)#no shutdown
R2(config)#int s0/1
R2(config-if)#ipv6 rip RIP1 enable
R2(config-if)#ipv6 address 2001:db8:100::2/126
```

```
R2(config-if)#no shutdown
R2(config-if)#exit

R3(config)#ipv6 unicast-routing
R3(config)#Ipv6 router rip RIP1
R3(config)#interface f0/0
R3(config-if)#ipv6 rip RIP1 enable
R3(config-if)#ipv6 address 2001:db8:100:3::1/64
R3(config-if)#no shutdown
R3(config)#int s0/0
R3(config-if)#ipv6 rip RIP1 enable
R3(config-if)#ipv6 address 2001:db8:100::6/126
R3(config-if)#no shutdown
R3(config)#int s0/1
R3(config-if)#ipv6 rip RIP1 enable
R3(config-if)#ipv6 address 2001:db8:100::A/126
R3(config-if)#no shutdown
R3(config-if)#exit
```

ANEXO I

IMPLEMENTACIÓN DE IPSEC EN LOS ENRUTADORES DE LAS SUCURSALES PARA LAS PRUEBAS REALIZADAS SOBRE UN TÚNEL VPN.

Tanto en R1 como en R3 configuramos una VPN-IPSEC al ingresar los siguientes comandos:

COMANDO	EXPLICACION
crypto isakmp policy 1	Empezamos la configuración de una política IKE con prioridad 1
authentication pre-share	Establecemos como modo de autenticación una clave precompartida
hash md5	Establecemos md5 como algoritmo de hash para garantizar la integridad
group 1	Especificamos el identificador de grupo de Diffie-Hellman en la política IKE
encryption 3des	Especificamos 3DES **como algoritmo de cifrado**
lifetime 86400	Especificamos el tiempo de vida en segundos para la Asociación de Seguridad, SA, es opcional
crypto isakmp key 0 cisco address ipv6 2001:db8:100: :6 /126 (en R1) crypto isakmp key 0 cisco address ipv6 2001:db8:100: :1 /126 (en R3)	Definimos la que será la clave precompartida, “cisco” , en texto plano, “0”, y la IP del que será el otro extremo del túnel en formato IPv6
crypto keyring ANILLO	Definimos el nombre del Keyring que se usará durante la autenticación
pre-shared-key address ipv6 2001:db8:100: :6 /126 key cisco (en R1)	Definimos la clave precompartida a usar durante la autenticación IKE

pre-shared-key address ipv6 2001:db8:100: :1 /126 key cisco (en R3)	
crypto ipsec transform-set TRANSFORMADA esp-3des	Definimos un transform-set, es decir, una combinación de protocolos y algoritmos que sea aceptable por enrutadores IPSec
crypto ipsec profile PERFIL	Define los parámetros que se van a usar para el cifrado IPSec entre los dos enrutadores
set transform-set TRANSFORMADA	Especifica el transform-set que se puede usar
interface tunnel 0	Empezamos la configuración de la interfaz virtual “tunnel 0”
ipv6 address 2001:DB8:100:100::1/64 (en R1) ipv6 address 2001:DB8:100:100::6/64 (en R3) ipv6 enable	Designamos direcciones ipv6 al “tunnel 0”
tunnel source 2001:db8:100: :1 (en R1) tunnel source 2001:db8:100: :6 (en R3)	Definimos el origen del túnel, en algunas IOS también podemos poner “tunnel source serial 0/0”)
tunnel destination 2001:db8:100: :6 (en R1) tunnel destination 2001:db8:100: :1 (en R3)	Definimos el destino del túnel
tunnel mode ipsec ipv6	Establecemos el modo de encapsulamiento para la interfaz tunnel 0
tunnel protection ipsec profile PERFIL	Asociamos la interfaz tunnel 0 con el perfil creado anteriormente
ipv6 route 2001:DB8:100:3::/64 tunnel 0 (en R1) ipv6 route 2001:DB8:100:1::/64 tunnel 0 (en R3)	Configuramos una ruta estática de forma que todo el tráfico que vaya de la red local de sucursal 1 hacia sucursal 2 y viceversa pase por el túnel.

ANEXO J

MUESTRAS DE CADA PRUEBA REALIZADA

Teniendo como base un paquete de un millón de bytes (1.000.000 bytes (976 KB) en las tres primeras pruebas de Disponibilidad, tanto en OSPF como en RIPNG.

Sin ataque en OSPF

Tiempo transmision (seg)	Velocidad transmision (KB/s)
8,053	121,197
7,906	123,45
7,833	124,601
8,152	119,725
7,974	122,397
7,811	124,951
7,905	123,466
7,999	122,015
7,871	123,999
8,197	119,067
7,726	126,326
7,749	125,951
8,054	121,182
8,013	121,802
8,412	116,024
8,206	118,937
7,955	122,69
7,738	126,13
8,053	121,197
7,769	125,627
8,232	118,561
7,785	125,369
8,088	120,672

7,775	125,53
8,36	116,746
7,705	126,67
7,779	125,465
7,889	123,716
7,703	126,703
7,768	125,643
7,716	126,49
7,953	122,72
7,789	125,304
7,923	123,185
7,75	125,935
7,675	127,166
7,958	122,643
7,98	122,305
7,681	127,066
7,945	122,844
8,086	120,702
8,106	120,405
7,996	122,061
7,752	125,903
8,009	121,863
7,879	123,874
7,856	124,236

Con ataque flood en OSPF

Tiempo transmision (seg)	Velocidad transmision (KB/s)
58,897	16,571
44,072	22,146
89,52	10,903
63,351	15,406
82,47	11,835
87,472	11,158
150,63	6,479
91,228	10,698

63,49	15,372
125,824	7,757
128,87	7,574
67,057	14,555
135,34	7,211
132,09	7,389
142,39	6,854
88,41	11,039
58,654	16,640
100,531	9,708
83,32	11,714
135,432	7,207
66,153	14,754
65,09	14,995
134,811	7,240
177,41	5,501
133,26	7,324
163,91	5,954
97,137	10,048
124,77	7,822
71,04	13,739
134,88	7,236
72,02	13,552
128,45	7,598
86,74	11,252
165,27	5,905
162,3	6,014
144,102	6,773
79,73	12,241
131,15	7,442
80,46	12,130
80,217	12,167
129,58	7,532
145,64	6,701
134,08	7,279
88,27	11,057
162,34	6,012
75,78	12,879
78,56	12,424

Convergencia vpn en OSPF (al recuperarse de una caída del servicio)

Tiempo (seg)
9,453
9,487
9,826
8,964
9,347
9,002
9,269
9,017
9,035
8,984
9,2531
24,23
24,551
12,804
28,128
12,458
10,423
10,037
19,106
11,858
17,001
22,89
18,611
9,36
10,408
9,314
10,085
13,604
9,204
9,712
9,45
13,001
12,697
11,22
13,706

13,27
8,95
9,006
8,711
9,81
9,43
9,302
9,002
9,501
9,19
9,5
9,525

Tiempo de convergencia al cambiar rutas con Quagga sobre OSPF

TIEMPO DE CONVERGENCIA (INTEGRIDAD)
10,032
10,020
10,000
10,004
10,008
10,016
10,020
10,042
10,120
10,034
10,002
10,041
10,019
10,022
10,015
10,101
10,084
10,045

10,022
10,023
10,024
10,004
10,123
10,134
10,034
10,009
10,067
10,021
10,024
10,017
10,109
10,082
10,046
10,029
10,022
10,020
10,012
10,187
10,087
10,033
10,012
10,066
10,022
10,033
10,012
10,066
10,022

Sin ataque en RIPng

Tiempo de transmision (seg)	Velocidad de transmision (KB/s)
7,726	126,327
7,699	126,770
8,177	119,359

7,656	127,482
7,657	127,465
8,266	118,074
7,851	124,315
8,323	117,265
7,98	122,306
7,636	127,816
7,887	123,748
8,015	121,772
8,08	120,792
8,152	119,725
7,949	122,783
7,975	122,382
8,083	120,747
7,897	123,591
7,608	128,286
8,216	118,793
7,945	122,845
8,046	121,303
8,162	119,579
8,214	118,822
8,312	117,421
8,258	118,188
7,892	123,670
7,764	125,708
7,849	124,347
7,743	126,049
9,205	106,029
7,775	125,531
7,71	126,589
7,883	123,811
8,022	121,665
8,11	120,345
7,733	126,212
7,936	122,984
8,087	120,688
8,119	120,212
8,129	120,064
8,183	119,272

8,033	121,499
7,861	124,157
8,463	115,326
8,657	112,741
8,274	117,960

Con ataque flood en RIPng

Tiempo de transmision (seg)	Velocidad de transmision (KB/s)
87,393	11,168
195,389	4,995
158,046	6,175
144,014	6,777
83,690	11,662
77,020	12,672
72,262	13,506
89,530	10,901
140,797	6,932
102,749	9,499
151,592	6,438
114,881	8,496
146,869	6,645
119,016	8,201
153,584	6,355
153,371	6,364
145,849	6,692
151,323	6,450
150,532	6,484
148,703	6,563
152,051	6,419
149,803	6,515
148,225	6,585
94,301	10,350
145,943	6,688
147,757	6,605

146,983	6,640
149,254	6,539
144,766	6,742
95,302	10,241
94,497	10,328
88,433	11,037
188,756	5,171
151,273	6,452
149,387	6,533
144,496	6,755
148,220	6,585
145,358	6,714
151,397	6,447
122,335	7,978
146,554	6,660
171,793	5,681
152,341	6,407
93,271	10,464
143,298	6,811
95,244	10,247
146,991	6,640

Convergencia vpn en RIPng (al recuperarse de una caída del servicio)

Tiempo (seg)
18.664
17.716
14.264
21.252
20.932
13.356
17.692
20.192
12.756
14.044

11.844
31.824
22.088
14.9
28.444
16.18
11.896
10.192
16.884
13.796
19.508
19.232
21.056
13.856
19.388
21.358
17.353
14.95
19.381
17.215
27.522
21.354
15.2
19.778
21.749
17.327
19.325
20.252
19.938
14.5
16.963
19.219
11.488
10.195
19.538
16.882
13.947

Tiempo de convergencia al cambiar rutas con Quagga sobre RIPng

TIEMPO DE CONVERGENCIA (INTEGRIDAD)
10,936
16,172
14,152
15,092
8,368
8,928
11,616
13,484
8,136
13,592
11,696
11,14
8,724
10,74
8,404
11,284
15,528
14,028
8,864
13,5
10,876
16,245
14,098
15,956
8,334
8,112
11,623
14,484
8,139
13,991
11,743
11,15
8,726
10,745

8,409
11,287
15,527
14,029
8,865
13,555
8,726
10,745
8,409
11,287
15,527
14,029
10,089