UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

T E S I S

ANÁLISIS Y PRUEBAS DEL SOPORTE DE IPSec CON IPv6

QUE PARA OBTENER EL TÍTULO DE

I N G E N I E R O E N C O M P U T A C I Ó N

P R E S E N T A:

JOSÉ LUIS MENDOZA PIÑEIRO

DIRECTOR
ING. AZAEL FERNÁNDEZ ALCÁNTARA

MÉXICO, D.F. ABRIL 2008

Agradecimientos

A la Dirección General de Servicios de Cómputo Académico (DGSCA) UNAM quien me abrió sus puertas para la realización de este tema, gracias por aceptarme como estudiante y apoyarme para alcanzar esta meta.

A mi director de tesis y amigo Ing. Azael Fernández Alcántara que solo con su apoyo y paciencia pude haber terminado este trabajo ¡Muchas gracias!

A mis compañeros y amigos del Netlab-DTD DGSCA UNAM con quienes compartí momentos de felicidad, diversión, entretenimiento, intercambio de conocimientos, además de su apoyo en la contribución y enriquecimiento del trabajo.

A todos mis amigos y personas que quiero, que me apoyaron y me alentaron a seguir adelante. ¡Muchas Gracias!

Dedicatorias

Este trabajo se lo dedico con todo mi corazón, admiración y respeto a las personas que siempre han estado a mi lado en todo momento.

A mi mamá, una señora extraordinaria que siempre ha estado conmigo apoyándome y enseñándome los verdaderos valores de la vida, así como la alegría de vivir. ¡Te quiero mucho mami!

A mi papá, que me ha apoyado en toda mi educación con un gran esfuerzo y me ha enseñado que la responsabilidad y dedicación es una de las principales armas para una mejor superación personal. ¡Te quiero mucho papi!

A mi hermana, gracias por todo tu cariño y comprensión, y recuerda que hay que seguir adelante en los momentos bueno y no tan buenos. ¡Te quiero mucho min!

A Isita, tengo toda la vida por delante para darte las gracias por estar conmigo y hacerme tan feliz, no hay palabras para decir lo mucho que te quiero y todo lo que siento por ti. Gracias por todo el apoyo, consejos y regaños. ¡Te quiero mucho flaquis!

A Alejandro Ordoñez, casi mi hermanito, que solo le quiero decir que en esta vida casi todo se puede, solo hay que tener mucha dedicación y compromiso. ¡Échale ganas!

Contenido

Índice de Figuras

Índice de Tablas

Introducción

La red Internet utiliza la familia de protocolos denominada TCP/IP, los cuales han gobernado, y seguramente seguirán haciéndolo, el funcionamiento de Internet, sin embargo, éstos han ido sufriendo cambios posteriores a sus definiciones originales para corregir imperfecciones o ajustarse a las necesidades actuales, de tal forma que el protocolo de Internet IP (creado en los años 70) que se designó IPv4 (IP versión 4), y como consecuencia del crecimiento de Internet a nivel mundial y el surgimiento de nuevas tecnologías, implicó el desarrollo de una nueva versión en 1996 denominada IPv6 (IP versión 6) o IPng (IP next generation) como se nombró en un principio.

El motivo principal por el que surge la necesidad de crear una nueva versión fue el evidente agotamiento de direcciones, mientras IPv4 ofrece teóricamente 4 mil millones de direcciones, IPv6 ofrece teóricamente 340 trillones de trillones de direcciones, con más del 70% de las direcciones IPv4 en uso asignadas, ya que los creadores de IPv4 no predijeron en ningún momento el gran éxito que este protocolo iba a tener en muy poco tiempo. Además en un principio no se tomó en cuenta la gran cantidad de aplicaciones, mecanismos y servicios que hoy en día se utilizan como son calidad de servicio, seguridad, movilidad, etcétera, para transporte de paquetes de voz, datos y video, que requirieron añadidos o *parches* en IPv4 para su correcto funcionamiento.

En lo que respecta a la seguridad informática, uno de los principales aspectos que hay que tomar en cuenta para el desarrollo de las redes IP, es el hecho de tener diversas implementaciones que dan protección a cada nivel de la pila TCP/IP con la finalidad de asegurar la información que se esté intercambiando entre las partes autorizadas, proporcionando servicios de seguridad como confidencialidad, integridad y autenticación, principalmente.

Cada implementación propuesta deberá contar con: algoritmos apropiados para llevar a cabo el cifrado y/o la autenticación; un correcto funcionamiento; así como una buena administración de llaves o claves en caso de ser necesario, que serán aspectos fundamentales para que puedan ser considerados como soluciones confiables.

Existen diferentes soluciones comerciales, que gracias al gran esfuerzo que han hecho sus desarrolladores, han tenido mucho éxito proporcionando distintos grados de seguridad dependiendo del nivel para donde fueron creados como SSH y S-HTTP a nivel de aplicación o SSL y TLS a nivel transporte, entre otros. Sin embargo, era conveniente una protección que se realizara a nivel de red y que ofreciera los servicios de seguridad mencionados, creándose así el protocolo IPSec que provee un nivel de seguridad común y homogéneo para todas las aplicaciones, además de ser independiente de la tecnología física empleada.
En el caso de IPv4, IPSec es utilizado de manera opcional, mientras que para IPv6 se encuentra soportado nativamente para obtener una seguridad robusta, pudiéndose utilizar o no, mediante 2 encabezados de extensión.

En IPv6 la forma en la que opera IPSec es con ayuda del encabezado para autenticación AH, usando algoritmos como MD5, SHA-1 y HMAC; y el encabezado para cifrado ESP utilizando algoritmos como DES y AES. Además, proporciona un método para crear las asociaciones de seguridad (parámetros para establecer una comunicación entre dos entidades) de forma dinámica entre quienes interactúan por medio de IKE e ISAKMP, usando un protocolo de intercambio de llaves Diffie-Hellman. Por otra parte, IPSec ha sido diseñado para funcionar en dos modos distintos dependiendo de lo que se quiera proteger: modo transporte para proteger la carga útil IP y modo túnel para proteger los paquetes IP.

Aunque IPSec ofrece una seguridad muy completa a nivel de red, también existen diversos mecanismos y protocolos para dar seguridad en diferentes servicios, aplicaciones y/o tecnologías pudiendo ser implementados en IPv4 o IPv6 dependiendo del soporte que se tenga para cada uno, por ejemplo, el uso del protocolo SEND para proporcionar seguridad en el descubrimiento de vecinos, DNSsec para proporcionar mejores mecanismos de autenticación, el uso de firewalls, filtros, listas de acceso, etcétera.

Si bien, las implementaciones de IPSec en IPv6 no se han estandarizado completamente, existen distintos esquemas donde se puede configurar IPSec para proporcionar a los usuarios los mejores beneficios y la mayor seguridad posible.

En este trabajo, se describen los protocolos de seguridad normalizados, su funcionalidad y aplicación. Se menciona el funcionamiento de IPv6; las características que presenta en relación con IPv4; los mecanismos de transición propuestos para llevar a cabo el paso de IPv4 a IPv6; y los métodos de seguridad que se siguen implementando en esta nueva versión, como IPSec, mencionando sus características, funcionamiento, beneficios y ventajas que presentan. Por último, se presentan escenarios de experimentación utilizando IPSec, donde se evaluaron diferentes configuraciones sobre distintas plataformas, aplicaciones y servicios, observando su comportamiento por medio de analizadores de tráfico, y con esto proporcionar los resultados y conclusiones sobre el uso de IPSec en IPv6.

Capítulo 1

Seguridad Informática

I.1 Introducción

La seguridad ha sido un elemento que tomó relevancia en la última década en los mundos de la computación y las comunicaciones, esto como un efecto natural en los ambientes que crecen de forma exponencial y que pierden confianza entre quienes interactúan.

Conviene mencionar que el Internet, surgido en el medio académico y siendo restringido por algunos años, actualmente es un ambiente más complejo y numeroso donde interactúan más de 60 millones de computadoras conectadas, más de 200 países de todos los continentes, y millones de usuarios de diferentes culturas con diferentes propósitos, inquietudes, edades, etcétera, trayendo como consecuencia un aumento exponencial de incidentes de seguridad informática. Por ejemplo, por estadísticas del CERT[1] (Computer Emergency Response Team) en el año 2003 se presentó un incremento de 55,435 incidentes de seguridad en computadoras, es decir, un aumento del 67.52% de incidentes con respecto al año 2002.

[1] Es un equipo de respuesta a incidentes de seguridad en cómputo que tiene por objetivo tomar las medidas oportunas de prevención, así como investigar y mejorar la seguridad de los sistemas.

La seguridad no fue un aspecto considerado en el diseño inicial de los protocolos de comunicación y sistemas operativos, lo que ha ocasionado oportunidades diversas para acceder de forma no autorizada a redes y sistemas que ha dado lugar a una de las actividades más populares en estos días: la intrusión de sistemas.

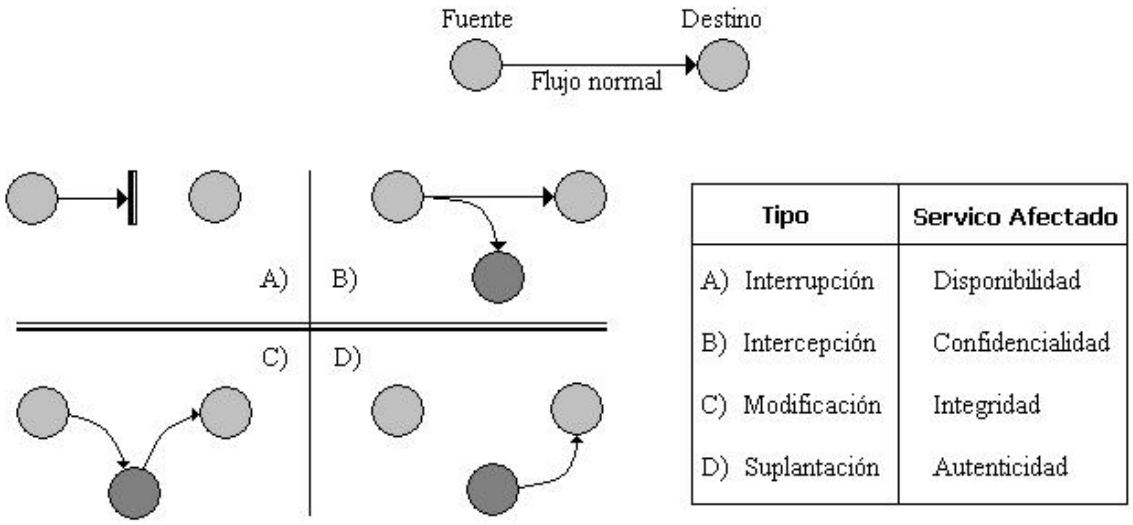
En los últimos años se ha desarrollado una gran variedad de herramientas, metodologías y técnicas para contrarrestar estas deficiencias de origen y proteger a los sistemas de intrusos, sobre todo a nivel de aplicaciones del usuario o del administrador.

La seguridad de la información tiene como finalidad la prevención y protección a través de ciertos mecanismos para evitar que ocurra de manera accidental o intencional la transferencia, fusión, modificación o destrucción **no** autorizada de la información.

Dentro del área de seguridad, se manejan diversos términos para identificar los factores que intervienen en la intrusión de sistemas, siendo los conceptos principales: vulnerabilidades, ataques, contramedidas y amenazas.

Vulnerabilidades: El software es desarrollado por humanos, quienes modelan e implementan programas con base en su criterio, concepto y conocimiento del lenguaje de programación que utilizan, en consecuencia, es posible encontrar imperfecciones en los sistemas, siendo éstas las que propician oportunidades para accesos no autorizados y las que se conocen como vulnerabilidades de los sistemas, es decir, se trata de las debilidades o aspectos atacables en un sistema o un punto del sistema que es susceptible de ser atacado o de dañar la seguridad del mismo.

Amenaza: Se puede definir como todo aquello que intenta, puede o pretende destruir. Entre los factores pueden estar los desastres naturales, fallas de hardware, fallas de software, códigos maliciosos, y el factor humano, donde adversarios motivados y capaces de montar ataques que exploten vulnerabilidades, podrían producir una violación a la seguridad (disponibilidad, confidencialidad, integridad y autenticidad). En la figura I.1 se muestran los diferentes tipos de amenazas y los servicios afectados.



Tipo	Servico Afectado
A) Interrupción	Disponibilidad
B) Intercepción	Confidencialidad
C) Modificación	Integridad
D) Suplantación	Autenticidad

Figura I.1 Tipos de amenazas y servicios afectados

Ataques: Se trata de la culminación de una amenaza cuando se explotan las vulnerabilidades. Se identifican dos tipos de ataques: extracción pasiva y extracción activa.

En la extracción pasiva el atacante sólo escucha sin modificar el mensaje o afectar la operación de la red, teniendo como objetivo la intercepción y el análisis de tráfico en la red. Generalmente no puede detectarse este tipo de ataque, pero se puede prevenir mediante mecanismos como el cifrado de información.

En la extracción activa el atacante modifica los mensajes o irrumpe la operación de la red con la finalidad de modificar los datos o bien crear tráfico falso. Este tipo de ataque por lo general puede detectarse pero no prevenirse.

Contramedidas: Lo más importante es contar con una política de seguridad, un documento legal y con apoyo directivo, que defina la misión, visión y objetivos de los recursos de red e información en cuestión. En una política se define lo que está o no permitido, así como los servicios de seguridad que se van a ofrecer para los recursos involucrados. Las contramedidas son entonces, las políticas de seguridad apoyadas por todos los medios técnicos o de procedimientos que se aplican y desarrollan para atender vulnerabilidades y frustrar ataques específicos.

I.2 Servicios de seguridad

Los servicios de seguridad son la respuesta a las amenazas, y responden al qué se debe hacer para satisfacer los requerimientos de seguridad de la organización y hacer frente a las amenazas.

La definición del estándar ISO[2] 7498-2 define cinco elementos básicos que constituyen la seguridad de un sistema: la confidencialidad de los datos, la autenticación de los datos, la integridad de los datos, el control de acceso (disponibilidad) y el no repudio.

La confidencialidad implica que la información sea accesible únicamente por las entidades, sistemas o personas autorizadas. La autenticación define mecanismos para garantizar la procedencia de la información. La integridad implica que los datos no han sido modificados o corrompidos de manera alguna desde su

[2] Es una organización no gubernamental encargada de producir normas internacionales industriales y comerciales con la finalidad de facilitar el comercio y el intercambio de información.

transmisión hasta su recepción, garantizando que el mensaje recibido es exactamente el mismo que se envió. El control de acceso establece la forma en que el recurso está disponible cuando es requerido, y garantiza que sólo accedan a la información y/o recursos los usuarios que tienen permiso para ello. El no repudio es la garantía de transmisión y recepción de información, busca proteger al emisor de que el receptor niegue haber recibido el mensaje, y proteger al receptor de que el transmisor niegue haber enviado el mensaje.

I.3 Criptografía

La criptología viene del griego kryptos (oculto) y logos (tratado o ciencia), es decir, es una ciencia encargada de ocultar, y cuyo ámbito es el estudio de los criptosistemas: sistemas que ofrecen medios seguros de comunicación en los que el emisor oculta o cifra el mensaje antes de transmitirlo para que sólo un receptor autorizado pueda descifrarlo. Se considera como un nombre genérico para dos disciplinas opuestas y complementarias: la ***criptografía*** y el ***criptoanálisis***, pudiéndose incluir la ***esteganografía*** como parte de esta ciencia aplicada.
Actualmente, una de las aplicaciones más extendidas estudiadas por la criptología es la autenticación de información digital (también llamada ***firma digital***).

La *criptografía* (del griego kryptos "ocultar" y grafos "escribir", literalmente "escritura oculta") es el arte o ciencia de cifrar y descifrar información utilizando técnicas matemáticas que hagan posible el intercambio de mensajes de manera que sólo puedan ser leídos por las personas a quienes van dirigidos. El *criptoanálisis* es una ciencia que trata de encontrar la información mediante un análisis de manera ilegítima o no autorizada.

La *esteganografía* (del griego stegos “cubierta”) es una rama de la criptografía que trata sobre la ocultación de mensajes en lugar de su contenido para evitar que se

perciba la existencia de los mismos, por lo general un mensaje de este tipo parece ser otra cosa, como una lista de compras, un artículo, una foto, etc.

La *firma digital* de un documento es el resultado de aplicar cierto algoritmo matemático, denominado función hash, a su contenido. Esta función asocia un valor dentro de un conjunto finito (generalmente los números naturales) a su entrada, por ejemplo, en caso de que la entrada sea un documento el resultado de la función es un número que identifica casi unívocamente al texto, además si se adjunta este número al texto, el destinatario puede aplicar de nuevo la función y comprobar su resultado con el que ha recibido.

En la criptografía, la información original que debe protegerse se *denomina texto plano*. El cifrado es el proceso de convertir el texto plano en un texto ilegible, denominado *texto cifrado* o *criptograma*, haciendo uso de un algoritmo de cifrado basado en la existencia de una llave (información secreta que adapta el algoritmo de cifrado para cifrar o descifrar el mensaje).
Las dos técnicas más básicas de cifrado en la criptografía clásica son la *sustitución* (que supone el cambio de significado de los elementos básicos del mensaje como letras, dígitos o símbolos) y la *transposición* (que supone una reordenación de las mismas). El descifrado es el proceso inverso que recupera el texto plano a partir del criptograma y la llave. En la Figura I.2 se muestra un esquema criptográfico.

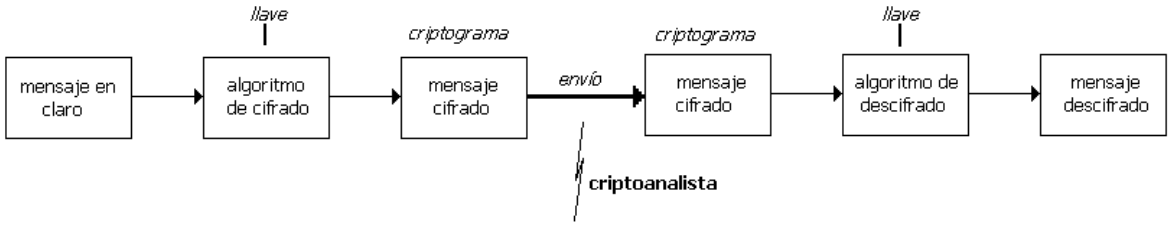


Figura I.2 Esquema criptográfico

I.3.1 Algoritmos simétricos y asimétricos

Existen dos grandes grupos de algoritmos de cifrado: los algoritmos que utilizan una única llave tanto en el proceso de cifrado como en el de descifrado denominados *simétricos*; y los que utilizan una llave para cifrar mensajes y una llave distinta para descifrarlos denominados *asimétricos* o también llamados *algoritmos de llave pública*.

En los algoritmos simétricos, Figura I.3, tanto el transmisor como el receptor deben de conocer la llave a utilizar, lo que puede causar un problema debido a que un secreto compartido no es un secreto por mucho tiempo. Un ejemplo muy sencillo es cifrar el mensaje "*Hola Mundo*" como "*Ipmb Nvñep*", desplazando las letras una posición. En este caso, la llave que ambas partes deben conocer es la cantidad de posiciones que se desplazaron las letras (no muy difícil de averiguar)[3]. Entre los algoritmos de cifrado simétrico más utilizados podemos nombrar a 3DES, Blowfish, IDEA, y CAST usando todos una longitud de llave de 128 bits, lo que significa que existen 2^{128} llaves posibles.

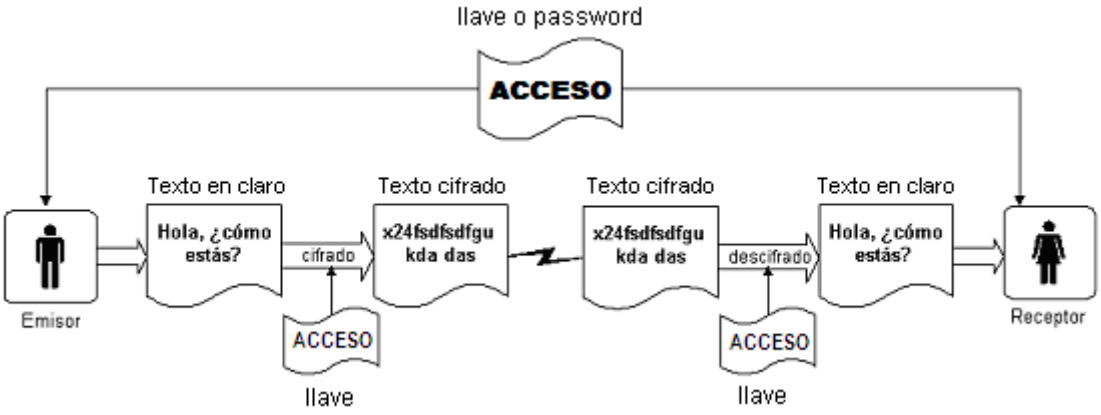


Figura I.3 Representación del funcionamiento de algoritmos simétricos

[3] Una variante de este algoritmo, conocida como ROT13 dado que desplaza las letras 13 posiciones, es muy utilizada para cifrar textos de forma simple.

En los algoritmos asimétricos, Figura I.4, una llave es pública y se puede entregar a cualquier persona, y la otra llave es privada y el propietario debe guardarla de modo que nadie tenga acceso a ella. El remitente usa la llave pública del destinatario para cifrar el mensaje, y una vez cifrado, sólo la llave privada del destinatario podrá descifrar ese mensaje. Para este tipo de algoritmos se recomienda el uso de llaves públicas de 1024 bits. Entre los algoritmos de cifrado asimétrico están Diffie-Hellman y RSA.

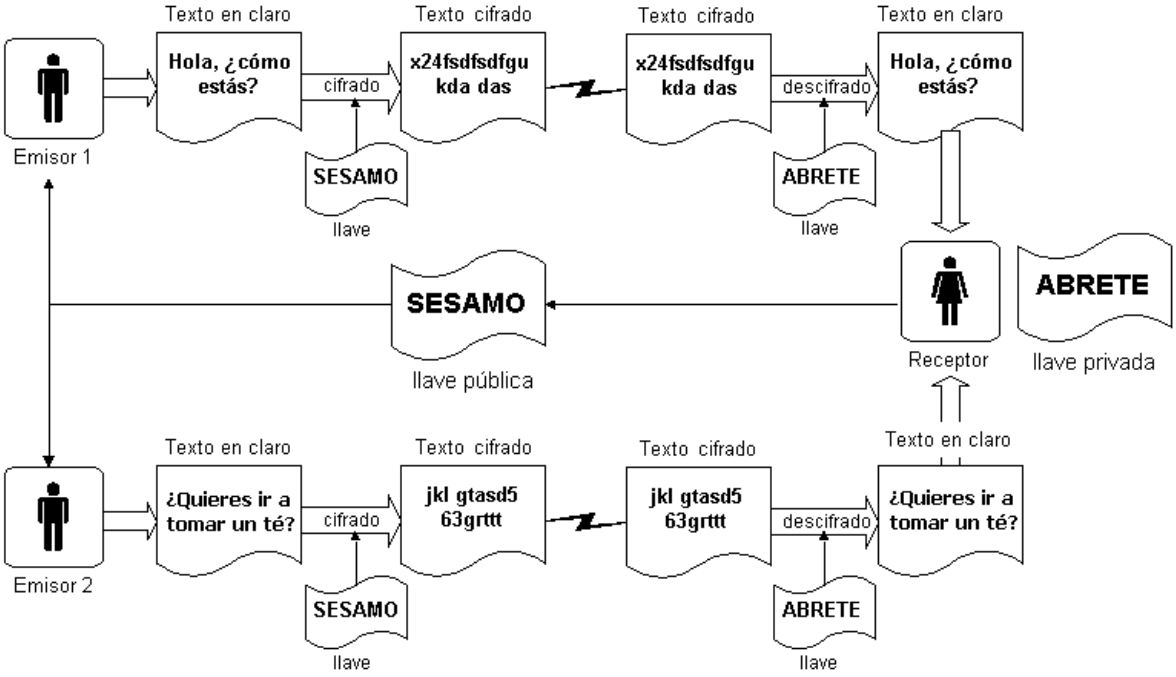


Figura I.4 Representación del funcionamiento de algoritmos asimétricos

El cifrado asimétrico aunque tiene mayor ventaja al usar dos llaves, presenta un mayor tiempo de procesamiento para una misma longitud de llave y mensaje, así como una longitud de llaves y mensajes de mayor tamaño que las simétricas haciendo muy costoso su uso, por lo cual generalmente se utiliza un esquema mixto, combinando cifrado simétrico y asimétrico, donde en una primera etapa de la comunicación emplee cifrado asimétrico para intercambiar de manera segura la llave, para que luego ésta se utilice junto con un algoritmo de cifrado simétrico para cifrar el resto de la comunicación.

En los algoritmos de cifrado también se pueden tener dos tipos de procesamiento: por bloques o por flujo. Los primeros toman bloques de tamaño fijo del texto en claro y producen un bloque de tamaño fijo de texto cifrado, generalmente del mismo tamaño que la entrada, aunque debe ser lo suficientemente grande para evitar ataques de texto cifrado; además hay que tomar en cuenta que la asignación de bloques de entrada a bloques de salida debe ser uno a uno para hacer el proceso reversible y que parezca de forma aleatoria. En el cifrado por flujo se cifra un bit (o byte) de texto en claro, pudiendo ser considerado como un cifrado por bloques de 1 bit (o byte) de tamaño. Este tipo de cifrado funciona realmente bien con datos en tiempo real como voz y video, donde en un momento sólo se conocen pequeñas partes de los datos.

I.3.2 Firma Digital

Una firma digital sirve básicamente para lo mismo que una firma realizada con una pluma o bolígrafo en un escrito, teniendo además la ventaja de probar que un documento está escrito por una determinada persona o institución, asegurando que el documento no fue alterado después de ser firmado.

El concepto de "firma digital" se basa en la criptografía de llave pública, pero el proceso es lo contrario, quien "firma" el mensaje lo cifra mediante su "llave privada", de forma que puede ser descifrado por cualquier persona siempre y cuando se posea la "llave pública" correspondiente a la llave privada que se utilizó para firmar el mensaje. Si la llave pública es capaz de descifrar un mensaje sólo puede ser por un motivo, que éste mensaje fuera cifrado por el poseedor de la llave privada, lo que autentica su identidad (además de asegurar que el mensaje original no ha sido alterado, de lo contrario no podría ser descifrado por la llave pública). En la Figura I.5 se muestra un esquema de su funcionamiento.

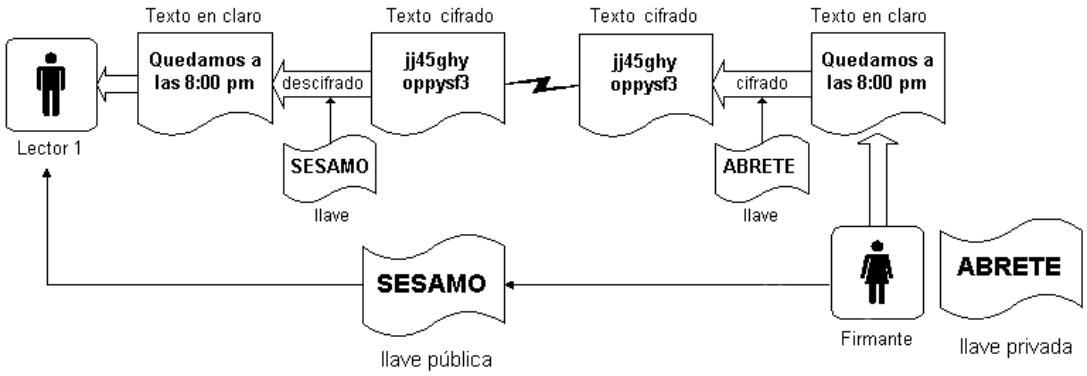


Figura I.5 Representación del funcionamiento de firmas digitales

I.4 Modelo OSI vs TCP/IP

Ante la necesidad de una norma de arquitectura de red para dar soporte a la infraestructura de las comunicaciones de los procesos distribuidos, a principios de 1977 se desarrolló el modelo OSI (Open System Interconnection).

El modelo conceptual OSI fue desarrollado por la ISO (International Standards Organization) como un modelo genérico de arquitectura de red (NO está ligado a Internet ni a ninguna red en particular), es utilizado por prácticamente la totalidad de las redes del mundo, y consta de 7 niveles o capas donde cada una de ellas define las funciones que deben proporcionar los protocolos con el propósito de intercambiar información entre diferentes sistemas.

Esta clasificación permite que cada protocolo, con propósitos de seguridad o no, sea desarrollado con una finalidad determinada, lo cual simplifica el proceso de implementación. Cada nivel depende de los que están debajo de él, y a su vez proporciona alguna funcionalidad a los niveles superiores.

Por otro lado, el modelo TCP/IP (Transmission Control Protocol / Internet Protocol), fue desarrollado inicialmente en 1973 por el informático estadounidense Vinton Cerf y adoptado como norma en 1983. Primeramente, el protocolo TCP se encarga de proveer un control de flujo de los paquetes enviados, asegurando que

lleguen a su destino de forma correcta y ordenada; y un protocolo más simple denominado IP se encarga de enviar paquetes individuales por la red hacia un nodo de destino. Para aquellas aplicaciones que no requieran un control tan estricto se diseñó el protocolo UDP (User Datagram Protocol), que al igual que TCP, utiliza los servicios del protocolo IP, pero sin dar fiabilidad.

Este modelo es usado en la práctica por su sencilla implementación, eliminando algunos niveles del modelo OSI, quedando en total 4 niveles o capas donde se evalúan las características más importantes para las comunicaciones.

En la Figura I.6 se muestra una comparación entre los dos modelos.

Figura I.6 Esquemas de comparación entre el modelo teórico OSI y la pila TCP/IP

Los protocolos que se mencionarán a continuación van a ser referidos respecto al modelo TCP/IP por tener un uso más común en la realidad de las redes.

I.5 Protocolos de seguridad en diferentes niveles de TCP/IP

Existen diversos protocolos en el ámbito de la seguridad informática, que pueden trabajar en diferentes niveles del modelo TCP/IP dependiendo de las funciones que realicen y para lo cual fueron creados. Algunos se mencionan a continuación.

I.5.1 Nivel de Aplicación

- **KERBEROS**

Kerberos es un sistema de autenticación en red desarrollado por el MIT (Massachusetts Institute of Technology) en 1983 y especificado posteriormente por la IETF[4] (Institute of Engineering Task Force), especialmente recomendado para los sistemas distribuidos proporcionando autenticidad en la información entre las entidades que participan en una comunicación.

El modelo de Kerberos está basado en un protocolo de autenticación a través de un servidor confiable, ya que este sistema considera que toda la red es una región de gran riesgo excepto por este servidor. Este servidor se denomina KDC (Kerberos Distribution Center), y provee dos servicios fundamentales: Servicio de Autenticación (Authentication Server, AS) y Servicio de Tickets (Tickets Granting Server, TGS). El primero tiene como función autenticar inicialmente a los clientes y el segundo proporcionar un ticket al cliente para que pueda acceder a algún servicio proveído por un servidor como se muestra en la Figura 1.7.

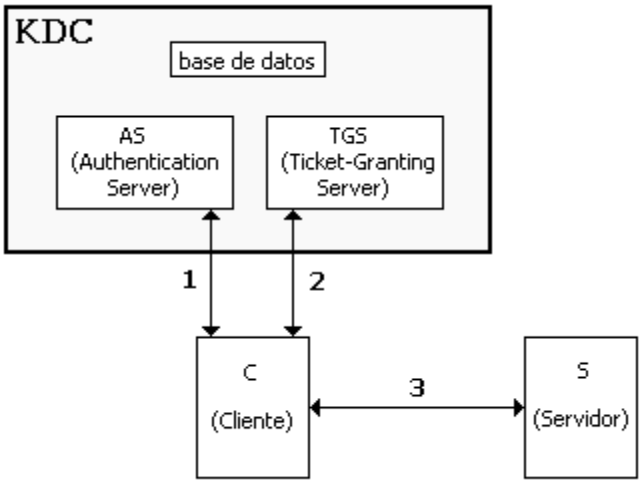


Figura I.7 Funcionamiento de Kerberos

[4] Grupo de individuos que participan en el desarrollo de las normas de Internet.

Uno de los problemas que presenta el protocolo se refiere a su implementación, donde cualquier programa que lo utilice debe ser modificado para poder funcionar correctamente siguiendo un proceso denominado "*kerberización*", implicando la disposición del código fuente de cada aplicación que se desee "*kerberizar*" y requiriendo de una inversión de tiempo considerable para aplicaciones complejas que no todas las organizaciones se pueden permitir.

En cuanto a su seguridad, para un correcto funcionamiento se debe de disponer en todo momento del servidor Kerberos, de forma que si la máquina que lo alberga falla, la red se convierte en inutilizable; esto es una contradicción con lo que nos dice la teoría de sistemas distribuidos, donde se recalca el uso de la distribución para mantener la disponibilidad del sistema, de manera que si un equipo falla el resto pueda seguir funcionando, si no a pleno rendimiento al menos correctamente. Por otro lado la seguridad de Kerberos reside en el servidor que mantiene la base de datos de las llaves, de forma que si éste se ve comprometido, la red entera estará también amenazada.

Otro problema es el timestamp (sello con la hora exacta del cliente que tiene que coincidir con la del servidor con un margen de error muy pequeño para indicar que el mensaje fue recientemente generado y evitar reenvíos) de los tickets, ya que se requiere que las máquinas donde se ejecutan los servicios autenticados mantengan sus relojes sincronizados, además que ese tiempo global debe de ser accesible en todas las estaciones y aunque en el diseño no se asume que todas mantengan la hora exacta, se les obliga a mantenerse dentro de los márgenes si desean solicitar tickets, por lo que se necesitan servidores de tiempo con los que los clientes puedan sincronizar periódicamente sus relojes.

Todos estos problemas han propiciado que el uso de Kerberos no esté muy extendido, de forma que su complejo modelo se vea sustituido por programas más simples como SSH.

- **SSH (Secure SHell)**

SSH es un programa para conectarse a otros equipos a través de la red para ejecutar comandos en una máquina remota y mover archivos de una máquina a otra, reemplazando a los comandos de acceso remoto no seguros como telnet, ftp, rlogin, resh y rcp; los cuales proporcionan gran flexibilidad en la administración de una red, pero presentan grandes riesgos de seguridad en un sistema.
Su seguridad estriba en el uso de criptografía fuerte de manera que toda la comunicación es cifrada y autenticada de forma transparente para el usuario, introduciendo varias mejoras como una autenticación más robusta de usuarios y hosts que la tradicionalmente ofrecida basada en direcciones IP y nombres de máquinas, una privacidad mayor para el usuario debido al uso de canales cifrados, y un entorno protegido contra ataques típicos como: el engaño de IP (IP spoofing), el origen de enrutamiento (IP source routing), el engaño de DNS (DNS spoofing), el fisgoneo (sniffing), X11 Server Spoofing, etc., utilizando algoritmos como IDEA, Blowfish, 3DES y RSA (ver capítulo III.6).

Después de una conexión inicial, el cliente puede verificar que se está conectando al mismo servidor durante sesiones posteriores y transmitir su información de autenticación al servidor, como el nombre de usuario y contraseña, en formato cifrado. Si el servidor usa la técnica del reenvío de puerto, los protocolos considerados como inseguros (POP, IMAP, etc.) se pueden cifrar para garantizar una comunicación segura.

SSH opera de la siguiente manera:

1. Se crea una capa de transporte segura para que el cliente sepa que está efectivamente comunicado con el servidor correcto, para después cifrar dicha comunicación entre ambos por medio de un código simétrico.
2. Con la conexión segura al servidor, el cliente se autentica ante el servidor sin preocuparse de que la información de autenticación pudiese ser vista.

3. Con el cliente autenticado ante el servidor, se pueden usar varios servicios diferentes, con seguridad a través de la conexión, como una sesión shell interactiva, aplicaciones X11 y túneles TCP/IP.

La ventaja más significativa de SSH es que no modifica mucho las rutinas. En todos los aspectos, iniciar una sesión de SSH es tan sencillo como (y similar a) iniciar una sesión de telnet, ya que tanto el intercambio de llaves, la autenticación, y el posterior cifrado de sesiones son transparentes para los usuarios.

- **PGP (Pretty Good Privacy)**

PGP es un protocolo desarrollado por Philip Zimmerman en 1991 usado para cifrar y descifrar mensajes de correo electrónico y firmas digitales, permitiendo al receptor comprobar la identidad del emisor, verificar que el mensaje no haya sido modificado, y en sus últimas versiones, permite cifrado de ficheros con llave simétrica, creación de VPNs, discos virtuales cifrados, y borrado seguro de datos; utilizando criptografía de llave pública.

Los servicios que ofrece PGP son confidencialidad, cifrado, autenticación, firmas digitales y compresión de datos usando algoritmos como RSA, IDEA, DES y MD5 (ver capítulo III.6).

En el caso de las firmas digitales, su uso involucra el uso de funciones hash y un algoritmo de cifrado de llave pública. La secuencia es como sigue: el transmisor crea el mensaje, el programa PGP genera un código hash del mensaje que después es cifrado por él mismo usando la llave privada del transmisor y añadida al mensaje, cuando el mensaje es recibido, PGP descifra el código hash usando la llave pública del transmisor y genera un nuevo código hash del mensaje recibido para después, compararlo con el código hash descifrado, si son idénticos el mensaje es aceptado como auténtico.

Para proporcionar confidencialidad PGP maneja un cifrado convencional para cifrar los mensajes a trasmitir o los datos a ser almacenados localmente, usando una llave única de 128 bits para cada mensaje. La comunicación se lleva a cabo de la siguiente manera: el transmisor crea el mensaje que al enviarlo PGP genera un número arbitrario para ser usado como una llave de sesión única para dicho mensaje, que después es cifrado con esa misma llave de sesión, después para proteger esta llave se cifra con la llave pública del receptor y se añade en el mensaje cifrado, al llegar el mensaje al receptor PGP descifra la llave de sesión usando la llave privada del receptor y después descifrar el mensaje usando esa llave de sesión.

También se puede tener el caso en donde los servicios de firma digital y confidencialidad pueden ser aplicados al mismo mensaje. Primeramente, la firma es generada por el mensaje y añadida al mensaje; después el mensaje más la firma son cifradas usando la llave de sesión convencional; finalmente, la llave de sesión es cifrada usando cifrado de llave pública y añadida al bloque de cifrado.

Para la compresión de datos PGP comprime el mensaje después de aplicar la firma digital pero antes del cifrado.

Usualmente en PGP parte del bloque a ser trasmitido es cifrado. Si es usado únicamente el servicio de firma digital, entonces el mensaje es cifrado (con la llave privada del transmisor). Si es usado el servicio de confidencialidad, el mensaje más la firma digital (si se presenta) son cifradas (con la llave convencional única), donde todo o parte del bloque resultante consiste en un conjunto de 8 bits; sin embargo, muchos sistemas de correo electrónico solo permiten el uso de bloques que consisten en texto ASCII, por lo que PGP proporciona un servicio para convertir este texto en caracteres ASCII. El esquema usado para este propósito es la conversión Radix-64 o “ASCII armor”, donde cada grupo de tres bytes del dato binario es mapeado en 4 caracteres ASCII.

Aunque el IETF ha trabajado con PGP, no se ha adoptado como norma todavía debido a que incorpora protocolos que tienen protecciones patentadas como IDEA y RSA.

La diferencia entre otras implementaciones como son Open/PGP o PGP/MIME se basa básicamente en el conjunto de algoritmos utilizados, con la finalidad de hacer más extensa la implementación al soportar varios algoritmos de cifrado o bien para que sea mejor soportado y con mayores servicios de seguridad al juntarse con otra aplicación como es el caso de PGP/MIME.

- **PEM (Privacy Enhanced Mail)**

El desarrollo de PEM se inició en 1985 por el PSRG (Privacy and Security Research Group) bajo la dirección de la IAB (Internet Architecture Board), cuyo objetivo era proporcionar seguridad en el correo electrónico a los usuarios de Internet dando servicios de confidencialidad, integridad, autenticación y no repudio.

PEM proporciona tres tipos de mensajes llamados *mic-clear*, *mic-only*, y *encrypted*. El tipo ***mic-clear*** está pensado para poder enviar correo electrónico a destinatarios que no tienen implementado PEM, incluyendo en estos mensajes un código MIC (Message Integrity Code) para su integridad y, aunque éste no va codificado es posible enviar un mensaje a una lista de distribución que contenga usuarios PEM y usuarios no PEM, pudiendo todos ellos leer el mensaje (puesto que no va codificado) aunque sólo los usuarios PEM pueden utilizar los servicios de integridad y autenticación. Los mensajes ***mic-only*** son como los anteriores, pero se les hace una codificación adicional para que no puedan ser modificados y no falle la prueba de integridad. El tipo ***encrypted*** añade la confidencialidad a los servicios de integridad y autenticación, realizando una codificación adicional a los

mensajes mic-only, ya que la salida binaria del proceso de codificación no puede transmitirse tal cual.

Con respecto a PGP la principal diferencia está en que PEM necesita entidades centralizadas para los certificados de llaves públicas, mientras que mediante PGP los usuarios intercambian certificados unos con otros sin necesidad de utilizar entidades de certificación ya que cada usuario genera sus propias llaves pública y privada.

- **MOSS (MIME Object Security Services)**

Se trata de una extensión de PEM que soporta la codificación de mensajes MIME (Multipurpose Internet Mail Extensions).
MIME es una serie de especificaciones que proporcionan un intercambio de diferentes archivos tales como texto, audio, video, etc., a través de Internet de forma transparente para el usuario. Una parte importante de MIME está dedicada a mejorar las posibilidades de transferencia de texto en distintos idiomas y alfabetos.
La especificación MIME añade estructuras de información al cuerpo del mensaje, que permite contener información no textual, sin embargo, no provee ningún servicio de seguridad.

Al usar PEM, los usuarios requieren tener certificados, mientras que con MOSS los usuarios solamente necesitan tener un par de llaves pública y privada.
Por otro lado MIME incluye operaciones de codificación de transferencia para asegurar el envío sin modificar las partes del cuerpo del mensaje, por lo tanto, en contraste con PEM, MOSS no necesita incluir estas funciones.

- **S/MIME (Secure Multipurpose Internet Mail Extensions)**

S/MIME está basado extensamente en la norma MIME y describe a un conjunto de especificaciones que proporcionan seguridad a los correos electrónicos añadiendo servicios de seguridad criptográficos a través de una encapsulación de MIME usando firmas digitales y objetos cifrados.
Los servicios de seguridad básicos que ofrece S/MIME son autenticación, no repudio, integridad y privacidad de los mensajes, y opcionalmente incluye recibos firmados, etiquetas seguras, listas seguras y un método para identificar los certificados firmados.
La especificación S/MIME cuenta con cuatro tecnologías fundamentales para dar formato y proteger a los mensajes de correo electrónico: algoritmos criptográficos, infraestructuras de llave pública PKI[5] (Public Key Infrastructure), sintaxis en los mensajes criptográficos (CMS) del formato de los datos, y MIME. La correcta implementación de estos mecanismos será esencial para dar seguridad e interoperabilidad en todos los clientes S/MIME.

Los algoritmos criptográficos que debe soportar S/MIME para un óptimo funcionamiento son RSA o DSA para firmas digitales con SHA o MD5 como algoritmos hash, Triple-DES o DES en modo CBC para cifrado, y Diffie-Hellman para intercambio de llaves (ver capítulo III.6).

- **S-HTTP (Secure - Hyper Text Transfer Protocol)**

S-HTTP fue posiblemente el primer protocolo de seguridad implementado para Internet y fue desarrollado por E. Rescorla y A. Schiffman de EIT (Enterprise Integration Technologies Inc.) en conjunto con RSA Data (los cuales formaron la

[5] Infraestructura de red formada por servidores y servicios que en base a llaves públicas gestionan de forma segura todas las transacciones realizadas a través de la red.

empresa Terisa Systems a este efecto), viendo la luz por primera vez en forma de borrador (draft) en Junio de 1994.
Hay que señalar que S-HTTP **no** es HTTPS (HTTP sobre SSL (ver sección I.5.2)), sino que es una extensión del protocolo HTTP que permite el intercambio seguro de archivos en la WWW (World Wide Web) y sus servicios sólo están disponibles para este protocolo, sin embargo, se puede considera como una alternativa a SSL.

S-HTTP es una extensión del protocolo HTTP cuya finalidad es la transmisión de datos de forma segura sobre la Web entre un cliente y un servidor usando para ello un sistema de cifrado basado en una pareja de llaves pública y privada. S-HTTP proporciona tres maneras distintas para la protección del mensaje: cifrado, autenticación en ambos extremos mediante el uso de firmas digitales y verifica la integridad de los datos usando MAC[6] (Message Authentication Code) permitiendo que las opciones de firma y cifrado se usen de forma opcional.

S-HTTP también incluye definiciones para proveer transferencias de llaves y certificados, así como funciones administrativas similares; aunque no cuenta con un esquema de certificación de llaves en particular, sino que incluye un soporte con RSA y Kerberos como intercambio de llaves.

La principal diferencia, en cuanto a diseño, con SSL es que S-HTTP fue concebido para la transmisión de mensajes individuales de forma segura mientras que SSL se diseñó para establecer una conexión segura permanente entre dos dispositivos. Además S-HTTP debido al nivel donde se encuentra es capaz de proporcionar características de no repudio de mensajes de forma individualizada a través de las firmas digitales y trabajar conjuntamente con un firewall[7], mientras que SSL no es capaz; sin embargo, los servicios de seguridad ofrecidos por SSL son transparentes al usuario y a la aplicación pudiendo ser usado por otros protocolos

[6] Algoritmo de autenticación utilizado para autenticar el origen de los mensajes, así como su integridad.
[7] Dispositivo electrónico de control de acceso utilizado para establecer políticas de control de flujo de información de un lado a otro permitiéndolas o negándolas.

aparte de HTTP. En cuanto al modo de operación S-HTTP es más robusto que SSL porque permite opciones de renegociación y reintento, mientras que su principal debilidad es el mecanismo inicial de intercambio de llaves.

- **SET (Secure Electronic Transaction)**

Protocolo que surgió en 1997 por VISA y MasterCard en colaboración con empresas del mundo informático como Microsoft, IBM, GTE, SAIC (Science Applications Internacional Corporation), Terisa Systems, y VeriSign.

Este protocolo garantiza la seguridad de transacciones financieras a través de Internet, donde la transacción se produce con una combinación de certificados y firmas digitales entre el comprador, el vendedor, el banco del vendedor y la entidad emisora de la tarjeta, asegurando la privacidad y confidencialidad de la información.

Ofrece servicios de confidencialidad de la información; integridad en los datos; autenticación de la cuenta del titular de la tarjeta con ayuda de las firmas digitales y los certificados X.509v3 utilizados para implementar esta función; autenticación mercantil dando a los titulares de las tarjetas verificar que no sólo son los poseedores legítimos de la tarjeta sino también tienen una relación con la institución financiera; e interoperabilidad entre hardware y software de varios fabricantes, permitiendo el uso de los titulares de la tarjeta u otros participantes.

SET no requiere un método particular de transporte, de manera que los mensajes pueden transportarse sobre HTTP en aplicaciones Web, correo electrónico o cualquier otro método donde los mensajes no necesitan transmitirse en tiempo real.

SET usa criptografía asimétrica (RSA) para las firmas digitales y para el cifrado tanto de las llaves de cifrado simétrico como de los datos bancarios, y criptografía simétrica (DES) para la transmisión del resto de los datos involucrados en la transacción. Las llaves usadas son de 128 bytes y SHA-1 es la función hash usada para la verificación de integridad de los mensajes.

Las principales ventajas frente a SSL son:

- Permite al cliente autenticar que el comerciante se encuentra autorizado para aceptar tarjetas de pago de forma segura y realizar transacciones económicas a través de Internet.
- Permite al vendedor autenticar la tarjeta de pago usada por el cliente en la transacción.
- Cada una de las tarjetas de pago, de cobro, mercancía o servicios comprados y dirección de entrega de los mismos es leída únicamente por el destinatario previsto (vendedor o banco).

Este protocolo también presenta diversas dificultades como un despliegue lento; no resulta fácil de implementar; exige software especial para clientes, vendedores y bancos; incompatibilidad, en algunos casos, de los productos que cumplen con el estándar SET; no permite pagos aplazados; y hasta puede llegar a causar un conflicto con los sistemas internos de los vendedores por la tarjeta del comprador cifrada.

I.5.2 Nivel de Transporte

- **SSL (Secure Sockets Layer)**

El protocolo SSL es un protocolo generalmente usado para intercambios seguros, sin embargo, es extensamente usado en aplicaciones de comercio electrónico.

Fue diseñado originalmente por Netscape Development Corporation e integrado en 1994 en su navegador para tener una comunicación segura entre un cliente y un servidor sobre una red abierta como Internet. La primera publicación de SSL fue la 2.0 (la versión 1.0 fue una versión de prueba usada solamente dentro de Netscape) y actualmente la versión 3.0 es la que está en uso.

SSL tiene una comunicación segura entre el cliente y el servidor de una manera transparente por funcionar entre la capa de aplicación y transporte del modelo TCP/IP debajo de protocolos específicos de aplicación tales como NTTP (Network News Transfer Protocol) para noticias, HTTP para Web y SMTP (Simple Mail Transfer Protocol) para correo electrónico, sin embargo, no opera sobre UDP ni IPX (Internet Protocol eXchange).

Adicionalmente, SSL está compuesto por cuatro subprotocolos: ***Handshake*** encargado de los algoritmos de cifrado, autenticación, secuencia, compresión, y establecimiento de llaves entre cliente y servidor; ***Change Cipher Spec (CCS)*** para invocar cambios síncronos en los mecanismos de seguridad y los parámetros llave entre cliente y servidor; ***Record*** encargado de transportar los mensajes encapsulados entre cliente y servidor; y ***Alerta*** que indica errores encontrados durante la verificación del mensaje, así como cualquier incompatibilidad que surga durante el Handshake. En la Figura I.8 se muestran estos protocolos.

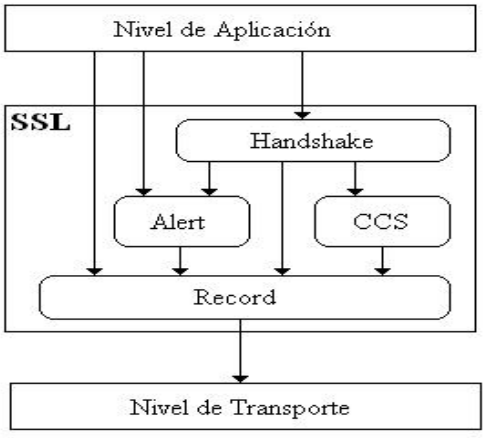


Figura I.8 Conjunto de protocolos que constituyen SSL

Cuando una conexión SSL se establece todas las conexiones entre el servidor y el navegador están cifradas incluyendo: la URL del documento pedido, el contenido del documento pedido, el contenido de cualquier campo de una forma electrónica, las cookies enviadas entre el navegador y el servidor, y el contenido del encabezado de HTTP.

Cuando el cliente deja una sesión SSL, generalmente la aplicación presenta un mensaje advirtiendo que la comunicación no es segura donde el cliente confirma que efectivamente desea abandonar la sesión SSL.

SSL ofrece tres servicios de seguridad: autenticación usando certificados X.509 versión 3; confidencialidad basada en algoritmos de cifrado simétricos con una llave de 40 u 80 bits como son DES, 3DES, DES40, RC2, RC4-128, RC4-40, IDEA, y SKIPJACK de Fortezza; e integridad usando MD5 o SHA como algoritmos hash (ver capítulo III.6). El intercambio de llaves se realiza por medio de los algoritmos RSA, Diffie-Hellman, y Fortezza (algoritmo desarrollado por la NSA[8] para aplicaciones criptográficas).

El principal problema de poner SSL en la capa de transporte es que no está específicamente afinado para el protocolo HTTP y por lo tanto podría no ser tan eficiente para navegar en la Web. Una limitación menor es que las conexiones de SSL deben de usar un socket de TCP/IP dedicado, además de trabajar solamente sobre TCP.

- **PCT (Private Communication Technology)**

PCT fue la respuesta de Microsoft a la supremacía de Netscape en el mundo de las comunicaciones seguras. El primer borrador del mismo apareció en septiembre

[8] Es la Agencia Nacional de Seguridad estadounidense dedicada especialmente a la seguridad informática.

de 1995 y fue diseñado para soportar transacciones comerciales de forma segura y espontánea incorporando mecanismos de llave pública/privada usando RSA para cifrado y autenticación en ambos extremos de la comunicación.

Las principales diferencias entre PCT y SSL están en la fase de negociación: Uno de los más novedosos mecanismos que aporta PCT es que separa las negociaciones y mecanismos de autenticación del cifrado, permitiendo que las llaves usadas en la autenticación excedan la longitud impuesta por las leyes de exportación de los Estados Unidos, no así las de cifrado. La segunda gran aportación de este protocolo es corregir un agujero de seguridad en el diseño de la fase Handshake de SSL.
A pesar de estas dos evidentes ventajas sobre su directo competidor, la gran implementación de SSL ha dejado a este nuevo protocolo sin ninguna posibilidad de competencia. En Mayo del año 2000, y como último intento, Microsoft anunció las especificaciones para una versión convergente de SSL y PCT que vendría a llamarse STLP (Secure Transport Layer Protocol) de la cual nunca más se ha vuelto a hablar y no ha llegado a tener ninguna implementación real.

- **TLS (Transport Layer Secure)**

TLS nace de la mano de la IETF en enero de 1999. Se construye a partir de las especificaciones de SSL 3.0 y son tan semejantes que a veces se le conoce como SSL 3.1. Por decirlo con las propias palabras de los autores de la especificación "[...] el protocolo TLS está basado en las especificaciones de SSL 3.0 publicadas por Netscape. Las diferencias entre este protocolo y SSL 3.0 no son grandes, pero si suficientes para que TLS 1.0 y SSL 3.0 no puedan interoperar (aunque TLS incorpora un mecanismo mediante el cual una implementación de TLS puede trabajar con SSL 3.0)".

Las principales diferencias entre SSL 3.0 y TLS 1.0 son las siguientes:

- En los mensajes *Certificate Request* y *Certificate Verify* del protocolo *Handshake*. En SSL 3.0 si el servidor solicita un certificado al cliente para que se autentique, éste debe responder con él o con un mensaje de alerta advirtiendo que no lo tiene. En TLS 1.0 si el cliente no posee certificado no responde al servidor de ninguna forma a este requerimiento.
- El mecanismo utilizado para construir las llaves de sesión es ligeramente diferente en TLS 1.0.
- TLS 1.0 no soporta el algoritmo de cifrado simétrico Fortezza que si es soportado por SSL 3.0. Esto es debido a que TLS busca ser íntegramente público mientras que Fortezza es un algoritmo propietario.
- TLS utiliza un mecanismo diferente y más seguro en el cálculo de la función hash MAC para autenticar el origen de los mensajes.
- TLS 1.0 introduce nuevos códigos de alerta no contemplados por SSL 3.0.
- TLS 1.0 introduce un nuevo mecanismo en el relleno de los bloques para frustrar ataques basados en el análisis de la longitud de los mensajes.

I.5.3 Nivel de Red

En este nivel del modelo TCP/IP un concepto muy importante es el de “tunneling”, el cual es un camino lógico por donde los datos son encaminados desde un extremo a otro del circuito que se crea formando una VPN[9] (Virtual Private Network) con la finalidad de ocultar la información.

Para que pueda establecerse un túnel es necesario que los extremos implicados utilicen los mismos protocolos de autenticación, cifrado y administración o generación de llaves.

[9] Es una red privada virtual que permite la transmisión de información privada sobre redes de uso público, como Internet, de manera segura utilizando conexiones virtuales.

Los protocolos que usan un túnel puede ser diseñados en niveles diferentes del modelo TCP/IP:

- Protocolos túnel capa 2: Éste corresponde al nivel de enlace, donde los túneles que se crean son similares a una sesión, los dos puntos terminales del túnel deben de estar de acuerdo con el túnel y deben negociar variables de configuración, los datos transferidos se mandan usando un protocolo basado en un datagrama, y el túnel debe ser creado, mantenido y después terminado. Los protocolos PPTP y L2TP operan en este nivel, y ambos encapsulan la carga útil en el datagrama PPP para ser enviados a través de Internet.

- Protocolos túnel capa 3. Estos protocolos trabajan en el nivel de red y usan paquetes como medio de intercambio, además se asume que todos los elementos de configuración fueron pre-configurados (frecuentemente por un proceso manual), y en este caso no es necesario mantener el túnel. IPSec opera en este nivel.

⌖ PPTP (Point-to-Point Tunneling Protocol)

Protocolo que fue desarrollado por un consorcio de fabricantes como Ascend Communications, ECI Telematics, U. S. Robotics y Microsoft para el envío de información en Internet a través de un túnel. Está basado en los protocolos PPP (Point to Point Protocol) y GRE (Generic Routing Encapsulation). Fue diseñado originalmente para proporcionar encapsulación, envolviendo los paquetes de información (IP, IPX o NetBEUI) dentro de paquetes IP para poder ser transmitidos a través de Internet usando GRE.

Este protocolo permite a los usuarios remotos tener acceso seguro a redes empresariales a través de Internet. Al contrario de PPP que está diseñado para admitir conexiones de acceso telefónico a Internet, PPTP no depende de

conexiones de acceso telefónico, pudiendo utilizarse para proporcionar conexiones a Internet de extremo a extremo por medio de un túnel seguro mediante otras tecnologías de acceso remoto, como el acceso a Internet por DSL[10] (Digital Subscriber Line).

La autenticación que ocurre durante la creación de una conexión VPN basada en PPTP utiliza los mecanismos de autenticación EAP, CHAP, SPAP y PAP (ver sección I.5.4) como se hace en PPP, además de soportar cifrado y filtrado de paquetes.
La gran desventaja de este protocolo reside en su diseño, que no es del todo seguro: antes de que el túnel GRE se establezca, parte del inicio de sesión, autenticación y demás se hace por el protocolo TCP en forma de texto claro, por lo que la información que pasa de este modo como es la IP del cliente y servidor, el nombre de usuario, la contraseña cifrada, etc., puede ser vista por cualquiera que esté en el medio.

PPTP establece el túnel pero no provee cifrado, por lo que puede ser usado en conjunto con la implementación de Microsoft denominada MPPE (Microsoft Point to Point Encryption), el cual es un protocolo para crear una VPN segura usando un algoritmo de cifrado RSA/RC4 para proporcionar confidencialidad con una longitud de llave para la sesión de 40 y 128 bits.
Sin embargo, con esta implementación se agregan un poco más de fallas debido a que la versión de 40 bits es demasiado débil para poder ser considerada segura y, además la llave se basa en la contraseña del usuario (de esta manera el usuario puede tener múltiples sesiones con su propia llave). El problema de esto es que la llave debería cambiarse cada determinado tiempo (mas aún cuando las sesiones PPTP son prolongadas) y esto realmente casi nunca sucede.

[10] Término para referirse de forma global a todas las tecnologías que proveen una conexión digital sobre la línea de la red telefónica local para permitir el envío de información a gran velocidad.

- ***L2F (Layer 2 Forwarding)***

Protocolo desarrollado por Cisco Systems para establecer túneles entre los usuarios remotos y sus sedes corporativas ofreciendo métodos de autenticación pero sin el cifrado de los datos. La principal diferencia con respecto al protocolo PPTP es que L2F puede trabajar sobre Frame-Relay[11] y ATM[12] (Asynchronous Transfer Mode), además de permitir que los túneles contengan más de una conexión.
L2F utiliza el protocolo PPP para la autenticación del usuario remoto, pero también implementa otros sistemas de autenticación como TACACS+ (Terminal Access Controller Access Control System) y RADIUS (Remote Authentication Dial-In User Service).

- **L2TP (Layer 2 Tunneling Protocol)**

Protocolo que fue desarrollado por la cooperación de Cisco y Microsoft, mezclando lo mejor de los protocolos PPTP de Microsoft y L2F de Cisco.
Los dos componentes principales del L2TP son: LAC (L2TP Access Concentrator), que es el dispositivo que físicamente termina una llamada; y el LNS (L2TP Network Server), que es el dispositivo que autentica y termina el enlace PPP.

L2TP utiliza redes conmutadas de paquetes para hacer posible que los extremos de la conexión estén ubicados en distintas computadoras.

Este protocolo no proporciona cifrado o autenticación por paquetes, sino que ha de combinarse con otro protocolo, como IPSec, para ofrecer integridad de datos y confidencialidad exigidos para una solución VPN.

[11] Tecnología utilizada para conectar distintas LANs entre sí de una manera rápida y eficiente.
[12] Tecnología de telecomunicación para la transmisión de paquetes.

- ***NLSP (Network Layer Security Protocol)***

NLSP es un protocolo desarrollado por la ISO para proporcionar seguridad al protocolo CLNP (Connectionless Network Protocol).
CLNP proporciona servicios orientados a la no conexión y no fiables en capas superiores del modelo OSI, por lo que el propósito del protocolo NLSP es implementar seguridad en los servicios de la capa de red y proporcionar servicios de seguridad en capas superiores.

- ***IPSec (Internet Protocol Security)***

IPSec proporciona autenticación, confidencialidad e integridad de datos, proporcionando protecciones efectivas contra la repetición de tramas y, gracias a su posición en la pila de protocolos, es capaz de trabajar con UDP y otros protocolos de la capa de transporte, uno de los inconvenientes de SSL.
Por estos motivos algunos autores presentan a IPSec como el verdadero sustituto de SSL mientras que otros insisten en que ambos coexistirán puesto que ofrecen soluciones óptimas para diferentes problemas.

I.5.4 Nivel de Enlace

- **WEP (Wired Equivalency Privacy)**

WEP fue el primer protocolo de cifrado introducido en la especificación 802.11 diseñado con la finalidad de proteger los datos que se transmiten en una conexión inalámbrica. Se basa en el algoritmo de cifrado RC4 con una llave secreta de 40 a

104 bits y un IV (Initialization Vector) o vector de inicialización de 24 bits para cifrar el mensaje junto con su checksum ICV (Integrity Check Value).
El IV debe ser aplicado a cada paquete para que los paquetes subsiguientes estén cifrados con llaves diferentes.

El protocolo WEP resuelve aparentemente el problema de cifrado de datos entre emisor y receptor; sin embargo, existen varias situaciones que hacen que WEP no sea seguro:

- Debilidades del algoritmo RC4 dentro del protocolo WEP debido a la construcción de la llave.
- No existe un método integrado de actualización de llaves, es decir, trabaja con llaves de cifrado estáticas (se configura una llave en el punto de acceso y ésta no se cambia nunca o muy de vez en cuando). Esto hace posible que un atacante acumule grandes cantidades de texto cifrado con la misma llave y pueda intentar un ataque por fuerza bruta.
- La longitud del IV que se utiliza es demasiado corto (24 bits), además que se permite la reutilización de éste implicando que no haya protección contra la repetición de mensajes.
- No existe una comprobación de integridad apropiada, ya que utiliza CRC32 (Cyclic Redundancy Checking de 32 bits) para la detección de errores y no es criptográficamente seguro por su linealidad.
- No ofrece servicio de autenticación; basta con que el equipo móvil y el punto de acceso compartan la llave WEP para que se tenga una comunicación.

En un principio, se había aceptado que WEP proporcionaba un nivel de seguridad aceptable sólo para usuarios domésticos y aplicaciones no críticas; sin embargo, se desvaneció con la aparición de ataques en el 2004 permitiendo que paquetes arbitrarios fueran descifrados sin necesidad de conocer la llave.

- ***WPA (Wi-Fi Protected Access)***

WPA soluciona todas las debilidades conocidas de WEP y se considera suficientemente seguro. Las principales características de WPA son: llaves generadas dinámicamente y distribuidas de forma automática evitando tener que modificarlas manualmente en cada uno de los elementos de la red cada cierto tiempo; utilización más robusta del IV con una longitud de 48 bits permitiendo generar 2^{48} combinaciones de llaves diferentes; nuevas técnicas de integridad como la implementación del código MIC en lugar del CRC-32; y autenticación sustituyendo el mecanismo de secreto compartido de WEP por 802.1X, EAP y RADIUS.

Una variante de WPA es el WPA2 que se introdujo en la norma 802.11i, donde su principal diferencia con la versión original es que soporta el algoritmo de cifrado AES (ver capítulo III.6.2)

- ***PAP (Password Authentication Protocol)***

Es un protocolo simple de autenticación debido a que no usa un método fuerte para la autenticación en una conexión PPP ya que la contraseña es enviada a través de la conexión en forma clara sin ninguna protección.

Este tipo de autenticación se utiliza cuando los datos en texto claro deben estar disponibles para simular un login de un host remoto, es decir, cuando el servidor (host) de acceso remoto pide el nombre de usuario y contraseña, y el cliente se los facilita en texto claro; estos datos son comparados por el servidor en su base de datos de llaves para permitir el acceso.

- **CHAP (Challenge Handshake Authentication Protocol)**

Es un protocolo de autenticación que funciona por medio del modelo de "desafío y respuesta" utilizado para conexiones PPP, mediante el esquema de cifrado estándar MD5, ofreciendo un esquema de autenticación más fuerte que PAP.

CHAP además de solicitar autenticación al cliente al comienzo de la sesión, envía retos a intervalos regulares para asegurarse de que el cliente no ha sido reemplazado por un intruso.

- **MS-CHAP (MicroSoft - CHAP)**

Es un protocolo creado por Microsoft para autenticar las estaciones de trabajo remotas de Windows NT 3.5, 3.51 y 4.0, y Windows 95, creándose posteriormente la versión 2, MS-CHAP v2, para soporte en Windows NT 4.0, 95, y posteriores.

- ***EAP (Extensible Authentication Protocol)***

El protocolo EAP se creó como una extensión al protocolo PPP para permitir la autenticación entre los nodos soportando múltiples mecanismos de autenticación.
EAP sirve como soporte a protocolos propietarios de autenticación, gestiona las contraseñas en mecanismos de desafío-respuesta y es capaz de trabajar con tecnología de llave pública.
EAP está soportado en la especificación 802 del IEEE y por lo tanto está soportado por los dispositivos inalámbricos adheridos al estándar IEEE 802.11x.

- ***LEAP (Lightweight EA P)***

Este protocolo es una solución propietaria de Cisco y se utiliza para proteger las redes inalámbricas de posibles intrusiones utilizando una fuerte autenticación entre el cliente y el servidor RADIUS por medio de llaves cifradas dinámicas por usuario y por sesión.

- ***PEAP (Protected EAP)***

Aunque EAP proporciona gran flexibilidad, todos los mensajes de autenticación intercambiados deben ser enviados en claro (sin cifrar) por lo que es posible que sean interceptados y manipulados por un atacante. Esto es más problemático todavía en el caso de las redes inalámbricas por la facilidad de intercepción que ofrecen. Dado que EAP ocurre antes de que comience el cifrado de paquetes sería conveniente disponer de algún sistema de protección previo. PEAP solventa este problema creando un canal de comunicación seguro que dispone de cifrado y control de integridad utilizando para ello TLS, esto es, después de establecer el canal seguro es cuando la negociación EAP comienza, es decir, primero se protege el intento de acceso del cliente y luego se produce el proceso de autenticación y autorización, permitiendo que los clientes que utilizan este sistema puedan usar contraseñas en lugar de certificados para autenticarse (usando MS-CHAP v2). Esta solución requiere un certificado digital (solicitado con una autoridad certificadora de terceros) en el servidor RADIUS pero no en los clientes inalámbricos teniendo la ventaja de no tener que distribuir los certificados a éstos últimos.

Cisco después presentó una nueva variante del EAP, denominada FAST (Flexible Authentication Secure Tunneling). La idea de FAST fue hacer un método que pudiera crear un túnel seguro como PEAP para proteger el intercambio de

credenciales pero sin la necesidad de utilizar certificados digitales, pues éstos son vistos por la mayoría de los usuarios como una complejidad y un costo mayor.

I.6 Ventajas y desventajas de los protocolos de seguridad en la pila de TCP/IP

Para tener más claramente cuáles serían las ventajas y desventajas de los protocolos de seguridad antes mencionados, en la Tabla I.1 se muestra un resumen con las características más sobresalientes de dichos protocolos para cada nivel del modelo TCP/IP.

Nivel	Ventajas	Desventajas	Protocolo
APLICACIÓN	Se puede extender la aplicación para brindar servicios de seguridad sin tener que depender del Sistema Operativo Facilita el servicio de no repudio	Los mecanismos de seguridad deben ser diseñados de forma independiente para cada aplicación Mayor probabilidad de cometer errores	Kerberos SSH PEM PGP MOSS S/MIME OpenPGP PGP/MIME SET S-HTTP
TRANSPORTE	En teoría no se requieren modificaciones por aplicación	Mantener el contexto del usuario es complicado TLS requiere que las aplicaciones sean modificadas	PCT SSL TLS
RED	Disminuye el flujo excesivo de negociación de llaves Las aplicaciones no requieren modificación alguna Permite crear VPNs e intranets	Difícil manejar el no repudio	NLSP PPTP L2F L2TP IPSec (AH, ESP)
ENLACE	Son más rápidos	No son soluciones estables y funcionan bien sólo para enlaces dedicados Los dispositivos deben estar físicamente conectados	WEP WPA PAP CHAP MS-CHAP EAP LEAP PEAP

Tabla I.1 Resumen de los protocolos de seguridad en la pila TCP/IP

Capítulo 2

Protocolo de Internet versión 6: IPv6

II.1 Introducción

Cuando se diseñó el actual protocolo de Internet, también denominado IPv4 (Internet Protocol version 4), a principios de los años 70, nunca se tuvo en cuenta el crecimiento exponencial que ha estado experimentando el uso de la red (Internet) en los últimos años (y lo que queda por crecer), así como la gran cantidad de usuarios que usarán Internet para necesidades diferentes, dando como resultado que el espacio de direcciones que proporciona IPv4 empiece a quedarse pequeño. Además, IPv4 nunca se diseñó pensando en una implementación a gran escala, de modo que los equipos de red encargados de encaminar los paquetes tienen que dedicar una gran parte de su procesamiento para lograr su objetivo, sin proporcionar ninguna funcionalidad adicional. Y por si fuera poco, no se consideraron aspectos como la seguridad, eficiencia, calidad de servicio, movilidad, etcétera, como características esenciales en la actualidad.

El problema de agotamiento de direcciones IP surgió en un principio por una mala asignación de direcciones. Con el actual protocolo IPv4, la dirección tiene un formato con cuatro números de 8 bits en notación decimal (0-255) separados por puntos (a.b.c.d), es decir, una dirección está compuesta por un total de 32 bits, con los que conseguimos un total de 4,300 millones de valores posibles.

La asignación de direcciones comenzó a hacerse de manera centralizada por el único centro de registro SRI-NIC[13] (Stanford Research Institute - Network Information Center), satisfaciendo casi todas las solicitudes sin necesidad de mayor trámite; sin embargo, cuando Internet comenzó a crecer de forma espectacular, ocasionó algunas de las siguientes consecuencias:

- Mal aprovechamiento del espacio de direcciones. Cada centro tendía a pedir una clase superior a la requerida, normalmente una clase B en vez de una o varias clases C, por puro optimismo en el crecimiento propio o por simple vanidad.

- Peligro de agotamiento de las direcciones de clase B. Las más solicitadas debido a la escasez de posibilidades de elección. La alerta sonó cuando se había agotado el 30% de esta clase y la demanda crecía exponencialmente.

- Síntomas de saturación en los enrutadores de backbone. Al imponerse restricciones severas en la asignación de clases B, las peticiones de múltiples clases C se hicieron masivas, lo que hizo aumentar de forma explosiva el número de prefijos en las tablas de direcciones de los enrutadores, alcanzando sus límites físicos debido a su capacidad de memoria y de proceso.

Por tanto, conforme la población conectada a la red se iba incrementando (en términos de equipos y redes conectadas), se estaba presentando un doble problema: agotamiento de direcciones y colapso de enrutadores debido a la saturación de rutas, teniéndose que tomar las siguientes medidas urgentes:

[13] Era un Instituto que por los años 80's desempeñaba funciones de administración y supervisión de algunos recursos de Internet (en aquel entonces ARPANET y NSFNET).

1. Imposición de políticas restrictivas de asignación de direcciones por parte de los centros de registros (actualmente el NIC[14]).

2. Uso de nuevos métodos como CIDR (Classless Inter-Domain Routing), cuya funcionalidad se basa en la agregación de prefijos que son adyacentes para dar lugar a prefijos menores (y por tanto más generales), reduciéndose el número de entradas en las tablas de los enrutadores. Por ejemplo, si a una Institución se le asignaba el bloque 193.144.0.0/15, tendría 131,072 direcciones para delegar entre sus miembros, anunciándose todo el bloque como un único prefijo.

3. Inicialización de asignación, según el modelo CIDR o sin-clase, de partes del espacio de direcciones que estaban reservadas, como 64.0.0.0/8 - 126.0.0.0/8 que suponen aproximadamente 1/4 del total del espacio asignable, estableciéndose en bloques de prefijo variable.

En cualquier caso, hay que entender que tanto CIDR como las políticas restrictivas de asignación de direcciones han sido sólo medidas temporales, dirigidas a afrontar problemas concretos y que no resuelven (en algunos casos hasta agravan) los problemas crónicos detectados en Internet. Así, se han llegado a plantear iniciativas como la devolución de direcciones, la obligatoriedad de cambiar de direcciones al cambiar de proveedor, la asignación dinámica de direcciones, el uso de traductores de direcciones como NAT[15] (Network Address Translation) que transforman un espacio privado de direcciones en otro perteneciente al proveedor, o incluso el cobrar una cantidad elevada por cada prefijo (no perteneciente al espacio del proveedor) que un cliente desee que su proveedor anuncie.

[14] Network Information Center: Institución encargada de asignar dominios de Internet bajo su dominio de red, que mediante un DNS pueden montar sitios de Internet mediante un proveedor.
[15] Es una aplicación para que un determinado dispositivo o aplicación de software sea capaz de cambiar la dirección IP de origen o destino por otra dirección definida previamente.

Aún con todo este tipo de sistemas y procedimientos, el caos y la desorganización han hecho que el espacio de direccionamiento actual resulte hoy por hoy insuficiente para acoger a nuevos dispositivos de tercera generación. Así, en la década de los 90's y previendo esta explosión por el interés de Internet, se hizo evidente que el IP tenía que evolucionar y volverse más flexible creándose en noviembre de 1994 el IPng (Internet Protocol next generation), RFC[16] 1752, y diseñado por el IETF, que posteriormente se denominaría como IPv6 (Internet Protocol version 6), RFC 2460, cuyo objetivo principal sería reemplazar de forma gradual la versión actual.

Cabe mencionar que esta nueva versión del IP se consideró con el número 6, debido a que la versión 5 se utilizó para describir al protocolo ST2+ (Internet Stream Protocol version 2), RFC 1819, como una extensión experimental, sin embargo, no se concluyó en nada quedándose en desuso; y para evitar posibles conflictos de numeración y/o confusión, se optó por elegir el número 6 para la nueva versión.

De esta manera, se justifica la revisión de la versión 4 del protocolo IP desde dos puntos de vista principalmente:

- *Técnicamente*: El sistema de direccionamiento es insuficiente para la demanda actual y futura prevista. Las tablas de enrutamiento (tablas de direcciones que almacenan los enrutadores de forma interna) son excesivamente grandes debido a la gran cantidad de direcciones existentes actualmente y al sistema de enrutamiento utilizado, que obliga a los enrutadores a mantener grandes cantidades de direcciones para conocer hacia dónde deben redireccionar los datagramas, ocasionando una lenta circulación por Internet por todas las consultas que deben hacer los enrutadores en las tablas para cada datagrama.

[16]Documentos de especificaciones que se exponen públicamente para su discusión.

- *Socialmente*: Las necesidades de los usuarios de Internet han aumentando espectacularmente, exigiendo nuevas opciones y capacidades (seguridad, privacidad, auto-configuración, velocidad, etc.) que la versión 4 no puede proporcionar de una manera eficiente.

II.2 Características principales de IPv6

1. Una de las características más importantes es un mayor espacio de direcciones. IPv4 tiene un espacio de direcciones de 32 bits, es decir, 2^{32} (4,294,967,296). Por otra parte, IPv6 ofrece un espacio de direcciones de 128 bits, 2^{128} (340,282,366,920,938,463,463,374,607,431,768,211,456), con lo cual se puede soportar más niveles jerárquicos de direccionamiento y más nodos direccionables.

2. Se simplifica el encabezado IP, es decir, algunos campos del encabezado IPv4 se eliminan o se cambian al campo donde se encuentran los encabezados de extensión opcionales (Siguiente encabezado).

3. Capacidad de auto-configuración “Plug & Play”. Los dispositivos pueden configurar sus propias direcciones IPv6 basándose en la información que reciban del enrutador más próximo.

4. Paquetes IP eficientes y extensibles sin que haya fragmentación en los enrutadores, alineados a 64 bits (preparados para su procesado óptimo con los nuevos procesadores de 64 bits), y con un encabezado de longitud fija, más simple, que agiliza su procesado por parte del enrutador.

5. Posibilidad de paquetes con carga útil (datos) de más de 65,355 bytes.

6. Soporte de seguridad en el núcleo del protocolo llamado IPSec, implementado de manera obligatoria, el cual proporciona seguridad en la capa de red, dando autenticación e integridad en los datos.

7. Capacidad de etiquetas de flujo. Puede ser usada por un nodo origen para etiquetar paquetes pertenecientes a un flujo de tráfico en particular que requieren un manejo especial por los enrutadores IPv6, como calidad de servicio o servicios en tiempo real (video conferencia).

8. Aplicaciones multicast: envío de *un* mismo paquete a un *grupo* de receptores; y anycast: envío de *un* paquete a *un* receptor dentro de *un* grupo.

9. Renumeración y multihoming[17], facilitando el cambio de proveedor de servicios.

10. Presenta ventajas en cuanto a movilidad, de tal manera que un nodo mantenga la misma dirección IP, a pesar de su movilidad.

11. Hace un enrutamiento más eficiente en el backbone de la red, debido a la jerarquía de direccionamiento basada en la agregación.

12. Ofrece Calidad de Servicio (QoS) y Clase de Servicio (CoS).

[17] El multihoming de sitio, definido en el RFC 3582, es la conexión de un host o sitio a más de un ISP a la vez.

II.3 Direccionamiento de IPv6

Frente a la estructura actual, en IPv6 se ha variado el formato de la dirección aumentando el número de bits a 128, permitiendo incrementar extraordinariamente el número de direcciones IP posibles.
Al ampliar el espacio de direcciones, se ha convenido en cambiar también la notación para facilitar su manejo, donde en lugar de octetos decimales separados por un punto, en IPv6 las direcciones se expresan como ocho grupos de números hexadecimales separados por dos puntos, dando una estructura más adecuada para un mejor enrutamiento mediante una jerarquía de prefijos que implican distintos tipos de asignación. Existen tres tipos de direcciones [RFC 4291]:

1. ***Unicast*** identifican a una única interfaz. Un paquete enviado a una dirección unicast es entregado sólo a la interfaz identificada con dicha dirección. Es el equivalente a las direcciones IPv4 actuales.

2. ***Anycast*** identifican a un conjunto de interfaces (típicamente pertenecen a diferentes nodos). Un paquete enviado a una dirección anycast es entregado en una (cualquiera) de las interfaces identificadas con dicha dirección (la que este más “cerca” en distancia). Lo que permite crear, por ejemplo, ámbitos de redundancia, de forma que varias máquinas puedan ocuparse del mismo tráfico según una secuencia determinada (por el enrutador), si la primera “cae”.

3. ***Multicast*** identifican a un grupo de interfaces (por lo general pertenecientes a diferentes nodos). Cuando un paquete es enviado a una dirección multicast es entregado a todas las interfaces del grupo identificadas por dicha dirección.

En IPv6 no existen direcciones **broadcast**, su funcionalidad ha sido suplantada por las direcciones multicast, cuya misión es la retransmisión múltiple de las aplicaciones.

II.3.1 Representación de las direcciones de IPv6

Existen tres formas de representar las direcciones IPv6 como *cadenas* de texto.

1. La forma convencional es x:x:x:x:x:x:x:x donde cada **x** es un valor hexadecimal de 16 bits. En la Tabla II.1 se mencionan dos ejemplos y, se puede observar que en caso de que **x** sea cero no es necesario escribirlos a la izquierda de cada campo, aunque al menos debe existir un número en cada uno (excepto en el caso 2).

FEDC:BA98:7654:3210:FEDC:BA98:7654:3210
1080:0:0:0:8:800:200C:417A

Tabla II.1 Ejemplos de direcciones IPv6 convencionales

2. Debido a que será común utilizar esquemas de direccionamiento con largas cadenas de bits en cero, existe la posibilidad de usar una sintaxis especial para representarlas. El uso de "**::**" indica múltiples grupos de 16 bits de ceros, el cual podrá aparecer una sola vez en cada dirección, como se muestra en la Tabla II.2.

Dirección IPv6		***Dir. IPv6 simplificada***	***Tipo de dirección***
0:0:0:0:0:0:0:0	➔	::	no especificada
0:0:0:0:0:0:0:1	➔	::1	loopback
1080:0:0:0:8:800:200C:417A	➔	1080::8:800:200C:417A	unicast
FF01:0:0:0:0:0:0:101	➔	FF01::101	multicast

Tabla II.2 Ejemplos de direcciones IPv6 en forma simplificada

3. Para escenarios con nodos IPv4 e IPv6 es posible utilizar la siguiente sintaxis: x:x:x:x:x:x:d.d.d.d, donde las **x** representan valores hexadecimales de las seis partes más significativas (de 16 bits cada una) que componen la dirección; y las **d** son valores decimales de las 4 partes menos significativas

(de 8 bits cada una) de la representación estándar del formato de direcciones IPv4. En la Tabla II.3 se muestran algunos ejemplos.

0:0:0:0:0:FFFF:129.144.52.38	➔	::FF:124.144.52.38
0:0:0:0:0:0:13.1.68.3	➔	:: 13.1.68.3

Tabla II.3 Ejemplos de direcciones IPv6 compatibles con IPv4

En relación a la representación de los prefijos de las direcciones IPv6, los prefijos de identificadores de subredes, enrutadores y rangos de direcciones IPv6 son expresados de la misma forma que en la notación CIDR utilizada en IPv4.

Un prefijo de dirección IPv6 se representa con la siguiente notación:

dirección-IPv6/longitud-prefijo, donde

dirección-IPv6: es una dirección IPv6 en cualquiera de las notaciones mencionadas anteriormente.
longitud-prefijo: es un valor decimal que especifica cuantos de los bits más significativos (bits contiguos de la parte izquierda), representan el prefijo de la dirección.

En la Tabla II.4, se muestra la representación válida con un prefijo de 60 bits para la dirección 12AB00000000CD30123456789ABCDEF.

12AB:0000:0000:CD30: 123:4567:89AB:CDEF/ 60			
donde	➔ 12AB:0:0:CD30:123:4567:89AB:CDEF	=	dirección
	12AB:0:0:CD30:: /60	=	subred (prefijo)

Tabla II.4 Ejemplo de una dirección IPv6 utilizando prefijos

Después de la experiencia de observar cómo se van agotando las direcciones IPv4 sin poder ampliar o reestructurar el direccionamiento de manera sencilla, los

diseñadores de la versión 6 optaron por no consumir todo el espacio direccionable de los 128 bits, realizando una partición en subgrupos independientes, como se muestra en Tabla II.5, para facilitar en un futuro la ampliación de tipos de direcciones o incluso un nuevo tipo de direccionamiento.

De esta forma, se han reservado algunos prefijos para aquellos grupos específicos de direcciones, como NSAP[18] (Network Service Access Point), que se prevé que en un futuro puedan necesitar un rango de direcciones separado del resto de direcciones IP.

Por lo ya asignado y con esta partición del espacio de direcciones, aún queda sin asignar más de un 80% del espacio total de direcciones. En la Tabla II.5 se muestra la distribución del espacio IPv6.

Grupo asignado	Prefijo	Fracción del espacio ocupado
No asignado	0000 0000	1/256
No asignado	0000 0001	1/256
Reservado para direcciones NSAP	0000 001	1/128
No asignado	0000 01	1/64
No asignado	0000 1	1/32
No asignado	0001	1/16
Direcciones globales unicast	001	1/8
No asignado	010	1/8
No asignado	011	1/8
No asignado	100	1/8
No asignado	101	1/8
No asignado	110	1/8
No asignado	1110	1/16
No asignado	1111 0	1/32
No asignado	1111 10	1/64
Direcciones unicast únicas locales	1111 110	1/128
No asignado	1111 1110 0	1/512
Direcciones unicast de enlace local	1111 1110 10	1/1024
No asignado (anteriormente para las direcciones unicast de sitio local)	1111 1110 11	1/1024
Direcciones multicast	1111 1111	1/256

Tabla II.5 Distribución del espacio de direcciones de IPv6

[18] Tipo de direcciones que identifican a un dispositivo.

II.4 Tipos de direcciones de IPv6

II.4.1 Direcciones unicast

Las direcciones unicast de IPv6 son agregables[19] con máscaras de bits contiguos similares a las direcciones IPv4 con CIDR.

En estas direcciones un host puede considerar que la dirección unicast (incluyendo la de sí mismo) no tiene una estructura interna, ver Figura II.1.

128 bits
Dirección del host

Figura II.1. Formato de dirección unicast sin estructura interna

Por otro lado, un host más sofisticado, considerando el prefijo de la subred donde se encuentra presentaría la estructura que se muestra en la Figura II.2.

n bits	**128-n bits**
Prefijo de la subred	ID de la interfaz

Figura II.2 Formato de dirección unicast con prefijo de subred

Existen diferentes formas de direcciones unicast asignadas en IPv6, las cuales se mencionan a continuación:

1. La dirección no especificada: Se compone por 16 bytes nulos (0:0:0:0:0:0:0:0) y sólo puede utilizarse como dirección inicial mientras se recibe una dirección fija. También puede utilizarse para funciones internas que requieran la especificación de una dirección IP.

[19] Es un procedimiento mediante el cual se pueden añadir direcciones a un nivel determinado para tener una mejor organización jerárquica en el enrutamiento de las redes globales.

2. La dirección interna o loopback (auto-retorno): Se define como 15 bytes nulos y un byte con el último bit a 1 (0:0:0:0:0:0:0:1). Esta dirección es interna y de ninguna forma puede circular por la red o ser dirección de origen o destino de un datagrama. Su utilidad viene dada para las PC que no dispongan de una conexión de red y deseen simular el comportamiento de conexión a una red mediante una dirección fantasma que nunca saldrá del propio host.

3. Direcciones IPv6 sobre IPv4:
   - Las “direcciones IPv6 compatibles con IPv4” permiten la retransmisión de tráfico IPv6 sobre infraestructuras IPv4 de forma transparente. Estas direcciones se representan como se muestra en la Figura II.3.

80 bits	**16 bits**	**32 bits**
0000.....0000	0000	Dirección IPv4

Figura II.3 Formato de direcciones IPv6 compatibles con IPv4

   - Las “direcciones IPv6 mapeadas desde IPv4” permiten que los nodos que sólo soportan IPv4 puedan seguir trabajando en redes IPv6. Tienen el formato de la Figura II.4.

80 bits	**16 bits**	**32 bits**
0000.....0000	FFFF	Dirección IPv4

Figura II.4 Formato de direcciones IPv6 mapeadas desde IPv4

4. Las direcciones de uso local: Este tipo de direcciones estaban subdivididas en dos tipos: direcciones de *enlace local y sitio local*.
   - Las direcciones de enlace local son direcciones que pueden utilizar los equipos conectados a una misma red local mientras se inicializan y no tienen asignada una dirección IP, pudiendo circular por la misma red local únicamente (en contraste con la dirección no especificada); por eso han sido diseñadas para direccionar un único enlace para propósitos de auto-configuración (mediante identificadores de interfaz),

descubrimiento de vecinos, o situaciones en donde no hay enrutadores (esta característica en IPv4 actúa conjuntamente con los protocolos ARP y RARP). Estas direcciones se construyen con el prefijo fe80::/10 y 64 bits que representan la dirección física (dirección MAC) de la tarjeta de red, y su formato se representa en la Figura II.5.

10 bits	**54 bits**	**64 bits**
1111111010	0	ID de la interfaz

Figura II.5 Formato de direcciones de enlace local

- Las direcciones de sitio local estaban reservadas para intranets. Estas direcciones como no eran válidas para Internet, sólo servían para que una organización pudiera tener la estructura de su red basada en un esquema TCP/IP sin la necesidad de estar conectados a Internet (en IPv4, existen diferentes clases reservadas para este mismo fin, como por ejemplo 192.168.xxx.yyy). Estas direcciones se construían con el prefijo fec0::/10 y su formato estaba representado según la Figura II.6.

10 bits	**38 bits**	**16 bits**	**64 bits**
1111111011	0	ID subred	ID de la interfaz

Figura II.6 Formato de direcciones de sitio local

- Sin embargo, la IETF definió un nuevo tipo de direcciones sustituyendo a las direcciones de sitio local llamadas *direcciones unicast únicas locales*, RFC 4193. Estas direcciones se construyen con el prefijo fc00::/7 y su formato está representado según la Figura II.7.

7 bits	**1 bit**	**40 bits**	**16 bits**	**64 bits**
1111110000	L	ID global	ID subred	ID de la interfaz

Figura II.7 Formato de direcciones de sitio local

Donde el bit "**L**" se configura a 1 si el prefijo es localmente configurado, mientras que para un valor 0 queda reservado para definirse en un futuro.

5. Las direcciones unicast globales agregables: Equivalen a las direcciones IPv4 públicas y son ruteables globalmente. Se identifican por el prefijo 2000::/3. En este tipo de direcciones los 64 bits más altos identifican a la red y los 64 más bajos al host.

   Este formato de direcciones ha sido diseñado para soportar el tipo de "agregación" que se utiliza hoy en día, *basado en proveedores* (provider-based) y un nuevo tipo de agregación denominado *basado en intercambios* (exchange-based). La combinación de ambos es la que permite un enrutamiento más eficiente.

   Existen tres tipos de direcciones unicast globales agregables:

   1. De prueba establecidas por el 6bone[20]: comienzan con 3ffe.
   2. Para emplear el método túnel 6to4: comienzan con 2002.
   3. De producción asignadas por un proveedor: comienzan con 2001.

   Además están organizadas en tres niveles de jerarquía:

   1. Topología Pública: Conjunto de proveedores e "intercambiadores" que proveen servicios públicos de tránsito Internet.
   2. Topología de Sitio: Es local a un sitio específico u organización que no provee servicio público a nodos fuera del sitio.
   3. Identificador de Interfaz: Un número único, al menos en el segmento local de la LAN, de 64 bits usualmente generado automáticamente, e identifica las interfaces en los enlaces.

[20] Fue el backbone experimental de IPv6 cuya función fue asistir en la evolución y desarrollo de IPv6.

El formato actual de las direcciones unicast globales agregables es el que se muestra en la Figura II.8, definido en el RFC 3587.

n bits prefijo de enrutamiento global	**m bits** ID de la subred	**128-n-m bits** ID de la interfaz

Figura II.8 Formato de direcciones unicast globales agregables

II.4.2 Direcciones anycast

Una dirección anycast identifica a múltiples interfaces. Los paquetes destinados a una dirección anycast se entregarán a una sola interfaz (la que esté más “cerca” en distancia, dentro del grupo de direcciones anycast), teniendo una comunicación “uno” a “uno-entre-muchos”.

Para que los paquetes se entreguen a la dirección anycast más “cercana”, el enrutador de la red debe conocer que interfaz tiene asignada a una dirección anycast, así como sus distancias.

Las direcciones anycast no tienen un espacio propio dentro del direccionamiento IPv6, utilizan el mismo espacio que las direcciones unicast (es decir, no podemos diferenciar entre direcciones unicast y anycast). El ámbito de las direcciones anycast se equipara con las de unicast, de tal modo que, pueden existir direcciones anycast de sitio, de enlace o global. Además, este tipo de direcciones sólo pueden usarse como direcciones de destino, jamás como fuente.

Existe una dirección anycast, requerida para cada subred, que se denomina “dirección anycast del enrutador de la subred”. Su sintaxis es equivalente al prefijo que especifica el enlace correspondiente de la dirección unicast, siendo el identificador de interfaz igual a cero, como se muestra en la Figura II.9.

n bits	128-n bits
prefijo de subred	Nulo (::0)

Figura II.9 Formato de dirección de tipo anycast del enrutador de la subred.

Todos los enrutadores deben de soportar esta dirección para las subredes a las que están conectados. Los paquetes enviados a la “dirección anycast del enrutador de la subred”, serán enviados a un enrutador de la subred.

La utilidad de estas direcciones se basa en la implementación de los siguientes mecanismos:

- Comunicación con el servidor más “cercano”: Estas direcciones permiten que un cliente pueda comunicarse con un servidor de entre un grupo, que sea el más cercano dentro de la red.
- Descubrimiento de Servicios: Al configurar un nodo con IPv6, no haría falta especificarle la dirección del servidor DNS, Proxy, etc., sino que podría existir una dirección anycast que identificara a estos servicios.
- Movilidad: Nodos que tienen que comunicarse con un enrutador dentro del conjunto disponible en su red.

El formato de este tipo de direcciones es muy sencillo debido a que toda la carga se centra en el sistema de enrutamiento; de esta forma, para cada enrutador se guarda un solo registro que le indica quién es el miembro más cercano a él de un grupo especificado, de tal manera que al recibir un datagrama con una dirección de destino anycast comprobará la existencia de este registro especial en su tabla de enrutamiento o bien encaminará de forma normal el datagrama.

II.4.3 Direcciones multicast

Una dirección multicast en IPv6 puede definirse como un identificador para un grupo de nodos (la misma dirección es compartida por todos los integrantes del grupo); de tal forma un nodo puede pertenecer a uno o varios grupos multicast, de manera que un datagrama enviado a esta dirección será distribuido a todos los integrantes del grupo. Su formato se muestra en la Figura II.10.

8 bits	4 bits	4 bits	112 bits
11111111	000T	alcance	identificador de grupo

Figura II.10 Formato de direcciones multicast

Los primeros 8 bits indican que se trata de una dirección multicast. El conjunto de 4 bits siguiente son las banderas, donde los tres primeros bits inicializados a “0” están reservados, y el bit T (transitorio) indica una dirección permanente (T=0) o una dirección temporal (T=1) asignada por la autoridad de numeración global de Internet. Los bits de alcance tienen el significado mostrado en la Tabla II.6.

0 reservado
1 alcance de interfaz local
2 alcance de enlace local
3 reservado
4 alcance de administración local
5 alcance de sitio local
6 no asignado
7 no asignado
8 alcance de organización local
9 no asignado
A no asignado
B no asignado
C no asignado
D no asignado
E alcance global
F reservado

Tabla II.6 Significado de los bits de ámbito en multicast

El “identificador de grupo”, determina el grupo de multicast concreto al que se hace referencia, ya sea temporal o permanente, dentro de un determinado ámbito.

Las principales direcciones multicast reservadas son las incluidas en el rango FF0x:0:0:0:0:0:0:0. Algunos ejemplos útiles, según su ámbito, son:

FF01: 0:0:0:0:0:0:1 → significa todos los nodos (ámbito local).
FF02: 0:0:0:0:0:0:1 → significa todos los nodos (ámbito enlace).
FF01: 0:0:0:0:0:0:2 → significa todos los enrutadores (ámbito local).
FF05: 0:0:0:0:0:0:2 → significa todos los enrutadores (ámbito de sitio).

II.5 Encabezados de IPv6 e IPv4

II.5.1 Descripción

El encabezado IPv4, Figura II.11, facilita un sistema “sin conexión”[21] y no fiable de entrega de datagramas entre dos equipos cualesquiera conectados a Internet, dando un servicio de entrega basado en el mejor intento.

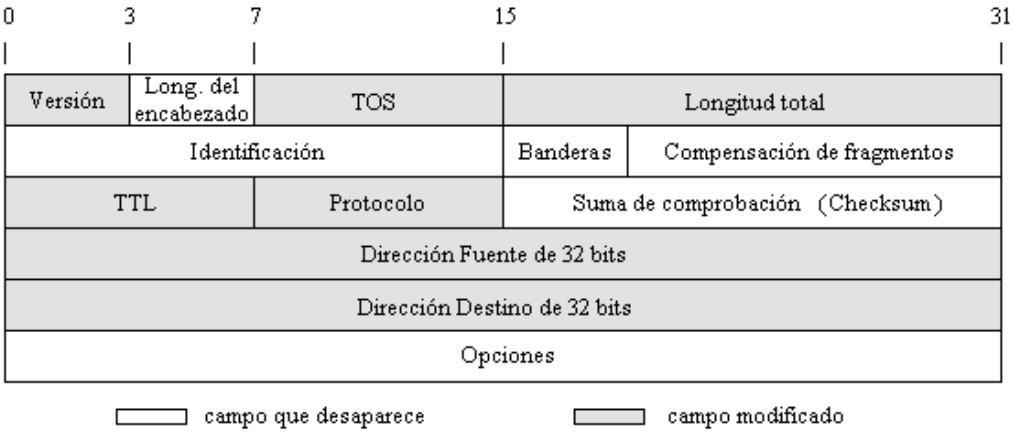


Figura II.11 Encabezado IPv4

[21] Significa que no existe ninguna conexión previa entre el origen y el destino para la transmisión de datos.

Los campos que conforman el datagrama de IPv4 son los siguientes:

- *Versión (4 bits)*: Indica el número de la versión del protocolo para permitir la evolución del mismo; en este caso es la versión 4.
- *Long. del encabezado (4 bits)*: Mide la longitud del encabezado en palabras (bloques) de 32 bits. El valor mínimo es de cinco, correspondiente a una longitud mínima del encabezado de 20 bytes.
- *TOS - Tipo de servicio (8 bits)*: Especifica los parámetros de seguridad, prioridad, retardo y rendimiento.
- *Longitud total (16 bits)*: Longitud total del datagrama en bytes.
- *Identificación (16 bits)*: Número de secuencia que se utiliza para identificar de forma única a un datagrama, por lo tanto, el identificador debe ser único tanto para la dirección origen, la dirección destino y el protocolo usado durante el tiempo en que el datagrama permanece activo.
- *Banderas (3 bits)*: Es un indicador de control que señala cómo será llevado a cabo el proceso de fragmentación. El bit 0 está reservado, el bit 1 especifica si el paquete puede ser fragmentado, y el bit 2 especifica si el paquete es el último fragmento de una serie de paquetes fragmentados.
- *Compensación de fragmentos (13 bits)*: Indica el lugar donde se sitúa el fragmento dentro del datagrama original para el reensamble, medido en unidades de 64 bits. Esto indica que todos los fragmentos excepto el último contienen un campo de datos con una longitud múltiplo de 64 bits.
- *TTL - Tiempo de vida (8 bits)*: Especifica cuánto tiempo, en segundos, se le permite a un datagrama permanecer en la red, limitado a 255 segundos. Cada dispositivo de enrutamiento que procesa el datagrama debe decrementar este campo al menos en una unidad, de forma que el tiempo de vida es de alguna manera similar a una cuenta de saltos.
- *Protocolo (8 bits)*: Indica el protocolo de nivel superior que se esta utilizando, por ejemplo TCP, UDP, ICMP, etc.
- *Checksum (16 bits)*: Código de detección y corrección de errores aplicado solamente al encabezado.

- *Dirección fuente (32 bits)*: Contiene la dirección del nodo fuente u origen.
- *Dirección destino (32 bits)*: Contiene la dirección del nodo destino.
- *Opciones (variable)*: Contiene las opciones solicitadas por el usuario que envía los datos, pueden ser opciones de seguridad, ruta pre-fijada desde el origen, registro de la ruta, registro de la hora, etc.

Como se observa, la longitud mínima del encabezado IPv4 es de 20 bytes (cada fila de la tabla supone 4 bytes), más los bytes que hay que añadir por el campo de opciones que dependen de cada caso.

En el caso de IPv6 su nuevo formato simplificado, Figura II.12, mejora la eficiencia en el enrutamiento al procesarse más rápido.

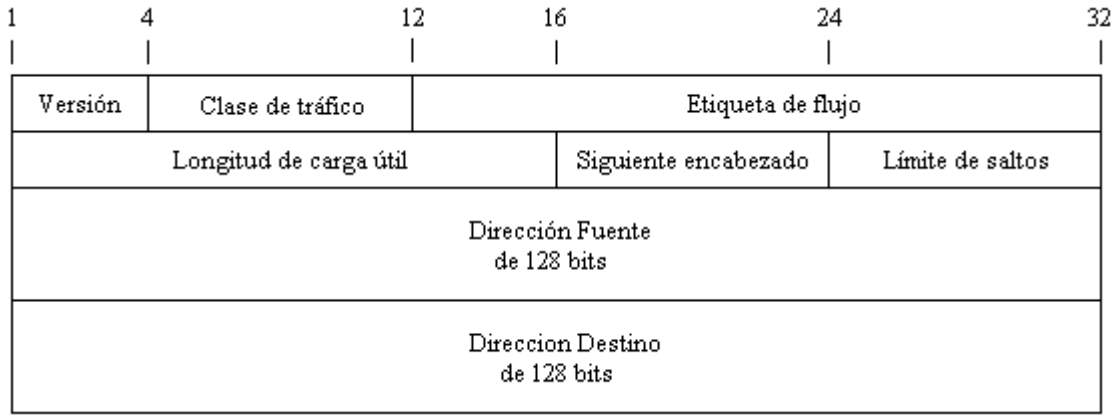


Figura II.12 Encabezado IPv6

Los campos que conforman el datagrama de IPv6 son los siguientes:

- *Versión (4 bits)*: Indica el número de la versión del protocolo, para permitir la evolución del mismo; en este caso es la versión 6.
- *Clase de tráfico (8 bits)*: Disponible para usarse por el nodo origen y/o los enrutadores de reenvío para identificar y distinguir entre diferentes clases o prioridades de paquetes IPv6.

- *Etiqueta de flujo (20 bits)*: Se utiliza para etiquetar aquellos paquetes que requieren un tratamiento especial dentro de la red, tal como la calidad de servicio no estándar o el servicio en “tiempo real”.
- *Longitud de la carga útil (16 bits)*: Mide la longitud, en octetos, de la carga, considerando a los encabezados de extensión como carga útil.
- *Siguiente encabezado (8 bits)*: Identifica el tipo de encabezado que sigue inmediatamente después del encabezado IPv6. Este campo se diseñó para decirle a los enrutadores si otro encabezado debe ser buscado y encaminar el paquete adecuadamente (ver sección II.6).
- *Límite de saltos (8 saltos)*: Número restante de saltos permitidos para este paquete. El límite de saltos es establecido por la fuente en algún valor máximo deseado, y se decrementa en 1 por cada nodo que reenvía el paquete, descartando al paquete si el límite de saltos se hace cero.
- *Dirección origen (128 bits)*: La dirección del transmisor del paquete.
- *Dirección destino (128 bits)*: La dirección del receptor del paquete. Puede que éste no sea en realidad el último destino deseado si está presente el encabezado de enrutamiento.

La longitud de este encabezado es de 40 bytes, el doble que en el caso de IPv4, pero con muchas ventajas al haberse eliminado campos redundantes.

II.5.2 Comparación y diferencias

Aunque el encabezado de IPv6 es más grande que la parte obligatoria del encabezado de IPv4 (40 bytes frente 20 bytes), contiene menos campos (8 frente a 13). De esta forma, los dispositivos de enrutamiento tienen que hacer menos procesamiento por paquete, lo que agiliza el enrutamiento eliminando una redundancia innecesaria.

Campos eliminados:

- Longitud del Encabezado: El encabezado de IPv6 tiene una longitud fija de 40 bytes haciendo innecesario este campo.
- Identificación, banderas y compensación de fragmentos: Estos tres campos en IPv4 se refieren al control de fragmentación y reensamblado de paquetes; mientras que para IPv6 estas funciones las realiza el encabezado de extensión denominado *fragmentación.*
- Checksum: En IPv4 se utiliza para comprobar la integridad del encabezado; sin embargo, las aplicaciones de capas superiores también calculan el checksum de todo el paquete, convirtiéndose este campo en redundante y no siendo necesario en IPv6. Si las aplicaciones requieren un alto grado de integridad, se puede alcanzar mediante el uso de los encabezados de extensión AH y ESP (ver capítulo III).
- Opciones: Se reemplaza por los encabezados de extensión en IPv6 debido a que en IPv4 causa un funcionamiento ineficiente en el enrutador, haciendo que cada nodo intermedio en la ruta examine este campo aunque las opciones en el mismo hagan referencia únicamente al nodo destino.

Campos renombrados:

- Longitud total ➔ Longitud de carga útil, que mide los datos que se encuentran después del encabezado, a diferencia de IPv4 que mide los datos y el encabezado. Además las cargas mayores a 65, 535 bytes están permitidas y son llamadas *jumbo cargas*.
- Protocolo ➔ Siguiente encabezado, dado que en lugar de utilizar encabezados de longitudes variables se emplean sucesivos encabezados encadenados.

- Tiempo de vida ➔ Límite de saltos, que a diferencia del IPv4 indica el máximo número de saltos que pueden ocurrir cuando el paquete es reenviado por varios nodos.

Campos agregados:

- Clase de Tráfico, también denominado Prioridad, o simplemente Clase. Este campo reemplaza las funciones que eran proporcionadas por el campo TOS en IPv4, permitiendo diferenciar los distintos tipos de paquetes dándoles un trato especial dependiendo de sus características.
- Etiqueta de flujo, para tráfico con requisitos particulares como el tiempo real.

Estos dos últimos campos son los que permiten unas de las características fundamentales e intrínsecas de IPv6: Calidad de Servicio (QoS), y Clase de Servicio (CoS), además de un poderoso mecanismo de control de flujo y de asignación de prioridades diferenciadas según los tipos de servicios.

II.6 Encabezados de extensión

En IPv6, la información opcional en un paquete está implementada en diferentes encabezados denominados “encabezados de extensión” localizados entre el encabezado principal IPv6 y el encabezado de Protocolos de Capa Superior.

Los 8 bits que contiene el campo “Siguiente encabezado” permiten identificar a 255 tipos diferentes de encabezados siguientes pero actualmente sólo se han desarrollado los que se mencionan a continuación:

1. Encabezado IPv6.
2. Encabezado de Opciones Salto a Salto.
3. Encabezado de Opciones de Destino (ser procesadas por el primer destino que aparece en el campo “Dirección Destino” del encabezado IPv6 más los destinos subsiguientes listados en el Encabezado de Enrutamiento.
4. Encabezado de Enrutamiento.
5. Encabezado de Fragmentación.
6. Encabezado de Autenticación.
7. Encabezado de Carga de Seguridad de Encapsulación.
8. Encabezado de Opciones de Destino (para ser procesado únicamente por el destino final del paquete).
9. Encabezado de Protocolos de Capa Superior.

Cuando más de un encabezado de extensión está presente en el mismo paquete deben aparecer en el orden anteriormente indicado.

A excepción del encabezado “Opciones Salto a Salto”, los encabezados de extensión son examinados o procesados solamente por el nodo destino (o varios nodos destino, en caso de usar multicast).
Cuando un nodo recibe un valor no reconocido en el campo “Siguiente Encabezado” del paquete, el nodo descarta el paquete y envía un “Problema de Parámetro” vía ICMP a la dirección origen del paquete con un código ICMP = 1.

Los encabezados “Opciones Salto a Salto” y “Opciones de Destino” contienen varias opciones de codificación TVL (Type-Length-Value) como se muestra en la Figura II.13.

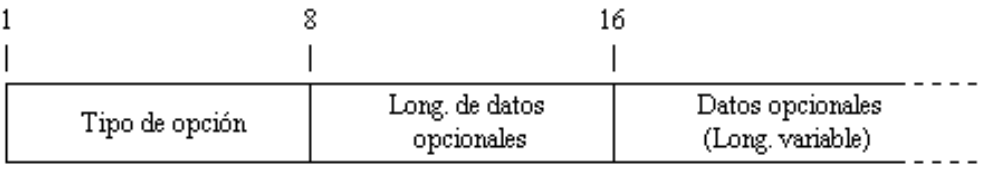


Figura II.13 Formato de opciones de codificación TVL

- *Tipo de opción (8 bits)*: Identificador para el tipo de opción.
- *Longitud de datos opcionales (8 bits)*: Longitud del campo “Datos opcionales” de esta opción en octetos.
- *Datos opcionales (variable)*: Datos opcionales de un tipo específico.

La secuencia de las opciones dentro del encabezado debe ser procesada estrictamente en el orden como aparecen en el encabezado.

Los identificadores en el campo “Tipo de opción” están internamente codificados como se muestra en la Tabla II.7, donde los dos bits más significativos especifican la acción que se debe tomar si durante el procesamiento de un paquete en un nodo IPv6 no se reconoce el tipo de opción.

Bits	***Acción a realizar***
00	Salta la opción y continúa el procesamiento del encabezado
01	Descarta el paquete
10	Descarta el paquete, y sin importar si la dirección destino del paquete fue una dirección multicast envía un mensaje de “Problema de Parámetro” vía ICMP, código 2, a la dirección origen del paquete
11	Descarta el paquete, y si solamente la dirección destino del paquete NO fue una dirección multicast envía un mensaje de “Problema de Parámetro” vía ICMP, código 2, a la dirección origen del paquete

Tabla II.7 Tipos de opciones codificadas

Los siguientes tres bits más significativos del campo “Tipo de opción” especifican si los “Datos opcionales” de determinada opción pueden cambiar la ruta de destino del paquete o no. Cuando el encabezado de Autenticación está presente en el paquete, para cualquier opción cuyos datos pueden cambiar la ruta, el campo de “Datos opcionales” debe ser considerado como octetos de valor cero debido a que la fuente del paquete ya calculó un valor de autenticación y lo colocó en el encabezado de Autenticación; por lo que cuando este conjunto de 3 bits es “0” los

"Datos opcionales" no cambian la ruta, mientras que para un valor de "1" sí pueden cambiarla.

Como los encabezados de extensión fueron diseñados para tener una longitud múltiplo de 8 octetos, en el campo de "Datos opcionales" se debe asegurar una alineación de 8 octetos como límite, por lo que se especifica en el campo "Tipo de opciones" un requerimiento de alineación de la forma *xn+y*, indicando que éste debe aparecer como un múltiplo entero "n", de "x" octetos desde el inicio del encabezado, más "y" octetos. Por ejemplo, una alineación 4*n*+2 indica que el campo "Tipo de opción" debe empezar en cualquiera de 4 octetos desplazados desde el inicio del encabezado, más 2 octetos: 2, 6, 10, 14, etc.

Además existen dos opciones de relleno, la opción Pad1 y la opción PadN, pudiendo ser usadas para forzar el contenido de las opciones del encabezado para alcanzar alguna longitud múltiplo de 8 octetos. La opción Pad1 es un caso especial y se usa para insertar un octeto con valor "0". Si más de un octeto de relleno es requerido la opción PadN es utilizada. En la Figura II.14 se muestra el formato del relleno Pad1 y PadN.

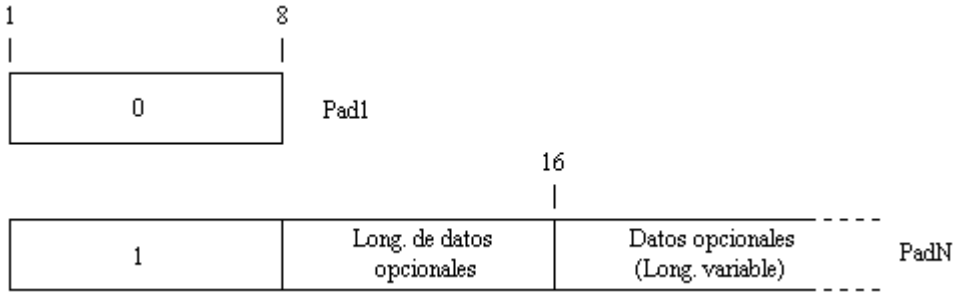


Figura II.14 Formato de relleno Pad1 y PadN

- ***Encabezado de Opciones Salto a Salto***

Este encabezado es identificado con el valor "0" en el campo "Siguiente encabezado" dentro del encabezado IPv6, y se encarga de llevar información

opcional que debe ser procesada por cada nodo a lo largo de la ruta por la cual se envía el paquete. El uso de este encabezado permite a los enrutadores examinar selectivamente los paquetes que necesitan un manejo especial, si es necesario. El formato se muestra en la Figura II.15.

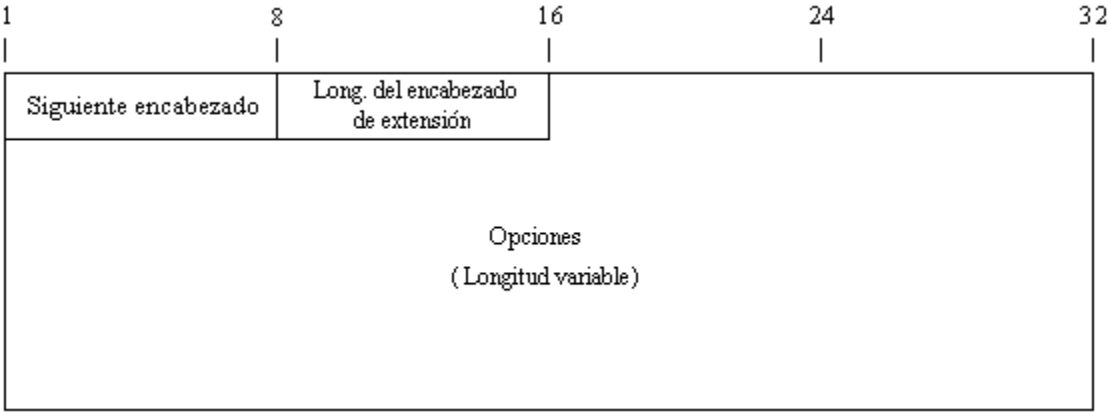


Figura II.15 Formato del encabezado de extensión Opciones Salto a Salto

- *Siguiente encabezado (8 bits)*: Indica el siguiente encabezado de extensión.
- *Longitud del encabezado de extensión (8 bits)*: Especifica el tamaño de este encabezado en bytes, no se incluyen los primeros 8 octetos.
- *Opciones (variable múltiplo de 8 octetos)*: Contiene una o más opciones codificadas TLV.

Las opciones que han sido definidas para este encabezado son: Alerta de Enrutamiento y Carga útil Jumbo. En la primera se ha definido el valor de “0” para indicar que un paquete contiene un mensaje de grupo ICMPv6, el valor de “1” para mensajes RSVP (protocolo usado para control de flujo punto a punto), y un valor de “2” para mensajes de actividad en la red. La segunda se utiliza para enviar paquetes mayores a 65,535 octetos, si el paquete es recibido con la opción de Carga útil Jumbo y la longitud de la carga útil Jumbo es menor o igual a 65,535 se envía un “Problema de Parámetro” vía ICMP a la dirección origen del paquete con un código ICMP = 0.

- ***Encabezado de Enrutamiento***

Este encabezado es identificado con el valor “43” en el campo “Siguiente encabezado” dentro del encabezado IPv6, y es utilizado por un nodo origen IPv6 para enlistar uno o más nodos intermedios que deben ser visitados en el camino del paquete hacia su destino. Su formato se muestra en la Figura II.16.

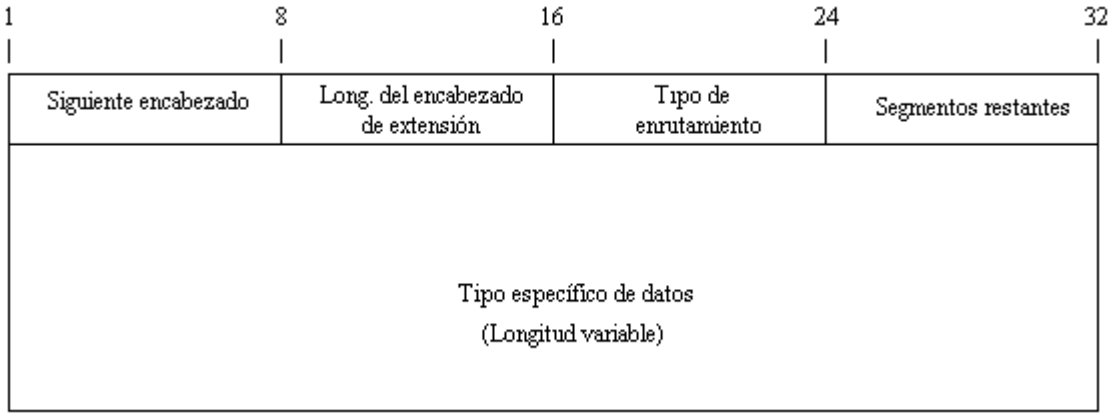


Figura II.16 Formato del encabezado de extensión Enrutamiento

- *Siguiente encabezado (8 bits)*: Indica el siguiente encabezado de extensión.
- *Longitud del encabezado de extensión (8 bits)*: Especifica el tamaño de este encabezado en bytes, no se incluyen los primeros 8 octetos.
- *Tipo de enrutamiento (8 bits*): Identifica un encabezado de enrutamiento particular.
- *Segmentos restantes (8 bits)*: Número de direcciones que quedan por visitar, es decir, el número de nodos intermedios explícitamente enlistados que deben ser visitados antes de alcanzar el destino final.
- *Tipo específico de datos (variable)*: Formato determinado por el tipo de enrutamiento y de longitud tal que el encabezado de enrutamiento sea un entero múltiplo de 8 octetos de longitud.

Cuando un nodo se encuentra con un valor “Tipo de enrutamiento” desconocido y el valor de “Segmentos restantes” es “0”, el nodo ignora el encabezado de

Enrutamiento y procede a procesar el siguiente encabezado. Sin embargo, si el campo “Segmentos restantes” no es “0” el nodo descarta el paquete y envía un “Problema de Parámetro” vía ICMP a la dirección origen del paquete con un código ICMP = 0.

El Tipo de enrutamiento = 0 se muestra en la Figura II.17.

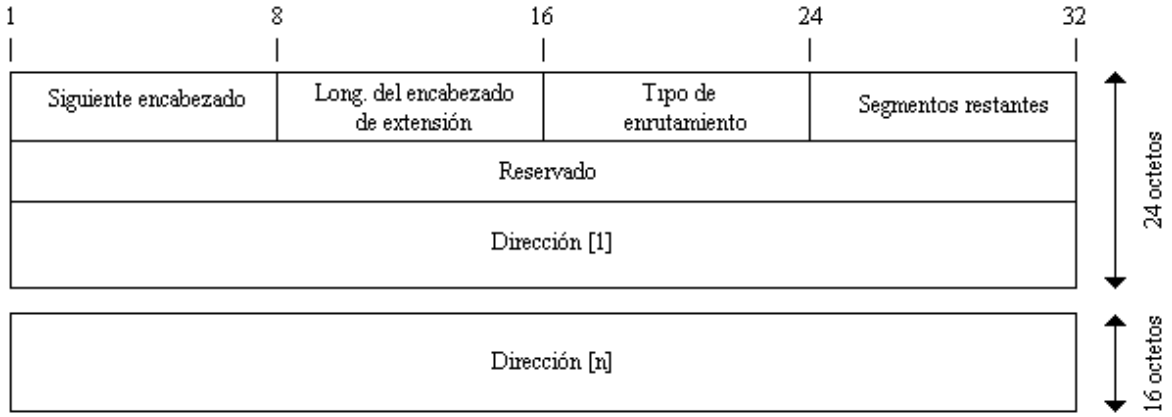


Figura II.17 Tipo de enrutamiento “0”

El campo Reservado de 32 bits se pone a cero y se ignora en la transmisión. Según se va enviando el paquete a cada nodo especificado en el encabezado de Enrutamiento, las direcciones visitadas se eliminan del paquete y se decrementa la cuenta de saltos, hasta que eventualmente el paquete llega a su destino final.

Las direcciones son un vector de 128 bits numeradas de 1 a n.

- ***Encabezado de Fragmentación***

Este encabezado es identificado con el valor “44” en el campo “Siguiente encabezado”. Su formato se muestra en la Figura II.18.

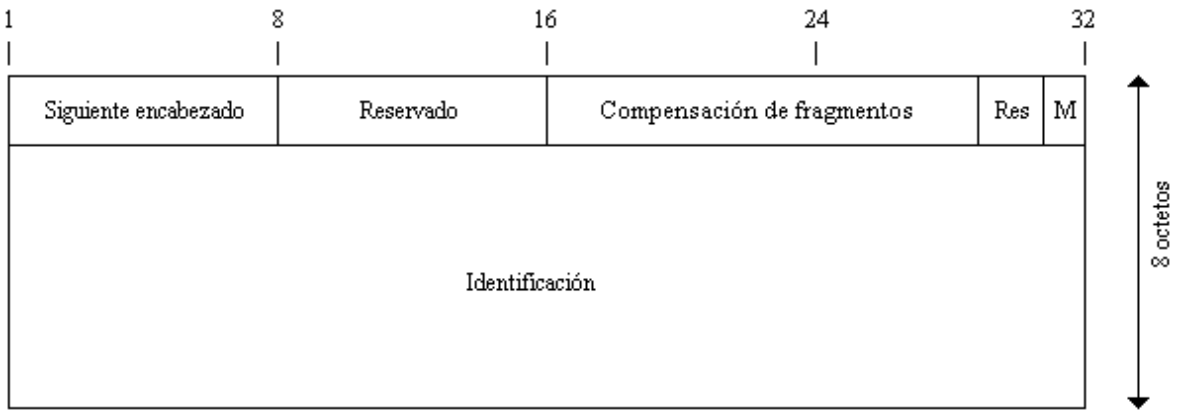


Figura II.18 Formato del encabezado de extensión Fragmentación

- *Siguiente encabezado (8 bits)*: Identifica el tipo de encabezado inicial de la parte fragmentada del paquete original.
- *Reservado (8 bits)*: Inicializado a cero en la transmisión e ignorado en la recepción.
- *Compensación de fragmentos (13 bits)*: Determina el orden de reensamblado en el nodo destino.
- *Res (2 bits)*: Inicializado a cero para la transmisión e ignorado en la recepción.
- *Bandera "M"*: Si se presenta un valor de "1" indica que hay más fragmentos, y si hay un valor de "0" indica el último fragmento.
- *Identificación (32 bits):* A cada fragmento se le asigna un valor único (identificador) para facilitar la retransmisión de paquetes perdidos.

En IPv6, el nodo de origen realiza la fragmentación, no los enrutadores como se realizaba en IPv4.

Para enviar un paquete que es demasiado grande para el tamaño óptimo de la MTU[22] (Maximum Transmission Unit) del origen hacia su destino, el nodo origen divide el paquete en fragmentos y envía cada fragmento como un paquete separado para ser reensamblados en el receptor.

[22] Término que expresa el tamaño máximo de un paquete que se puede enviar por IP.

Para cada paquete que es fragmentado, el nodo origen genera un identificador que debe ser diferente para cualquier otro paquete fragmentado recientemente con las misma Dirección Origen y Dirección Destino, sin embargo, la presencia de un encabezado de Enrutamiento puede requerir que los nodos intermedios fragmenten el paquete como resultado de una MTU distinta en la ruta ya que, como cada uno de los saltos se convierte en nodo de origen, según se envía el paquete a la siguiente dirección, al nodo sólo le interesará la MTU del enlace entre él mismo y el siguiente destino, en lugar de conocer la MTU de todos los enlaces de la red.

Cada paquete fragmento está compuesto de:

- La parte no fragmentada del paquete original, que consiste en el encabezado IPv6 y sus encabezados de extensión (si existen) que han de tratarse en cada nodo en la ruta de envío.
- El encabezado de Fragmentación, que contiene el valor del “Siguiente encabezado” que identifica el primer encabezado de la parte fragmentada del paquete original; y la “Compensación de fragmentos” que contiene el desplazamiento del fragmento en unidades de 8 octetos.
- El fragmento mismo. Las longitudes de los fragmentos deben ser elegidas como el resultado de los paquetes fragmentados adecuados a la MTU de la ruta desde el origen hasta el destino final.

En el destino se reensambla el paquete original a partir de los paquetes fragmentados que tienen la misma dirección origen, dirección destino, e identificación del fragmento.

- ***Encabezado de Autenticación (AH).***

Este encabezado es identificado con el valor “51” en el campo “Siguiente encabezado” dentro del encabezado IPv6, y se usa para proporcionar autenticación e integridad en el origen de los datos y opcionalmente servicios de anti-réplica a los paquetes IP, aunque **no** proporciona ninguna garantía de confidencialidad al no proveer cifrado de datos.
Además, proporciona autenticación al encabezado IPv6, a los encabezados de protocolos de capa superiores, los datos del usuario, y a los encabezados de extensión que no cambian la ruta; por ejemplo, el campo “Dirección Destino” en el encabezado IPv6 cambia en todos lo saltos cuando en el encabezado de enrutamiento tipo 0 es utilizado, por lo que en este caso AH no proporciona autenticación en éste. (ver capítulo III.9.1).

- ***Encabezado de Carga de Seguridad de Encapsulación (ESP).***

Este encabezado es identificado con el valor “50” en el campo “Siguiente encabezado” dentro del encabezado IPv6, y es usado para proporcionar *confidencialidad*, para ello especifica el modo de cifrar los datos que se desean enviar y cómo este contenido cifrado se incluye en un paquete IP. Adicionalmente, puede ofrecer servicios de anti-réplica, integridad y autenticación del origen de los datos incorporando un mecanismo similar a AH. (ver capítulo III.9.2).

- ***Encabezado de Opciones de Destino.***

Este encabezado es identificado con el valor “60” en el campo “Siguiente encabezado” dentro del encabezado IPv6 y es casi idéntico al encabezado “Opciones Salto a Salto”, con la diferencia que en éste encabezado sólo se

examina en el nodo destino el paquete y no en los nodos intermedios de la ruta; los nodos móviles utilizan este encabezado.

El resto de los campos son idénticos a los del encabezado “Opciones Salto a Salto” como se muestra en la Figura II.19.

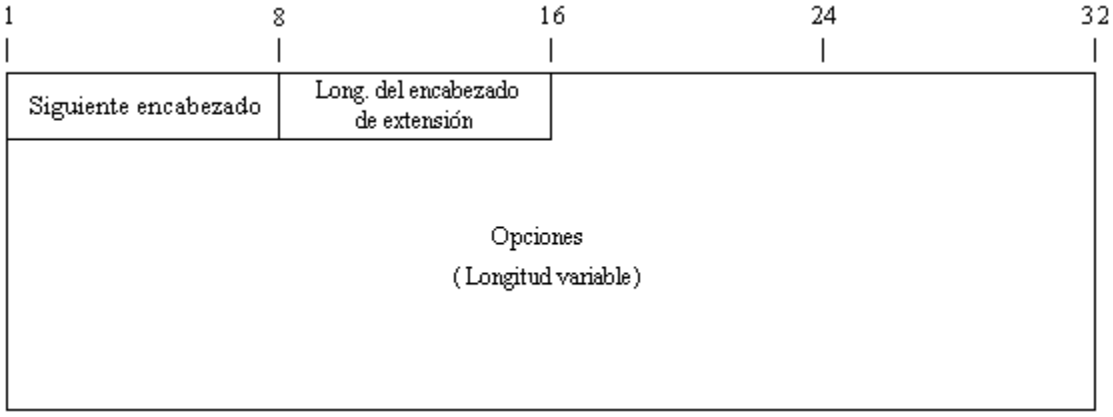


Figura II.19 Formato del encabezado de extensión Opciones de Destino

- *Siguiente encabezado (8 bits)*: Indica el siguiente encabezado de extensión.
- *Longitud del encabezado de extensión (8 bits)*: Especifica el tamaño de este encabezado en bytes, no se incluyen los primeros 8 octetos.
- *Opciones (variable múltiplo de 8 octetos)*: Contiene una o más opciones codificadas TLV.

Encabezado de Protocolos de Capa Superior

Los encabezados de capa superior (nivel transporte) son los encabezados típicos usados dentro del paquete para transportar los datos, como por ejemplo TCP con un valor de “6”, UDP con un valor de “17” o ICMP con un valor de “58”.

II.7 Formas de coexistencia en ambas versiones IP

Los diseñadores de IPv6 reconocieron que la transición de IPv4 a IPv6 tomaría muchos años y que habría organizaciones o hosts dentro de organizaciones que continuarían con el uso de IPv4 para siempre; sin embargo, a pesar que la transición es a largo plazo, existen ciertas consideraciones que permitirán la coexistencia de ambas versiones, donde en la especificación original, RFC 1752, se definen los siguientes criterios de transición.

- Los hosts existentes con IPv4 podrán ser actualizados en cualquier momento, independientemente de la actualización de los otros hosts o enrutadores.
- Los nuevos hosts, utilizando sólo IPv6, pueden ser agregados en cualquier momento, sin dependencias de otros hosts o infraestructura de enrutamiento.
- Los hosts existentes con IPv4, con IPv6 instalado, pueden continuar utilizando direcciones IPv4 sin necesitar direcciones adicionales.
- Se requiere de una pequeña preparación ya sea para actualizar de IPv4 a IPv6 o para implementar nuevos nodos IPv6.

La inherente falta de dependencia entre los hosts de IPv4 e IPv6 y la infraestructura de enrutamiento IPv4 e IPv6, se requiere de mecanismos que permitirán la coexistencia de las dos versiones de manera transparente.

II.7.1 Tipos de nodos

Aunado a los criterios de transición de IPv4 a IPv6, en el RFC 4213 se definen los siguientes tipos de nodos:

- Nodo sólo con IPv4: Es un nodo con IPv4 y que sólo tiene direcciones IPv4. Este tipo de nodo no soporta IPv6. La mayoría de los hosts y enrutadores instalados hoy en día son sólo con IPv4.
- Nodo sólo con IPV6: Es un nodo con IPv6 y que sólo tiene direcciones IPv6. Este nodo sólo se puede comunicar con nodos y aplicaciones IPv6. Este tipo de nodo no es muy común hoy en día.
- Nodo IPv6/IPv4: Es un nodo que tiene implementado tanto IPv4 como IPv6. Este nodo permite IPv6 sólo si tiene configurado una interfaz IPv6.
- Nodo IPv4: Es un nodo con IPv4, puede enviar y recibir paquetes IPv4. Un nodo IPv4 puede ser un nodo sólo con IPv4 o un nodo IPv6/IPv4.
- Nodo IPv6: Es un nodo con IPv6, puede enviar y recibir paquetes IPv6. Un nodo IPv6 puede ser un nodo sólo con IPv6 o un nodo IPv6/IPv4.

II.7.2 Modos de operación

- Operación sólo con IPv6: El nodo IPv6/IPv4 trabaja con la pila IPv6 habilitada y la pila IPv4 deshabilitada.
- Operación sólo con IPv4: El nodo IPv6/IPv4 trabaja con la pila IPv4 habilitada y la pila IPv6 deshabilitada.
- Operación IPv6/IPv4: El nodo IPv6/IPv4 trabaja con las dos pilas habilitadas.

Para que ocurra la coexistencia, los nodos en mayor número (IPv4 o IPv6) deberán comunicarse utilizando infraestructura IPv4, una infraestructura IPv6 o una infraestructura que sea una combinación de IPv4 e IPv6. La verdadera transición se realizará cuando todos los nodos IPv4 sean convertidos a nodos sólo con IPv6; sin embargo, en el presente, la transición práctica está dada por los nodos sólo con IPv4 convertidos en nodos IPv6/IPv4.

II.7.3 Mecanismos de transición

Para coexistir con una infraestructura IPv4 y proporcionar una transición eventual a una infraestructura sólo con IPv6, los siguientes mecanismos son utilizados:

- Doble pila.
- Túneles.
- Traductores.

II.7.3.1 Doble pila

La Doble Pila representada en la Figura II.20, es una implementación de la pila de protocolos TCP/IP que incluye tanto IPv4 como IPv6. Los nodos IPv6 que proporcionan implementaciones de IPv4 e IPv6 son llamados nodos IPv6/IPv4 y tienen la habilidad de enviar y recibir paquetes IPv4 e IPv6.

Todos los protocolos de capas superiores en una implementación de doble pila pueden comunicarse sobre IPv4 (interoperar con nodos IPv4 utilizando paquetes IPv4), IPv6 (interoperar con nodos IPv6 utilizando paquetes IPv6), o ambos (IPv6 con túnel en IPv4).

Las aplicaciones que se comuniquen a través de una doble pila podrán elegir en utilizar IPv4 o IPv6, implicando que en los nodos IPv6/IPv4 se configuren direcciones IPv6 e IPv4, tener implementado un sistema DNS[23] (Domain Name System) tanto para las direcciones IPv6 utilizando registros AAAA (los registros A6 fueron registros solo para uso experimental, RFC 3363) como para las direcciones IPv4 usando los registros A, además de configurar requerimientos particulares en la comunicación y el tipo de tráfico que se vayan a utilizar.

[23] Es una base de datos distribuida y jerárquica que almacena información asociada a nombres de dominio en redes como Internet.

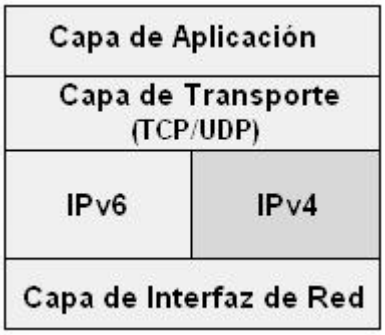


Figura II.20 Arquitectura de Doble Pila

II.7.3.2 Túneles

El túnel IPv6 sobre IPv4 es la encapsulación de paquetes IPv6 con un encabezado IPv4 para que los paquetes IPv6 puedan ser enviados sobre una infraestructura IPv4, como se muestra en la Figura II.21.

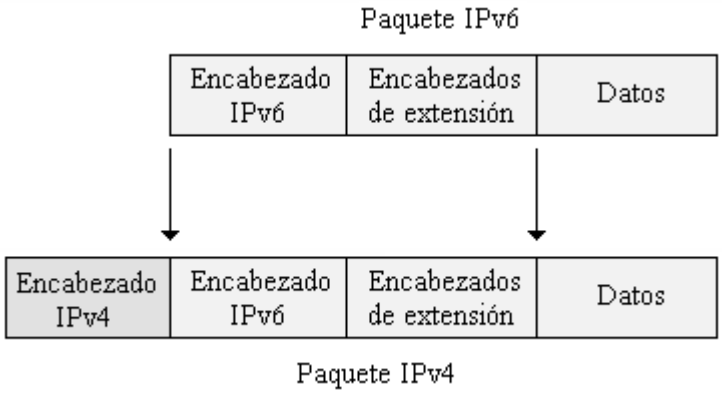


Figura II.21 Estructura de paquetes para túnel IPv6 sobre IPv4

Los nodos IPv6/IPv4 pueden soportar la configuración de túnel o no, existiendo tres maneras de funcionamiento:

- Nodo IPv6/IPv4 que no soporta el funcionamiento de túnel.
- Nodo IPv6/IPv4 que funciona sólo con un túnel manualmente configurado.

- Nodo IPv6/IPv4 que funciona con túnel manualmente configurado y con configuración automática.

Los túneles pueden ser configurados de 4 formas diferentes para que se pueda transportar el tráfico IPv6 entre los nodos IPv6/IPv4 dentro de la infraestructura IPv4.

1.- Enrutador a Enrutador. Enrutadores con doble pila (IPv6/IPv4) que se conectan mediante una infraestructura IPv4 y transmiten tráfico IPv6 a través de un túnel que comprende un enlace punto a punto por donde se envían los paquetes, en la Figura II.22 se muestra este tipo de configuración.

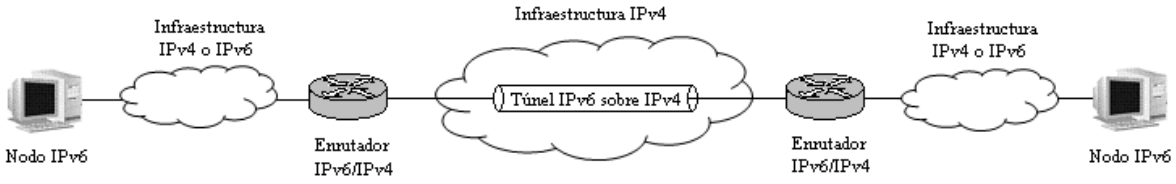


Figura II.22 Configuración de túnel Enrutador a Enrutador

2.- Host a Enrutador. Hosts con doble pila que se conectan a un enrutador intermedio (también con doble pila) alcanzable mediante una infraestructura IPv4 mediante un túnel que comprende el primer segmento de la ruta punto a punto por donde se envían los paquetes.

3.- Enrutador a Host. Enrutadores con doble pila que se conectan a hosts también con doble pila por medio de un túnel correspondiente al último segmento de la ruta punto a punto por donde se envían los paquetes.
En la Figura II.23 se tiene el esquema Host a Enrutador (para el tráfico que va del nodo A al nodo B) y Enrutador a Host (para el tráfico que va del nodo B al nodo A).

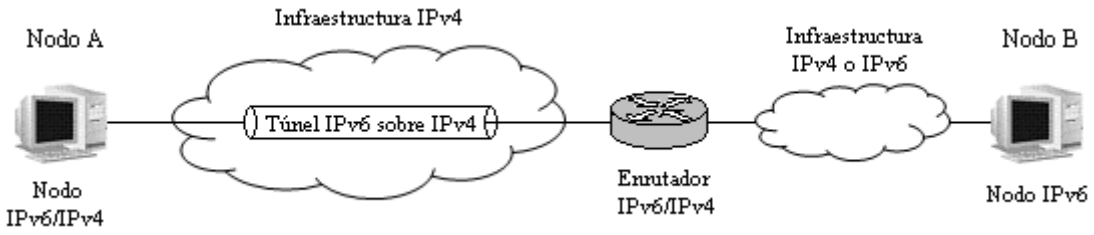


Figura II.23 Configuración de túnel Host a Enrutador y Enrutador a Host

4.- Host a Host. Hosts con doble pila que se conectan mediante una infraestructura IPv4 y transmiten tráfico IPv6 a través de un túnel que comprende la ruta completa punto a punto por donde se envían los paquetes, en la Figura II.24 se muestra esta configuración.

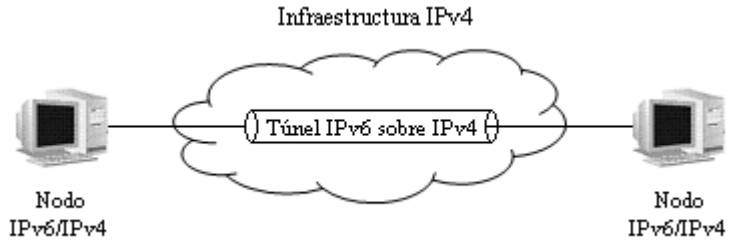


Figura II.24 Configuración de túnel Host a Host

Existen dos tipos de túneles: **configurados** y **automáticos**. En los primeros se requiere una configuración manual en los puntos finales del túnel, típicamente la configuración del modo enrutador a enrutador es manualmente configurado. En los túneles automáticos no se requiere una configuración manual sino que los puntos finales del túnel se determinan por el uso de las interfaces lógicas involucradas en el túnel, rutas y direcciones IPv6 de origen y destino.

Dentro de los túneles automáticos se encuentran los siguientes tipos:

- Túnel automático IPv6
- 6to4
- ISATAP
- 6over4
- Teredo

Túnel automático IPv6

El túnel automático en IPv6 ocurre cuando las direcciones compatibles con IPv4 son utilizadas, teniendo el formato ::a.b.c.d donde a.b.c.d es una dirección pública de IPv4. Este túnel automático es un túnel host-host entre dos hosts de doble pila.
Para realizar la prueba de conectividad se puede usar el comando *ping*, por ejemplo: ping ::10.0.0.1.

6to4

Es un mecanismo definido en el RFC 3056, y determinado al modo enrutador-enrutador utilizado para proporcionar conectividad unicast en IPv6 entre sitios y hosts IPv6 a través de una infraestructura IPv4. Utiliza el prefijo global 2002::/16 de la forma 2002:AABB:CCDD/48 donde AABB:CCDD es la representación hexadecimal de una dirección IPv4 pública asignada al sitio o host (a.b.c.d).
La dirección completa 6to4 entonces es 2002:AABB:CCDD:ID_subred:ID_interfaz.

Este mecanismo define los siguientes términos mostrados en la Figura II.25

- Host 6to4: Cualquier host IPv6 que es configurado con al menos una dirección 6to4 (el direccionamiento global tiene el prefijo 2002::/16). Los hosts 6to4 no

requieren configuración manual y crean las direcciones 6to4 usando el formato mencionado anteriormente.

- Enrutador 6to4: Es un enrutador de doble pila que soporta el uso de interfaces 6to4 y es utilizado para enviar tráfico con direcciones 6to4 entre hosts 6to4 dentro de un sitio y enrutadores 6to4 o enrutadores retransmisores 6to4 sobre una infraestructura IPv4.
- Enrutador retransmisor 6to4: Enrutador de doble pila utilizado para enviar tráfico con direcciones 6to4 entre enrutadores 6to4 y no 6to4 sobre IPv4 e IPv6.

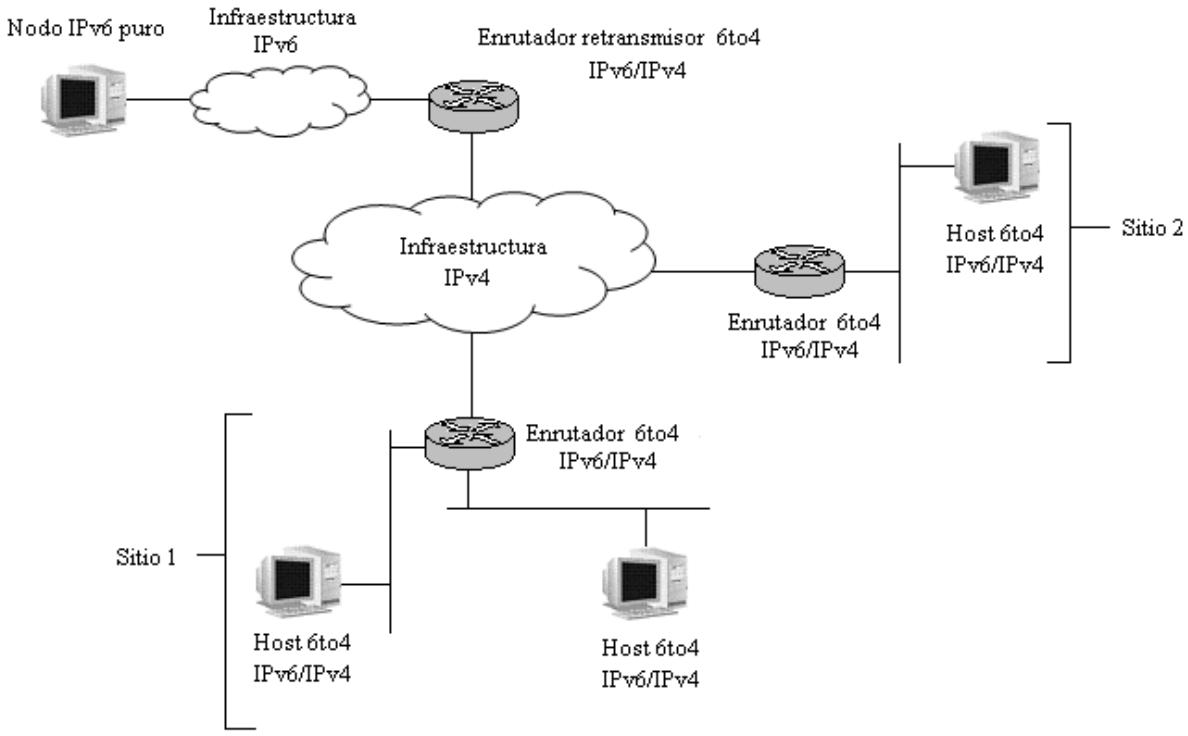


Figura II.25 Elementos de una red utilizando 6to4

- ***ISATAP (Intra-Site Automatic Tunnel Addressing Protocol)***

Es un mecanismo asignado a los modos host-host, host-enrutador y enrutador-host utilizada para proporcionar conectividad unicast en IPv6 entre hosts IPv6 a

través de una intranet IPv4, o bien entre nodos IPv6/IPv4 a través de una infraestructura IPv4, definida en el RFC 4214 (experimental).

El prefijo que utiliza es el ::0:5EFE:a.b.c.d donde a.b.c.d es una dirección IPv4 unicast asignada a una interfaz, conteniendo direcciones privadas y públicas.

El identificador de una interfaz ISATAP puede ser combinada con cualquier prefijo de 64 bits válido para las direcciones unicast en IPv6, incluyendo el prefijo de la dirección de enlace local (FE80::/64) y prefijos globales (incluyendo el prefijo 6to4).

En la Figura II.26 se muestra un ejemplo de configuración usando direcciones de enlace local ISATAP, donde el host A se configura con la IPv4 192.168.41.30 y el host B con la IPv4 10.40.1.29. Cuando el encabezado IPv6 es habilitado los hosts automáticamente configuran sus direcciones ISATAP, FE80::5EFE:192.168.41.30 y FE80::5EFE: 10.40.1.29, respectivamente.

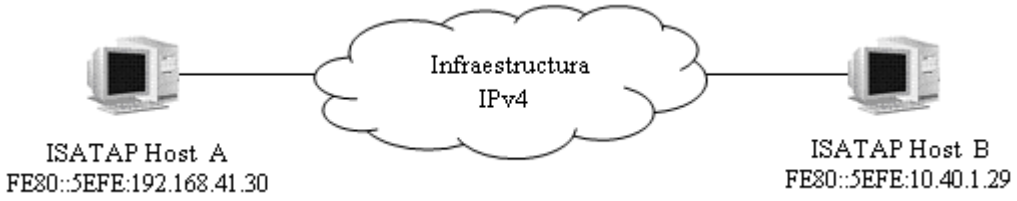


Figura II.26 Ejemplo de una configuración ISATAP

- ***Teredo***

También es conocido como un traductor de direcciones de red IPv4 (NAT) para IPv6 y es utilizado en un modelo host-host para una conectividad unicast en IPv6 a través de una infraestructura IPv4, siempre y cuando los hosts IPv6/IPv4 estén localizados detrás de uno o múltiples NATs IPv4. Para atravesar los NATs IPv4, los paquetes IPv6 son enviados como mensajes UDP basados en IPv4.
Este mecanismo define los siguientes elementos, mostrados en la Figura II.27.

- Cliente Teredo: Es un nodo IPv6/IPv4 que soporta túneles con base en Teredo a través de los cuales se envían los paquetes hacia otro Cliente Teredo o nodos basados en IPv6 (a través del retransmisor Teredo).
- Servidor Teredo: Es un nodo IPv6/IPv4 que está conectado tanto en IPv6 como en IPv4 ayudando a la configuración inicial de los Clientes Teredo y facilitar la comunicación inicial entre ellos con los hosts sólo con IPv6.
- Teredo retransmisor: Es un enrutador IPv6/IPv4 que puede reenviar paquetes entre los Clientes Teredo sobre IPv4 y los hosts sólo con IPv6.
- Teredo retransmisor de un host en específico: Es un nodo IPv6/IPv4 que está conectado tanto en IPv6 como en IPv4 para poder comunicarse directamente con Clientes Teredo sobre IPv4, sin la necesidad de un Teredo retransmisor intermedio.

Es importante señalar que Teredo se diseñó como un último recurso en los mecanismos de transición para la conectividad con IPv6, ya que si existe conectividad con una infraestructura IPv6, 6to4 o ISATAP y los nodos que se quieran comunicar, Teredo no es utilizado. Por otro lado, como muchos NATs IPv4 se actualizan constantemente para soportar 6to4, una conectividad con IPv6 y Teredo se usará con menor frecuencia hasta que eventualmente se deje de utilizar, aunque en ambientes donde se tengan implementados múltiples NATs no es conveniente trabajar con 6to4 ya que no está soportado.

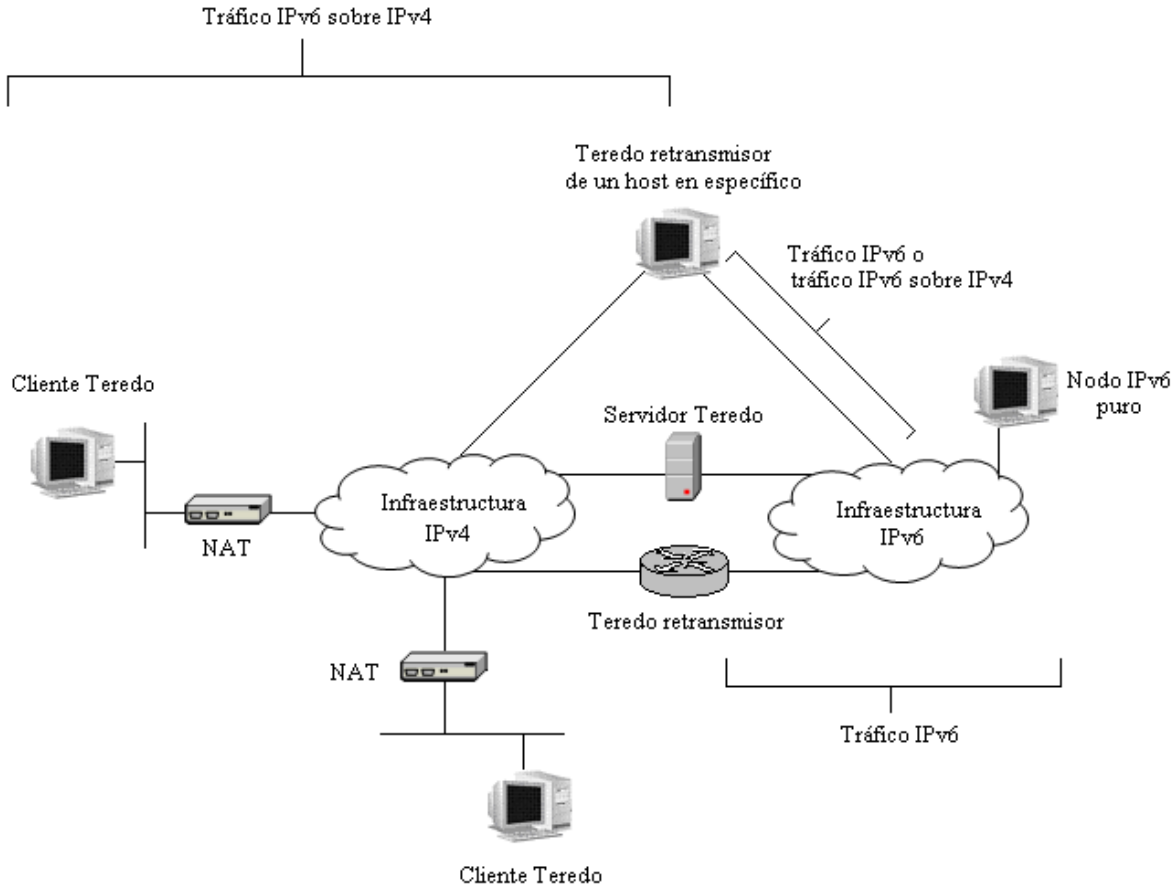


Figura II.27 Elementos de una red utilizando Teredo

- ***6over4***

Este mecanismo, también conocido como túnel multicast de IPv4, se utiliza en configuraciones host-host, host-enrutador y enrutador-host para proporcionar conectividad unicast y multicast en IPv6 entre nodos IPv6 que atraviesan una intranet IPv4, RFC 2529. Los hosts 6over4 usan un prefijo válido de 64 bits para las direcciones unicast teniendo el formato ::AABB:CCDD donde AABB:CCDD es la representación hexadecimal de a.b.c.d, una dirección IPv4 unicast asignada a una interfaz. Por defecto, los hosts configuran automáticamente la dirección de enlace local en cada interfaz 6over4, por ejemplo una dirección 6over4 de enlace local de la dirección IPv4 131.107.4.92 sería FE80::836B:45C.

6over4 considera una infraestructura IPv4 como un enlace único con capacidades multicast, es decir, se utiliza multicast IPv4 como su "ethernet virtual", de esta forma los hosts IPv6 no requieren direcciones IPv4 compatibles, ni túneles configurados, y los extremos finales del túnel se determinan mediante el proceso de descubrimiento de vecino ND (Neighbor Discovery); aunque es imprescindible que la infraestructura IPv4 soporte multicast y lo tenga habilitado. En la Figura II.28 se muestra este tipo de configuración.

Para facilitar la comunicación multicast en IPv6 sobre la infraestructura IPv4 con multicast habilitado, se definió el siguiente mapeo para traducir las direcciones multicast en IPv6 a direcciones multicast en IPv4:
239.192.[del segundo al último byte de la dirección IPv6].[último byte de la dirección IPv6]

Por ejemplo:

- FF02::1 se mapea a 239.192.0.1 → Dirección multicast que representa todos los nodos de ámbito local.
- FF02::2 se mapea a 239.192.0.2 → Dirección multicast que representa todos los enrutadores de ámbito local.
- FF02::1:FF28:9C5A se mapea a 239.192.156.90 → Solicitud de un nodo usando una dirección multicast.

No obstante, cabe mencionar que este mecanismo es poco utilizado ya que se requiere de multicast en IPv4.

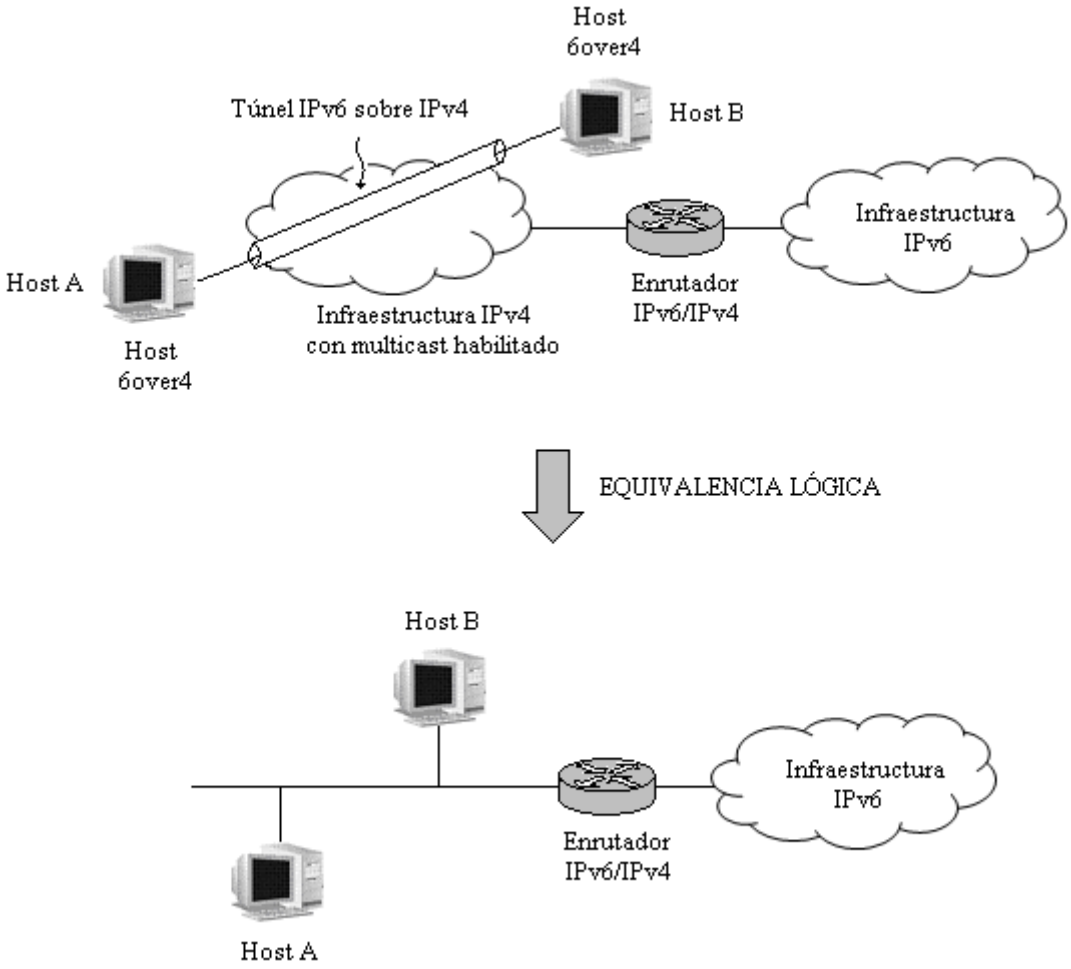


Figura II.28 Elementos de una red utilizando 6over4

Por otro lado, se tiene el concepto de "**tunnel broker**", RFC 3053, presentado por primera vez por el IETF en Orlando en diciembre de 1998, y se refiere a ISPs con IPv6 "virtuales" que proporcionan una conectividad para acceder a redes y sitios IPv6 por medio de un túnel a usuarios que ya tienen conectividad IPv4, Figura II.29, sin la necesidad de configurar un enrutador.

El "tunnel broker" es el lugar donde el usuario se conecta para registrar y activar su túnel y tiene la función de gestionar la creación, modificación y eliminación del túnel por parte del usuario.

El "**tunnel server**" es un enrutador de doble pila conectado a Internet que siguiendo órdenes del "tunnel broker" crea, modifica o elimina los servicios asociados a un determinado túnel.

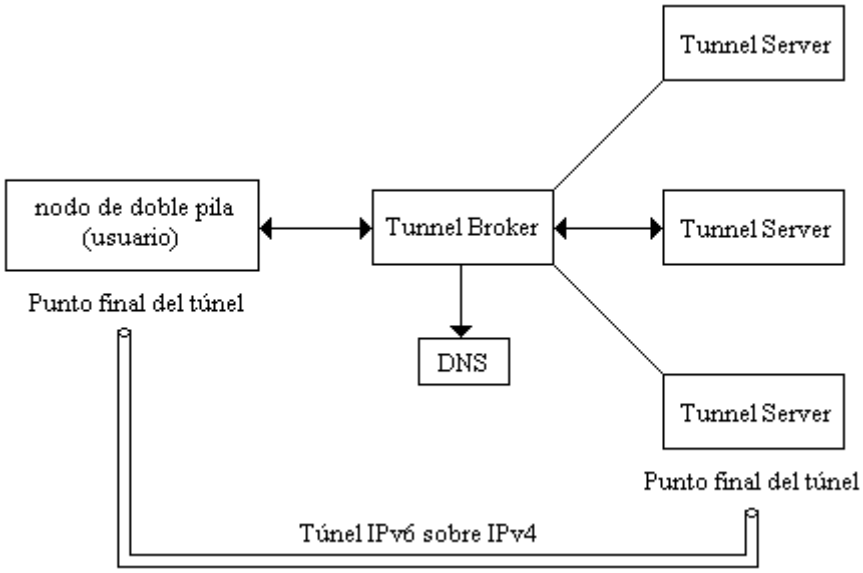


Figura II.29 Modelo “Tunnel Broker”

La lista de “tunnel brokers” son referenciados como páginas Web con soporte IPv6 “bien conocidas” (por ejemplo, http://www.ipv6.org) para permitir a los usuarios escoger el más cercano, el más económico, o algún otro.

II.7.3.3 Traductores

Los mecanismos de traducción pueden utilizarse de dos formas diferentes: Por un lado, el tráfico IPv6 puede traducirse a IPv4 y posteriormente convertirse nuevamente a IPv6, proporcionando un medio de comunicación entre dos hosts IPv6 aislados, o bien, proporciona los medios para que un host sólo con IPv6 pueda comunicarse con un host sólo con IPv4, por ejemplo, en escenarios donde existe una red completamente nueva con hosts que soportan IPv6 o en escenarios

donde ya existe una red y se añaden hosts IPv6 que se quieren comunicar con hosts IPv4.

Actualmente existen diversos mecanismos de traducción que ofrecen diferentes soluciones, como los que se mencionan a continuación:

- ***SIIT (Stateless IP/ICMP Translator)***

Es un mecanismo, definido en el RFC 2765, especifica la traducción de encabezados IP/ICMP entre nodos sólo con IPv4 e IPv6, haciendo la traducción para todos los paquetes no tomando en cuenta el estado del paquete (stateless).

SIIT se limita a no traducir las opciones del encabezado IPv4, así como los encabezados de Enrutamiento, “Opciones Salto a Salto” y “Opciones de Destino” del encabezado IPv6, además de ser imposible aplicar técnicas para tráfico multicast.

En la Figura II.30 se muestra el uso de SIIT para una sola subred con hosts IPv6, y en la Figura II.31 se muestra el uso de SSIT para hosts sólo con IPv6 o una infraestructura IPv6/IPv4 que contiene hosts IPv6 y hosts IPv4.

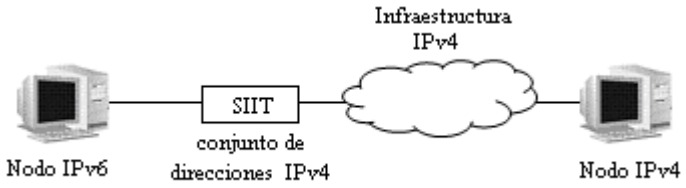


Figura II.30 Uso de SIIT para una subred con nodos IPv6

Figura II.31 Uso de SIIT para una infraestructura IPv6/IPv4

- ***BIS (Bump in the Stack)***

Este mecanismo de doble pila, RFC 2767, permite a los hosts IPv4 comunicarse con otros hosts IPv6 utilizando aplicaciones IPv4 y viceversa. Para lograr este propósito a cada host IPv4 se le agregan tres módulos: extensiones al resolvedor de nombres, un mapeador de direcciones, y un traductor (utilizando el mecanismos de conversión IP SIIT) a su pila en lugar de aplicaciones IPv6.
Sin embargo, los inconvenientes que presenta son: tener que modificar a todos los nodos, no permitir comunicaciones multicast, y no permite traducir opciones de IPv4 e IPv6 (exceptuando los encabezados de Fragmentación y Enrutamiento). En la Figura II.32 se observa su composición.

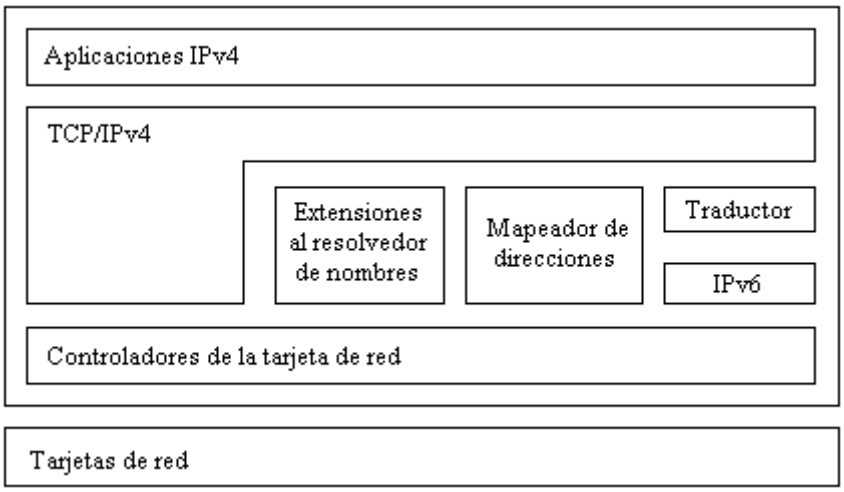


Figura II.32 Estructura BIS

- ***BIA (Bump in the API)***

Es un mecanismo definido como experimental en el RFC 3338 para hosts de doble pila, Figura II.33, en donde se inserta un traductor API[24] (Application Programming Interface) entre el módulo del socket[25] API y el módulo TCP/IP en los hosts de doble pila, de modo que se traduzcan las funciones del socket API IPv4 dentro de las funciones del socket API IPv6 y viceversa, es decir, cuando las aplicaciones IPv4 en una doble pila se comunican con otros hosts IPv6, el traductor API detecta las funciones del socket API para comunicarse con los hosts IPv6, y viceversa.
Con este mecanismo, la traducción puede ser simplificada sin la traducción del encabezado IP; sin embargo, necesita traducir direcciones IP embebidas[26] en protocolos de nivel de aplicación, por ejemplo FTP, de tal modo que podría no trabajar para nuevas aplicaciones con direcciones IP embebidas en la carga útil; solamente soporta comunicaciones unicast; y como los APIs IPv6 son una nueva característica son difíciles de traducir ciertos tipos dentro de APIs IPv4.

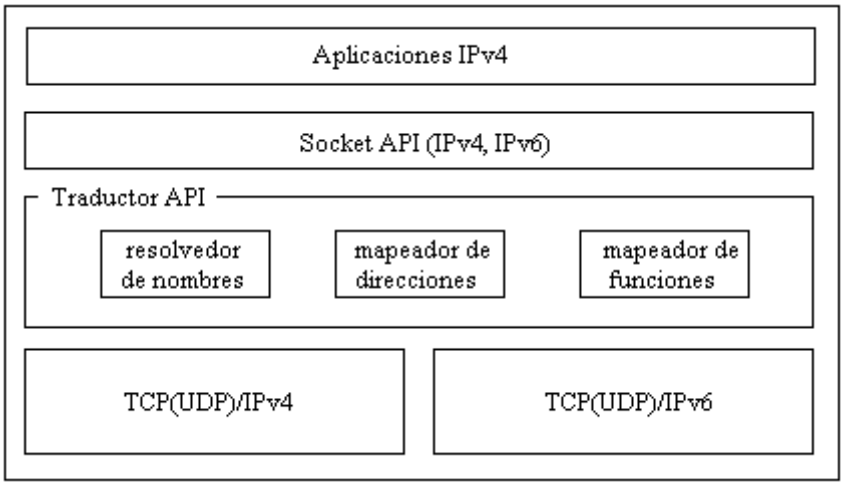


Figura II.33 Estructura BIA

[24] Es un conjunto de rutinas que permiten que una aplicación se ejecute en un determinado sistema operativo, es decir, representa un interfaz de comunicación entre componentes software.
[25] Interfaz de comunicación que ofrece un mecanismo de comunicación general entre dos procesos cualquiera que pertenezcan a un mismo sistema o dos sistemas diferentes.
[26] Se refiere a que los hosts IPv6 se les puede asignar direcciones compatibles con IPv4 y las direcciones IPv4 se mapean a IPv6.

- ***NAT-PT (Network Address Translation - Protocol Translation)***

NAT-PT, definido en un principio en el RFC 2766, se coloca como una puerta de enlace entre dos redes para traducir todas las direcciones de los paquetes que pasan a través de él, es decir, permite a los hosts dentro de una red IPv6 comunicarse con hosts dentro de una red IPv4, y viceversa, Figura II.34.
Este mecanismo combinado con SIIT y ALGs, proporciona una completa solución para permitir la comunicación entre nodos sólo con IPv6 y sólo con IPv4.

Al igual que BIA la traducción puede ser simplificada sin la traducción del encabezado IP; sin embargo, necesita traducir direcciones IP embebidas en protocolos del nivel de aplicación.

Este mecanismo ha pasado a ser solamente informativo para limitar la posibilidad de que sea implementado inapropiadamente, RFC 4966.

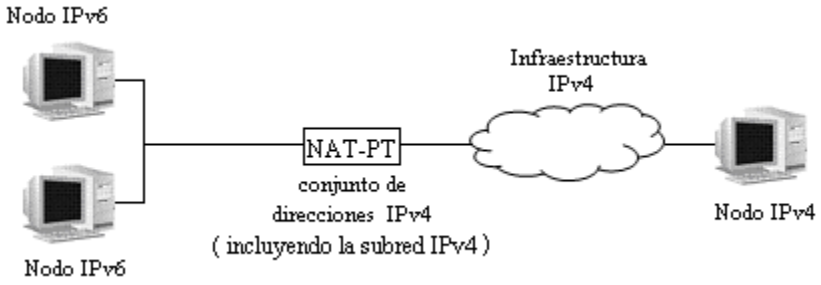


Figura II.34 Uso de NAT-PT

- ***ALG (Application Level Gateway)***

Un ALG, RFC 2962, es una aplicación que permite la traducción de ciertos paquetes que contienen información de direcciones IP, como pasa por ejemplo con los protocolos FTP y DNS que usan las direcciones IP como parte del propio protocolo.

- ***SOCKS - based IPv6/IPv4 Gateway***

Es una puerta de enlace (gateway) entre dos redes que permite que ciertas aplicaciones se comuniquen con sus contrapartes en la otra red, en este caso desde una red IPv4 a una IPv6 o viceversa.
La comunicación a través de un servidor SOCKS es dependiente de la aplicación, esto quiere decir que, si alguna aplicación no tiene soporte para SOCKS no se va a poder comunicar con su contraparte.
Se encuentra definido en el RFC 3089 y está basado en SOCCKSv5.

- ***TRT (Transport Relay Translator)***

Este mecanismo, definido en el RFC 3142, se encarga de traducir conexiones TCP y paquetes UDP. Se coloca también como una puerta de enlace entre las dos redes obteniendo una comunicación bidireccional directa y transparente.
Entre una de sus ventajas se puede mencionar que no es necesaria una modificación extra en los nodos sólo con IPv6; y como desventajas es que sólo soporta conexiones bidireccionales, necesita un sistema stateful entre los que realizan la comunicación, y no soporta IPSec entre otros protocolos.

Dado que los mecanismos de traducción dependen normalmente del proceso de conmutación de paquetes, experimentan limitaciones significativas en el rendimiento, limitaciones de escalabilidad y puntos únicos de falla, que deberán tenerse en cuenta cuando se integren en un despliegue planificado de IPv6.

Todos los mecanismos anteriormente mencionados están diseñados para ser usados por hosts y enrutadores IPv6 que necesitan interoperar con hosts IPv4 y utilizar infraestructuras de enrutamiento IPv4, estimando que muchos nodos

necesitarán ésta compatibilidad por mucho tiempo o quizás indefinidamente. No obstante, IPv6 también puede ser usado en ambientes donde no se requiere interoperabilidad con IPv4 sin necesidad de usar ni implementar ninguno de estos mecanismos.

En la Tabla II.8 se presenta una comparación entre las versiones de IP.

Parámetro	***IPv4***	***IPv6***
Longitud de direcciones	32 bits	128 bits
Formato del encabezado	Variable	Fijo
Campos del encabezado	13	8
Longitud del encabezado	20-60 bytes	40 bytes
Registros DNS	A	AAA
Campo Checksum	Sí	No
Campos de fragmentación	Sí	No
Encabezados de extensión	No	Sí
Direcciones anycast	Sí	Sí
Soporte nativo de Seguridad (IPSec)	No	Sí
Soporte nativo para movilidad	No	Sí
Auto-configuración	No	Sí
Descubrimiento entre enrutadores	No	Protocolo de descubrimiento de vecinos
Fragmentación en los enrutadores	Sí	Solo en el inicio (arranque)
Requerimiento de traducción de direcciones (NAT/PAT)	Sí	No

Tabla II.8 Resumen comparativo entre IPv4 e IPv6

Capítulo 3

Protocolo de Seguridad en Internet: IPSecurity (IPSec)

III.1 Introducción

IPSec (Internet Protocol Security) es un grupo de extensiones al protocolo de Internet IP y se trata de un protocolo estándar desarrollado por la IETF definido en el RFC 4301, que aborda las carencias en cuanto a seguridad del IP, considerándose una tendencia a largo plazo para las redes que implementen seguridad al proporcionar una línea de defensa robusta frente a los ataques.

IPSec tiene por objetivo proteger el contenido de los paquetes IP punto a punto mediante un filtrado de paquetes haciendo uso de algoritmos criptográficos y una administración dinámica de llaves (si es implementada), para proporcionar protección a la comunicación entre equipos de redes privadas, dominios, sitios, sitios remotos, extranets[27] y/o clientes de acceso telefónico, pudiendo incluso bloquear la recepción y/o transmisión de determinados tipos de tráfico.

IPSec se basa en un modelo de seguridad completo, estableciendo la confiabilidad y seguridad desde una dirección IP origen hasta una dirección IP destino. Los

[27] Es una red privada virtual resultante de la interconexión de dos o más intranets de la misma compañía que utilizan Internet como medio de transporte para compartir información entre sus usuarios (suministradores, vendedores, socios, clientes u otros negocios) de manera segura.

únicos equipos que deben conocer que el tráfico está protegido son el remitente y el receptor, donde cada uno trata la seguridad en su extremo respectivo y supone que el medio a través del cual tiene lugar la comunicación no es seguro; los equipos que se limitan a enrutar datos desde el origen hasta el destino no necesitan ser compatibles con IPSec, salvo en el caso de que se filtren paquetes hacía algún equipo en específico por el cual se requiera que pase éste, o bien, cuando se use una traducción de direcciones de red entre los dos equipos.

III.2 Características de IPSec

IPSec es un mecanismo estándar, robusto y con posibilidades de expansión, para proporcionar seguridad al protocolo IP y protocolos de capas superiores (UDP, TCP, ICMP, etc.), dando servicios de seguridad tales como el control de acceso, autenticación, integridad y confidencialidad.

IPSec puede proteger paquetes IP entre hosts, gateways, o hosts y gateways, pudiendo ser implementado en IPv4 de manera opcional, aunque se introdujo de manera obligatoria en IPv6.

Otra característica destacable de IPSec es su carácter de estándar abierto. Se complementa perfectamente con la tecnología PKI permitiendo excluir de la solicitud del certificado el nombre de la entidad emisora de certificados, evitando la posible revelación a intrusos de información importante como el nombre de la compañía propietaria del equipo o el dominio al cual pertenece dicho equipo. Por otro lado, aunque establece ciertos algoritmos comunes, por razones de interoperabilidad, permite integrar algoritmos criptográficos más robustos que pueden ser diseñados en un futuro.

III.3 Arquitectura de IPSec

La arquitectura de IPSec específica la base en la cual todas las implementaciones serán construidas y define los servicios de seguridad proveídos por IPSec, cómo y dónde pueden ser usados, cómo serán los paquetes construidos y procesados, y la interacción del procesamiento de IPSec con las políticas de seguridad.
Esta arquitectura define la granularidad con la que el usuario puede especificar su política de seguridad, permitiendo que cierto tráfico sea identificado para recibir el nivel de protección deseado, como se ve en la Figura III.1.

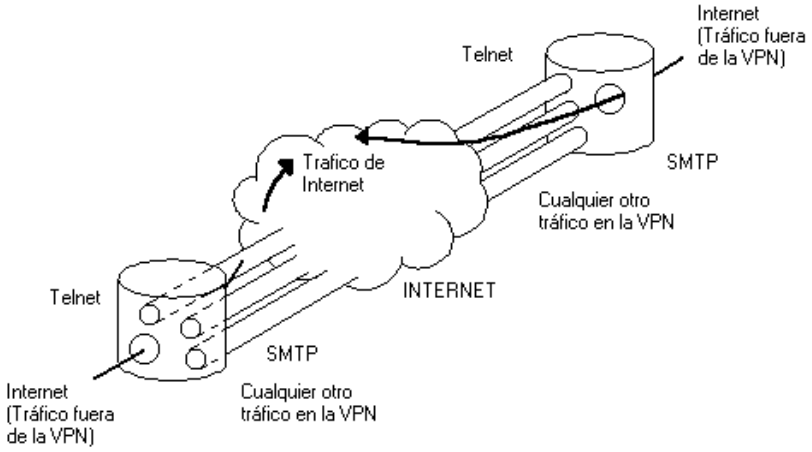


Figura III.1 Flujo protegido por IPSec entre redes separadas

IPSec está diseñado para proporcionar seguridad interoperable de alta calidad basada en criptografía, tanto para IPv4 como para IPv6. Se compone de dos encabezados para proveer seguridad en el tráfico: AH (Authentication Header) y ESP (Encapsulating Security Payload), además de protocolos para la generación y administración de llaves cifradas, por ejemplo IKE (Internet Key Exchange) e ISAKMP (Internet Security Association and Key Management Protocol), respectivamente. AH provee autenticación en el origen de los datos, integridad de los datos y la protección contra la réplica. Por otro lado, ESP además de proveer los servicios que proporciona AH, adicionalmente provee confidencialidad en los datos (cifrado) y confidencialidad limitada en el flujo de tráfico.

El esquema de interoperabilidad de IPSec se maneja a través de SAs (Security Association) las cuales son controladas por un SPI (Security Parameter Index), y regidas por SPs (Security Policy), ver sección III.10, previamente configuradas; tanto las SAs como las SPs son almacenadas en sus respectivas bases de datos: SAD para las asociaciones de seguridad y SPD para las políticas. La arquitectura de IPSec también define la interacción que hay entre estas bases de datos con las diferentes funciones de procesamiento de IPSec (cifrado y descifrado) y define cómo varias implementaciones de IPSec pueden existir.

Los parámetros que se negocian para establecer los canales seguros se indican bajo políticas pre-establecidas dentro de un esquema de funcionamiento estático con valores fijos y previamente establecidos, o bien, en un esquema de funcionamiento dinámico utilizando un protocolo de administración de llaves como ISAKMP (Internet Security Association and Key Management Protocol). Estas políticas determinan si dos entidades son capaces de comunicarse entre sí y cuál sería la transformación a usar en un caso dado. En la Figura III.2 se muestran los elementos que conforman IPSec, donde el DOI (Domain of Interpretation) contiene los valores necesarios para que los componentes de IPSec se relacionen entre sí.

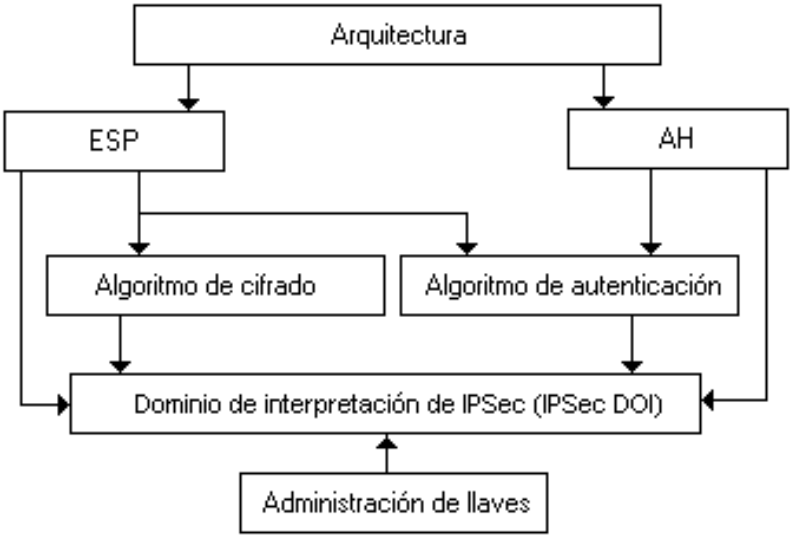


Figura III.2 Relación de los componentes de IPSec

III.4 Servicios de seguridad ofrecidos por IPSec

IPSec proporciona los siguientes servicios de seguridad:

A. Control de acceso

Previene el uso no autorizado de recursos, garantizando que sólo acceden a la información y a los recursos los usuarios que tienen permiso para ello.

B. Integridad

Implica que los datos no puedan ser modificados o corrompidos de manera alguna desde su transmisión hasta su recepción en una comunicación.

C. Autenticación

Define mecanismos para garantizar la procedencia de la información, de modo que se puedan verificar que realmente es el remitente autorizado quien la envió, asegurando la legitimidad de dicha información.

D. Protección a la réplica

Asegura que una transacción sólo se pueda llevar a cabo una vez, a menos que se autorice una repetición de la misma. Nadie debería poder grabar una transacción para luego replicarla al pie de la letra con el propósito de aparentar múltiples transacciones del remitente original, por ejemplo, en caso de que el atacante conociera el motivo del tráfico sin la necesidad de descifrarlo, y que el tráfico causara sucesos favorables para él, como depositar dinero en su cuenta, se tendría que asegurar que no se pueda replicar este tráfico más tarde.

E. Confidencialidad

Implica que la información sea accesible únicamente por las entidades, sistemas o personas autorizadas, asegurando la privacidad de la información al no ser consultada por terceras personas.

F. Confidencialidad limitada en el flujo de tráfico

Este servicio se refiere a ocultar las direcciones fuente y destino, la longitud del mensaje, o la frecuencia de la comunicación. En el contexto de IPSec, usando ESP en modo túnel, especialmente en un gateway de seguridad, puede proporcionar un cierto nivel de confidencialidad en el flujo de tráfico.

En la Tabla III.1 se muestra una comparación de los servicios de seguridad, anteriormente mencionados, con los encabezados de IPSec.

	AH	**ESP (sólo cifrado)**	**ESP (cif + aut)**
Control de acceso	Sí	Sí	Sí
Integridad sin conexión	Sí	Sí	Sí
Autenticación del origen de los datos	Sí	No	Sí
Rechazo de paquetes repetidos	No	Sí	Sí
Confidencialidad de los datos	No	Sí	Sí
Confidencialidad limitada en el flujo de tráfico	No	Sí	Sí

Tabla III.1. Comparación de los distintos servicios de seguridad ofrecidos por IPSec para los dos encabezados con las configuraciones correspondientes.

III.5 Beneficios de IPSec

IPSec proporciona un nivel de seguridad común y homogéneo para todas las aplicaciones, además de ser independiente de la tecnología física empleada.

Cuando se implementa IPSec en un firewall o enrutador, éstos proporcionan una fuerte seguridad que puede ser aplicada a todo el tráfico que cruza el perímetro.

Por otro lado, al estar implementado en la capa de red, debajo de los protocolos TCP/UDP resulta “transparente” para las aplicaciones, es decir, no hay necesidad de realizar alguna configuración desde el punto de vista de usuario final ni del servidor.
También IPSec tiene la capacidad de ofrecer seguridad individual si ésta fuese indispensable, resultando útil para los empleados que accedan a la red desde el exterior vía telefónica. Además, es posible asegurar una subred virtual dentro de una organización para las aplicaciones más sensibles.

Facilita el comercio electrónico de negocio a negocio al proporcionar una infraestructura segura sobre la cual realizar transacciones usando cualquier aplicación, por ejemplo las extranets.

III.6 Algoritmos criptográficos que ofrece IPSec

Los algoritmos permitidos para la protección con IPSec, tanto los usados para autenticación como los usados para cifrado, idealmente desempeñan dos metas incompatibles: proveer máxima protección contra una gran variedad de ataques matemáticos, de análisis criptológico y de fuerza bruta; y por otro lado, requerir un procesamiento mínimo en el lado de cada participante dentro de la comunicación.
Aunque los documentos de IPSec decretan algoritmos específicos para proveer un grado estándar, con seguridad interoperable, se pueden implementar algoritmos adicionales ya sea para dominio público o privado.

Todos los algoritmos son algoritmos de bloque, empiezan en el inicio del mensaje y cada bloque es procesado uno a la vez. El tamaño del bloque es parte de la

definición de cada algoritmo, donde el más común es de 8 bytes (64 bits). Cada bloque pasa de cierto modo por algún procesamiento repetitivo donde cada iteración de ese procesamiento es conocido como *ciclo*. El número de ciclos es algunas veces considerado como una característica importante en la criptografía de un algoritmo. Cada ciclo, en turno, consiste de una *función de ciclo*, la cual es un procesamiento que constituye cada ciclo del cifrado. La función de ciclo puede ser simple y sencilla, o extremadamente compleja. Algunos algoritmos tienen múltiples funciones de ciclo que se pueden aplicar a uno o más ciclos.

En muchos algoritmos, la llave secreta más completa no es usada como función hash (ver capítulo III.6) o para cifrar cada bloque, sino para generar múltiples *subllaves*, o *ciclos de llave* donde a su vez cada ciclo puede incorporar una o más subllaves. Si cada bloque fuera cifrado o manejado por una función hash separadamente, se presentarían ataques más fáciles, ya que el contenido de algunas partes del paquete de Internet serían conocidas. En el caso de una función hash, el hash final se debe reflejar en todos los bits de todo el bloque de entrada, no solo en el último bloque. En el caso de un algoritmo de cifrado, si cada bloque es descifrado separadamente, sin hacer referencia a ningún otro bloque, los bloques previsibles pueden ser atacados más fácilmente una vez que la llave fue conocida y todo el bloque puede ser descifrado. Por esta razón, todo algoritmo de manera obligatoria en IPSec incorpora dentro de su definición un mecanismo de retroalimentación, es decir, el cifrado o autenticación de cada bloque tiene como una de sus entradas la salida calculada criptográficamente del bloque previo.

Existen un gran número de operaciones usadas comúnmente en los algoritmos, entre las cuales están: la operación exclusiva OR (XOR), corrimiento circular y la aritmética modular.

a) La operación de corrimiento circular consiste básicamente en cambiar de posición los bits a la izquierda o derecha de forma circular como se muestra en la Figura III.3.

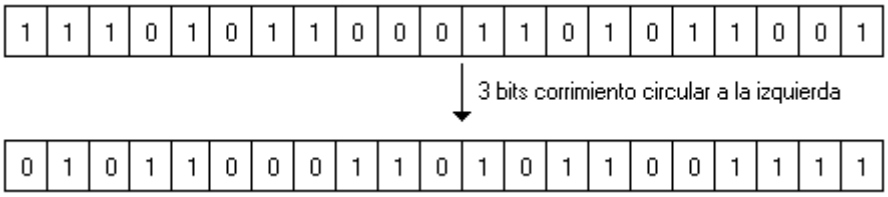


Figura III.3 Operación de corrimiento circular

b) La operación XOR consiste en una comparación bit a bit de dos cantidades numéricas, donde el resultado de la XOR contendrá un bit “0” si las dos cantidades de entrada tienen el mismo valor y un bit “1” si son diferentes. La Figura III.4 muestra un ejemplo.

Figura III.4 Operación exclusiva OR (XOR)

c) La aritmética modular (suma, resta, multiplicación y exponenciación) frecuentemente son usadas en algoritmos criptográficos. En la Figura III.5 se muestra el modulo aritmético 16 (2^4).

$$7+8=15_{\text{módulo10}}=15_{\text{módulo16}}$$

$$7+8=25_{\text{módulo10}}=(25-16)_{\text{módulo16}}=9_{\text{módulo16}}$$

$$7+8=35_{\text{módulo10}}=(35-2*16)_{\text{módulo16}}=3_{\text{módulo16}}$$

Figura III.5 Aritmética modular: adición modular 16 (2^4)

La seguridad de los algoritmos criptográficos dependerá de la complejidad de su criptografía y de su robustez. Sin embargo, un algoritmo criptográfico no es

suficiente para garantizar la seguridad de las comunicaciones debido a que varios factores juegan un papel muy importante, como por ejemplo, la implementación en hardware o software, o bien, la generación de llaves secretas que deberán tener una apropiada longitud, complejidad y ser generadas, intercambiadas, administradas y almacenadas de una manera segura.

El protocolo IPSec ha sido diseñado en forma modular de modo que se puedan seleccionar determinados algoritmos para cifrado y autenticación sin afectar a otras partes de la implementación. Sin embargo, han sido definidos algunos algoritmos de manera estándar para soportar todas las implementaciones y asegurar la interoperabilidad en el mundo global de Internet, como son AES (en etapa de evaluación) para sustituir a DES y 3DES, considerados actualmente para cifrado, así como MD5 y SHA-1 como funciones hash para autenticación. Además es perfectamente posible usar otros algoritmos que se consideren más seguros o más adecuados para un entorno específico, como por ejemplo IDEA o Blowfish.

III.6.1 Algoritmos de autenticación

Para los algoritmos de autenticación se utilizan las funciones hash (o primitivas hash), cuya funcionalidad es usada principalmente para resolver el problema de integridad y autenticidad del origen de los mensajes.

Una función hash o “función resumen” es un algoritmo que, aplicado a un mensaje determinado, crea una representación digital o hash de una longitud fija mucho menor que el mensaje original, pero substancialmente único a él, de tal manera, que no sea factible, dado solamente el hash, reconstruir el mensaje original, es decir, las funciones hash son de una sola dirección. Un simple ejemplo de una función hash sería contar el número de letras del mensaje, si es par asociamos un 0 y si es impar un 1, Figura III.6. El principal inconveniente de este sistema es que pueden existir colisiones (dos mensajes diferentes producen la misma salida) por

lo que conviene que las funciones tengan un rango de salida lo suficientemente grande (128 bits o más) para poder considerarlas libres de colisión.

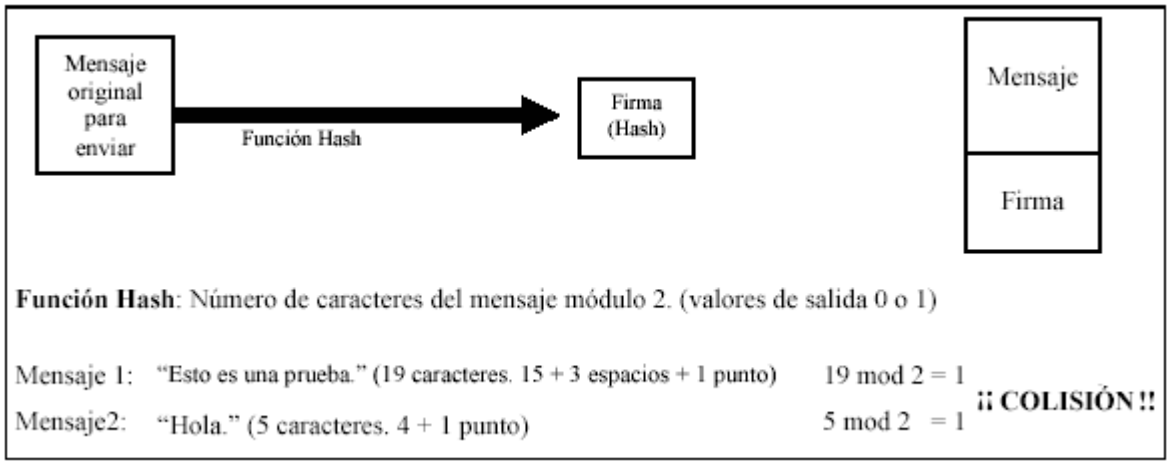


Figura III.6 Ejemplo de una función hash y colisión.

Estas funciones hash pueden operar como: MDC (Modification Detection Code) para resolver el problema de integridad de la información, o MAC (Message Authentication Code) para autenticar el origen de los mensajes (junto con la integridad). En nuestro estudio nos enfocaremos a la operación con MAC.

Algoritmo MD5

MD5 (Message Digest) es el más antiguo hash de una serie inventada por Ronald L. Rivest con un tamaño de bloque de 64 bytes (512 bits) y una longitud de llave de 128 bits, generando un hash de 16 bytes (128 bits).

El hash MD5 de un mensaje se calcula de la siguiente manera:

1. El mensaje será extendido hasta que su longitud en bits sea congruente con 448 módulo 512, esto es, el mensaje se extenderá hasta que se forme el menor número múltiplo de 512 bits. Esta extensión se realiza siempre, incluso si la longitud del mensaje es ya congruente con 448 módulo 512.

La extensión se realiza como sigue: un solo bit "1" se añade al mensaje, y después bits "0" se añaden hasta que la longitud en bits del mensaje extendido se haga congruente con 448 módulo 512. En todos los mensajes se añade al menos un bit y como máximo 512.

2. Una representación de 64 bits de 'b' (la longitud del mensaje antes de añadir los bits) se concatena al resultado del paso anterior. En el supuesto no deseado de que 'b' sea mayor que 2^{64}, entonces sólo los 64 bits de menor peso de 'b' se usarán.

   En este punto el mensaje resultante (después de rellenar con los bits y con 'b') tiene una longitud que es un múltiplo exacto de 512 bits. A su vez, la longitud del mensaje es múltiplo de 16 palabras (32 bits por palabra).
3. Un búfer de cuatro palabras (A, B, C, D) se usa para calcular el hash del mensaje. Cada una de las letras representa un registro de 32 bits.
4. Se definen cuatro funciones auxiliares que toman como entrada tres palabras de 32 bits y su salida es una palabra de 32 bits.

   $$F(X,Y,Z) = (X \wedge Y) \vee (\neg X \wedge Z)$$
   $$G(X,Y,Z) = (X \wedge Z) \vee (Y \wedge \neg Z)$$
   $$H(X,Y,Z) = X \oplus Y \oplus Z$$
   $$I(X,Y,Z) = Y \oplus (X \vee \neg Z)$$

   Los operadores son las funciones XOR, AND, OR y NOT respectivamente.

   En cada posición de cada bit F actúa como un condicional: si X, entonces Y si no Z. Las funciones G, H e I son similares a la función F, ya que actúan "bit a bit en paralelo" para producir sus salidas de los bits de X, Y y Z, en la medida que si cada bit correspondiente de X, Y y Z son independientes y no sesgados, entonces cada bit de F(X,Y,Z), G(X,Y,Z), H(X,Y,Z) e I(X,Y,Z) serán independientes y no sesgados. Este paso usa una tabla de 64 elementos T[1 ... 64] construida con la función **Seno**.
5. El hash del mensaje es la salida producida por A, B, C y D, esto es, se comienza el byte de menor peso de A y se acaba con el byte de mayor peso de D.

MD5 es utilizado para proporcionar la seguridad de que un archivo descargado de Internet no se ha alterado, comparando una suma MD5 publicada con la suma de comprobación del archivo descargado, protegiendo al usuario contra virus que algún otro usuario malicioso pudiera incluir en el software o bien contra descargas corruptas o incompletas. También puede ser utilizado, por ejemplo, en sistemas UNIX y GNU/Linux para cifrar las llaves de los usuarios o para comprobar que los correos electrónicos no han sido alterados usando llaves públicas y privadas.

Algoritmo SHA-1

SHA-1 (Secure Hash Algorithm) fue definido originalmente por la NSA (Nacional Security Agency), y fue adoptado por el NIST (National Institute of Standards and Technology) como hash unidireccional prescrito para usarse con DSA (Digital Signature Algorithm). Tiene un tamaño de bloque de 64 bytes (512 bits) y una longitud de llave de 160 bits, generando un hash de 20 bytes (160 bits).

El funcionamiento de SHA-1 se basa en principios similares a los usados por el profesor Ronald L. Rivest en el diseño del algoritmo hash MD5. Sin embargo, la resistencia del algoritmo SHA-1 se ha visto comprometida a lo largo del año 2005. Después de que MD5, entre otros, quedara seriamente comprometido en el 2004 por parte de un equipo de investigadores chinos (Xiaoyun Wang, Yiqun Lisa Yin y Hongbo Yu) que también han demostrado que es capaz de romperse el SHA-1 en al menos 2^{69} operaciones, unas 2000 veces más rápido que un ataque de fuerza bruta (que requeriría 2^{80} operaciones), quedando su tiempo de vida comprometido.

Algoritmo HMAC

El algoritmo HMAC (Hashed MAC) utiliza una función hash existente aplicando un proceso iterativo a la función hash del mensaje y a la llave secreta dos veces, fortaleciendo a una función hash creada sin un incremento de nivel significante en los recursos computacionalmente requeridos.

La diferencia de HMAC con respecto a MAC radica en que HMAC usa un algoritmo hash en combinación con una llave secreta compartida que se añade a los datos para aplicarles una función hash. Esto hace al hash más seguro debido a que ambas partes tienen la misma llave secreta compartida para verificar la autenticación de los datos.

HMAC no especifica la longitud de la llave. Si la longitud de la llave excede la longitud del bloque (64 bytes) se hace un hash de la llave con la función hash subyacente para producir una nueva llave que va a ser el tamaño del hash de salida. Para una seguridad apropiada, la longitud de la llave secreta no debe ser más pequeña que el tamaño del hash de salida (16 bytes para MD5, 20 bytes para SHA-1); sin embargo, se debe tomar en cuenta que una mayor longitud de llave no aumenta una seguridad de forma apreciable. Por lo tanto, para el uso en el encabezado AH de IPSec, una llave de 16 bytes se asigna para HMAC-MD5 y una llave de 20 bytes para HMAC-SHA-1 de una manera estándar. El cálculo de HMAC se presenta a continuación.

1. La llave se rellena con ceros, en caso de ser necesario, hasta obtener una longitud de 64 bytes (tamaño del bloque), se genera "*llave"*.
2. Se hace una "*XOR"* de la llave generada en el paso 1 con una constante especial ("*entrada completa"*), se genera "*llave 1*".
3. Se anexan los "*datos del mensaje*" al resultado del paso 2.
4. Se aplica la función hash al resultado del paso 3, se genera *"llave con hash"*.

5. Se calcula otra operación *"XOR"* entre la llave expandida del paso 1 y otra constante especial (*"salida completada"*), se genera *"llave 2"*.
6. La salida del paso 4 se añade a la salida del paso 5.
7. Se aplica la función hash al resultado del paso 6, que acorde a la función hash empleada será de 16 o 20 bytes.

En la Figura III.7 se ilustra el funcionamiento de HMAC con MD5 como hash.

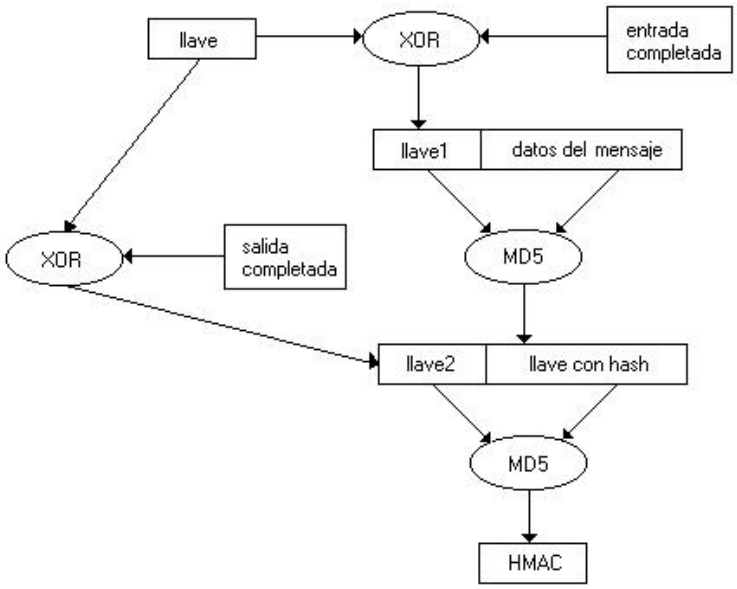


Figura III.7 Cálculo de HMAC, usando MD5 como el hash subyacente.

Otros algoritmos de autenticación

RIPEMD-160 (RACE Integrity Primitives Evaluation Message Digest), se definió para usarse con los encabezados de IPSec. Es un hash de cinco ciclos con un tamaño de bloque de 64 bytes (512 bits) y con una longitud de llave de 160 bits, generando un hash de 20 bytes (160 bits). Los creadores de RIPEMD, Hans Dobbertin, Antoon Bosselaers, y Bart Preneel, originalmente pretendieron hacer una versión más segura de MD4 (precursor de MD5). RIPEMD-160 es una versión de reingeniería de RIPEMD con una estructura similar pero con características adicionales de seguridad que consiste en dos cálculos paralelos como los de MD5,

que diferencian uno del otro, en el orden de aplicación de las funciones, las palabras del mensaje seleccionadas para el hash y las constantes agregadas; después del ciclo final, los dos hash resultantes se agregan para producir el hash final.

III.6.2 Algoritmos de cifrado

Todos los algoritmos de cifrado para el encabezado de ESP son algoritmos orientados al bloque. Cada bloque del texto de entrada, o texto plano, se transforma conforme se usa el algoritmo de cifrado en conjunto con una llave secreta en sus contrapartes cifradas conocidas como texto cifrado. El mecanismo de encadenamiento usado por los algoritmos de cifrado ESP es denominado modo CBC (Cipher Block Chaining). En modo CBC, antes del cifrado, cada bloque no cifrado realiza una XOR con el texto cifrado del bloque anterior. Un valor análogo también se necesita para el primer bloque; dicho valor es referido como el vector de inicialización IV (Initialization Vector). Debido a que el primer bloque de texto cifrado generalmente contiene campos múltiples (por ejemplo, campos del encabezado TCP) cuyos valores son conocidos e invariantes a partir de un paquete al siguiente, el uso de un IV asegura que los campos idénticos que se cifraron con llaves idénticas varíen de un paquete al siguiente.

Cabe mencionar que en los algoritmos de cifrado las llaves débiles no proporcionan un nivel de seguridad generalmente atribuido al algoritmo en cuestión, por ejemplo, dos cifrados sucesivos con una llave débil pueden reproducir el texto plano original, o un cifrado único puede dar lugar al texto cifrado que es más vulnerable para ataques conocidos. Las SAs de IPSec no son creadas para ser establecidas con llaves débiles conocidas para un algoritmo seleccionado de cifrado.

Actualmente se ha propuesto a AES como el algoritmo obligatorio de cifrado para ESP, RFC 4835, para sustituir a DES y 3DES como se mencionaba en el RFC 4305. Aunque también existen otros algoritmos de cifrado que pueden utilizarse en el encabezado ESP como se mencionará mas adelante.

Algoritmo DES

DES (Data Encryption Standard), definido originalmente por IBM, fue adoptado por el NIST como un algoritmo estándar de cifrado del gobierno de EUA para los datos no clasificados. Consiste en 16 ciclos, tiene un tamaño de bloque de 8 bytes (64 bits), y genera una versión cifrada de un mensaje que, en la mayoría, aumenta la longitud del mensaje de modo que sea un múltiplo exacto del tamaño de bloque.

El algoritmo DES, como originalmente se definió, tiene cuatro modos. El primero, DES plano cifra cada bloque de entrada separadamente y constituye el modo ECB (Electronic Codebook). Los otros tres modos, el modo de CBC, el modo CFB (Cipher Feedback) y el modo OFB (Output Feedback) incorporan alguna forma de retroalimentación. Cada texto cifrado del bloque es una función no solamente del texto original y de la llave secreta para ese bloque, sino contiene además el texto cifrado de uno o más bloques anteriores. El modo DES requerido por IPSec es el modo CBC. El encadenamiento de los bloques cifrados produce una cierta protección limitada contra ataques “cortar y pegar” (Figura III.8), la repetición de ataques y la información repetitiva disfrazada, así como incrementa la robustez criptográfica a la salida.

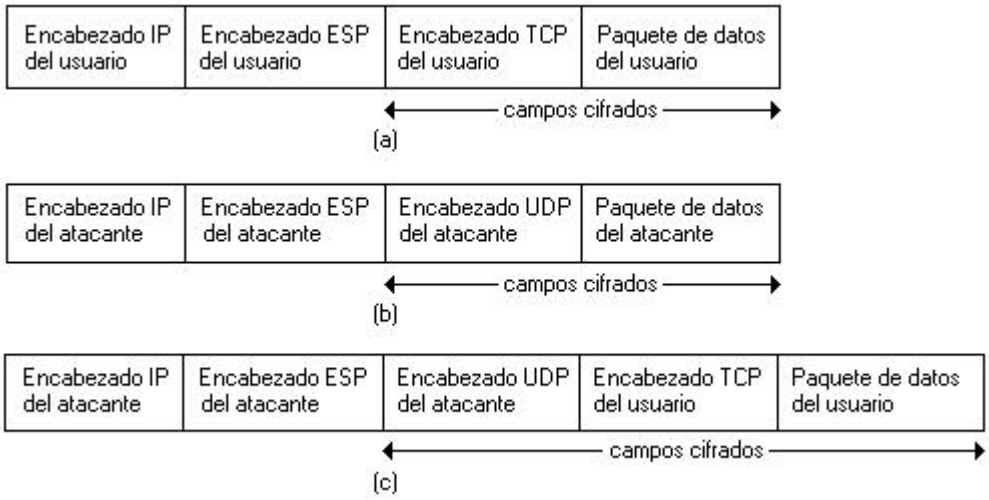


Figura III.8 Ataque “cortar y pegar”

(a) mensaje original del usuario.
(b) mensaje original del atacante.
(c) mensaje modificado por el atacante.

DES requiere una llave secreta de 64 bits de longitud, pero solamente 56 de esos bits son bits de llave real; los 8 bits restantes son los bits de paridad que aseguran la consistencia interna de cada byte de la llave. El algoritmo DES consiste en 16 ciclos, cada uno de los cuales utiliza una llave de 48 bits diferente para trabajar satisfactoriamente. La llave original de 56 bits es transformada en 16 llaves de 48 bits como sigue.

1. Usando una tabla, los 56 bits de la llave se permutan, dando por resultado dos valores de 28 bits, la parte izquierda y derecha, de la llave.
2. Para obtener la llave de cada ciclo, tanto en la parte izquierda como en la parte derecha se hace un corrimiento circular izquierdo de 1 o 2 bits, obteniendo una nueva parte izquierda y una parte derecha. La llave del ciclo actual se obtiene al realizar una permutación en la concatenación de la parte izquierda actual y la parte derecha actual, obteniendo una llave de ciclo de 48 bits.

DES es un algoritmo extremadamente complejo. Después de que la llave para cada ciclo fue calculado, el calculo verdadero comienza. Para el modo CBC, el primer bloque realiza una XOR con el IV y los bloques restantes hacen una XOR con el texto cifrado del bloque anterior. La salida de la XOR se permuta y después se divide en una mitad izquierda y una mitad derecha. La mitad derecha y la llave del ciclo se utilizan como las entradas de una manipulación numérica compleja; su resultado es entonces una XOR con la mitad izquierda. La salida con la XOR se convierte en la nueva mitad derecha, y la vieja mitad derecha se convierte en la nueva mitad izquierda, y el ciclo siguiente comienza.

Después de 16 de estos ciclos, la mitad izquierda y la mitad derecha finales se intercambian y concatenan, permutándose una vez más, y se obtiene el texto cifrado. Más detalladamente, el procesamiento de cada bloque es como sigue.

1. Para satisfacer el modo CBC, se hace una XOR entre el texto a ser cifrado del bloque actual y el texto cifrado del bloque anterior. En el caso del primer bloque del mensaje se realiza una XOR con el IV, que para IPSec es un valor de 64 bits generado aleatoriamente. La salida de esta operación se convierte en el bloque actual de la entrada del algoritmo DES.

2. Los bits del bloque de entrada se reacomodan en una *permutación inicial*; el bloque permutado entonces se divide en dos mitades, la mitad izquierda inicial *L0* y la mitad derecha inicial *R0*.

3. Cada uno de los 16 ciclos consiste en los siguientes pasos.
    a. La mitad derecha del ciclo anterior (para el primer ciclo la mitad derecha inicial) se almacena en la mitad izquierda.
    b. Los 32 bits de la mitad derecha actual se permutan, y algunos de los bits de entrada aparecen más de una vez en la salida, dando por resultado una salida de 48 bits.

c. Con los 48 bits de salida del paso 3(b) se hace una XOR con la llave de 48 bits del ciclo actual.
d. La salida de 48 bits del paso 3(c) se divide en ocho valores de 6 bits. Cada uno de esos ocho valores se utiliza como índice en una de las ocho tablas, cada una con 4 filas y 16 columnas. El primer bit y el último bit de cada valor constituyen la fila, y los 4 bits centrales la columna. Cada uno de los ocho valores de 6 bits es sustituido por los 4 bits de la tabla de entrada referida por la fila y el índice de la columna derivados del valor de 6 bits. La salida de este paso es la concatenación de los ocho valores de los 4 bits derivados, dando por resultado una salida de 32 bits. Las ocho tablas usadas para transformar los valores de 6 bits en valores de 4 bits se denominan tablas S o cajas S.
e. La salida de 32 bits del paso 4(d) son permutados, dando por resultado, la mitad derecha actualizada.
f. Con la mitad derecha actualizada se hace una XOR con la mitad izquierda del ciclo anterior, y el resultado es almacenado en la mitad derecha.

4. Después de la terminación de 16 ciclos, se intercambia la mitad izquierda con la mitad derecha. Con la concatenación de estos dos valores (la mitad derecha seguida por la mitad izquierda) se aplica una permutación inversa realizada en el paso 2. La salida de esa permutación es el cifrado DES del bloque actual.

La definición DES contiene tablas que definen las permutaciones de las llaves iniciales e intermedias, el número de operaciones de corrimiento para ser aplicadas a la llave de cada ciclo, la permutación del bloque inicial y su inversa, la permutación que expande la mitad derecha de cada bloque de 32 bits a 48 bits, las ocho tablas S, y la permutación aplicada a la salida de las tablas S. En la Figura III.9 se muestra el funcionamiento de este algoritmo.

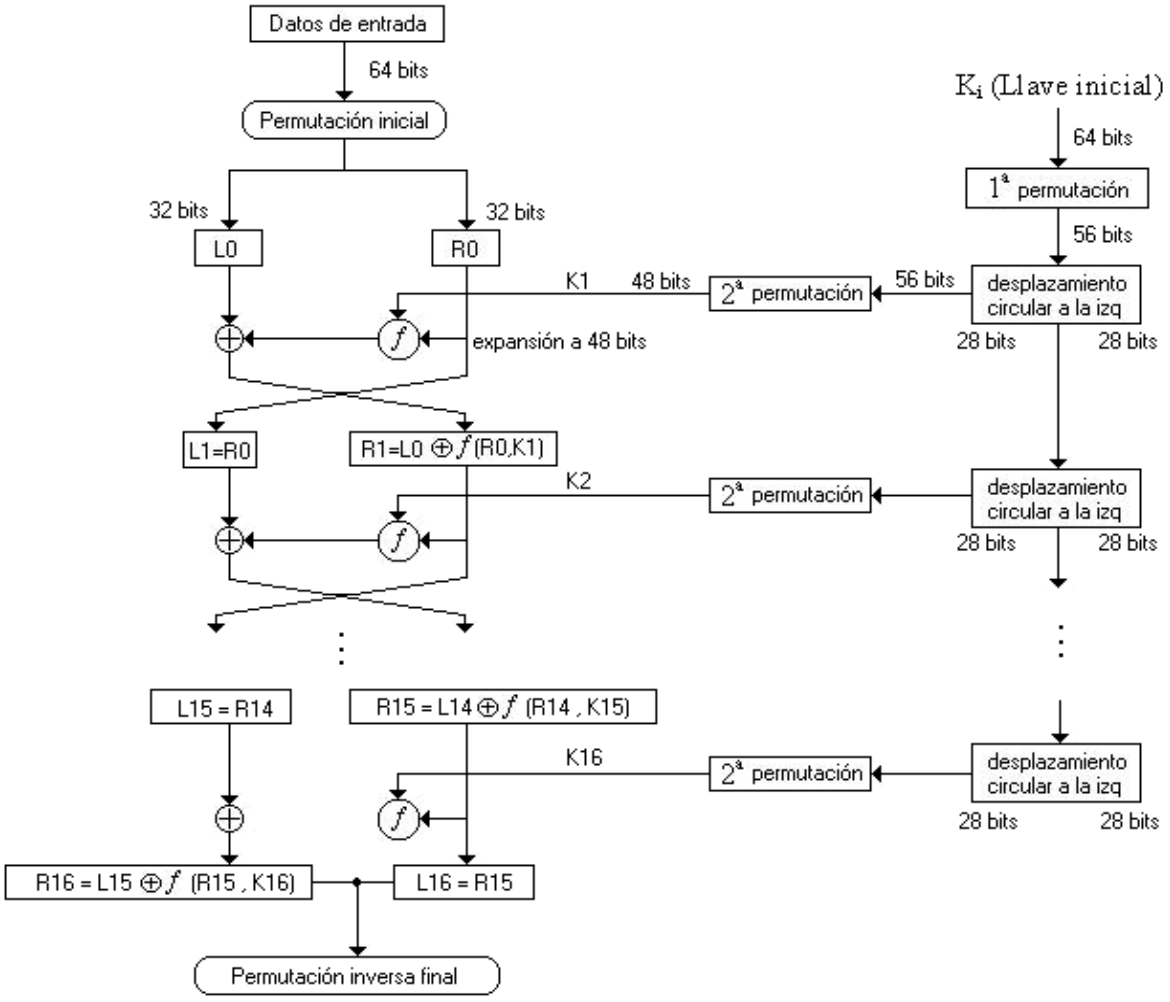


Figura III.9 Lógica general del cifrado de DES

Un mensaje que ha sido cifrado con DES es descifrado usando el mismo algoritmo, sólo ajustando un cambio: las subllaves son usadas en un orden opuesto; es decir, el último par de subllaves cifradas son utilizadas para el primer ciclo de descifrado. Esto es invirtiendo el orden de las aplicaciones de las permutaciones y aplicando a cada llave un corrimiento circular derecho, en lugar de un corrimiento circular izquierdo, para obtener la siguiente subllave.

Además, existe una versión reforzada de DES, denominada Triple DES, utilizada más comúnmente y definida en octubre de 1999 por el NIST como un estándar del gobierno de EUA.

Algoritmo Triple DES

Debido a que depende de DES, Triple DES también tiene un tamaño de bloque de 8 bytes (64 bits). El tamaño de la llave tiene una longitud de 192 bits, donde 1 bit de cada 8 es un bit de paridad para asegurar el estado interno de cada byte de la llave, dando lugar a una llave secreta de 168 bits de longitud. Operacionalmente, la llave se analiza en tres llaves del tamaño de DES de 56 bits cada una. Para IPSec, el modo CBC de Triple DES es el modo de valor por defecto.

Cada bloque del mensaje de entrada se procesa como sigue.

1. Se satisface el modo CBC, igual que DES (paso 1)
2. Se utiliza el algoritmo DES para cifrar la salida del paso 1 con la primera llave secreta de 56 bits.
3. Se utiliza al algoritmo DES para descifrar la salida del paso 2 con la segunda llave secreta de 56 bits.
4. Se utiliza al algoritmo DES para cifrar la salida del paso 3 con la tercer llave secreta de 56 bits.

Algoritmo AES

El algoritmo DES se está acercando al final de una carrera larga y gloriosa. Para seleccionar su reemplazo, el NIST llevó a cabo una competencia de varios años, anunciada en enero de 1997. De los 15 algoritmos de cifrado que fueron admitidos como candidatos de AES (Advanced Encryption Standard) en agosto de 1998, 5 fueron señalados como finalistas: MARS, RC6, Rijndael, Serpent, y Twofish. Todos los finalistas presentaban un tamaño de bloque de 128 bits y pueden manejar un tamaño de llave de 128, 192, y 256 bits. El análisis, la discusión y la comparación de los candidatos continuaron hasta que la selección final se realizó

el 2 de octubre del año 2000, donde el NIST anunció oficialmente que Rijndael sería el nuevo AES.

Rijndael tiene una tamaño de llave variable y consiste en 10, 12, o 14 ciclos de cifrado, dependiendo del tamaño de la llave. Cada ciclo consiste en 4 pasos: substitución de bytes usando una tabla S, mezcla de datos entre las columnas, corrimiento de bytes sobre un desplazamiento variable, y una XOR con la llave de ciclo. La opción fue hecha en base a la seguridad, eficiencia computacional y requisitos de memoria con gran variedad de software y hardware incluyendo tarjetas inteligentes, flexibilidad, y simplicidad. AES será un algoritmo de cifrado designado al gobierno de EUA para información no clasificada y adoptado indudablemente para usarse en negocios e instituciones financieras. El grupo de IPSec muy probablemente declarará AES para ser un algoritmo de cifrado obligatorio para el encabezado ESP.

El NIST también ha definido tres nuevos algoritmos hash que son apropiados para usarse con los tres tamaños de llave requeridas para AES y que substituirán a SHA-1. Estos algoritmos son SHA-256, SHA-384, y SHA-512; donde cada uno genera un hash cuya longitud en bits sea conmensurada con el nombre del hash. El NIST está considerando si define nuevos modos de encadenamiento para sustituir o suplir los definidos para DES.

Algoritmo NULL

El algoritmo de cifrado NULL es el algoritmo con el cual el encabezado ESP puede ser usado para proveer autenticación sin cifrado. Se define en el RFC 2410 donde se describe su historia, funcionamiento y su uso.

Otros Algoritmos de cifrado

Blowfish

- El algoritmo Blowfish fue inventado por el criptógrafo Bruce Schneier. Es un cifrado de 16 ciclos capaz de manejar una llave de longitud variable; para IPSec una llave de 128 bits se define como el valor por defecto. Los cálculos de la subllave de Blowfish son absolutamente complejos, pero la porción del cifrado del algoritmo es extremadamente directa. Como DES, prototipo de cifrados Feistel, Blowfish consiste en permutaciones y una función de ciclo que contiene sustituciones múltiples de tablas S. A diferencia de DES, las permutaciones son llaves dependientes, al igual que las cuatro tablas S. Las salidas de las tablas S, que son de longitud de 32 bits, se calculan usando la función XOR y la adición modular.

CAST

- El algoritmo CAST es nombrado por sus inventores originales, Carlisle Adams de Entrust Technologies y Stafford Tavares de la Universidad Queen en Canadá. Este algoritmo tiene una llave de tamaño variable y consiste de 12 o 16 ciclos del cifrado, dependiendo del tamaño de la llave. La versión del algoritmo seleccionado para usarse con el encabezado ESP es conocido como CAST-128, utilizando una llave de 128 bits y un total de 16 ciclos. El cifrado utiliza ocho tablas S, cuatro para el cálculo de las llaves usadas para los ciclos individuales y cuatro para el cifrado actual. Hay realmente dos llaves para cada ciclo, la “llave para el enmascaramiento” y “una llave de rotación.” La llave para el enmascaramiento se combina con los datos del bloque a través de la adición modular, la substracción modular, o una XOR, para después realizar un corrimiento de bits dictados por la llave de rotación.

IDEA

- IDEA (International Data Encryption Algorithm) es el invento de Xuejia Lai y de James L. Massey del Instituto de Tecnología Federal Suizo. Es un cifrado de 8

ciclos con un tamaño de bloque de 64 bytes (512 bits) y una longitud de llave de 128 bits. Este algoritmo difiere de otros cifrados de ESP en que no es un cifrado Feistel y su uso es patentado. Cada uno de los 8 ciclos usa 6 subllaves e implica series XOR, adiciones y multiplicaciones modulares, combinando las subllaves con porciones del bloque de entrada o resultados calculados de ciclos anteriores. Después del ciclo final, un paso adicional utiliza 4 subllaves más. Aunque no es un cifrado Feistel, se diseñó de modo que las operaciones de cifrado y descifrado sean iguales; el descifrado se logra usando diferentes subllaves, generadas de las subllaves del cifrado.

RC5

- RC5 (Ron's Code or Rivest's Cipher), creado por Ronald Rivest del MIT, es un cifrado sencillo y elegante que se puede utilizar con gran variedad de tamaños de llave, tamaños de bloque, y número de ciclos. La versión de RC5 especificada para usarse con ESP tiene una llave de 128 bits, un tamaño de bloque de 64 bits, y 16 ciclos. Una diferencia de RC5 es la rotación de datos dependiente, la cual es una rotación circular a la izquierda de un elemento de datos por un número variable de bits, según el valor de otro elemento de datos. Las funciones de ciclo combinan estas rotaciones de datos dependientes, las funciones XOR, y la adición modular.

III.7 Implementaciones de IPSec

IPSec puede ser implementado en hosts, en conjunto con un enrutador, o con un firewall (para crear gateways de seguridad). La implementación es configurada dependiendo de los requerimientos de seguridad de los usuarios.

A continuación se menciona la implementación de IPSec en varios dispositivos de red (hosts y enrutadores). La implementación en hosts es más útil cuando se desea una seguridad punto a punto; sin embargo, en casos cuando la seguridad

se desea sobre una parte de la red, es mejor la implementación en enrutadores que incluyen VPNs e intranets.

III.7.1 Implementación en hosts

La implementación en hosts tiene las siguientes ventajas:

- Provee una seguridad punto a punto.
- Capacidad de implementarse en todos los modos de IPSec.
- Proporciona seguridad en el flujo de datos.
- Capacidad para conservar la autenticación de los usuarios en las conexiones establecidas por IPSec.

Esta implementación puede ser clasificada en dos distintas sub-implementaciones:

1. *Implementación integrada con el Sistema Operativo (OS):* Como IPSec es un protocolo de nivel de red, puede ser implementado como parte del mismo como se muestra en la Figura III.10, donde IPSec necesita los servicios del nivel IP para construir el encabezado IP. Este modelo es idéntico a la implementación de otros protocolos del nivel de red como ICMP.

Figura III.10 Niveles de la pila IPSec con OS

2. Implementación que se coloca entre los niveles de red y de enlace de la pila del protocolo como se muestra en las Figuras III.11a y III.11b. Se denomina

implementación BITS (Bump in the Stack), y es utilizado para que las compañías encargadas de dar soluciones en VPN e intranets puedan proporcionan una solución completa, dado que la solución que se integra con el OS limita las capacidades para proporcionar soluciones avanzadas.

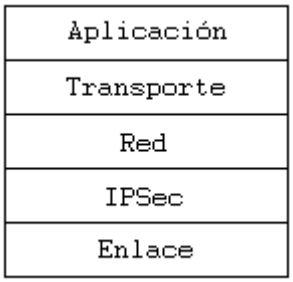


Figura III.11a Niveles de la pila IPSec con implementación BITS

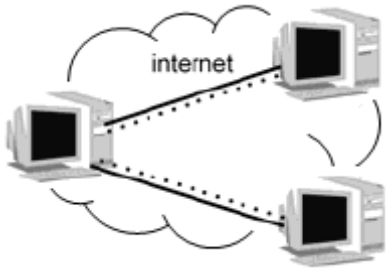


Figura III.11b Implementación tipo BITS

III.7.2 Implementación en enrutadores

La implementación en enrutadores tiene la capacidad de proporcionar seguridad al flujo de paquetes entre dos redes sobre una red pública, como lo es Internet, por medio de un túnel; además de autenticar y autorizar a los usuarios que entran a la red privada para comunicarse sobre Internet construyendo sus VPN o intranets.

Existen dos tipos de implementación en enrutadores:

1. Implementación nativa: Esta implementación es análoga a la implementación en hosts integrada con el OS. En este caso, IPSec es integrado con el software del enrutador, ver Figura III.12.

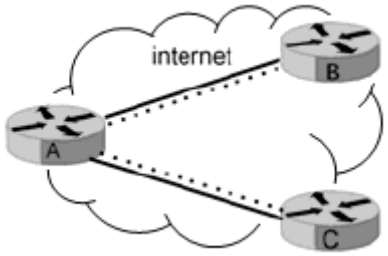


Figura III.12 Implementación nativa

2. “Bump in the Wire” (BITW): Es similar a la implementación BITS, pero en este caso IPSec es implementado en un dispositivo de cifrado externo dedicado conectado a la interfaz física del enrutador. Este dispositivo normalmente no ejecuta ningún algoritmo de ruteo, sino solamente es usado para asegurar los paquetes, ver Figura III.13.

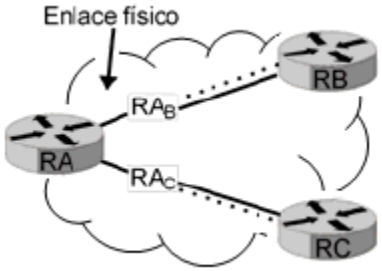


Figura III.13 Implementación tipo BITW

A la fecha existen diversas implementaciones; sin embargo, la mayoría limitadas a la aplicación de VPNs únicamente de forma nativa, por lo que IPSec es denominado por algunos como el "protocolo VPN". En los últimos años han emergido proyectos para implementar seguridad en sistemas operativos, usando esquemas BITS, en busca de brindar una plataforma base

de seguridad que sea independiente de las aplicaciones utilizadas por el usuario.

III.8 Modos de procesamiento

IPSec ha sido diseñado para funcionar en dos modos diferentes para sus distintos encabezados, AH y ESP, los cuales son: modo transporte y modo túnel.
La diferencia entre estos tipos de modos radica en la unidad que se está protegiendo, para el modo transporte se protege la carga útil de IP (capa de transporte), mientras que para el modo túnel se protegen los paquetes IP (capa de red)
Existen cuatro combinaciones posibles de implementar estos modos junto con los encabezados de IPSec: AH en modo transporte, AH en modo túnel, ESP en modo transporte, y ESP en modo túnel. En la práctica, AH en modo túnel no es usado, debido a que se protegen los mismos datos que si se usara en modo transporte.
Cabe mencionar que los encabezados AH y ESP no cambian por utilizar un modo u otro, sino que su diferencia se basa en lo que están protegiendo: los paquetes IP o la carga útil IP.

III.8.1 Modo Transporte

El modo transporte se aplica a nivel de hosts. AH y ESP en este modo interceptan los paquetes procedentes de la capa de transporte a la capa de red, aplicando la seguridad que previamente se configuró. En la Figura III.14 se observa un esquema de IPSec en modo transporte, en donde A y B son dos hosts que han sido previamente configurados. Si esta configuración define que los paquetes

deben ser cifrados, se utiliza ESP en modo transporte. Si sólo se requiere autenticación se usará AH en modo transporte

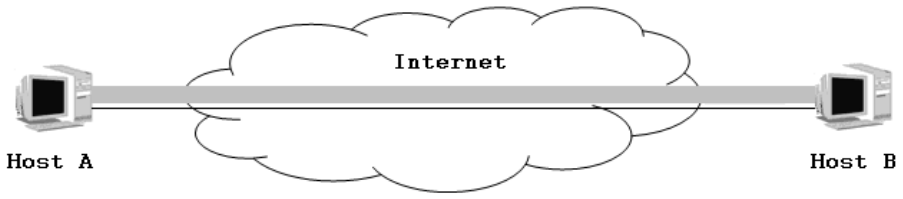


Figura III.14 IPSec en modo transporte entre dos hosts

En este modo los paquetes de la capa de transporte como TCP y UDP pasan a la capa de red, donde agregan su encabezado IP y pasan a las capas inferiores. Cuando se habilita IPSec, los paquetes de la capa de transporte pasan al encabezado de IPSec (que es implementado como parte de la capa de red, en el caso de sistemas operativos), donde se agregan los encabezados AH y/o ESP, y el encabezado IP de la capa de red; por tanto, el encabezado IPSec se inserta inmediatamente a continuación del encabezado IP y antes de los datos de los niveles superiores que se desean proteger.

En el caso que se apliquen ambos protocolos, primero debe aplicarse el encabezado de ESP y después AH, para que la integridad de datos se aplique a la carga útil de ESP que contiene la carga útil de la capa de transporte, esto se ilustra en la Figura III.15.

Datagrama IP

Encabezado IP original (IPv4 o IPv6)	Encabezado TCP

Datagrama en modo Transporte

Encabezado IP original (IPv4 o IPv6)	Encabezado AH	Encabezado ESP	Carga útil TCP

Figura III.15 Datagrama en modo transporte

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

III.8.2 Modo túnel

El modo túnel se utiliza cuando la seguridad es aplicada por un dispositivo diferente al que genera los paquetes, como el caso de las VPNs, o bien, cuando el paquete necesita ser asegurado hacia un punto seguro como destino y es diferente al destino final, como la implementación BITS o BITW.
En la Figura III.16, el flujo de tráfico es entre A y B, e IPSec puede aplicarse con una asociación de seguridad entre RA y RB, o bien, una asociación de seguridad entre A y RB.

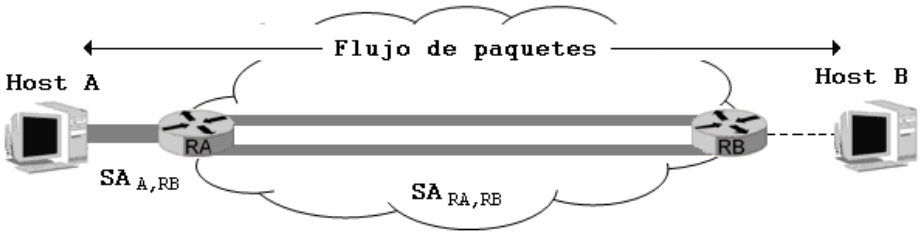


Figura III.16 Aplicación de IPSec en modo túnel

IPSec en modo túnel, tiene dos encabezados IP, uno interior y otro exterior. El encabezado interior es creado por el host para añadir un encabezado de IPSec; y el encabezado exterior es agregado por el dispositivo que está proporcionando los servicios de seguridad para encaminar los paquetes a través de la red. IPSec encapsula el paquete IP con los encabezados de IPSec y agrega un encabezado exterior de IP como se ilustra en la Figura III.17.

Datagrama IP

Encabezado IP original (Ipv4 o IPv6)	Encabezado TCP

Datagrama en modo Túnel

Encabezado IP nuevo (Ipv4 o IPv6)	ESP	Encabezado IP original	Carga útil de la capa de red

Figura III.17 Datagrama en modo túnel

El modo túnel es empleado principalmente por gateways seguros de IPSec, como se mencionó, con el objetivo de identificar a la red que se protege bajo una misma dirección IP y centralizar de este modo el procesado del tráfico de IPSec en un equipo. Cuando se utiliza junto con ESP, se utiliza para ocultar la identidad de los nodos que se están comunicando. Además el modo túnel, tanto con ESP como con AH, es usado para poder establecer VPN a través de redes públicas, es decir, interconectar de forma segura redes de área local, incluso en el caso de que éstas usen direccionamiento privado.

IPSec también soporta túneles anidados, aunque no son recomendados por lo complicado de su construcción, mantenimiento y consumo de recursos de red. La Figura III.18 muestra dos túneles, el nodo A envía un paquete al nodo B, la política indica que debe ser autenticado por el enrutador RB, además existe una VPN entre RA y RB, de tal forma que el paquete que ve RB es el que se muestra en la Figura III.19, el encabezado exterior es un paquete ESP encapsulado y contiene un paquete AH encapsulado, el paquete AH contiene el paquete IP para el nodo B generado por el nodo A.

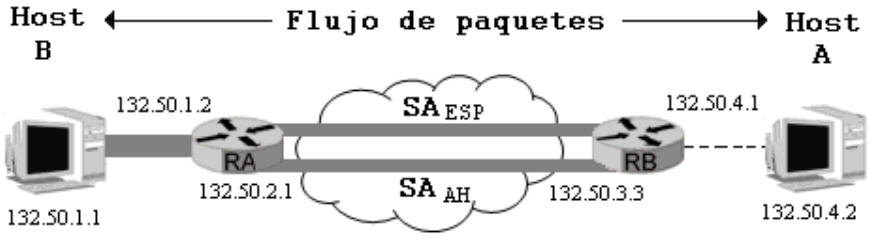


Figura III.18 Túneles anidados

Encabezado IP	ESP	Encabezado IP	AH	Encabezado IP	Datos
Fuente 132.50.2.1 *Destino* 132.50.3.3		*Fuente* 132.50.1.1 *Destino* 132.50.3.3		*Fuente* 132.50.1.1 *Destino* 132.50.4.2	

Figura III.19 Formato del paquete del túnel anidado

III.9 Encabezados de Seguridad de IPSec

III.9.1 Encabezado de Carga de Seguridad de Encapsulación (ESP)

III.9.1.1 Introducción

El Encabezado de Carga de Seguridad de Encapsulación (ESP), definido en el RFC 4303, tiene como objetivo principal proporcionar *confidencialidad*, especificando el modo de cifrar los datos que se desean enviar y como este contenido cifrado se incluye en un datagrama IP. Adicionalmente, puede ofrecer servicios de anti-réplica, integridad y autenticación del origen de los datos incorporando un mecanismo similar a AH. El encabezado ESP se inserta después del encabezado IP y antes del encabezado del protocolo de capa superior (modo transporte) o antes del encabezado IP encapsulado (modo túnel)

En la Figura III.20 se observa cómo el protocolo ESP permite enviar datos de forma confidencial. El emisor toma el mensaje original, lo cifra utilizando una llave determinada y lo añade en un paquete IP seguido del encabezado ESP. Durante el tránsito hasta su destino, si el paquete es interceptado por un tercero sólo

obtendrá un conjunto de bits no legibles. En el destino, el receptor aplica de nuevo el algoritmo de cifrado con la misma llave 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 llave, que es conocida únicamente por el emisor y el receptor.

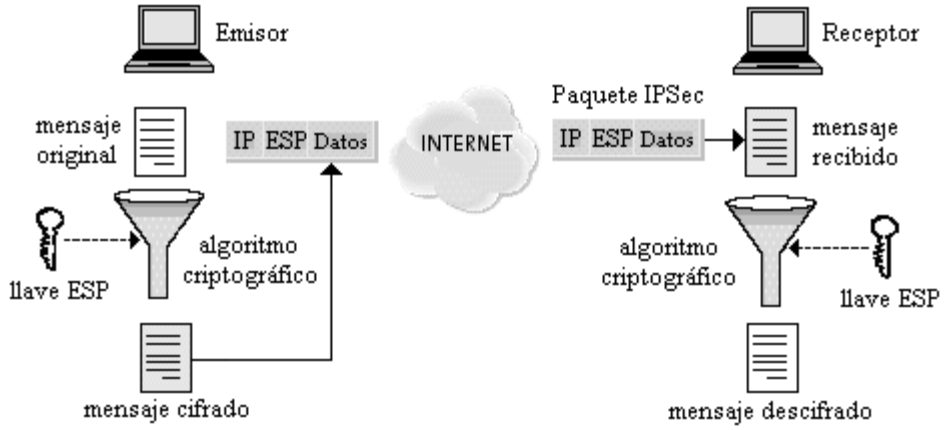


Figura III.20 Funcionamiento de ESP

La función de cifrado dentro del protocolo ESP es desempeñada por un algoritmo de cifrado de llave simétrica. Típicamente se usan algoritmos de cifrado por bloques (DES), de modo que la longitud de los datos a cifrar tenga que ser un múltiplo del tamaño de bloque (8 o 16 bytes, en la mayoría de los casos). Por esta razón existe un campo de relleno cuya función es añadir caracteres de relleno al campo de datos para ocultar así su longitud real y, por tanto, las características del tráfico. Por ejemplo, un atacante suficientemente hábil podría deducir cierta información a partir del análisis de ciertos parámetros de las comunicaciones, aunque estén cifradas, tales como el retardo entre paquetes y su longitud. La función de relleno está pensada para dificultar este tipo de ataques.

III.9.1.2 Formato del datagrama

La IANA ha asignado al protocolo ESP el número decimal 50. Esto implica que en el campo Protocolo (en IPv4) o “Siguiente encabezado” (en IPv6) del encabezado IP contendrá el valor 50, mientras que dentro del mensaje ESP se indica la naturaleza de los datos. Puesto que este campo, al igual que la carga útil, está cifrado, un hipotético atacante que intercepte el paquete no podrá saber si el contenido es TCP o UDP; esto es completamente normal ya que el objetivo que se persigue es, precisamente, ocultar la información. En la Figura III.21 se muestra el formato del encabezado ESP.

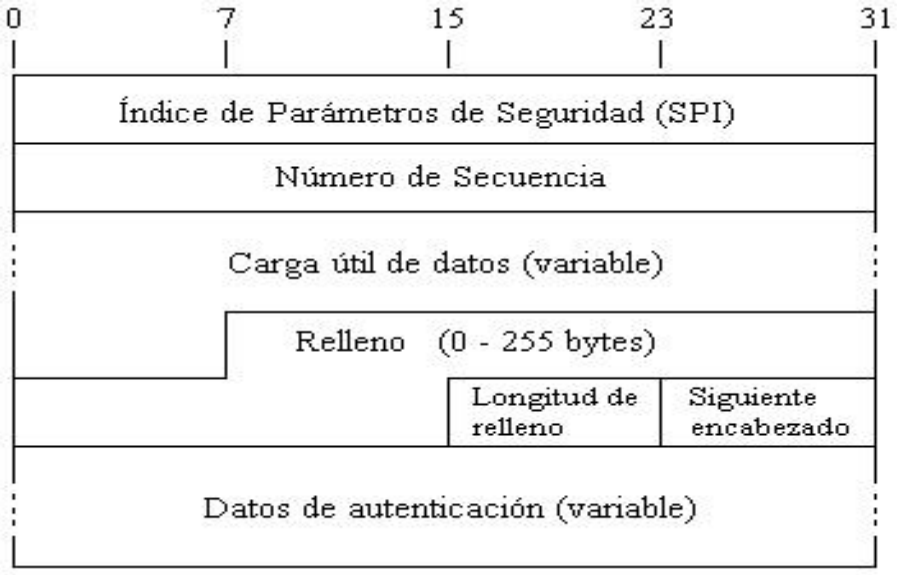


Figura III.21 Datagrama del encabezado ESP

Campos del encabezado ESP

- *Índice de Parámetros de Seguridad (SPI)*: Es un valor arbitrario de 32 bits que, en combinación con la dirección IP destino y el protocolo de seguridad (ESP), únicamente identifica a la Asociación de Seguridad para ese datagrama. El conjunto de valores de SPI del 1 al 255 es reservado por la IANA para uso futuro.

- *Número de secuencia*: Es un campo de 32 bits que se incrementa de forma secuencial por cada paquete. Este número siempre está presente incluso si el receptor no elige habilitar el servicio de anti-réplica para una Asociación de Seguridad específica.
- *Carga útil de datos*: Campo de longitud variable que contiene datos descritos por el campo “Siguiente encabezado”. Contiene los datos que se van a proteger.
- *Relleno*: Se utiliza en ESP por varias circunstancias: algunos algoritmos criptográficos requieren que el elemento de entrada sea un múltiplo del tamaño de su bloque; también se utiliza para asegurar que el resultado del texto cifrado tenga 4 bytes como límite con el propósito de establecer que la autenticación de los datos sea de 4 bytes como lo especifica el encabezado ESP; y para esconder el tamaño real de la carga útil.
- *Longitud de Relleno*: Campo de 8 bits que indica el número de bytes de relleno, donde un valor de cero indica que no hay bytes de relleno.
- *Siguiente encabezado*: Campo de 8 bits que identifica el tipo de dato contenido en el campo “Carga útil de datos”; por ejemplo, un encabezado de extensión o un protocolo de capa superior.
- *Datos de autenticación*: Es un campo de longitud variable múltiplo de 32 bits que contiene el ICV (Integrity Check Value). Este campo es opcional, y es solo incluido si el servicio de autenticación ha sido seleccionado por la Asociación de Seguridad.

III.9.1.3 Modos de procesamiento

El encabezado ESP puede ser implementado en dos modos posibles: modo transporte y modo túnel.

ESP aplicado en modo transporte sólo se utiliza en implementaciones en host y provee protección a los protocolos de capas superiores, pero no al encabezado IP.

El encabezado ESP se inserta después del encabezado IP y antes del protocolo superior (TCP, UDP, ICMP, etc.) o antes de cualquier encabezado IP que haya sido previamente insertado. En la Figura III.22 se ilustra la transformación del paquete IP al aplicar ESP en modo transporte para IPv4 y en la Figura III.23 se muestra el caso para IPv6.

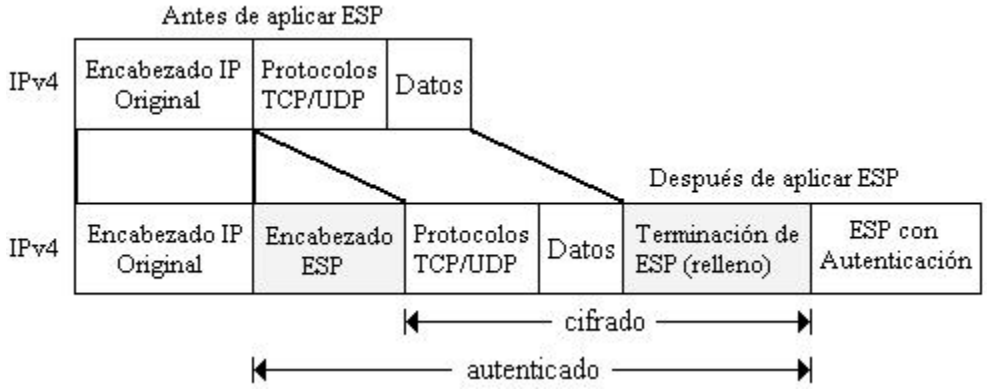


Figura III.22 Transformación del paquete IPv4 al aplicar ESP en modo transporte

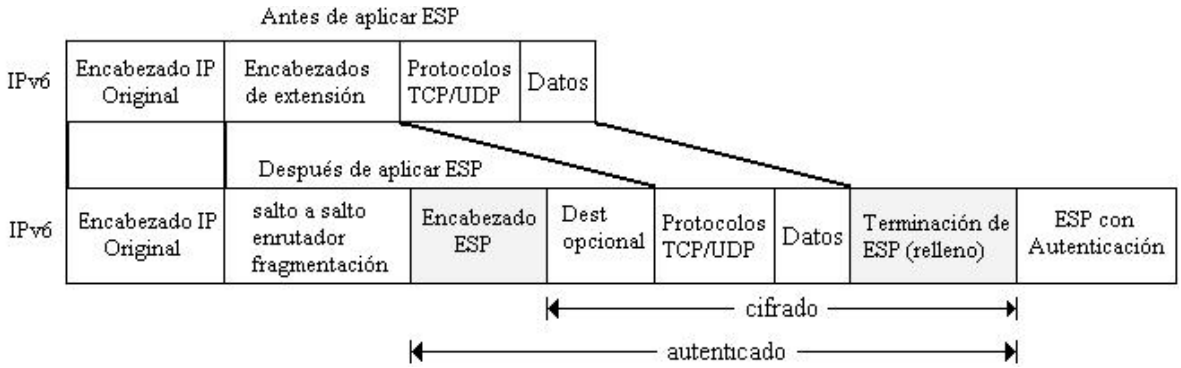


Figura III.23 Transformación del paquete IPv6 al aplicar ESP en modo transporte

En modo túnel, ESP puede ser empleado en hosts o en gateways. El encabezado IP interior contiene las direcciones del destino y origen del paquete, y el encabezado exterior puede contener direcciones diferentes, comúnmente direcciones de gateways de seguridad en el camino entre el origen y destino. La posición de los encabezados ESP en modo túnel con respecto a los encabezados

IP exteriores es igual que en modo transporte. En las Figuras III.24 y III.25 se muestran los encabezados ESP para IPv4 e IPv6 en modo túnel respectivamente.

Figura III.24 Transformación del paquete IPv4 al aplicar ESP en modo túnel

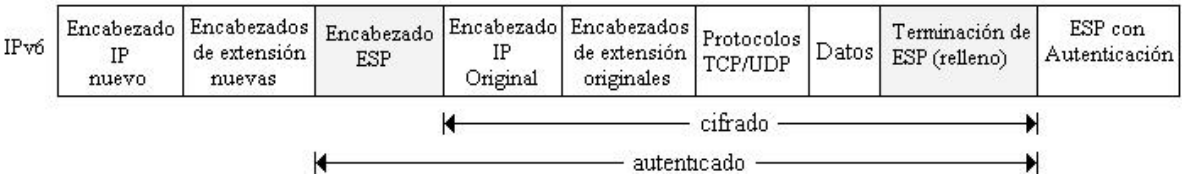


Figura III.25 Transformación del paquete IPv6 al aplicar ESP en modo túnel

III.9.2 Encabezado de Autenticación (AH)

III.9.2.1 Introducción

El Encabezado de Autenticación (AH), definido en el RFC 4302, es un encabezado de IPSec usado para proporcionar integridad en los datos, autenticación en el origen de los datos y opcionalmente servicios de anti-réplica a los datagramas IP. Sin embargo *no proporciona ninguna garantía de confidencialidad*, es decir, los datos transmitidos pueden ser vistos por terceros.

Se suele situar justo antes de los datos, de forma que los proteja de posibles atacantes. No obstante ha sido diseñado de forma muy versátil, pudiendo incluirse

antes de otros encabezados (opciones, encaminamiento, etc.) para asegurar así que las opciones que acompañan al datagrama sean correctas.

De esta forma, la presencia de un encabezado de autenticación no modifica el funcionamiento de los protocolos de nivel superior (TCP, UDP, etc.) ni el de los enrutadores intermedios que simplemente encaminan el datagrama hacia su destino.

En la Figura III.26 se muestra el modo en que funciona el encabezado AH. El emisor calcula un extracto del mensaje original, el cual se copia en uno de los campos del encabezado 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. 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.

Si analizamos con detalle el protocolo AH, podemos concluir que su seguridad reside en que el cálculo del extracto (MAC) es imposible sin conocer la llave, y que dicha llave sólo la conocen el emisor y el receptor.

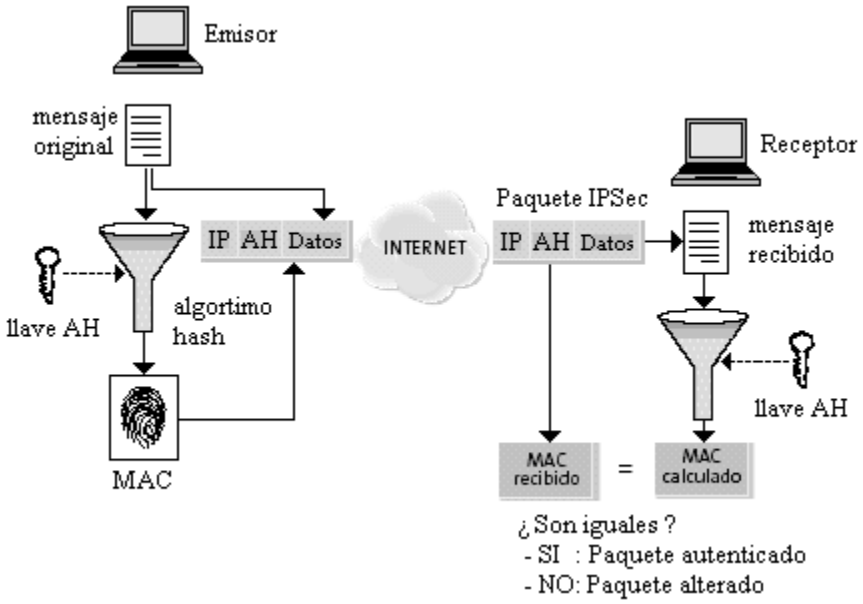


Figura III.26 Funcionamiento del encabezado AH

III.9.2.2 Formato del datagrama

La IANA le ha asignado a AH el número decimal 51. Esto significa que para IPv4 en el campo *Protocolo* o para IPv6 en el campo “Siguiente encabezado” del encabezado IP contiene el valor 51. Es importante destacar que AH asegura la integridad y autenticidad de los datos transportados y del encabezado IP, excepto los campos variables: TOS, TTL, flags, offset y checksum.

El formato del datagrama del encabezado AH se muestra en la Figura III.27.

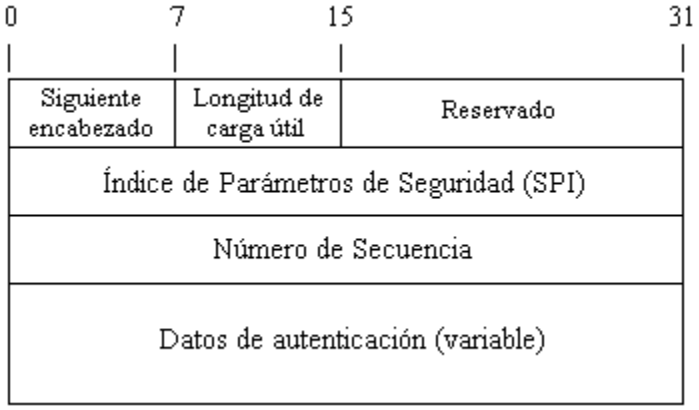


Figura III.27 Datagrama del encabezado AH

Campos del encabezado AH:

- *Siguiente encabezado*: Es un campo de 8 bits que identifica la siguiente carga útil después de la autenticación
- *Longitud de carga útil*: Es un campo de 8 bits que especifica el tamaño del encabezado de autenticación (AH) en palabras de 32 bits.
- *Reservado*: Es un campo de 16 bits reservado para usos posteriores. Debe ser colocado en “cero”.
- *Índice de Parámetros de Seguridad (SPI)*: Es un valor arbitrario de 32 bits que, en combinación con la dirección IP destino y el protocolo de seguridad (AH), únicamente identifica a la Asociación de Seguridad para ese

datagrama. El conjunto de valores de SPI del 1 al 255 es reservado por la IANA para uso futuro.

- *Número de secuencia*: Es un campo de 32 bits que se incrementa de forma secuencial por cada paquete. Este número siempre está presente incluso si el receptor no elige habilitar el servicio de anti-réplica para una Asociación de Seguridad específica.
- *Datos de autenticación*: Es un campo de longitud variable múltiplo de 32 bits que contiene el ICV. Este campo puede incluir un relleno explícito. Este relleno es colocado para asegurar que la longitud del encabezado AH es un múltiplo de 32 bits para IPv4 o 64 bits para IPv6.

III.9.2.3 Modos de Procesamiento

De la misma manera que ESP, AH se puede implementar tanto en modo transporte como en modo túnel. En modo transporte es insertado después del encabezado IP y antes de los protocolos de capa superior, o antes de cualquier otro encabezado de IPSec que hubiese sido insertado. En la Figura III.28 se muestra AH en modo transporte para IPv4 y en la Figura III.29 para IPv6.

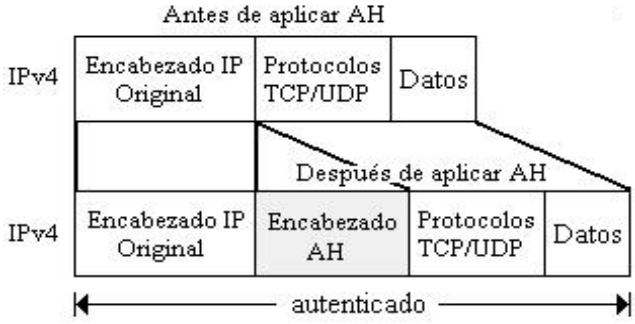


Figura III.28 Transformación del paquete IPv4 al aplicar AH en modo transporte

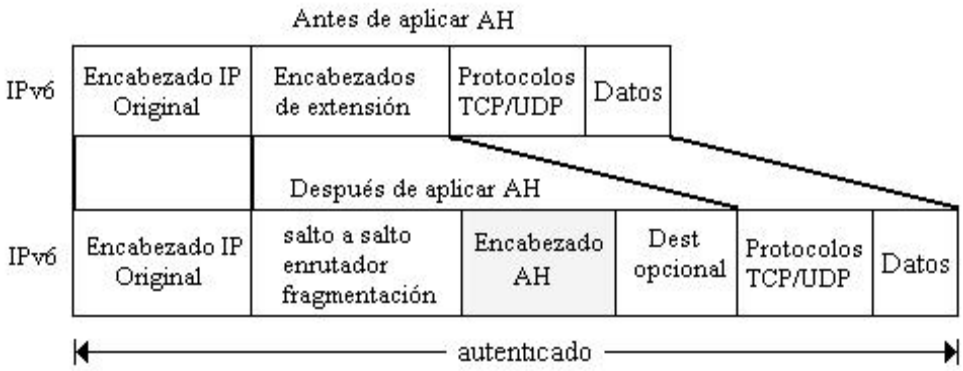


Figura III.29 Transformación del paquete IPv6 al aplicar AH en modo transporte

La aplicación de AH en modo túnel tiene una ubicación similar a la de ESP, en la Figura III.30 se muestra la transformación de los paquetes IP al aplicar AH en modo túnel para IPv4 y en la Figura III.31 para IPv6.

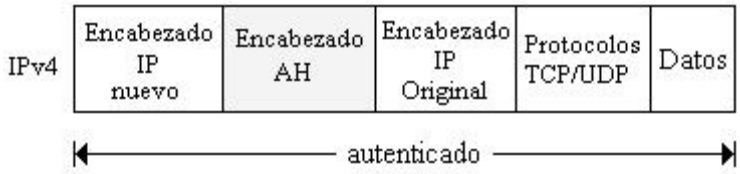


Figura III.30 Transformación del paquete IPv4 al aplicar AH en modo túnel

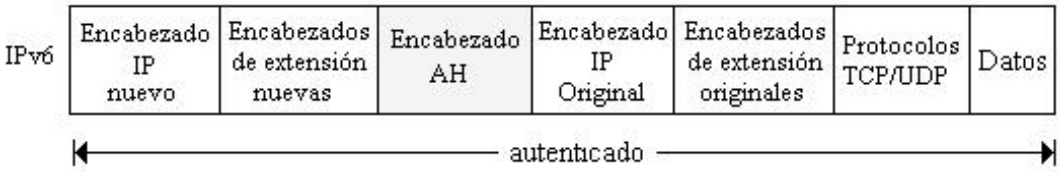


Figura III.31 Transformación del paquete IPv6 al aplicar AH en modo túnel

III.10 Asociaciones de Seguridad (SA)

Una SA es la forma básica de comunicación con IPSec refiriéndose a un contrato entre dos entidades que desean comunicarse en forma segura. Las SA

determinan los encabezados de IPSec a utilizar, las transformaciones, las llaves y la duración de validez de dichas llaves. Las SA son de un solo sentido (unidireccionales), es decir, cada entidad con IPSec tiene una SA para el tráfico que entra, y otra SA para el tráfico que sale; además, son específicas para cada encabezado, esto es, existe una SA tanto para AH como para ESP de manera independiente. Cuando se implementa IPSec se crea una base de datos de las SA denominada SAD donde se almacenaran todas las SA de dicha implementación.

III.10.1 Índice de Parámetros de Seguridad (SPI)

El SPI es una entidad de 32 bits que identifica de manera única a una SA. Es un contrato por el cual dos entidades se comunican de manera segura e indica los parámetros usados, como llaves y algoritmos. Es el mecanismo concebido para que en una comunicación segura, la fuente identifique cual SA utilizará para asegurar el paquete que enviará, y el destino identifique cual SA utilizará para verificar la seguridad del paquete recibido. El SPI forma parte en los encabezados ESP y AH para identificar de forma única a la SA.

III.10.2 Administración de las SAs

Para el manejo de las SAs se establecen dos tareas principalmente: creación y eliminación; que a su vez se pueden ejecutar de manera manual o dinámica a través de un protocolo de intercambio de llaves como IKE (ver sección III.12).

La creación de las SAs es un proceso de dos etapas: 1) negociación de parámetros de la SA y, 2) actualización de la SAD. El manejo manual de llaves es obligatorio en toda implementación, el proceso de asignación del SPI y la negociación de parámetros es totalmente manual y permanecerán hasta que sean

manualmente borrados. En el manejo dinámico de llaves se utiliza IKE donde es invocado por el kernel de IPSec cuando en la política se establece una comunicación segura y no encuentra una SA, entonces IKE negocia la SA con el destino o con el siguiente salto (host o enrutador) dependiendo de la política, creándose la SA en la SAD. Cuando en la política se establecen múltiples SAs, la colección de estas SAs se denominan “Paquete de SA” o “SA *bundle*”.

En el proceso de eliminación, las SAs pueden ser borradas manualmente o a través de IKE. Una SA es eliminada por varias razones: el tiempo de vida ha expirado, llaves comprometidas, el número de bytes utilizado excede un umbral especificado en la política, solicitud explícita para eliminarse la SA.

III.10.3 Parámetros

Los parámetros por negociar en una SA, tanto para AH como para ESP, son los siguientes:

- Número de secuencia: Campo de 32 bits utilizado en el procesamiento de paquetes de salida que es parte de los encabezados de AH y ESP. Su valor inicial es 0 y se incrementa en uno cada vez que la SA es utilizada. Se utiliza para detectar ataques de réplica o repetición.
- Sobreflujo del número de secuencia: Campo utilizado en el procesamiento de paquetes de salida y se establece cuando hay un sobreflujo en el campo de número de secuencia. La política determina si la SA puede ser aún usada para procesar paquetes adicionales.
- Ventana de anti-réplica: Campo utilizado en el procesamiento de paquetes de entrada. Se activa si IPSec detecta paquetes retransmitidos por hosts sospechosos.
- Tiempo de vida: Es el tiempo de validez asociado a una SA que no puede ser usada, se especifica en términos de bytes asegurados con la SA y, no se recomienda enviar más de 4Gb de paquetes utilizando la misma SA. Para

evitar la pérdida de la conexión segura se manejan los límites *soft* y *hard*. Al llegar al límite *soft* el kernel es notificado para que inicie una nueva negociación antes del límite *hard* que es cuando la SA expira.

- Modo: Los valores pueden ser túnel, transporte o indistinto. Si el valor es indistinto la SA puede ser utilizada para modo túnel o modo transporte.
- Destino del túnel: Campo utilizado para modo túnel que indica la dirección IP de destino del encabezado exterior.
- Parámetros PMTU: IPSec no fragmenta o reensambla paquetes; sin embargo, agrega un encabezado IPSec y por lo tanto impacta la longitud del PMTU (Protocol Maximum Transfer Unit). IPSec debe participar en la determinación del PMTU debido a que una SA mantiene dos valores: el PMTU y el campo de edad.

III.11 Políticas de Seguridad en IPSec

La política es uno de los componentes más importantes en la arquitectura de IPSec, determina los servicios de seguridad que serán aplicados a un paquete. Las políticas de seguridad son almacenadas también en una base de datos (SPD) indexada por seleccionadores.

La SPD es consultada tanto para el procesamiento de los paquetes IP de salida como los de entrada, y requiere de un administrador de la SPD para agregar, borrar y modificar las políticas; no hay un estándar que lo defina, pero se propone que los seleccionadores contengan los siguientes campos:

- Dirección fuente: puede ser indistinta, un rango de direcciones, un prefijo de red, o una dirección IP específica. Se utiliza la dirección indistinta cuando la política es la misma para todos los paquetes; el rango de direcciones y prefijo de red para los gateways de seguridad y para VPN; y la dirección específica para un host con varias direcciones, o en un gateway cuando los requerimientos de seguridad de algún host sean específicos.

- Dirección destino: puede ser indistinta, un rango de direcciones, un prefijo de red, o una dirección IP específica (homologada o no). Las tres primeras opciones se usan para hosts que están detrás de los gateways de seguridad, y la dirección específica como índice para la SPD.
- Nombre: Identifica a una política específica para un nombre válido de usuario o sistema. Se utiliza únicamente durante la negociación de IKE y no durante el procesamiento del paquete.
- Protocolo: Especifica el protocolo de transporte.
- Puertos de capas superiores: Son los puertos fuente y destino sobre los que se aplica la política.

III.12 Intercambio de Llaves por Internet (IKE)

III.12.1 Introducción

El protocolo IKE no es parte de IPSec, sino que es una alternativa para crear las Asociaciones de Seguridad de forma dinámica entre las partes de una comunicación. Las implementaciones de IPSec por lo general están forzadas a soportar el manejo manual y sólo algunas de ellas consideran a IKE debido a que ha resultado demasiado complejo e inapropiado, aunque ha estado en constate desarrollo definiéndose hasta el momento la versión 2, IKEv2, RFC 4306.

Una característica importante de IKE es que su utilidad no se limita a IPSec, sino que es un protocolo estándar de gestión de llaves que podría ser útil en otros protocolos, como por ejemplo, OSPF o RIPv2.

IKE utiliza un protocolo de intercambio de llaves denominado Diffie-Hellman (también conocido como acuerdo de llave exponencial) desarrollado por Diffie y

Hellman en 1976. El protocolo permite a dos usuarios intercambiar una llave secreta sobre un canal inseguro sin la necesidad de secretos previos.

El protocolo tiene dos parámetros de sistema *p* y *g* públicos y pueden ser usados por todos los usuarios de un sistema. El parámetro *p* es un número primo y el parámetro *g* (normalmente llamado generador) es un entero menor que *p* con la siguiente propiedad: para cada número *n* entre 1 y p-1 hay una potencia *k* de *g* tal que $n = g^k \bmod p$.

Supongamos que Alicia y José quieren acordar una llave secreta compartida usando el protocolo Diffie-Hellman, por lo que procederían de la siguiente manera: Primero, Alicia genera un valor aleatorio privado **a** y José genera un valor aleatorio privado **b**. Tanto **a** como **b** son escogidos de entre los enteros. Entonces deben derivar sus valores públicos usando los parámetros **p** y **g** y sus valores privados. El valor público de Alicia es $\mathbf{g^a \bmod p}$ y el de José es $\mathbf{g^b \bmod p}$.

Finalmente Alicia realiza $\mathbf{g^{ab} = (g^b)^a \bmod p}$, y José realiza $\mathbf{g^{ba} = (ga)b \bmod p}$. Debido a que $\mathbf{g^{ab} = g^{ba} = k}$, Alicia y José ahora tienen una llave secreta compartida **k**.

La seguridad del protocolo depende del problema del logaritmo discreto. Asume que el cálculo de la llave compartida $\mathbf{k = g^{ab} \bmod p}$ es inasequible computacionalmente dados los valores públicos $\mathbf{g^a \bmod p}$ y $\mathbf{g^b \bmod p}$ cuando el número primo **p** es suficientemente grande.

III.12.2 Protocolos que definen el IKE

IKE, definido en el RFC 4109, es un protocolo híbrido que utiliza parte del protocolo Oakley y parte del protocolo SKEME que, en combinación con ISAKMP, sirve para autenticar a los participantes en una comunicación, para después negociar las asociaciones de seguridad y escoger las llaves secretas a usar.

Oakley es un protocolo de intercambio de llaves basado en una versión modificada del algoritmo Diffie-Hellman, el cual describe un conjunto de métodos de

intercambio de llaves, llamados “modos”, proporcionando servicios de seguridad para cada uno, por ejemplo: envío correcto de la llave en secreto, protección de las identidades de las partes en la negociación, y autenticación. Por otro lado, SKEME es un protocolo encargado de proporcionar características de seguridad importantes como son: anonimidad, repudiabilidad, y una actualización de llaves de una manera rápida.

Tanto Oakley como SKEME definen un método para establecer un intercambio autenticado de llaves, es decir, cada uno debe de contener información sobre la construcción de la carga útil, la carga útil de acarreo, el orden de procesamiento, y como son usadas las llaves.

ISAKMP define cómo se pueden comunicar dos entidades, como los mensajes que se usan para comunicarse son construidos, y los estados de transición que se utilizarán para una comunicación segura. Además proporciona autenticación, intercambio de información del intercambio de llaves y negocia servicios de seguridad. Sin embargo, no define cómo se realiza un intercambio de llaves autenticado y los atributos necesarios para establecer una SA.

Mientras Oakley define los *modos*: principal, dinámico y rápido, ISAKMP se encarga de definir las 2 *fases* que existen. La relación entre los dos es muy indispensable para que IKE pueda definir el intercambio de llaves y negociar los servicios de seguridad. El resultado final de este intercambio realizado por IKE es una llave autenticada junto con los servicios de seguridad acordados, que en otras palabras, es una Asociación de Seguridad de IPSec (IPSec SA).

III.12.3 Fases para establecer una conexión

Para realizar la negociación de los parámetros necesarios para establecer una IPSec SA entre la conexión de dos entidades, ISAKMP define dos *fases* de negociación. En la primera fase, las dos entidades establecen un canal seguro y autenticado. El resultado de esta fase es el establecimiento de una ISAKMP SA.

Esta asociación de seguridad será usada en la siguiente fase para proteger la negociación de los parámetros de seguridad asociados a un protocolo determinado (AH o ESP).

ISAKMP también define varios tipos de carga útil, los cuales son usados para transferir las SA, así como los datos del intercambio de llaves en formatos definidos por el DOI. El número, los tipos y el orden de esta carga útil durante una negociación ISAKMP son especificados en el campo “tipo de intercambio” de ISAKMP. Actualmente hay cinco tipos de intercambios definidos, donde cada uno fue diseñado para proporcionar un conjunto particular de servicios de seguridad.

Para la primera fase el canal seguro se consigue mediante el uso de un algoritmo de cifrado simétrico y un algoritmo HMAC. Las llaves necesarias se derivan de una llave maestra que se obtiene mediante un algoritmo de intercambio de llaves Diffie-Hellman. Para esta fase se usan los modos principal y dinámico para establecer un intercambio de llaves autenticado. El modo principal se utiliza para proteger la identidad de cada una de los nodos; sin embargo, en caso de no ser necesario se utiliza el modo dinámico para reducir ciclos de comunicación.

Durante la segunda fase se negocian las características de la conexión ESP o AH y todos los parámetros necesarios mediante el modo rápido. El equipo que ha iniciado la comunicación ofrecerá todas las posibles opciones que tenga configuradas en su política de seguridad y con la prioridad que se hayan configurado. El sistema receptor aceptará la primera que coincida con los parámetros de seguridad que tenga definidos. Así mismo, ambos nodos se informarán del tráfico que van a intercambiarse a través de dicha conexión, del funcionamiento del protocolo IKE y del modo en que se obtiene una llave de sesión, que es la que se utiliza para proteger las conexiones ESP o AH.

Capítulo 4

Aspectos de seguridad contemplados en IPv6

El uso de IPSec da un mecanismo robusto y extensible para asegurar los datagramas IP a nivel de red permitiendo obtener soluciones de comunicaciones independientemente de cual sea el medio de transporte (FR, PPP, xDSL , ATM, etc.) y el tipo de aplicación que se utilice, teniendo la ventaja de que se extiende universalmente ofreciendo un nivel de seguridad homogéneo. Sin embargo, no es el único mecanismo para ofrecer seguridad a IPv6, lo cual se discutirá en este capítulo.

IV.1 Mecanismos que ofrecen seguridad en IPv6

Existen diversos mecanismos que ofrecen determinados grados de seguridad en distintos protocolos, implementaciones y herramientas de IPv6, como por ejemplo:

a) Modelo seguro punto a punto

La arquitectura de IPv6 se puede adaptar fácilmente a modelos punto a punto de manera segura, donde los puntos finales (hosts, servidores o enrutadores) tienen la responsabilidad de proporcionar los servicios de seguridad necesarios para

proteger cualquier tráfico de datos entre ellos. Esto resulta en una gran flexibilidad para crear dominios confiables basados en políticas donde cada dispositivo o nodo final puede ser miembro de múltiples dominios con distintas políticas de seguridad. Cuando cualquier par de dispositivos finales quieren comunicarse de manera segura, los dispositivos pueden iniciar un intercambio confidencial y autenticado.
Por otra parte existen escenarios híbridos que combinan arquitecturas seguras punto a punto y redes centralizadas, incorporando el modelo de "firewalls distribuidos" que se fundamentan en la administración de firewalls basados en hosts proporcionando una mayor protección, aparte de contar con el modelo convencional de firewall perimetral.

b) Consideraciones de seguridad en la asignación de direccionamiento

Una de las ventajas de IPv6 es el gran espacio de direccionamiento, 128 bits, con el que cuenta, siendo casi imposible recordar alguna dirección IPv6.
Una consideración crítica en el diseño de IPv6 es cómo los hosts conectados en una red IPv6 crean su identificador de interfaz, puesto que esto tiene varias implicaciones de seguridad, siendo que las direcciones IPv6 son formadas combinando prefijos de red con éste; el cual puede ser configurado manualmente o utilizando la auto-configuración stateless o stateful.

La configuración stateless (sin intervención o descubrimiento automático) habilita una configuración básica de las interfaces de IPv6 en la ausencia de un servidor DHCPv6, permitiendo al sistema generar sus propias direcciones locales y globales y verificar la duplicidad de direcciones. Este método utiliza el protocolo de descubrimiento de vecinos NDP para el intercambio de información; sin embargo, existen diversas amenazas que comprometen la comunicación como el envío de información falsa, denegación de servicio, y ataques para redirigir los paquetes a otro destino, por consiguiente para resolver este tipo de vulnerabilidades se hace uso de SEND descrito en el siguiente inciso.

En la configuración stateful o configuración predeterminada, el host obtiene la dirección de la interfaz y/o la información y parámetros de configuración a través de un servidor DHCPv6.
DHCPv6 ofrece un excelente método para tener un control central de la asignación de las direcciones IP para conseguir una mayor seguridad y políticas de QoS. Por ejemplo, DHCPv6 puede usarse para asignar, rotar, y almacenar asignaciones de direcciones generadas aleatoriamente para alcanzar una privacidad de direcciones.

c) SEND (SEcure Neighbor Discovery)

Los nodos IPv6 usan NDP (Neighbor Discovery Protocol), definido en el RFC 4861, para el descubrimiento de vecinos, auto-configuración de direcciones, resolución de direcciones, detección de vecinos inalcanzables, detección de direcciones duplicadas, y redirección para que un enrutador informe a un nodo el mejor primer salto para alcanzar un destino en particular; todo con la finalidad de mantener información actualizada en relación a los vecinos activos. Si no se utiliza la seguridad en este protocolo puede llegar a ser vulnerable, por consiguiente se diseñó el protocolo SEND, RFC 3971, para controlar las amenazas contra NDP sin la necesidad de utilizar IPSec, aunque existen normas donde se especifica que el descubrimiento de vecinos de IPv6 y los mecanismos de auto-configuración de direcciones pueden ser protegidos con IPSec AH; sin embargo, en la práctica se limita a que las SAs de IPSec deben ser pre-configuradas manualmente siendo inviable en implementaciones a gran escala.
Básicamente SEND se encarga de proporcionar seguridad basada en mensajes y solicitudes de información y certificación de rutas, utilizados para descubrir una ruta certificada. Para asegurar que el emisor del mensaje, para el descubrimiento de vecinos, es el “propietario” de la dirección enviada hace uso de CGAs (Cryptographically Generated Addresses) para garantizar la integridad en todos los

mensajes, y para proporcionar autenticación de la identidad del emisor se usa la firma de llave pública RSA, evitando reenvíos al usar “timestamps” (para tráfico multicast) y “nonce” (para tráfico entre un par en una comunicación).

La implementación de CGA, definida en los RFCs 4581 y 4982, es una alternativa al uso de PKI para realizar una autenticación de llave pública y se refiere a la generación de un identificador de interfaz, por ejemplo los 64 bits menos significativos, de una dirección IPv6 utilizados para almacenar un hash criptográfico de una llave pública.

El uso de SEND ayuda a proteger contra mensajes engañosos que crean falsas entradas en la cache de los vecinos (spoofing), fallas en la detección de vecinos no alcanzables, ataques de DoS (Denial of Service) por detección de direcciones duplicadas en el descubrimiento de vecinos, ataques de información y solicitud a enrutadores, y ataques de réplica.

d) DNSsec (Domain Name System Security)

El DNS es un conjunto de protocolos y servicios (base de datos distribuida) que permite a los usuarios utilizar nombres en lugar de direcciones IP para acceder a un determinado servidor.
En el RFC 4472 se mencionan los elementos y consideraciones de operación para el DNS con IPv6, donde muchas consideraciones son inseguras sufriendo vulnerabilidades como son: modificación en los datos, denegaciones de servicio, obtención de datos, re-direccionamiento de consultas, suplantación de servidores DNS, entre otros. Para prevenir estas vulnerabilidades se puede implementar el DNSsec (DNS Security) proporcionando mejores mecanismos de autenticación basados en firmas criptográficas para validar la integridad y el origen de los datos del DNS.

Además se consideran algunos puntos de seguridad para DNS IPv6, como por ejemplo, las direcciones locales no deben ser publicadas

e) MIPv6 (Movilidad versión 6) con IPSec o RR (Return Routability)

Se entiende por movilidad a la capacidad que tiene un MN (Mobile Node), dispositivo móvil de una red, para mantener la misma dirección IP a pesar de que se desplace físicamente a otra red y que, sin importar su ubicación en la misma, puede seguir siendo accesible a través de su misma dirección IP original.

Hay que tomar en cuenta que el concepto de “movilidad” es distinto al de “portabilidad”; mientras la primera consiste en proporcionar a los usuarios móviles la capacidad de cambiar de punto de acceso mientras mantienen su conexión de red, la segunda consiste únicamente en estar conectado a Internet (en determinados sitios), habitualmente de manera inalámbrica.

La definición del protocolo que permite movilidad en IPv6, MIPv6 (Mobile IPv6), esta definido en el RFC 3775, el cual especifica que el MN siempre pretenderá ser accesible desde su dirección principal o *home address*, sin importar si se encuentra en su HN (Home Network), red origen o vinculo principal, o si se encuentra en una FN (Foreign Network), red visitada o vinculo externo. La dirección principal es la dirección IP fija (para permitir que el nodo sea accesible mediante una entrada válida en el DNS y esconder la movilidad a las capas superiores) que le corresponde al MN en su HN.

Mientras el MN se encuentre en su HN, los paquetes destinados a su *home address* son ruteados utilizando los mecanismos estándares de ruteo de Internet. La característica de movilidad se invoca cuando el MN se desplaza a una FN donde el MN adquiere una nueva dirección denominada *care-of-address,* con igual

prefijo de red al de la red visitada. Una vez configurada esta dirección, debe informársela a un nodo ubicado en su red origen llamado HA (Home Agent). Este proceso de asociar la *home address* con la *care-of address* se conoce como *binding*, y se realiza cuando el MN envía un mensaje BU (Binding Update) a su HA informándole de su nueva dirección, en contestación al BU se envía un BA (Binding Acknowledgement).
Estos son unos de los muchos mensajes MIPv6 que se codifican en un nuevo encabezado de extensión de IPv6 llamado Mobility Header

A partir de ahora, el HA comienza a funcionar como proxy del MN. Cualquier paquete enviado a la *home address* del MN será recibido por su HA y éste reenviará los paquetes, formando un túnel bidireccional, a la *care-of address* del MN, es decir, cuando el MN envía un paquete, primero lo manda, utilizando el túnel, al HA quien lo desencapsula obteniendo el mensaje original, y lo reenvía hacia su destino final CN (Correspondent Node), nodo fijo o móvil por el cual el MN quiere establecer la comunicación. Esto se conoce como “Ruteo Triangular” y se muestra en la Figura IV.1.

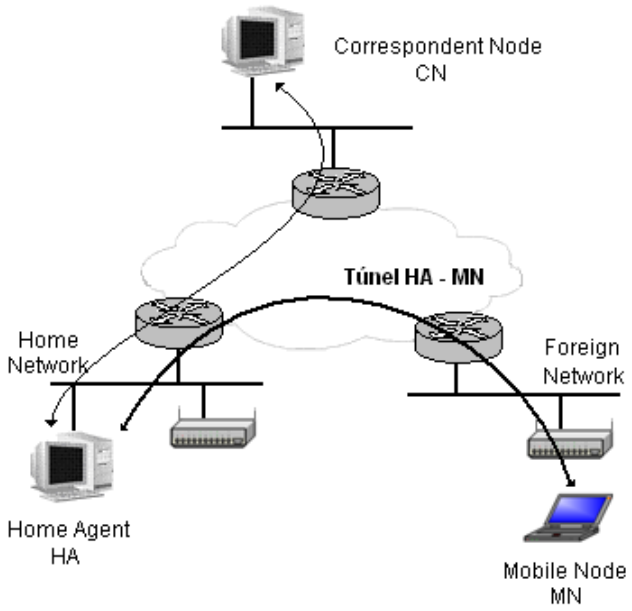


Figura IV.1 Movilidad con ruteo no optimizado

En este modelo todo el tráfico entre el MN y el CN debe pasar por el HA, lo cual se convierte en un cuello de botella y en un punto central de falla, ya que si el HA falla todas las conexiones se pierden, dando como resultado que este tipo de ruteo sea ineficiente, sin embargo, para solucionar este aspecto existe otro modo denominado “Route Optimization”, donde el MN debe registrar su ubicación actual al CN (además de registrarse con el HA) enviándole un BU. A partir de ahora, el tráfico entre estos dos nodos se enviará directamente, sin pasar por el HA y los paquetes enviados por el CN tendrán la *care-of address* en su dirección destino.

El primer problema de seguridad se encuentra en el momento en el que el MN registra su movimiento con su HA, *home registration*. Por ejemplo, un atacante podría enviar un BU al HA indicándole una *care-of address* falsa y diciéndole que el MN está en una ubicación distinta de la cual está. Para este caso, con el fin de evitar cualquier tipo de ataque, los dos nodos deben definir una asociación de seguridad, usando IPSec, para proteger la autenticidad e integridad de los mensajes intercambiados en el proceso (BU y BA).

El uso de IPSec para la comunicación entre el MN y el HA, RFC 4877, se basa en la implementación con ESP con cifrado NULL en modo transporte para el cifrado. Para evitar ataques de réplica es necesario contar con el protocolo IKEv1 (IKEv2 se encuentra en etapa de revisión: draft-ietf-mip6-ikev2-ipsec-06.txt). Se puede hacer uso de MOBIKE (Mobility and Multihoming IKEv2 Protocol) definido en el RFC 4555, el cual es una extensión de IKEv2 para ser usado en escenarios donde se hace uso de VPNs. Además se encuentra en etapa de revisión proporcionar seguridad entre la comunicación del MN y CN usando IPSec (draft-ietf-mip6-cn-ipsec-02.txt) en lugar de RR (Return Routability) otro mecanismo de seguridad que se menciona a más adelante.

Se puede implementar una alternativa en el Protocolo de Autenticación de IPv6 definido en el RFC 4285, el cual es un documento informativo cuyo propósito es dar una solución alternativa de IPSec para tener un medio seguro donde puedan transitar los mensajes BU y BA entre el MN y HA usando una opción de autenticación en un mensaje móvil que es incluido en esos mensajes. El

mecanismo para autenticar el MN con el HA o con el AAAH (Authentication, Authorization, and Accounting server in the Home Network) se fundamenta en una asociación de seguridad móvil basada en llaves compartidas creadas de manera manual o automática entre el MN y su respectiva identidad de autenticación

El segundo problema se presenta en el proceso de “Route Optimization”. Si los BU no son autenticados, el nodo correspondiente puede ser utilizado como cómplice involuntario del ataque. Por ejemplo, un nodo podría enviarle un BU al CN indicándole una nueva *care-of address* para una *home address* determinada implicando que todo el tráfico sea redirigido hacia esta nueva dirección. La *care-of address* elegida por el atacante podría ser su propia dirección IP o cualquiera que él desee. Para este problema se integró con el proceso “Route Optimization” un mecanismo básico de seguridad llamado RR que aunque no elimina todas las amenazas si limita a los posibles atacantes que son capaces de monitorear la ruta entre el HA y el CN, además de permitir que el CN tenga una seguridad razonable en cuanto a que el MN es direccionable en su HA como en su *care-of address*. Únicamente después que este procedimiento tuvo éxito, el CN procesa el BU y realiza la optimización del ruteo.

El mecanismo básico de RR consiste de dos chequeos diferentes: el de la *home address* y de la *care-of address*. Para el primero, el MN envía un mensaje Home Test Init (HoTI), a través del HA, al CN, quien le contesta con un mensaje Home Test (HoT), también por medio del HA. Para el segundo, el MN envía un mensaje Care-of Init Test (CoIT) directamente al CN, quien le contesta con un Care-of Test (CoT). Una vez que el MN ha recibido las dos contestaciones le envía el BU al CN.

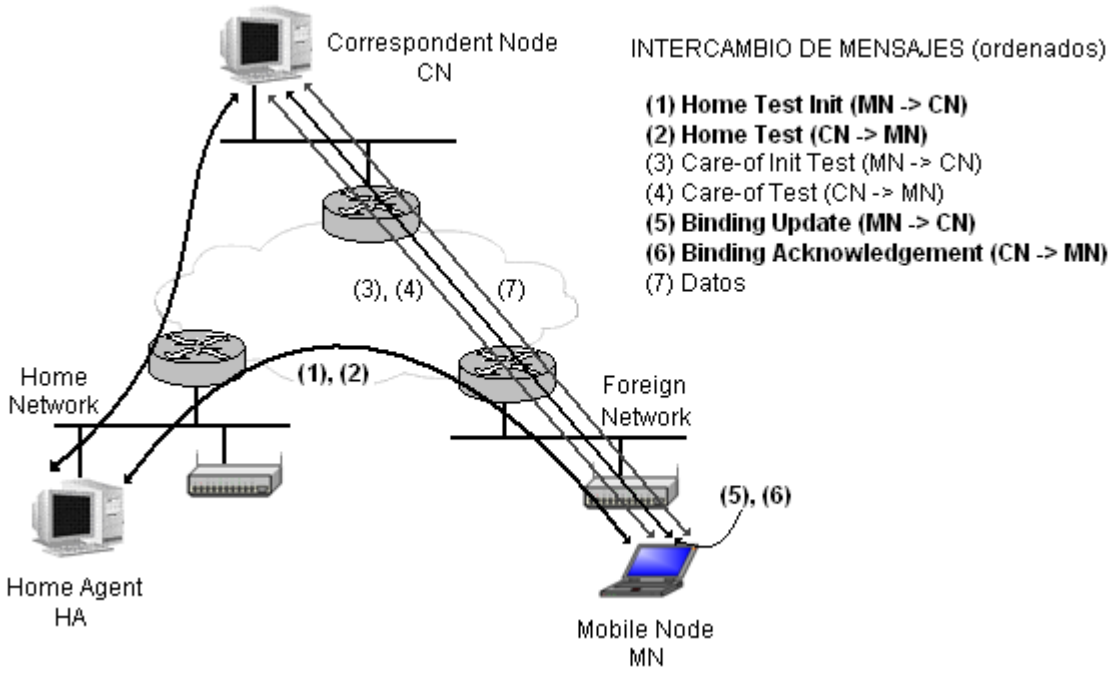


Figura IV.2 Movilidad con ruteo no optimizado

Asimismo, para optimizar el proceso de “Route Optimization” y usar muy poco intercambio de mensajes, se definió una norma definida en el RFC 4449 la cual describe un mecanismo de seguridad móvil usando llaves compartidas estáticas denominado “Securing Mobile IPv6 Route Optimization”

La implementación de movilidad sobre IPv4 comparte muchas de las características que su contraparte en IPv6, sin embargo, la movilidad sobre IPv6 ofrece una serie de ventajas.

- No hay necesidad de enrutadores especiales. La implementación de movilidad en IPv6 funciona en cualquier lugar físico sin la necesidad de características especiales en el enrutador local.
- El soporte de optimización en el ruteo es intrínseco a la implementación de IPv6, no así en IPv4 que requiere una serie de parches externos.
- La implementación de Movilidad sobre IPv6 esta totalmente desacoplada de la capa de enlace, usando IPv6 Neighbor Discovery, con lo cual le otorga mayor robustez al protocolo.

f) Enrutamiento y Fragmentación

Debido a que todos los nodos IPv6 (incluyendo hosts) deben de tener habilitado el proceso enrutamiento para el envío y recepción de paquetes existen varios aspectos de seguridad asociados como son DoS o spoofing. Por ejemplo, se pueden tener ataques repitiendo una dirección varias veces hacia una misma ruta especificada en el encabezado de enrutamiento, o bien, se podrían alternar las direcciones hacia varias rutas especificadas dando como resultado una re-transmisión del paquete en el enrutador o firewall. Estos ataques pueden ser contrarrestados asegurándose que en los encabezados de enrutamiento no se tenga la misma dirección para una ruta especificada más de una vez, así como observar cuál es el ingreso/egreso del filtro para verificar que la dirección fuente sea la apropiada para la dirección destino.

Algunos de estos problemas de seguridad pueden ser resueltos por medio del "Enrutamiento Tipo 2" para MIPv6 descrito en el RFC 3775.

Además, es importante que los nodos implementen diferentes tipos de enrutamiento apropiadamente, ya que es posible aplicar reglas de filtrado por separado para los diferentes tipos de encabezado de enrutamiento. Por diseño, los hosts deben de procesar el Tipo 2 para soportar MIPv6, pero para los enrutadores no es necesario, por lo que es deseable no permitir o limitar el procesamiento Tipo 0 en hosts y algunos enrutadores.

En relación a la fragmentación la actual especificación de IPv6 en el RFC 2460, no define un tamaño mínimo del paquete para los fragmentos del paquete antes del último, con excepción ante la necesidad de colocar la parte no fragmentada en cada paquete fragmentado; abriendo la posibilidad a ataques de DoS por el envío de un gran número de pequeños fragmentos sin fragmento terminal, ocasionando gran consumo en los recursos del procesamiento y sobrecarga de los buffers, sin embargo, al definir un tamaño para los fragmentos del paquete se puede reducir este impacto limitando la razón de llegada de los fragmentos y el número de fragmentos que necesitan ser procesados.

Los paquetes con fragmentos solapados son considerados un mayor riesgo, pero la especificación no define un comportamiento para minimizar este efecto.

g) ICMPv6 (Internet Control Message Protocol versión 6) y Multicast

A diferencia de IPv4, ICMPv6 permite una respuesta para notificar algún error cuando un paquete no fue procesado y es enviado a una dirección multicast. Estas respuestas pueden ser porque se recibió un paquete muy grande que es limitado por el MTU (Packet too big), o porque el paquete recibido contiene opciones no reconocidas en los encabezados “Opciones Salto a Salto” y “Opciones de Destino” (Parameter Problem), ocasionando un consumo de recursos en los enrutadores en una red con multicast habilitado por la generación de múltiples respuestas.

h) Extensiones para la privacidad de direcciones

El propósito de las extensiones de privacidad para auto-configuración stateless de direcciones (ietf-ipv6-privacy-addrs-v2) es cambiar el identificador de interfaz (y por lo tanto las direcciones de ámbito global generadas por este cambio), haciendo más difícil identificar el nodo especifico donde se esta realizando la conexión. Estas direcciones modificadas son topológicamente correctas.
Cabe señalar que aunque un nodo sea considerado como conocido, un cambio en la dirección implica definir reglas o políticas basadas en direcciones (por ejemplo: listas de control de acceso). Sin embargo, los nodos que emplean direcciones privadas no tienen que usar este tipo de mecanismo para todas sus comunicaciones.

i) Otros aspectos de seguridad

También se tienen otras formas de implementar seguridad en las dos versiones de IP. Por ejemplo: implementación de filtros como IPFilter (ipf) disponible en sistemas UNIX o PF (Paket Filter) en OpenBSD con funciones más avanzadas; configuración de firewalls (en el caso de Linux se denominan ip6tables, y en ambiente BSD y MacOS IPFirewall o ipfw); configuración de listas de acceso; autenticación con servidores, entre otros.

IV.2 Aplicaciones prácticas de IPSec

Los encabezados de IPSec proporcionan una solución viable para una interconexión segura de redes locales (intranet), acceso seguro de usuarios remotos, o una conexión de una corporación con sus patrocinadores, proveedores y/o consumidores (extranet), además de aumentar la seguridad en el comercio electrónico debido al nivel donde es implementado (nivel de red del modelo OSI).
Existen diferentes esquemas de seguridad donde se puede configurar IPSec para los diferentes ámbitos antes mencionados. Cuando la implementación de IPSec radica en un host o sistema final, los paquetes pueden ser asegurados punto a punto, es decir, desde el origen de los datos hasta su destino final. La Figura IV.3 muestra este esquema, donde cada paquete que sale del host está asegurado, implicando que todo paquete que no haya sido asegurado por IPSec sea eliminado, asimismo se puede definir el tipo de tráfico (telnet, SMTP, HTTP, etc.) que deba ser asegurado a través de la definición explícita de una SA. Generalmente para este esquema de seguridad punto a punto se utiliza IPSec en modo transporte debido a que los puntos finales de la comunicación son también los puntos finales de IPSec, aunque también el modo túnel se puede usar añadiendo un encabezado extra por definición al usar este tipo de modo.

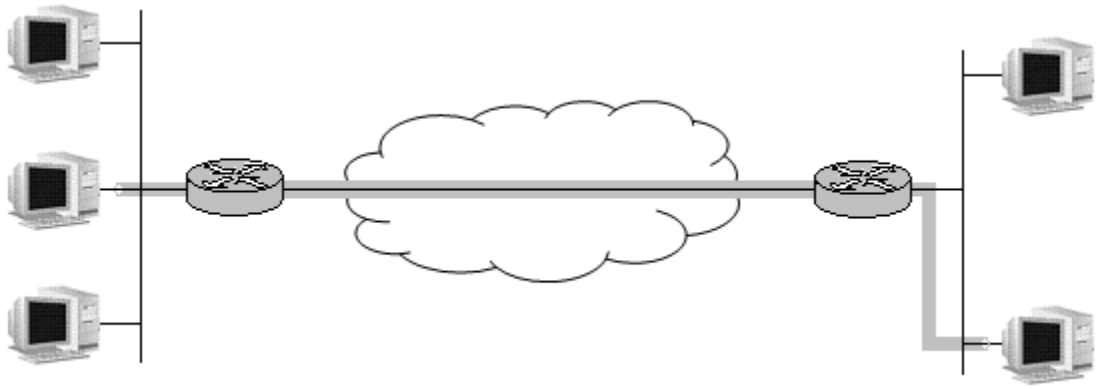


Figura IV.3 Esquema de configuración segura punto a punto a través de la red

Algo importante de mencionar de la seguridad punto a punto es que puede afectar el funcionamiento de otras aplicaciones que requieran inspeccionar los paquetes en tránsito (Firewalls, QoS, Monitoreo de tráfico, etc.) y no puedan hacerlo debido a que solo verían paquetes con ESP. Quizá la implementación más común son las VPNs que han sido vistas como una excelente alternativa de ahorro en lugar de contratar líneas dedicadas para utilizar la red pública con servicios de seguridad. Se considera una VPN cuando dos enrutadores establecen túneles a través de los cuales envían tráfico desde una subred localmente protegida hacia otra subred remotamente protegida, en la Figura IV.4 se muestra un esquema de este tipo.

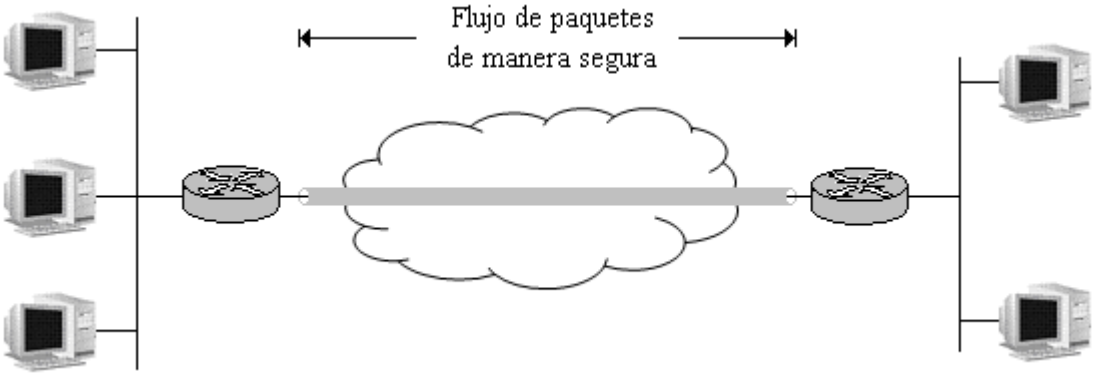


Figura IV.4 Esquema de configuración de una VPN a través de la red

Debido a que las VPNs protegen el tráfico que circula en una red protegida, se debe usar IPSec en modo túnel, a menos que el tráfico se envíe por un túnel vía otro protocolo, como L2TP, en donde se usará el modo transporte.

Existe otro tipo de implementación denominado “Road Warrior”, que es una combinación entre la configuración punto a punto donde un host cifra y descifra el tráfico que envía y recibe, y una VPN donde un enrutador realiza también esta función, es decir, una computadora y un enrutador implementan IPSec para asegurar los paquetes en su comunicación punto a punto. La Figura IV.5 ilustra este esquema.

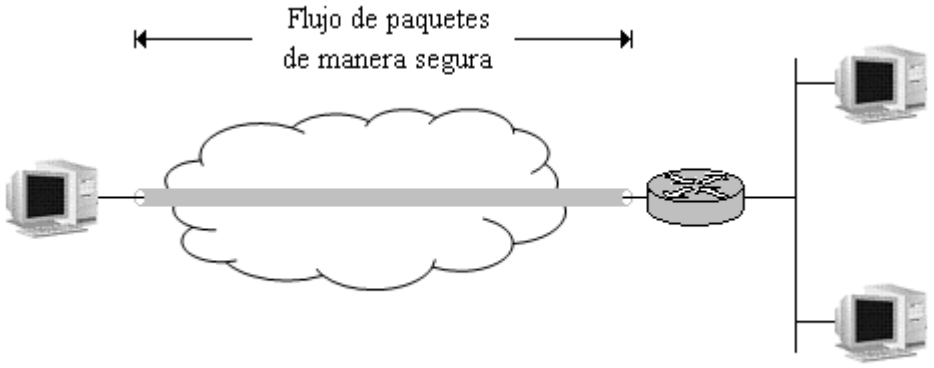


Figura IV.5 Esquema de configuración de un “Road Warrior”

La implementación “Road Warrior” es utilizada para los usuarios que requieren un acceso remoto a una red protegida para acceder a los recursos corporativos a cualquier hora y en cualquier lugar de manera transparente y confiable.

También es posible la implementación de túneles anidados, un ejemplo podría ser una institución que tiene un gateway de seguridad para proteger su red de ataques externos, pero además tiene otro gateway de seguridad para su red interna contra ataques internos. La Figura IV.6 muestra este esquema, el cual es difícil de mantener y establecer, pero quizá es útil y necesario para ciertas necesidades entre instituciones con instalaciones remotas.

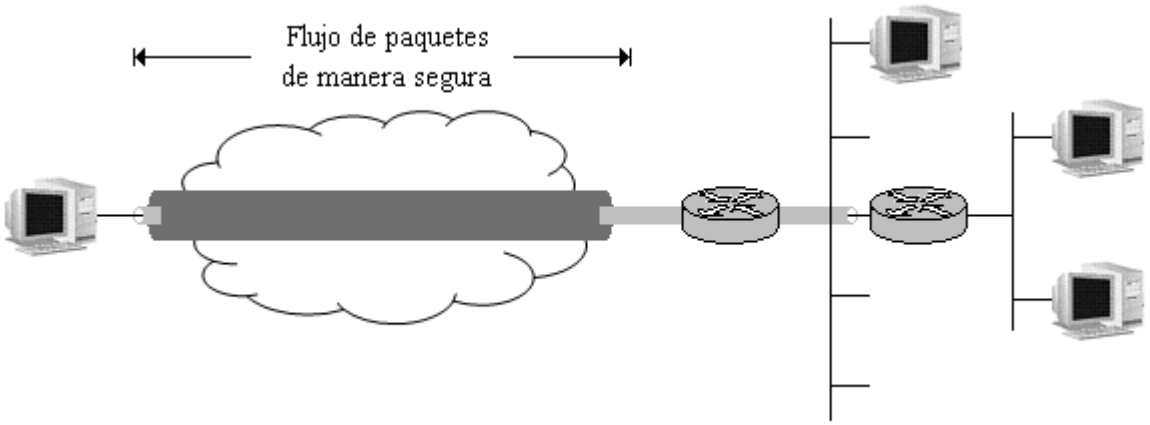


Figura IV.6 Esquema de configuración de túneles anidados

IV.3 Ventajas y limitaciones de IPSec

En cuanto a las ventajas que tiene IPSec está el desarrollo de APIs para facilitar su uso. El primer socket de IPSec que surgió para ser estandarizado en la familia de protocolos fue definido en 1998 con el nombre de PF-KEY para la comunicación entre la administración de llaves de las aplicaciones con la administración de llaves internas del sistema operativo (SADB), sin embargo este socket nunca fue concluido.

Existe un Socket Avanzado API para IPv6 (RFC 3542) donde se omitió el control para IPSec; sin embargo, los proyectos WIDE (www.wide.org) y KAME (www.kame.net) han continuado con los trabajos iniciales del API de IPSec y producido implementaciones estables para IPSec sobre IPv6 e IKEv1 en sistemas basados en Unix BSD (FreeBSD, NetBSD, OpenBSD y BSDi). Estas implementaciones incluyen el socket PF-KEY definido en el RFC 2367 utilizado para acceder al procesamiento de la administración de llaves de IPSec, así como controlar el procesamiento de las políticas de IPSec (SPD y SAD).

Un API más reciente, es el SAPI (Service API) que proporciona una interfaz genérica para configurar y administrar las reglas de las bases de datos de IPSec (SPD y SAD). Estas bases de datos o reglas contienen varios atributos permitiendo una implementación dada de IPSec para determinar cuales son los paquetes que entran y salen. Además IPSec SAPI permite a una aplicación de un cliente recibir notificaciones indicando cambios de estado, alertas, y otros datos informativos.

Cabe mencionar que constantemente se están haciendo esfuerzos en el desarrollo de las APIs para que sean consolidadas y se establezca una norma interoperable. Además, con las recientes actualizaciones en la arquitectura de seguridad para el protocolo de Internet (RFC 4301) e IKEv2 (RFC 4306) se puede considerar que el protocolo ha sido definido lo suficientemente bien, lo cual simplificará grandemente el trabajo requerido por las aplicaciones de los desarrolladores y facilitará la transición de las aplicaciones existente de IPv4 para usarse en conjunto con IPv6 con servicios seguros utilizando IPSec.

Otra de las ventajas proporcionadas por IPSec es el beneficio que tiene en relación al protocolo SSL, esto es, debido a que SSL trabaja por encima de TCP implica que si hay una interrupción en la sesión de TCP es suficiente para también tener una interrupción en la comunicación con SSL, aunque, afortunadamente esta interrupción sólo bloquea la comunicación no permitiendo a un atacante colocar datos falsos o descifrar información confidencial; IPSec, por otro lado, al estar por debajo de TCP no presentaría este problema, siendo capaz de rechazar paquetes interrumpidos ante cualquier protocolo de capa superior, además de que los paquetes que sean falsificados por un atacante sean rechazados por IPSec basándose en las políticas de seguridad.

Entre sus limitaciones se puede nombrar la falta de una PKI utilizado para verificar la identidad entre las partes y establecer una comunicación confiable. Actualmente

no está implementado en certificados digitales pero se trabaja en ello para una autenticación más fácil de usar y configurar.
También hay que considerar que IPSec necesita llevar información almacenada tanto de cifrado como de autenticación para cada paquete individual, implicando una cantidad significante de sobrecarga.

Finalmente una de las razones por la que IPSec no ha ganado mucha atención como son las VPNs, probablemente se deba a que al operar en un nivel bajo (capa de red), por ejemplo, una aplicación que quiere usar SSL puede ser redistribuida con su propio código SSL sin impactar a otras aplicaciones, aspecto que no sucede con IPSec.

Capítulo 5

Requerimientos y pruebas de IPSec con IPv6

Al ser IPSec un encabezado que se implementó de manera obligatoria en el protocolo IPv6 se deben de considerar el soporte que se tiene para éste en las distintas plataformas existentes (Windows, Linux, BSD, etcétera.) tomando en cuenta los algoritmos criptográficos soportados y las distintas configuraciones en las que puede operar; además se debe analizar cuales son los requerimientos de hardware en el caso de dispositivos como enrutadores, switches, etcétera, si es que se quiere aplicar IPSec.

En este capítulo se mencionaran los requerimientos necesarios para implementar seguridad usando IPSec para algunas plataformas, así como las pruebas que se realizaron en distintos escenarios para comparar la interoperabilidad entre los dispositivos implicados, incluyendo los resultados que se obtuvieron.

V.1 Soporte en Hardware y Software existente

En cuestiones de Hardware existen equipos o módulos para equipos con soporte IPSec para IPv6. En la tabla V.1 se menciona el hardware y software que está evaluado para operar sobre cualquier red de manera estandarizada de acuerdo al programa de certificación “IPv6 Ready Logo Committee”, sin embargo, también

existen equipos de diversas compañías como son 3Com, Cisco, Allied Telesyn, Foundry, Nokia, etcétera, que pueden operar bajo sus protocolos propietarios o internos.

De forma particular, se hicieron pruebas con un Switch Allied Telesyn capa 3 Modelo AT-8948, con versión 2.6.2 y parche 89262-09.paz para tener el soporte de IPSec en el dispositivo. Las pruebas que se realizaron con este equipo se mencionan más adelante.

Fecha de aprobación	Proveedor	País	Nombre del producto	Versión	Descripción del producto	Categoría
2006-03-13	NEC Corporation	Japón	IX1000/IX2000/IX3000 Series (Typified by IX2015)	7.3.21	Enrutador http://www.sw.nec.co.jp/ixseries/ix1k2k/index.html (JP) http://www.neaxnet.com/products/ix.html (US)	IPSec Gateway de Seguridad
2006-03-07	NEC Corporation	Japón	IX1000/IX2000/IX3000 Series (Typified by IX2015)	7.3.21	Enrutador http://www.sw.nec.co.jp/ixseries/ix1k2k/index.html (JP) http://www.neaxnet.com/products/ix.html (US)	IPSec Nodo final
2005-11-11	Panasonic	Japón	IPv6 Stack	2.2.0.f	Pila para consumidores electrónicos	IPSec Nodo final
2005-05-27	Hewlett-Packard Company	E.U.A.	Jetdirect Print Server	J7961A, J7961G	Servidor de impresora HP Jetdirect IPv6/IPv4/IPSec	Host
2005-09-06	SECUi.COM Corporation	Corea	NXG2000	V1.3 (SecuiOS)	Dispositivo multi-gigabit con seguridad integrada que proporciona stateful firewalling, IPSec VPN, protección contra intrusiones, e inspección de contenido.	Enrutador
2006-03-22	Oki Electric Industry Co., Ltd	Japón	OKI IPv6/v4 Dual Stack	1.10.0	Pila desarrollada para sistemas integrados, por ejemplo, Telefonía IP.	IPSec Nodo final
2006-03-31	Oki Information Systems Co., Ltd	Japón	IPv6 TCP/IP Stack For ESWare	V1.0	Pila para sistemas integrados	IPSec Nodo final
2006-05-30	kernel.org	E.U.A.	Linux	2.6.15	Linux (similar al kernel del Sistema Operativo Unix), actuando como host	IPSec Nodo final
2006-11-10	Hewlett-Packard	E.U.A.	HP-UX IPSec	A.02.01	HP-UX IPSec	Nodo final
2007-04-17	Hewlett-Packard	E.U.A.	HP-UX IPSec	A.02.01.01	HP-UX IPSec	Nodo final
2007-04-17	Hewlett-Packard	E.U.A.	HP-UX IPSec for HP-UX 11i version 3	A.02.01.01	Sistema Operativo HP-UX11i	Host
2006-03-10	Samsung Electronics Co. Ltd India Software Operations	India	SISOv6 Stack	3.7.0	Pila IPv6 sobre Ethernet (10/100 Base-T) / (Host/Enrutador)	IPSec Nodo final

Tabla V.1 Hardware y Software que está estandarizado para operar con IPSec para IPv6

En relación a Software se debe de considerar la plataforma y la versión del Sistema Operativo que se va a utilizar para implementar seguridad. En la tabla V.2 se menciona un resumen de distintos S.O. en relación al soporte que tienen con IPv6, así como diversos paquetes que se pueden instalar para un mejor soporte o una mayor seguridad con IPSec.

Sistema operativo (Empresa)	Versión	Soporte / Descripción
Windows (Microsoft)	3.X 95/98 ME NT 4.0	Microsoft no soporta IPv6 en estas plataformas, sin embargo existen alternativas para su uso: 1) Instalando WinSock v5.0 con pila IPv6 de Trumpet. 2) Implementando un protocolo denominado Toolnet6 de Hitachi (solo para algunas tarjetas de red).
	XP	Todas las versiones de XP incluyen IPv6 instalado, pero es preciso habilitarlo. Es necesario tener instalado el Service Pack 1 o posterior. El firewall incluido en el SP2 es diferente al que tiene el ANP. También se puede instalar un paquete adicional llamado Advanced Networking Pack (ANP) que incluye un firewall, un cliente de Teredo y soporte para redes Windows punto a punto. El soporte de IPSec esta limitado debido a que no cuenta con el soporte de IKE o cifrado de datos, además las políticas de seguridad, las asociaciones de seguridad, y las llaves son configuradas a través de archivos de texto y son activadas a través de líneas de comando con la ayuda de ipsec6.exe
	2000	Con soporte desde SP1 instalado o posterior. La instalación varia dependiendo del SP instalado.
	Server 2003	Desde su publicación incluyo la pila IPv6. Con el SP1 se incluyeron características adicionales provenidas por el SP2 de Windows XP. El soporte de IPSec es el mismo utilizado en Windows XP.
	Vista y Server 2008	Nuevas versiones de Microsoft con soporte IPv6 instalado y habilitado por defecto para todas sus versiones (Home Basic, Home Premium, Business, Enterprise, y Ultimate para Windows Vista). El uso de IPSec soporta IKE y cifrado de datos con AES 128/192/256, y en lo referente a las políticas de seguridad se configuraran mediante las consolas de Políticas de Seguridad o el nuevo Firewall con Seguridad Avanzada.
	CE 5.0 Mobile 6 Pocket PC	Utilizados para dispositivos móviles como teléfonos, PDAs, Smart phones, dispositivos embebidos y similares con soporte opcional para IPv6.
Linux (Software libre)	RedHat Debian Fedora SuSe Mandrake etc.	En Linux, IPv6 se implementa como un módulo del kernel. La versión de kernel que se debe de utilizar es la 2.6.x, ya que para la versión 2.2.x el soporte de IPv6 esta obsoleto y para la versión 2.4.x solo cuenta con un soporte limitado. El soporte de IPSec puede ser implementado mediante software adicional como las distribuciones del proyecto S/WAN: FreeS/WAN, OpenS/WAN, StrongS/WAN. Sin embargo, FreeS/WAN detuvo su desarrollo en 2004. Además de estos proyectos también existe un proyecto denominado USAGI (UniverSAI playGround for Ipv6) el cual es una implementación como alternativa para uso de IPv6 con soporte para IPSec y Movilidad
*BSD (Software libre)	FreeBSD 4.0 NetBSD 1.5 OpenBSD 3.0	Sistemas con soporte IPv6 e IPSec para las versiones mencionadas y posteriores.
(Solaris) Sun Microsystems	8	Se cuenta con soporte IPv6 e IPSec para esta versión y posteriores.
Macintosh (Apple Computer)	OS X	Se tiene el desarrollo de IPv6 e IPSec a partir de esta versión. Desde la versión Mac OS X v10.3 "Panther" IPv6 esta habilitado por defecto.
Netware (Novell)	6.1	A partir de esta versión se implementa IPv6
AIX z/OS (IBM)	AIX 4.3 z/OS 1.4	El sistema operativo AIX 4.3 fue la primera plataforma comercial con soporte para IPv6.Para el z/OS se cuenta con soporte de IPSec

Tabla V.2 Soporte de software para IPv6 e IPSec.

También cabe mencionar que hay proyectos como TAHI (http://www.tahi.org), KAME (http://www.kame.org) y WIDE (http://www.wide.org), que están en una colaboración constante para proporcionar una pila IPv6, IPSec o MIPv6 para distribuciones Linux y BSD, en cooperación con el proyecto USAGI (http://www.linux-ipv6.org).

Finalmente, existen diversas aplicaciones que ya tienen implementado el soporte con IPv6 como se muestra en la Tabla V.3.

Aplicación	Categoría	Plataforma o Sistema Operativo
FTP	cliente/consola	Linux, BSD, MacOS y Windows
TELNET	cliente/consola	Linux, BSD, MacOS y Windows
SSH	cliente/consola	Linux, BSD, MacOS. En Windows “SecureCRT SSH” y “Putty”
Internet Explorer	Navegador	Las versiones anteriores a la 7.0 para Windows no soporta las direcciones literales IPv6 (alfa-numéricas) en la URL. Macintosh no soporta IPv6.
Konqueror	Navegador	Sistemas basados en UNIX
Netscape	Navegador	Todas las distribuciones o versiones, deshabilitado por defecto en Mac
Mozilla Firefox	Navegador	Todas las distribuciones o versiones, habilitado en pocas versiones de Linux
Opera	Navegador	El soporte para Mac se añadió en Opera 9.0
Mozilla Thunderbird	Cliente de correo	Sistemas UNIX/Linux y Windows. En Mac a partir de la v2.0 y OS X 10.4.9
Microsoft Outlook	Cliente de correo	Windows no lo soporta
Apple Mail	Cliente de correo	Mac
Windows Media Player	Multimedia	Windows

Tabla V.3 Soporte de IPv6 en distintas aplicaciones

Aplicación	Categoría	Soporte
VLC Video LAN Client	Multimedia	Linux, BSD, MacOS y Windows, entre otros
iTunes	Multimedia	Windows, MacOS
Apache v2	Servidor Web	Todas las distribuciones o versiones
IIS	Servidor Web	Todas las distribuciones o versiones
Tight VNC	Virtual Network Computing	Linux, Windows Experimental (opcional)
BIND	Servidor DNS	Linux, BSD y Windows a partir de la versión 9
Pidgin	Messenger	Linux
Quagga	Software de ruteo	Software de enrutamiento

Tabla V.3 Soporte de IPv6 en distintas aplicaciones (continuación…)

V.2 Pruebas de interoperabilidad realizadas

Las pruebas que se realizaron consistieron en verificar el comportamiento de los encabezados de seguridad sobre diferentes plataformas y versiones de Sistemas Operativos, así como parches y/o paquetes adicionales sobre los mismos. Además se utilizaron analizadores de tráfico para observar el tipo de intercambio de paquetes, y se probaron servicios como telnet y servidores Web como Apache para analizar el comportamiento con este tipo de tráfico al aplicar IPSec. En la tabla V.4 se muestran los S.O., parches y paquetes utilizados.

Estas pruebas básicamente radicaron en configurar un túnel entre dos dispositivos por el cual circularía la información habilitando y configurando IPSec en sus distintas modalidades.

SO / Versión	Versión de Kernel	Service Pack	Paquete adicional
Windows XP	No aplica	SP1 SP2	Advanced Networking Pack
Windows 2003	No aplica	SP1	No aplica
Linux Fedora Core 4	2.6.11 2.6.15	No aplica	FreeS/WAN USAGI
FreeBSD	6.1	No aplica	No aplica

Tabla V.4 Software utilizado durante las pruebas

Para la configuración de los encabezados de IPSec los valores posibles que se configuraron para un soporte en el S.O. Windows se mencionan en la tabla V.5.

Parámetro	Tipo de configuración
Encabezado	AH ESP
Modo de operación	TUNNEL TRANSPORT
Algoritmo de autenticación	HMAC-MD5 HMAC-SHA1 NULL
Acción a ejecutar	APPLY BYPASS
Dirección	INBOUND OUTBOUND

Tabla V.5. Parámetros disponibles en la configuración de los encabezados de IPSec sobre el S.O. Windows

Las pruebas que se realizaron fueron las siguientes:

1. Comunicación entre una computadora con sistema operativo Windows y un Switch de capa 3 en el mismo segmento.
2. Comunicación entre dos computadoras con sistema operativo Windows en el mismo segmento.
3. Comunicación entre dos computadoras con sistema operativo Windows en diferente segmento.
4. Comunicación entre una computadora con sistema operativo Windows y una con distribución Linux en el mismo segmento.
5. Comunicación entre una computadora con sistema operativo Windows y una con distribución BSD en el mismo segmento.

1. **Pruebas entre una computadora con sistema operativo Windows y un Switch de capa 3 en el mismo segmento.**

Esta prueba consistió en la configuración e implementación de un túnel con IPSec entre una PC con Windows XP y SP1 con el complemento Advanced Networking Pack y un switch de capa 3 en el mismo segmento como se muestra en la figura V.1, utilizando direcciones de enlace local y direcciones configuradas manualmente.

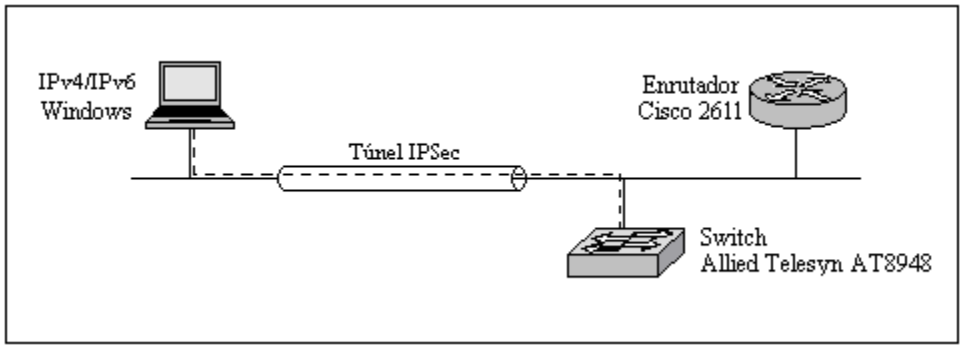


Figura V.1 Esquema de configuración para la prueba 1 entre una PC y un switch capa 3

Para la configuración en la PC, con IPv6 habilitado, primero se configuró una dirección local o manual sobre la interfaz 2 y después se añadió una ruta por defecto que iba hacia el switch, En el siguiente ejemplo se configuró una dirección manual (3ffe:8070:fee1::1), como se muestra a continuación:

c:\>ipv6 adu 2/3ffe:8070:fee1::1 (añade la dirección IPv6 en la interfaz 2)

c:\>ipv6 if 2 (muestra la interfaz 2 de IPv6)

Interfaz 2: Pseudo-interfaz de protocolo de túnel automático
GUID {48FCE3FC-EC30-E50E-F1A7-71172AEEE3AE}
no usa descubrimiento de vecinos
no usa descubrimiento de enrutador
preferencia de enrutamiento 1
Dirección IPV4 incrustada EUI-64: 0.0.0.0
dirección de capa de enlace de enrutador: 0.0.0.0
preferred global **3ffe:8070:fee1::1**, duración infinite (manual)
preferred link-local fe80::5efe:1.1.11.1, duración infinite
enlace MTU 1280 (enlace MTU 65515)
límite de saltos actual128
tiempo alcanzable 38000ms (base 30000ms)
intervalo de retransmisión 1000ms
transmisiones DAD 0
longitud de prefijo de sitio predeterminada 48

c:\>ipv6 rt (muestra la tabla de ruteo de IPv6)

3ffe:831f::/32 -> 5 pref 2if+8=10 duración infinite (configuración automática)
2002:84f8:6cfe::/48 -> 4 pref 8 duración 29d23h57m42s (configuración automática)
3ffe:8070::/28 -> 4 pref 8 duración 29d23h57m42s (configuración automática)
2001:448::/35 -> 4 pref 8 duración 29d23h57m42s (configuración automática)
::/0 -> 4/fe80::2d0:58ff:fef3:6d41 pref 256 duración 27m42s (configuración automática)

c:\>**ipv6 rtu ::/0 2/::1.1.1.2 (añade una ruta por defecto por la interfaz 2)**

c:\>ipv6 rt

::/0 -> 2/::1.1.1.2 pref 1if+0=1 duración infinite (manual)
3ffe:831f::/32 -> 5 pref 2if+8=10 duración infinite (configuración automática)
2002:84f8:6cfe::/48 -> 4 pref 8 duración 29d23h59m11s (configuración automática)
3ffe:8070::/28 -> 4 pref 8 duración 29d23h59m11s (configuración automática)
2001:448::/35 -> 4 pref 8 duración 29d23h59m11s (configuración automática)
::/0 -> 4/fe80::2d0:58ff:fef3:6d41 pref 256 duración 29m11s (configuración automática)

Después se prosiguió a configurar los parámetros para habilitar IPSec en la PC, en este caso con direcciones configuradas manualmente, como se muestran en la figura V.2 para las Asociaciones de Seguridad y en la figura V.3 para las Políticas de Seguridad.

```
Security Association List

SAEntry   SPI            SADestIPAddr                                DestIPAddr
                         SrcIPAddr                                   Protocol    DestPort
  SrcPort     AuthAlg     KeyFile        Direction   SecPolicyIndex
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------
2         3001           3ffe:8070:fee1::2                           POLICY
                         POLICY                                      POLICY      POLICY
  POLICY      HMAC-MD5    test.key       OUTBOUND    2            ;
1         3000           3ffe:8070:fee1::1                           POLICY
                         POLICY                                      POLICY      POLICY
  POLICY      HMAC-MD5    test.key       INBOUND     2            ;
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------
```

Figura V.2 Parámetros de configuración de las SAs en una PC para la prueba 1

```
Security Policy List

Policy      RemoteIPAddr                                    LocalIPAddr
            Protocol      RemotePort    LocalPort   IPSecProtocol  IPSecMode   RemoteGWIPAddr
                              SABundleIndex  Direction   Action      InterfaceIndex
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------
2         - 3ffe:8070:fee1::2                                - *
          - *           - *           - *           AH              TRANSPORT   *
                              NONE           BIDIRECT    APPLY       0               ;
1         - *
          - *           - *           - *           NONE            *           *
                              NONE           BIDIRECT    BYPASS      0               ;
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------

- = Take selector from policy.
+ = Take selector from packet.
```

Figura V.3 Parámetros de configuración de las SPs en una PC para la prueba 1

En relación a la configuración del switch de capa 3 se realizó lo siguiente después de habilitar IPv6.

Creación del túnel entre los dos equipos.

(se añade un túnel entre la PC y el Switch sobre la interfaz virtual 1)
SecOff > add ipv6 tunnel local=1.1.1.2 target=1.1.1.1 interface=virt1 ipaddress=3ffe:8070:fee1::2

Info (1066284): v6 over v4 tunnel successfully created.

SecOff > show conf dyn=ipv6 (se muestra la configuración de IPv6)
#
IPv6 configuration
#
enable ipv6
add ipv6 6to4 ip=1.1.1.3
add ipv6 tunnel local=1.1.1.2 target=1.1.1.1 ip=3ffe:8070:fee1::0002 int=virt1
create ipv6 int=vlan1
add ipv6 int=vlan1 ip=2001:0448:0004::0001/64 type=unicast
set ipv6 nd int=vlan1 hop=64
add ipv6 route=::/35 next=2002:84f8:6cfe::1 int=vlan1 pref=360

Creación de la ruta por defecto que apunta hacia la PC

(se añade una ruta por default)

SecOff > add ipv6 route=::/0 next=3ffe:8070:fee1::1 int=vlan1 pref=360
#
Info (1066267): IPV6 Route successfully added.

SecOff > show conf dyn=ipv6
IPv6 configuration
#
enable ipv6
add ipv6 6to4 ip=1.1.1.3
add ipv6 tunnel local=1.1.1.2 target=1.1.1.1 ip=3ffe:8070:fee1::0002 int=virt1
create ipv6 int=vlan1
add ipv6 int=vlan1 ip=2001:0448:0004::0001/64 type=unicast
set ipv6 nd int=vlan1 hop=64
add ipv6 route=::/0 next=3ffe:8070:fee1::1 int=vlan1 pref=360
add ipv6 route=::/35 next=2002:84f8:6cfe::1 int=vlan1 pref=360

Información general en relación a la interfaz virt1 y a las rutas configuradas

SecOff > sh ipv6 interface=virt1 (se muestra la interfaz virtual 1)

IPV6 Interface Configuration

Interface..... virt1
Ipv6 Interface Index 2
Link-layer address..... ipv4 tunnel
Link-layer state ..... ipv4 tunnel
EUI-64 Interface Identifier ..... ipv4 tunnel
IPSec ..... No
True MTU/Link MTU ..... -/1280
Multicast status..... Enabled
Send Router Advertisements? ..... No
Ipv6 Interface Addresses :

Int	Addresses		PLen	Decrement		
Type	Scope	State	Enabled	Valid	Preferred	Publish
0	**3ffe:8070:fee1::0002**		**/64**	**No**		

unicast global preferred Yes infinite infinite No

```
SecOff > show ipv6 route   (se muestra la tabla de ruteo)
IPV6 Routing Table Entries

Destination prefix    --->   Next Hop
Int.   Age   Policy Protocol       Metric  Pref Tunnel  DLCI  Flags
-------------------------------------------------------------------------
::/0 ---> 3ffe:8070:fee1::1
vlan1  no    0      static        1       360  no      -
::/35 ---> 2002:84f8:6cfe::1
vlan1  no    0      static        1       360  no      -
2001:448::/35 ---> 2001:448::
vlan1  yes   0      neighdisc     1       250  no      -
2001:448:4::/64 ---> ::
vlan1  no    0      interface     1       0    no      -
2002:84f7:fd0e::/48 ---> ::
virt0  no    0      interface     1       0    yes     -
2002:84f8:6cfe::/48 ---> 2002:84f8:6cfe::
vlan1  yes   0      neighdisc     1       250  no      -
3ffe:8070::/28 ---> 3ffe:8070::
vlan1  yes   0      neighdisc     1       250  no      -
3ffe:8070:fee1::/64 ---> ::
virt1  no    0      interface     1       0    yes     -
-------------------------------------------------------------------------
Codes: P=publish, D=default, A=addrconf, S=stale, L=onlink
N=nonexthop, C=cache, F=flow, Y=policy, U=unknown
```

Los resultados que se obtuvieron en esta etapa no fueron los esperados ya que al comprobar la conectividad vía PING implementando IPSec hacia cada uno de los equipos no se tuvo una conectividad continua, teniendo una pérdida de paquetes.

2. Pruebas entre dos computadoras con sistema operativo Windows en el mismo segmento.

Estas pruebas consistieron en la implementación de un túnel con IPSec entre dos computadoras sobre distintas versiones de Windows y/o parches instalados en el mismo segmento utilizando direcciones locales, manuales y auto-configuradas siguiendo el esquema que se muestra en la figura V.4.

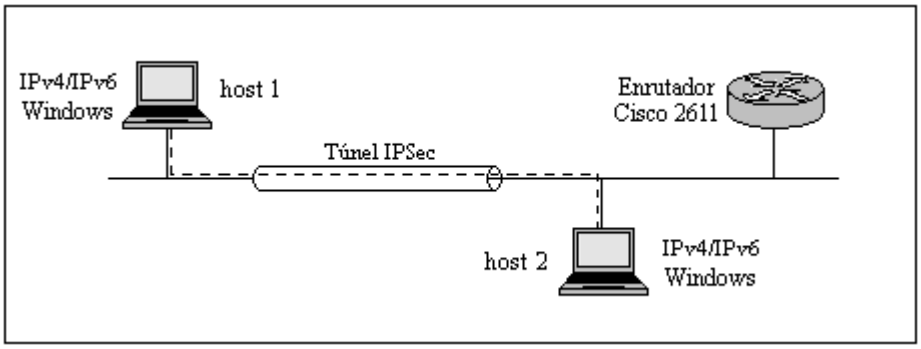


Figura V.4 Esquema de configuración para la prueba 2 entre dos PC en el mismo segmento

Para realizar esta prueba se creó un túnel entre las dos PC, y se configuraron las SAs como se muestran en las Tablas V.6 y V.7, y las SP en la Tabla V.8.

Número de Asociación	1
SPI	3000
Dir. IP destino de SA	fe80::0211:11ff:fe2b:40f2
Dirección IP destino	POLICY
Dirección IP origen	POLICY
Protocolo	POLICY
Puerto destino	POLICY
Puerto origen	POLICY
Alg. de autenticación	HMAC-MD5
Archivo llave	test.key
Dirección	INBOUND
Índice de SP	2

Tabla V.6 Parámetros de configuración para la prueba 2 en la creación de SAs en la PC (host1)

Número de Asociación	2
SPI	3001
Dir. IP destino SA	fe80::0211:11ff:fe2b:40f2
Dirección IP destino	POLICY
Dirección IP origen	POLICY
Protocolo	POLICY
Puerto destino	POLICY
Puerto origen	POLICY
Alg. de autenticación	HMAC-MD5
Archivo llave	test.key
Dirección	OUTBOUND
Índice de SP	2

Tabla V.7 Parámetros de configuración para la prueba 2 en la creación de SAs en una PC (host2)

Número de política	2
Dirección IP destino	- fe80::208:0dff:fed2:161
Dirección IP local	- fe80::211:11ff:fe2b:40f2
Protocolo	- *
Puerto destino	- *
Puerto local	- *
Encabezado IPSec	AH
Modo IPSec	TRANSPORT
Dirección IP del gateway de seguridad destino	*
Índice del conjunto de las SA	0
Dirección	BIDIRECT
Acción	APPLY
Índice de la interfaz	0

Tabla V.8 Parámetros de configuración para la prueba 2 en la creación de SPs en una PC (host1 y host2)

En esta etapa, cabe mencionar que para el host1 si una de las SA se configuró como INBOUND, en el host2 se debe de configurar como OUTBOUND, mientras que para las SP se debe tomar en cuenta que para la dirección destino y la dirección local se conservará el orden en las dos computadoras, es decir, si la dirección destino es Dir1=A y la dirección local es Dir2=B, la configuración en **ambas** máquinas será Dir1=A y Dir2=B.

En la figura V.5 se observa el tráfico analizado con Ethereal cuando se configuró el encabezado AH en modo túnel.

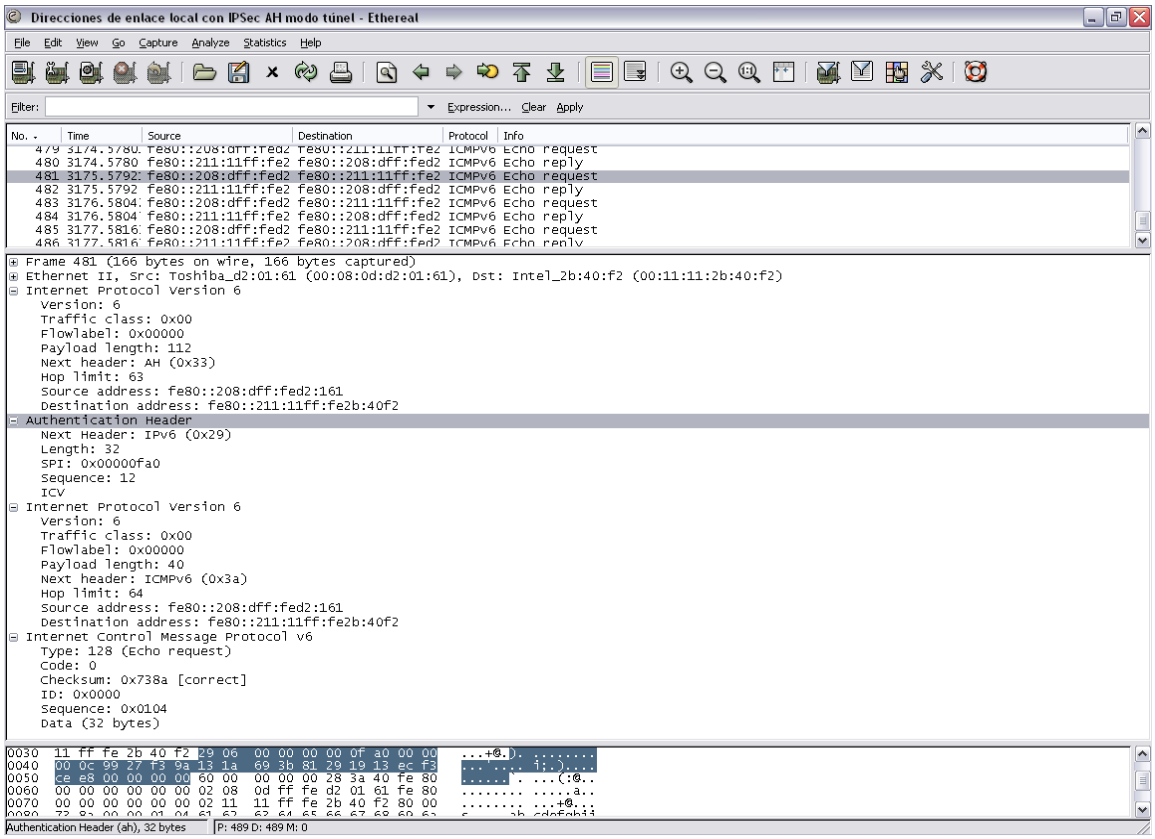


Figura V.5 Análisis de tráfico para la prueba 2 usando direcciones de enlace local entre dos PC utilizando el encabezado AH de IPSec en modo túnel.

Para estas pruebas los resultados fueron los siguientes

- La comunicación entre las máquinas con Windows XP y SP1 fueron correctas.
- La comunicación entre la máquina con Windows XP y SP2 no fueron correctas debido a que el Firewall de Windows bloqueaba los paquetes y se tenía que apagar para tener conectividad. Este dispositivo no se pudo configurar para IPv6 debido a que no lo soporta.
- La comunicación entre la máquina con Windows 2003 sin ningún SP instalado fue correcta, sin embargo, con el SP1 instalado no se logró la conectividad debido al Firewall que viene integrado junto con el SP1, y se

tenía que apagar para tener una correcta comunicación de la máquina con Windows XP a la máquina con Windows 2003, igual que el caso anterior.

- Utilizando las direcciones de enlace local no se pudo acceder con el servidor Web Apache y el servicio telnet debido a que no soportan este tipo de direcciones.

3. Pruebas entre dos computadoras con sistema operativo Windows en diferente segmento.

Para esta tercera prueba se hizo básicamente lo mismo que la prueba anterior, con la diferencia que las computadoras se encontraron en distinto segmento, y por consiguiente sólo se utilizaron sus direcciones manuales y auto-configuradas siguiendo el esquema que se muestra en la figura V.6

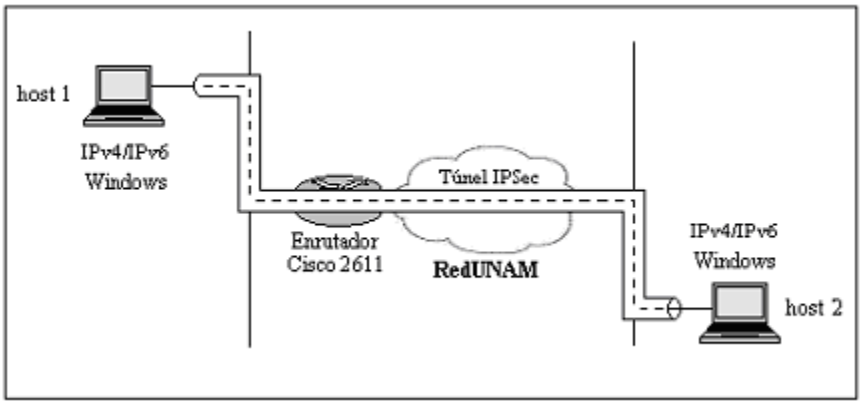


Figura V.6 Esquema de configuración para la prueba 3 entre dos PC en diferente segmento.

En la figura V.7 se observa el tráfico analizado por Ethereal cuando se configuró el encabezado AH en modo transporte utilizando direcciones auto-configuradas y en la figura V.8 cuando se configuro el encabezado ESP en modo transporte. Cabe señalar que para ESP los paquetes se ven de la misma manera para modo transporte y modo túnel debido a que se cifra la trama, teniendo como referencia la carga útil para diferenciarlos (el modo túnel tendrá mayor carga útil o payload).

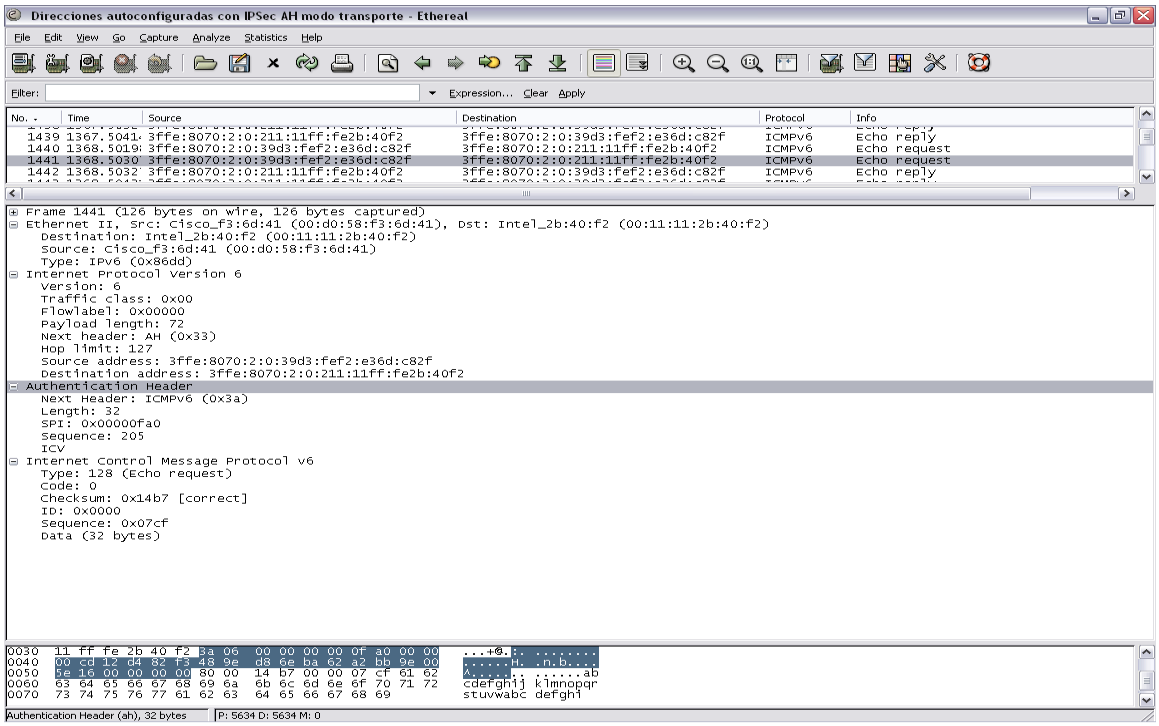


Figura V.7 Análisis de tráfico para la prueba 3 usando direcciones auto-configuradas entre dos PC utilizando el encabezado AH de IPSec en modo transporte.

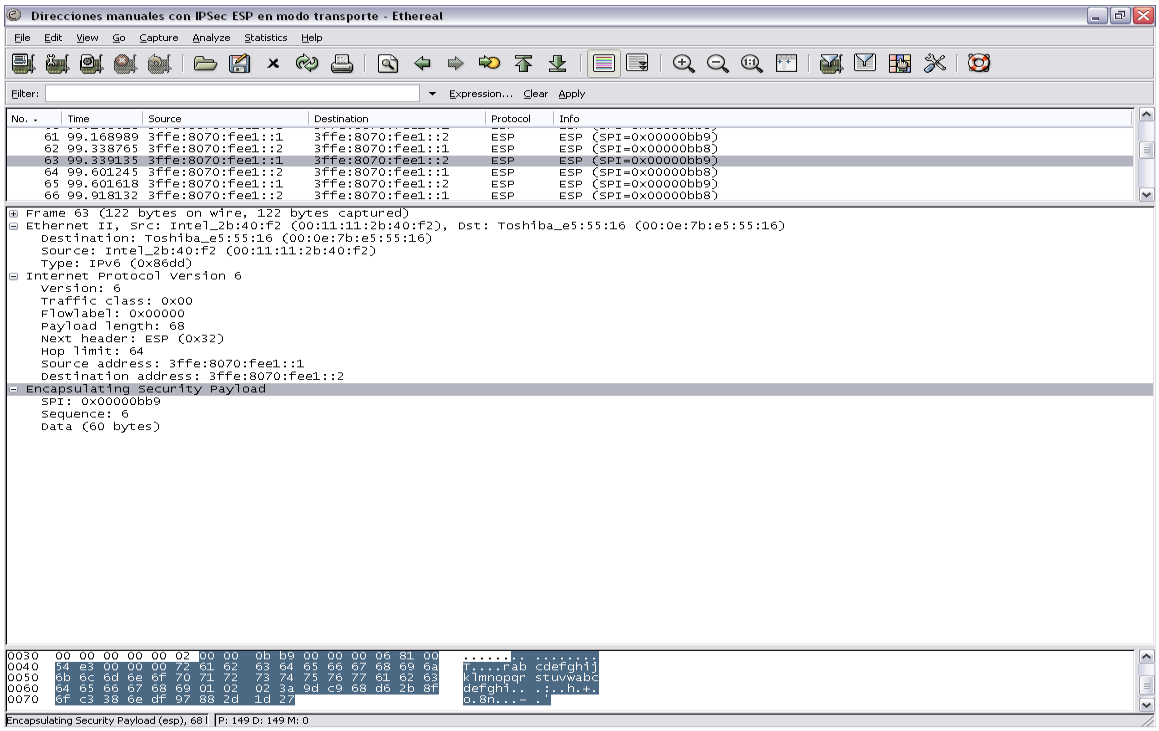


Figura V.8 Análisis de tráfico para la prueba 3 usando direcciones manuales entre dos PC utilizando el encabezado ESP de IPSec en modo transporte

Para estas pruebas los resultados fueron los siguientes

- La comunicación entre las máquinas con Windows XP y SP1 fueron correctas.
- La comunicación entre la máquina con Windows XP y SP2 no fueron correctas debido a que el Firewall de Windows bloqueaba los paquetes y se tenia que apagar para tener conectividad. Este dispositivo no se pudo configurar para IPv6 debido a que no lo soporta.
- La comunicación entre la máquina con Windows 2003 sin ningún SP instalado fue correcta, sin embargo con el SP1 instalado no se logro la conectividad debido al Firewall que viene integrado junto con el SP1 y se tenia que apagar para tener una correcta comunicación de la máquina con Windows XP a la máquina con Windows 2003, igual que el caso anterior.

4. Pruebas entre una computadora con sistema operativo Windows y una con distribución Linux en el mismo segmento.

En la cuarta prueba se utilizó la distribución de Linux Fedora Core 4 y Windows XP SP1 en un mismo segmento. En la figura V.9 se tiene el esquema que implementó.

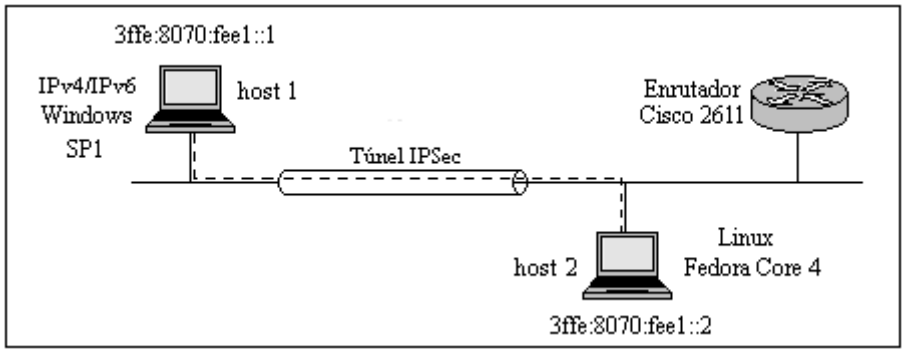


Figura V.9 Esquema de configuración para la prueba 4 entre dos PC en el mismo segmento

Los detalles de la configuración en la creación del túnel utilizando IPv6 sin IPSec se muestran a continuación:

En la PC con Fedora Core 4

[netlab@voip ~]$ /sbin/ifconfig –a (se muestran las interfaces)

```
eth0      Link encap:Ethernet  HWaddr 00:13:20:4C:33:10
          inet addr:1.1.1.2  Bcast:1.1.1.255  Mask:255.255.255.224
          inet6 addr: 3ffe:8070:1:6:213:20ff:fe4c:3310/64 Scope:Global
          inet6 addr: fe80::213:20ff:fe4c:3310/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:7060 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4100 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:989312 (966.1 KiB)  TX bytes:420441 (410.5 KiB)
          Interrupt:177 Memory:ff720000-0

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1227 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1227 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2029592 (1.9 MiB)  TX bytes:2029592 (1.9 MiB)

sit0      Link encap:IPv6-in-IPv4
          NOARP  MTU:1480  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

[root@voip ~]# ping6 3ffe:8070:1:6:213:20ff:fe74:123c (comprobación de conectividad por PING)

```
PING 3ffe:8070:1:6:213:20ff:fe74:123c(3ffe:8070:1:6:213:20ff:fe74:123c) 56 data bytes
64 bytes from 3ffe:8070:1:6:213:20ff:fe74:123c: icmp_seq=0 ttl=63 time=2.70 ms
64 bytes from 3ffe:8070:1:6:213:20ff:fe74:123c: icmp_seq=1 ttl=63 time=2.62 ms
64 bytes from 3ffe:8070:1:6:213:20ff:fe74:123c: icmp_seq=2 ttl=63 time=2.52 ms
--- 3ffe:8070:1:6:213:20ff:fe74:123c ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2017ms
rtt min/avg/max/mdev = 2.525/2.619/2.704/0.084 ms, pipe 2
```

En la PC con Windows XP SP1

c:\>ping 3ffe:8070:1:6:213:20ff:fe4c:3310 (comprobación de conectividad por PING)

Haciendo ping a 3ffe:8070:1:6:213:20ff:fe4c:3310 con 32 bytes de datos:

Respuesta desde 3ffe:8070:1:6:213:20ff:fe4c:3310: tiempo=1ms
Respuesta desde 3ffe:8070:1:6:213:20ff:fe4c:3310: tiempo<1m
Respuesta desde 3ffe:8070:1:6:213:20ff:fe4c:3310: tiempo<1m
Respuesta desde 3ffe:8070:1:6:213:20ff:fe4c:3310: tiempo<1m

Capturando tráfico con la herramienta de tcpdump de Fedora

[root@voip ~]# tcpdump -t -n -i eth0 -s 512 -vv ip6 or proto ipv6
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 512 bytes
fe80::213:20ff:fe74:123c > fe80::213:20ff:fe4c:3310: [icmp6 sum ok] icmp6: echo request seq 19 (len 40, hlim 128)
fe80::213:20ff:fe4c:3310 > fe80::213:20ff:fe74:123c: [icmp6 sum ok] icmp6: echo reply seq 19 (len 40, hlim 64)
fe80::213:20ff:fe74:123c > fe80::213:20ff:fe4c:3310: [icmp6 sum ok] icmp6: neighbor sol: who has
3ffe:8070:1:6:99d3:407:2442:eb70 > 3ffe:8070:1:6:213:20ff:fe4c:3310: [icmp6 sum ok] icmp6: echo request seq 21 (len 40, hlim 127)
3ffe:8070:1:6:213:20ff:fe4c:3310 > 3ffe:8070:1:6:99d3:407:2442:eb70: [icmp6 sum ok] icmp6: echo reply seq 21 (len 40, hlim 64)
…

En la figura V.10 se observa el tráfico con el analizador Ethereal y con tráfico HTTP cuando se crea un túnel sobre IPv6 sin IPSec.

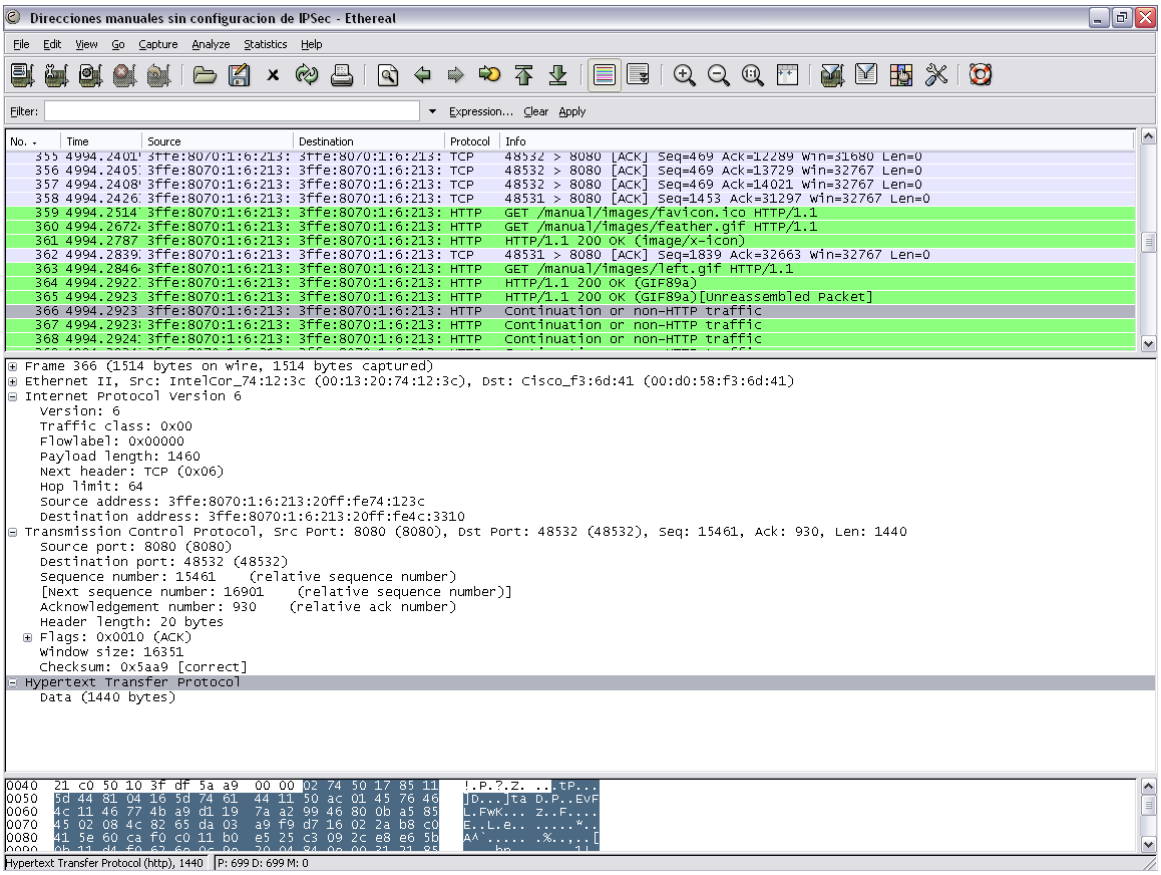


Figura V.10 Análisis de tráfico para la prueba 4 entre dos PC con tráfico HTTP

Para la implementación de IPSec primero se utilizó el software FreeS/WAN y después se implementó USAGI, donde no se pudo tener conectividad entre estos distintos sistemas operativos.

5. Pruebas entre una computadora con sistema operativo Windows y una con distribución BSD en el mismo segmento.

En esta última prueba se utilizó el sistema FreeBSD v6.1 como se muestra en la figura V.11; sin embargo, debido a complicaciones con hardware, software y la misma distribución solo se comprobó una correcta conectividad entre una PC con Windows y otra con BSD utilizando IPv6.

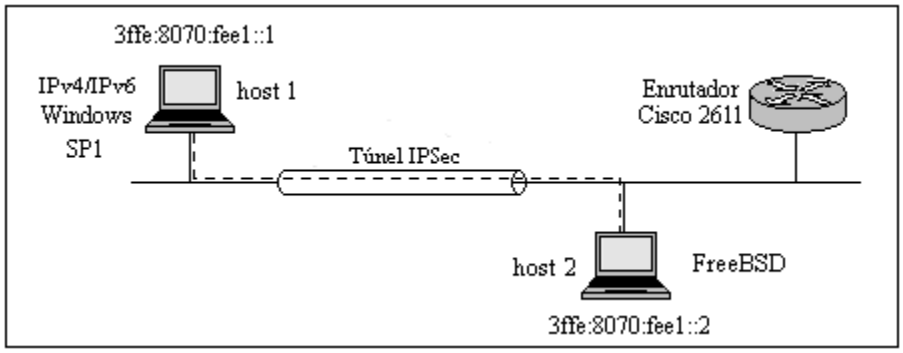


Figura V.11 Esquema de configuración para la prueba 5 entre dos PC en el mismo segmento

Los detalles de la configuración en la creación del túnel utilizando IPv6 sin IPSec se muestran a continuación:

En la PC con FreeBSD v6.1

[netlab@ /usr/home/netlab]$ ifconfig –a

```
myk0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        options=2b<RXCSUM,TXCSUM,VLAN_MTU,JUMBO_MTU>
        inet6 fe80::213:20ff:fe74:123c%myk0 prefixlen 64 scopeid 0x1
        inet 1.1.1.2 netmask 0xfffffffe0 broadcast 1.1.1.255
        ether 00:13:20:74:12:3c
        media: Ethernet autoselect (10baseT/UTP <half-duplex>)
        status: active
plip0: flags=108810<POINTOPOINT,SIMPLEX,MULTICAST,NEEDSGIANT> mtu 1500
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x3
        inet 127.0.0.1 netmask 0xff000000
```

Para habilitar de IPv6 e IPSec se edita el archivo rc.conf

[root@ /etc/]# **pico rc.conf (abrir el archive rc.conf con el editor de texto "pico")**

```
# -- sysinstall generated deltas -- # Fri May 26 07:17:54 2006
# Created: Fri May 26 07:17:54 2006
# Enable network daemons for user convenience.
# Please makes all changes to this file, not to /etc/defaults/rc.conf.
# This file now contains just the overrides from /etc/defaults/rc.conf.

#Valores habilitados por la instalación de FreeBSD
inetd_enable="YES"
keymap="spanish.iso.acc"
linux_enable="YES"
moused_enable="YES"
saver="fire"
sshd_enable="YES"
usbd_enable="YES"
```

#Para habilitar auto configuración de IPv6
ipv6_enable="YES"

```
#Configuración de Red
defaultrouter=1.1.1.3
ifconfig_myk0="inet 1.1.1.2 netmask 255.255.255.224"
network_interfaces="gif0"

# interfaces que usaran ipv6
ipv6_network_interfaces="gif0"

# interface del túnel
gif_interfaces="gif0"

#Se crea el túnel con las direcciones IPv4
gifconfig_gif0="1.1.1.2 1.1.1.1"

#Dirección ipv6 de nuestro lado del túnel
/sbin/ifconfig gif0 inet6 2001:db8::2 prefixlen 64

#la ruta de ipv6 por defecto es
/sbin/route add -inet6 default 2001:db8::1
```

#Para habilitar soporte de IPSec
ipsec_enable="YES"
ipsec_file="/etc/ipsec.conf"

En el archivo ipsec.conf

```
# basic configuration
config setup
        interfaces="ipsec0=eth0"
        klipsdebug=none
        plutodebug=none
        plutoload=%search
        plutostart=%search

# sample connection
conn deep-mail
        left=208.164.186.1
        leftsubnet=192.168.1.0/24
        leftnexthop=205.151.222.250
        right=208.164.186.2
        rightsubnet=192.168.1.0/24
        rightnexthop=205.151.222.251
        keyingtries=0
        auth=ah
        auto=start

spdadd 1.1.1.1/32 1.1.1.2/32 ipencap -P out ipsec esp/tunnel/1.1.1.1-1.1.1.2/require;
spdadd 1.1.1.1/32 1.1.1.2/32 ipencap -P in ipsec esp/tunnel/1.1.1.1-1.1.1.2/require;
```

Para permitir el paso de paquetes se añaden reglas en el firewall

```
ipfw add 1 allow esp from 1.1.1.1 to 1.1.1.2
ipfw add 1 allow esp from 1.1.1.2 to 1.1.1.1
ipfw add 1 allow ipencap from 1.1.1.1 to 1.1.1.2
ipfw add 1 allow ipencap from 1.1.1.2 to 1.1.1.1
```

Verificación de conectividad sobre IPv6

[root@voip ~]# ping6 3ffe:8070:1:6:213:20ff:fe74:123c (comprobación de conectividad por PING)

```
PING 3ffe:8070:1:6:213:20ff:fe74:123c(3ffe:8070:1:6:213:20ff:fe74:123c) 56 data bytes
64 bytes from 3ffe:8070:1:6:213:20ff:fe74:123c: icmp_seq=0 ttl=63 time=2.70 ms
64 bytes from 3ffe:8070:1:6:213:20ff:fe74:123c: icmp_seq=1 ttl=63 time=2.62 ms
64 bytes from 3ffe:8070:1:6:213:20ff:fe74:123c: icmp_seq=2 ttl=63 time=2.52 ms
--- 3ffe:8070:1:6:213:20ff:fe74:123c ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2017ms
rtt min/avg/max/mdev = 2.525/2.619/2.704/0.084 ms, pipe 2
```

El soporte de algoritmos de cifrado y autenticación para IPSec para la distribución de FreeBSD se muestran en la tabla V.9, donde se puede observar que por ser de código abierto cuentan con un mayor soporte.

Autenticación		Cifrado	
Algoritmo	**Num. de bits**	**Algoritmo**	**Num. de bits**
hmac-md5	128		
hmac-sha1	160	des-cbc	64
keyed-md5	128	3des-cbc	192
keyed-sha1	160	null	0 a 2048
null	0 a 2048	blowfish-cbc	40 a 448
hmac-sha2-256	256	cast128-cbc	40 a128
hmac-sha2-384	384	des-deriv	64
hmac-sha2-512	512	3des-deriv	192
hmac-ripemd160	160	rijndael-cbc	128/192/256
aes-xcbc-mac	128	aes-ctr	160/224/288
tcp-md5	8 a 640		

Tabla V.9 Algoritmos soportados para IPSec FreeBSD

V.3 Resultados obtenidos

Si bien los resultados que se obtuvieron no fueron del todo satisfactorios en todos los sistemas operativos, debido por ejemplo a que sobre la plataforma Windows XP con el SP2 y Windows 2003 con SP1 no se tuvo conectividad entre los equipos a causa del Firewall que se tiene instalado por defecto, no pudiéndose configurar el mismo para permitir el tráfico de IPv6 sin que lo dejara de filtrar. Además, como la versión de Internet Explorer 6.1 instalada no tiene soporte para direcciones literales IPv6 se instaló el navegador Netscape v8.1. Hoy en día ya está disponible esta funcionalidad para la versión 7.0 del IE.

En relación a la conectividad de Windows XP con SP1 no hubo ningún problema al implementar IPSec, así como con el Windows 2003 sin ningún SP.
Además se comprobó que para los servicios de Web (HTTP) y Telnet no se puede acceder a ellos utilizando direcciones de enlace local sino solamente direcciones configuradas manualmente o auto-configuradas.

Por otra parte, para las plataformas de Linux y BSD no se obtuvieron los resultados esperados ya que tanto la instalación y configuración de estos sistemas operativos como el software/kernel con soporte IPSec se llevo mucho tiempo por la poca documentación que existe y/o por problemas de compatibilidad, lo que implicaba estar probando distintos kernels para adecuarlos a las necesidades. Además influyeron otros factores como poder habilitar la tarjeta de red instalada para que funcionara en estas plataformas. Sin embargo, se comprobó que el soporte de IPSec es más completo en estas plataformas con más algoritmos de autenticación y cifrado.

Por lo tanto, se puede decir que en general, en los sistemas operativos probados se tiene un soporte aún no lo suficientemente maduro para IPSec e IPv6, debido a que para las distintas distribuciones no se tienen las características completas de ambos protocolos, que se espera pronto se solucione en las nuevas versiones, lo que implicó que existieran limitaciones para la comunicación al implementar ambos protocolos; sin embargo, bajo estas limitantes se pueden tener escenarios donde resulten eficientes, tal como lo muestran algunos resultados de la tabla V.10, en donde se presenta un resumen de las pruebas realizadas con los sistemas operativos Windows con distintos parches o paquetes instalados adicionalmente, verificando cómo es el tráfico habilitados los servicios de Telnet y Apache como servidor Web.

Equipo o Sistema Operativo empleado		Service Pack		Parámetros		Tipo de dirección IPv6		Uso del Servidor Web Apache	habilitación de IPSec				Conectividad (resultados obtenidos)		
						Local	Global		m. túnel		m. transporte				
PC1	PC2	PC1	PC2	sentido	segmento	Enlace	M / A Man/Auto	Versión utilizada	ESP	AH	ESP	AH	ping ping6	Servidor Apache	Telnet
Win XP	Switch	SP1+	-	PC1 ↔ PC2	mismo	Enlace	M	-	No	No	No	No	Si	-	-
Win XP	Switch	SP1+	-	PC1 ↔ PC2	mismo	Enlace	M	-	Si	No	No	Si	No	-	-
Win XP	Switch	SP1+	-	PC1 ↔ PC2	diferente		M	-	Si	No	No	Si	No	-	-
Win XP	Win XP	SP1+	SP1	PC1 ↔ PC2	mismo	Enlace	M y A	-	No	No	No	No	Si	-	-
Win XP	Win XP	SP1+	SP1	PC1 ↔ PC2	mismo	Enlace	M y A	-	Si	Si	Si	Si	Si	-	-
Win XP	Win XP	SP1+	SP2	PC1 ↔ PC2	mismo	Enlace		PC1 v1.0.37 PC2 v2.054	No	No	No	No	Si	No	No
Win XP	Win XP	SP1+	SP2	PC1 ↔ PC2	mismo	Enlace		PC1 v1.0.37 PC2 v2.054	Si	Si	Si	Si	No	No	No
Win XP	Win XP	SP1+	SP2	PC1 ↔ PC2	mismo		M y A	PC1 v1.0.37 PC2 v2.054	No	No	No	No	Si	Si	Si
Win XP	Win XP	SP1+	SP2	PC1 ↔ PC2	mismo		M y A	PC1 v1.0.37 PC2 v2.054	Si	Si	Si	Si	No	No	No
Win XP	Win XP	SP1	SP1	PC1 ↔ PC2	mismo	Enlace		v2.0.54	No	No	No	No	Si	No	No
Win XP	Win XP	SP1	SP1	PC1 ↔ PC2	mismo	Enlace		v2.0.54	Si	Si	Si	Si	Si	No	No
Win XP	Win XP	SP1	SP1	PC1 ↔ PC2	mismo		M y A	v2.0.54	No	No	No	No	Si	Si	Si
Win XP	Win XP	SP1	SP1	PC1 ↔ PC2	mismo		M y A	v2.0.54	Si	Si	Si	Si	Si	Si	Si
Win XP	Win XP	SP1	SP1	PC1 ↔ PC2	diferente		M	v2.0.54	No	No	No	No	Si	Si	Si
Win XP	Win XP	SP1	SP1	PC1 ↔ PC2	diferente		M	v2.0.54	Si	Si	Si	Si	Si	Si	Si
Win XP	Win XP	SP2	SP2	PC1 ↔ PC2	mismo	Enlace		v2.0.54	No	No	No	No	Si	No	No
Win XP	Win XP	SP2	SP2	PC1 ↔ PC2	mismo		M	v2.0.54	No	No	No	No	Si	Si	Si
Win XP	Win XP	SP2	SP2	PC1 ↔ PC2	mismo	Enlace	M	v2.0.54	Si	Si	Si	Si	No	No	No
Win XP	Win XP	SP2	SP2	PC1 ↔ PC2	diferente		M	v2.0.54	No	No	No	No	Si	Si	Si
Win XP	Win XP	SP2	SP2	PC1 ↔ PC2	diferente		M	v2.0.54	Si	Si	Si	Si	No	No	No
Win XP	Win 2003	SP1	No	PC1 ↔ PC2	mismo	Enlace		PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	Si	No	No
Win XP	Win 2003	SP1	No	PC1 ↔ PC2	mismo	Enlace		PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	Si	No	No
Win XP	Win 2003	SP1	No	PC1 ↔ PC2	mismo		M y A	PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	Si	Si	Si
Win XP	Win 2003	SP1	No	PC1 ↔ PC2	mismo		M y A	PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	Si	Si	Si
Win XP	Win 2003	SP1	No	PC1 ↔ PC2	diferente		M	PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	Si	Si	Si
Win XP	Win 2003	SP1	No	PC1 ↔ PC2	diferente		M	PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	Si	Si	Si
Win XP	Win 2003	SP1	SP1	PC1 → PC2	mismo	Enlace		PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	No	No	No
Win XP	Win 2003	SP1	SP1	PC1 ← PC2	mismo	Enlace		PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	Si	No	No
Win XP	Win 2003	SP1	SP1	PC1 ↔ PC2	mismo	Enlace		PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	Si	No	No
Win XP	Win 2003	SP1	SP1	PC1 → PC2	mismo		M y A	PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	No	No	No
Win XP	Win 2003	SP1	SP1	PC1 → PC2	mismo		M y A	PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	No	No	No
Win XP	Win 2003	SP1	SP1	PC1 ← PC2	mismo		M y A	PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	Si	Si	Si
Win XP	Win 2003	SP1	SP1	PC1 ← PC2	mismo		M y A	PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	Si	Si	Si
Win XP	Win 2003	SP1	SP1	PC1 → PC2	diferente		M	PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	No	No	No
Win XP	Win 2003	SP1	SP1	PC1 → PC2	diferente		M	PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	No	No	No
Win XP	Win 2003	SP1	SP1	PC1 ← PC2	diferente		M	PC1 v2.0.54 PC2 v1.3.27	No	No	No	No	Si	Si	Si
Win XP	Win 2003	SP1	SP1	PC1 ← PC2	diferente		M	PC1 v2.0.54 PC2 v1.3.27	Si	Si	Si	Si	Si	Si	Si

Tabla V.10 Resultados de pruebas de IPSec con IPv6 en Windows

Conclusiones

Con la finalidad de entender las aplicaciones y funcionamiento de los diferentes protocolos de seguridad existentes, examinándose a detalle el IPSec, se estudiaron los modelos de comunicación OSI y TCP/IP, así como los algoritmos criptográficos y los servicios de seguridad que ofrecen; observando que se tienen diferentes beneficios y/o perjuicios dependiendo del nivel donde estén implementados.

Dado que el objetivo principal de este trabajo fue realizar pruebas y un análisis del soporte de IPSec con IPv6, primeramente se hizo un estudio de la versión anterior de IP, IPv4, con el propósito de dar a conocer los elementos necesarios para poder entender el porqué de su evolución, siendo la principal causa el agotamiento de direcciones ante las necesidades tecnológicas actuales. Con IPv6 se tiene un direccionamiento prácticamente infinito para conectar cualquier dispositivo a la red, además, su configuración resulta ser más sencilla y amigable para los usuarios; se simplifica el encabezado IP original permitiendo un procesamiento más rápido y eficiente en los enrutadores, utilizándose encabezados de extensión que ayudan a la flexibilidad del protocolo; se evita el uso de *broadcast;* se realizan mejoras en las comunicaciones inalámbricas; se emplea IPSec como protocolo de seguridad definido en el núcleo de esta nueva versión, entre otras características.

Así mismo, se describieron varios mecanismos de transición existentes para llevar a cabo una convivencia y una migración progresiva de IPv4 a IPv6, en donde IPv6 se irá imponiendo poco a poco y es solo cuestión de tiempo para que la versión actual de IP sea prácticamente reemplazada por la nueva ante la necesidad incuestionable, que se va haciendo más urgente, conforme el rango de direcciones IP actuales se vaya consumiendo.

Esta transición puede resultar no trivial, sobre todo porque para muchas de las empresas representa una gran inversión de dinero en cuanto a hardware y software, y adicionalmente puede representar un riesgo en las aplicaciones que pudieran ser o no compatibles con ésta nueva versión de IP.
Además, se tendrían que planificar los cambios en los nombres asociados y direcciones de red en los distintos dispositivos, revisar el diseño de las redes (enrutadores con doble pila, Firewalls, NATs, DNS, etc.), plantear la posibilidad de que convivan sistemas con IPv4 e IPv6, contar con un plan de contingencia en caso de falla por un mal soporte de IPv6, entre otros.

El uso de IPSec es fundamental para una seguridad completa en IPv6 y, aunque no es el único mecanismo que proporciona seguridad en esta versión de IP, es indispensable para conexiones punto a punto donde se requiera una seguridad sólida para los que interactúan en una comunicación sin importar el tipo de tráfico que se transmita. Este protocolo, como los otros estudiados, no ofrece una medida única para combatir las diferentes vulnerabilidades existentes, sino que es parte de la solución para brindar seguridad en una red, pudiéndose utilizar, por ejemplo, junto con SSH para establecer conexiones seguras a nivel de usuario.

IPSec no detiene todos los ataques del tipo DoS, aunque sí representa una solución para ataques de tipo spoofing. IPSec cifra y autentica paquetes IP, como se pudo constatar, contra el análisis de tráfico para proteger el contenido del paquete, aunque los encabezados quedan visibles pudiendo representar información útil para los intrusos.

Por otra parte, el manejo de una VPN no representa una solución completa y no brinda protección total a una red, ya que protege únicamente el canal por donde transita la información de un extremo a otro de la VPN, donde si uno de los extremos se compromete, se perdió la protección.
Aunque en un principio la única tecnología en VPN disponible fue IPSec VPN, más adelante surgió SSL VPN como una capacidad en los navegadores. La principal

diferencia entre un tipo y el otro, radica en que IPSec realiza la autenticación a través de certificados y cuenta con diferentes niveles de cifrado para una mayor seguridad, aunque se vuelve más compleja su administración, además de implicar un mayor costo. SSL, por el contrario, no tiene un cliente por software haciendo su administración más sencilla, sin embargo, se tiene una sola opción de cifrado. Así una aplicación que requiera usar SSL puede ser redistribuida con su propio código SSL sin impactar a otras aplicaciones, aspecto que no sucede con IPSec.

Además hay que considerar que si se hace uso de SSL los usuarios tengan acceso a los navegadores, de lo contrario este protocolo no es posible, aunque se tiene la posibilidad de correr los dos tipos de VPN en una misma red.
En cuestiones de cifrado SSL utiliza DES, mientras que IPSec usa 3DES que será sustituido por AES por trabajos elaborados recientemente que, si bien, para la mayoría de las aplicaciones DES es adecuado, para requerimientos donde se quiera tener una seguridad mayor, como por ejemplo en asuntos militares, 3DES/AES sea una mejor opción.

Como parte del análisis del protocolo IPSec, se desarrollaron escenarios y pruebas entre diversos dispositivos y plataformas aplicando las configuraciones que soportaban IPSec. Las pruebas y resultados se mostraron en el Capítulo V, donde se puede ver que para la implementación en sistemas Unix no se obtuvieron los resultados esperados por cuestiones de instalación y compatibilidad con el kernel para soporte de IPSec en IPv6, aunado con la poca documentación existente; sin embargo, se comprobó que su soporte es más completo en comparación con Windows.
Para la configuración en los sistemas Unix se buscaron paquetes adicionales que permitieran un mayor soporte para trabajar con IPSec como son los proyectos USAGI o FreeS/WAN, que aunque representan una muy buena opción, no se cuenta con documentación suficiente y resultan incompatibles.
En relación a las pruebas que se realizaron con el sistema Windows, en sus versiones XP y 2003, se observó que el Firewall propio de Windows (integrado en

algunas versiones de Service Pack) no presentaba opciones de configuración para soporte de IPSec e IPv6, provocando que al estar habilitado éste último, se tuviera una pérdida de paquetes en la comunicación que se tenía con IPSec. Además, se probaron de manera exitosa conexiones con tráfico real, implementando el servicio de telnet y Apache como servidor Web, comprobando que el funcionamiento de IPSec fuera correcto con el uso de analizadores de red, donde se pudieron observar los encabezados de este protocolo al configurarlo en sus diferentes modos.

Adicionalmente se comprobó que para Internet Explorer versión 6 no se tiene soporte para direcciones literales (alfa-numéricas) IPv6 en la URL del navegador, aspecto que ya es contemplado en su más reciente versión.

En cuanto a las líneas futuras de investigación, como el protocolo IPSec está en etapa de exploración y comprensión para muchos, constantemente se encuentra en proceso de re-evaluación emitiendo propuestas a nivel de *drafts*. Hay mucho trabajo por hacer en varios aspectos de IPSec como son: el análisis de vulnerabilidades en su arquitectura y operación; realizar estudios y pruebas sobre la administración dinámica de llaves con IKE e ISAKMP, investigar y experimentar su implementación con una arquitectura de llave pública (PKI) para establecer y mantener un entorno de red seguro, a través de la generación y distribución de llaves y certificados digitales, para su uso en aplicaciones de firma digital; así como la integración de nuevos protocolos en las redes de siguiente generación como lo es MPLS. Finalmente, hay que considerar que se debe de contar con un mayor soporte de hardware y software, en relación con el costo, con el propósito de que sea rentable para cualquier negocio.

Este trabajo de tesis representa para mí un fuerte crecimiento profesional y personal al abordar uno de los temas más importantes, como lo es la seguridad en las redes, donde la principal aportación es dar a conocer un panorama teórico – práctico sobre el uso y características de IPSec en IPv6 para que sirva en líneas futuras de investigación con el fin de implementarse de forma confiable y transparente al usuario final.

Glosario de Términos

3DES - Triple DES: Algoritmo de cifrado, *véase pág. 111.*

6bone: Fue el backbone de IPv6, cuya función fue asistir en la evolución y desarrollo de IPv6. Su creación se formalizó en marzo de 1996 en una reunión del IETF en Los Ángeles formando una red experimental, informal y cooperativa de alcance mundial (http://www.6bone.net).

6over4: Tipo de túnel automático, *véase pág. 80.*

6to4: Tipo de túnel automático, *véase pág. 76.*

Abstracción: Consiste en aislar un elemento de su contexto o del resto de los elementos que lo acompañan. En programación, el término se refiere al énfasis en el "¿qué hace?" más que en el "¿cómo lo hace?".

AES (Advanced **E**ncryption **S**tandard / Cifrado avanzado estándar**):** Algoritmo de cifrado, *véase pág. 111.*

AH (Authentication **H**eader / Encabezado de autenticación**):** Encabezado de IPSec, *véase capítulo 3.*

ALG (Application **L**evel **G**ateway / Traducción a nivel aplicación**):** Tipo de traductor, *véase pág. 87.*

ANP (Advanced **N**etworking **P**ack / Paquete de red avanzado**):** Complemento para el Service Pack 1 de Windows XP.

Anycast: Envío de información al "mejor" destino, sólo un receptor, desde el punto de vista de la topología de red.

Apache: Servidor de páginas Web de código abierto para diversas plataformas desarrollado por la Apache Software Foundation.

API (Application **P**rogramming **I**nterface / Interfaz del programa de aplicación**):** Conjunto de llamadas a determinadas bibliotecas que ofrecen acceso a ciertos servicios desde los procesos y representa un método para conseguir abstracción en la programación.

ASCII (American Standard Code for Information Interchange / Código Estadounidense Estándar para el Intercambio de Información**):** Código de caracteres utilizado por computadoras para representar todas las letras.

ATM (Asynchronous Transfer Mode / Modo de Transferencia Asíncrona**)**: Tecnología de telecomunicación para una comunicación a altas velocidades, donde la información no se transmite y conmuta a través de canales asignados en permanencia, sino en forma de cortos paquetes (celdas ATM) de longitud constante, pudiendo ser enrutados individualmente.

Base de datos: Conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su uso posterior.

BIA (Bump **I**n the **A**PI / Ataque en el API**):** Tipo de traductor, *véase pág. 86.*

BIND (Berkeley **I**nternet **N**ame **D**omain / Nombre de dominio de Internet Berkeley**):** Servidor de DNS más comúnmente usado en Internet creado por Paul Vixie en 1988 a partir de un proyecto en la Universidad de Berkeley y cuenta con el apoyo de la Internet Systems Consortium.

BIS (Bump **In** the **S**tack / Ataque en la pila**):** Tipo de traductor, *véase pág. 85.*

Blowfish: Algoritmo de cifrado, *véase pág. 113.*

Broadcast: Envío de información de un emisor hacia todos los posibles receptores, todas las estaciones en la red, de manera simultanea.

BSD (Berkeley Software Distribution / Distribución de software Berkeley**):** Iniciales para identificar un sistema operativo derivado del sistema Unix nacido a partir de las aportaciones realizadas a ese sistema por la Universidad de California en Berkeley.

CAST (por sus creadores **C**arlisle **A**dams y **S**tafford **T**avares**):** Algoritmo de cifrado, *véase pág. 113.*

CERT (Computer **E**mergency **R**esponse **T**eam / Equipo de Respuesta a Emergencias Informáticas**):** Creado por DARPA (Defense Advanced Research Projects Agency) en 1988 en respuesta a las necesidades requeridas durante un famoso incidente conocido como el "Gusano de Internet" que infectó a más de 600 computadoras en una red. El CERT trabaja para facilitar las respuestas a incidentes de seguridad que afectan a Internet con el objetivo de tomar las medidas oportunas de prevención, además de investigar y mejorar la seguridad de los sistemas que existen.

CHAP (Challenge **H**andshake **A**uthentication **P**rotocol / Protocolo de autenticación por desafío mutuo**):** Protocolo de seguridad, *véase pág. 33.*

CIDR (Classless **I**nter**-D**omain **R**outing / Enrutamiento inter-dominios sin clases): Es un estándar de red para la interpretación de direcciones IP, donde no importa la clase sino el prefijo utilizado facilitando el enrutamiento.
CoS (Class **o**f **S**ervice / Clase de servicio**):** Se refiere a la diferenciación del tráfico, es decir, la habilidad de tratar los paquetes de forma diferente basados en la importancia del paquete.

CRC (Cyclic **R**edundancy **C**hecking / Comprobación de redundancia cíclica**):** Comprobación que se suele añadir a los datos transmitidos en una comunicación permitiendo detectar si se ha producido algún error en la transmisión.

Criptoanálisis: Ciencia que trata de encontrar la información mediante un análisis de manera ilegítima o no autorizada.

Criptografía: Arte o ciencia de cifrar y descifrar información utilizando técnicas matemáticas que hagan posible el intercambio de mensajes

Datagrama: Paquete de información que se envía de forma "no orientada a conexión" y "no confiable".

DES (Data **E**ncryption **S**tandard **/** Cifrado de datos estándar**):** Algoritmo de cifrado, *véase pág. 106.*

DHCP (Dynamic **H**ost **C**onfiguration **P**rotocol / Protocolo de configuración de host dinámico**):** Software que asigna automáticamente las direcciones IP para las estaciones cliente que conecta a la red TCP/IP.

Diffie-Hellman: Protocolo de intercambio de llaves, *véase pág. 136.*

DNS (Domain **N**ame **S**ystem / Sistema de nombre de dominios**):** Conjunto de protocolos y servicios (base de datos distribuida) que permite a los usuarios utilizar nombres en vez de tener que recordar direcciones IP numéricas.

Doble pila: Se refiere a tener implementado tanto IPv4 como IPv6.

DoS (Denial **o**f **S**ervice / Denegación de servicio**):** Se refiere a un tipo de ataque que pretenden bloquear uno o varios servicios.

Draft (Borrador): Documento de especificaciones que se expone públicamente para su discusión.

DSA (Digital **S**ignature **A**lgorithm / Algoritmo de firma digital**):** Algoritmo propuesto por el Instituto Nacional de Normas y Tecnología para firmas digitales

DSL (Digital **S**ubscriber **L**ine / Línea de abonado digital**):** Término utilizado para referirse de forma global a todas las tecnologías que proveen una conexión digital sobre línea de abonado de la red telefónica local: ADSL, SDSL, HDSL y VDSL.

EAP (Extensible **A**uthentication **P**rotocol / Protocolo de autenticación extensible**):** Protocolo de seguridad, *véase pág. 33.*

Ehtereal: Analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones.

ESP (Encapsulating **S**ecurity **P**ayload / Encabezado de Carga de Seguridad de Encapsulación**):** Encabezado de cifrado para IPSec, *véase capítulo 3.*

Esteganografía: Rama de la criptografía que trata sobre la ocultación de mensajes en lugar de su contenido para evitar que se perciba la existencia de los mismos.

Extranet: Es el resultado de ampliar la Intranet de una organización para que ésta incluya la red de uno o más socios, por ejemplo comerciales.

Firewall (cortafuegos): Sistema de seguridad encargado de proteger una red contra accesos no autorizados

Firma digital: Código digital que se puede adjuntar a un mensaje transmitido por medios electrónicos y que identifica de manera exclusiva al remitente.

FORTEZZA: Sistema de cifrado usado por el gobierno de los EUA para manejar información sensible pero no clasificada.

Frame-Relay: Técnica de comunicación mediante retransmisión de tramas.

FTP (**F**ile **T**ransfer **P**rotocol / Protocolo de transferencia de archivos**):** Protocolo para transferir archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor.

Función hash: Es una función para resumir o identificar probabilísticamente un gran conjunto de información, dando como resultado un conjunto imagen finito generalmente menor.

Gateway: Equipos para interconectar redes con protocolos y arquitecturas completamente diferentes a todos los niveles de comunicación.

GRE (**G**eneric **R**outing **E**ncapsulation / Encapsulación genérica de enrutamiento**):** Protocolo para el establecimiento de túneles a través de Internet

Hash: Se refiere a una función o método para generar llaves que representen de manera casi unívoca a un documento, registro, archivo, etc., resumir o identificar un dato a través de la probabilidad, utilizando una *función hash* o *algoritmo hash*. Un hash es el resultado de dicha función o algoritmo.

HMAC (**H**ashed **MAC**)**:** Algoritmo de autenticación, *véase pág. 103.*

Host: Nombre único que se le da a un dispositivo conectado a una red. Puede ser un ordenador, un servidor de archivos, un dispositivo de almacenamiento por red, una máquina de fax, impresora, etcétera.

HTTP (Hyper**T**ext **T**ransfer **P**rotocol / Protocolo de transferencia de hipertexto**):** Es el protocolo de la Web (WWW), usado en cada transacción. El hipertexto es el contenido de las páginas web, y el protocolo de transferencia es el sistema envío/respuesta para acceder a una página web.

IANA (Internet **A**ssigned **N**umber **A**uthority / Agencia de Asignación de Números Internet): Era el antiguo registro central de los protocolos Internet, como puertos, números de protocolo y empresa, opciones y códigos.

ICMP (Internet **C**ontrol **M**essage **P**rotocol / Protocolo de Mensajes de Control de Internet): Protocolo de diagnóstico y notificación de errores

IDEA (**I**nternational **D**ata **E**ncryption **A**lgorithm / Algoritmo internacional para el cifrado de datos)**:** Algoritmo de cifrado, véase pág. 113.

IEEE (**I**nstitute of **E**lectrical and **E**lectronics **E**ngineers / Instituto de Ingenieros Eléctricos y Electrónicos)**:** Asociación estadounidense dedicada a la estandarización.

IETF (**I**nternet **E**ngineering **T**ask **F**orce / Grupo de Trabajo en Ingeniería de Internet)**:** Organización internacional abierta que participa en el desarrollo de los estándares de Internet (protocolos, algoritmos, etc.)

IIS (**I**nternet **I**nformation **S**ervices / Servidor de Información de Internet)**:** Es el servidor Web de Microsoft que corre sobre plataformas Windows.

IKE (**I**nternet **K**ey **E**xchange ***/*** Intercambio de llaves por Internet): *véase pág. 135.*

IMAP (**I**nternet **M**essage **A**ccess **P**rotocol / Protocolo de Acceso a Mensajes de Internet)**:** Protocolo de red de acceso a mensajes electrónicos almacenados en un servidor

Intranet: Es una red interna o privada de una organización que utiliza la tecnología de Internet; se puede decir que es un "Internet Privado" que sólo puede ser usado por las computadoras conectadas a esa red.

IP (**I**nternet **P**rotocol)**:** Protocolo de capa 3 desarrollado bajo el financiamiento del Departamento de Defensa de Estados Unidos a mediados de los 1970s. Está instrumentado en una gran variedad de equipos y plataformas, haciéndolo mayormente disponible e independiente de marcas comerciales. Actualmente se denomina IPv4 debido a la versión nombrada IPv6 destinada a sustituirlo.

IPSec (**I**nternet **P**rotocol **S**ecurity): Protocolo de seguridad, *véase capítulo 3.*

IPX (**I**nternetwork **P**acket e**X**change / Intercambio de paquetes inter-red): Protocolo de comunicaciones NetWare que se utiliza para encaminar mensajes de un nodo a otro sin garantizar la entrega de un mensaje completo.

ISAKMP (Internet Security Association and Key Management Protocol / Asociación Segura de Internet y Protocolo de Administración de Llaves)**:** *véase pág. 137.*

ISATAP: Tipo de túnel automático, *véase pág. 77.*

ISO (**I**nternational **S**tandards **O**rganization / Organización Internacional de Estándares)**:** Organización no gubernamental fundada en 1947 con sede en Ginebra Suiza encargada de producir normas internacionales con la finalidad de facilitar el comercio y el intercambio de información.

ISP (**I**nternet **S**ervice **P**rovider / Proveedor de Servicios de Internet)**:** Término que designa a la compañía que provee el servicio de Internet.

iTunes: Aplicación multimedia creada por Apple Computer.

KAME: Proyecto para el desarrollo de IPv6 en plataformas BSD (http://www.kame.net).

Kerberos: Mecanismo de autenticación de usuarios, véase *pág. 12*.

Konqueror: Navegador de Internet para plataformas Unix/Linux.

L2F (**L**ayer **2 F**orwarding)**:** Protocolo de seguridad, *véase pág. 29.*

L2TP (**L**ayer **2 T**unneling **P**rotocol): Protocolo de seguridad, *véase pág. 29*.

LEAP (**L**ightweight **EAP**)**:** Protocolo de seguridad, *véase pág. 34.*

Llave o clave: Pieza de información que habitualmente es una secuencia de números o letras para controlar la operación en un algoritmo de criptografía.

MAC (**M**essage **A**uthentication **C**ode / Código de Autenticación de Mensajes)**:** Código para autenticar el origen de los mensajes.

MAC Address (**M**edia **A**ccess **C**ontrol **Address** / Dirección de Control para el Acceso al Medio)**:** Dirección única que llevan las tarjetas de red grabadas en una ROM para identificarse y diferenciarse de las demás.

MD5 (**M**essage **D**igest **v**ersión **5**)**:** Algoritmo de autenticación, *véase pág. 100.*

MDC (**M**odification **D**etection **C**odes / Código de Detección de Modificaciones)**:** Un método en que puede operar una función hash para la integridad de los mensajes.

MIC o Michael (**M**essage **I**ntegrity **C**ode / Código de Integridad del Mensaje): Código para verificar la integridad de los datos en las tramas.

MIPv6 (**M**obility **I**nternet **P**rotocol **v6** / Movilidad en el Protocolo de Internet v6): Protocolo utilizado para movilidad basado en IPv6.

MIT (**M**assachusetts **I**nstitute of **T**echnology / Instituto Tecnológico de Massachusetts): Institución dedicada a la ciencia, ingeniería e investigación en los Estados Unidos.

MOSS (**MIME O**bject **S**ecurity **S**ervices): Protocolo de seguridad, *véase pág. 18*.

Mozilla: Navegador de Internet para plataformas Unix/Linux, Windows, Mac.

MPLS (**M**ulti-**P**rotocol **L**abel **S**witching): Mecanismo de transporte de datos estándar diseñado para unificar el servicio de transporte de datos para las redes basadas en circuitos y las basadas en paquetes en base a etiquetas.

MPPE (Microsoft **P**oint to **P**oint **E**ncryption / Cifrado Punto a Punto diseñado por Microsoft): Protocolo para crear una VPN segura usando un algoritmo de cifrado RSA/RC4.

MS-CHAP (Micro**S**oft **CHAP** / CHAP creado por Microsoft)**:** Protocolo de seguridad, *véase pág. 33.*

MTU (Maximum **T**ransfer **U**nit / Unidad Máxima de Transferencia): Término que expresa el tamaño en bytes del datagrama más grande que puede pasar por una capa de un protocolo de comunicaciones.

Multicast: Envío de información a uno o más receptores específicos de manera simultánea.

Multihoming de sitio: Conexión de un host o sitio a más de un ISP a la vez, y es hoy por hoy un componente esencial para muchos sitios conectados a Internet.

NAT (**N**etwork **A**ddress **T**ranslation / Traductor de Dirección de Red): es una aplicación por que determinado dispositivo o aplicación de software sea capaz de

cambiar la dirección IP de origen o destino por otra dirección definida previamente. Se puede utilizar para dar salida a redes públicas a ordenadores que se encuentran con direccionamiento privado o para proteger máquinas públicas.

NAT-PT (**N**etwork **A**ddress **T**ranslation - **P**rotocol **T**ranslation / Traducción de Dirección de Red – Protocolo de traducción): Tipo de traductor, *véase pág. 87.*

NDP (**N**eighbor **D**iscovery **P**rotocol / Protocolo de Descubrimiento de Vecinos)**:** Protocolo para descubrir a otros nodos en la misma red local, determinar su dirección de nivel de enlace, encontrar enrutadores y mantener información de la ruta hacia otros nodos activos.

Netscape: Navegador de Internet para plataformas Unix/Linux, Windows y Mac.

NIC (**N**etwork **I**nformation **C**enter / Centro de Información de Red)**:** Institución encargada de asignar dominios de Internet bajo su dominio de red, a personas naturales o empresas que mediante un DNS pueden montar sitios de Internet mediante un proveedor.

NIST (**N**ational **I**nstitute of **S**tandards and **T**echnology / Instituto Nacional de Normas y Tecnología): Organismo Federal no regulador que forma parte del Departamento de Comercio de los Estados Unidos, donde una de sus misiones investigar e innovar cuestiones relacionados a la tecnología.

NLSP (**N**etwork **L**ayer **S**ecurity **P**rotocol / Protocolo de Seguridad en capa de Red)**:** Protocolo de seguridad, *véase pág. 30.*

Nodo: Es cualquier dispositivo conectado a una red.

NSA (**N**acional **S**ecurity **A**gency / Agencia Nacional de Seguridad)**:** Agencia estadounidense dedicado especialmente en la seguridad informática.

NSAP (**N**etwork **S**ervice **A**ccess **P**oint / Punto de Acceso al Servicio de Red)**:** Es un tipo de direcciones que se asocian a un dispositivo en particular.

OAKLEY: Protocolo de intercambio de llaves, *véase pág. 137.*

Opera: Navegador de Internet para plataformas Unix/Linux.

OSI (**O**pen **S**ystem **I**nterconnected / Interconexión de Sistemas Abiertos)**:** Modelo teórico propuesto por la ISO que describe cómo deberían conectarse las distintas computadoras a diferentes tipos de red para poder comunicarse entre sí.

PAP (**P**assword **A**uthentication **P**rotocol ***/*** Protocolo de Autenticación por Contraseña)**:** Protocolo de validación de usuarios, *véase pág. 32.*

Paquete: Conjunto de datos que son enviados hacia un emisor por Internet.

PAT: (**P**ort **A**ddress **T**ranslation / Traductor de Dirección de Puerto)**:** Es una característica de NAT que traduce conexiones TCP y UDP hechas por un host y un puerto en una red externa a otra dirección y puerto de la red interna permitiendo que una sola dirección IP sea utilizada por varias máquinas de la intranet.

PCT (**P**rivate **C**ommunication **T**echnology / Tecnología de Comunicación Privada)**:** Protocolo de seguridad, *véase pág. 24.*

PEAP (**P**rotected **EAP /** EAP Protegido)**:** Protocolo de seguridad, *véase pág. 34.*

PEM (**P**rivacy **E**nhanced **M**ail / Correo Privado Mejorado)**:** Protocolo de seguridad, *véase pág. 17*.

PGP (**P**retty **G**ood **P**rivacy / Privacidad Muy Buena)**:** Protocolo de seguridad, *véase pág. 15.*

Pidgin: Messenger con soporte IPv6.

PKI (**P**ublic **K**ey **I**nfrastructure / Infraestructura de Llave Pública)**:** Es una combinación de hardware y software, políticas y procedimientos que permiten asegurar la identidad de los participantes en un intercambio de datos usando criptografía pública.

POP (**P**ost **O**ffice **P**rotocol / Protocolo de Oficina de Correos)**:** Protocolo usado para la recuperación de correo electrónico.

PPP (**P**oint-to-**P**oint **P**rotocol / Protocolo Punto a Punto)**:** Protocolo permite establecer una comunicación a nivel de enlace entre dos computadoras.

PPTP (**P**oint-to-**P**oint **T**unneling **P**rotocol / Protocolo de Túnel Punto a Punto)**:** Protocolo de seguridad*, véase pág. 27*.

Protocolo: Conjunto de reglas que dos dispositivos deben de seguir para intercambiar mensajes.

Proxy: Hace referencia a un programa que realiza una acción en representación de otro.

Puerto: Es una forma genérica de denominar a una interfaz por la cual diferentes tipos de datos pueden ser enviados y recibidos.

QoS (**Q**uality of **S**ervice: / Calidad de Servicio)**:** Se refiere a como deben ser tratados los distintos tipos de datos (datos, voz, video) para ofrecer una calidad optima por medio de parámetros como son velocidad, ancho de banda, etc.

Quagga: Es un software libre para poder usar los sistemas basados en Unix como enrutadores.
RADIUS (**R**emote **A**ccess **D**ial **I**n **U**ser **S**ervice / Servicio de Usuario de Marcado con Autenticación Remota): Es un protocolo de autenticación y autorización para aplicaciones de acceso a la red.

RC5 (**R**ivest **C**ipher v5 / Cifrado de Rivest v5)**:** Algoritmo de cifrado, *véase pág. 114*.

RFC (**R**equest **F**or **C**omments: Documentos de especificaciones que se expone públicamente para su discusión.

RIPEMD (**RACE I**ntegrity **P**rimitives **E**valuation **M**essage **D**igest): Algoritmo de autenticación, *véase pág. 104.*

Router (enrutador o encaminador): Dispositivo de hardware o software de interconexión de redes de computadoras que opera en la capa 3 (nivel de red) del modelo OSI. Este dispositivo interconecta segmentos de red o redes enteras. Hace pasar paquetes de datos entre redes tomando como base la información de la capa de red.

RSA (**R**ivest, **S**hamir y **A**dleman): Algoritmo de cifrado de llave pública, *véase pág. 14*.

S/MIME (**S**ecure **M**ulti-purpose **I**nternet **M**ail **E**xtensions / Extensiones de correo Internet multipropósito seguras): Protocolo de seguridad, *véase pág. 19*.

S/WAN: Proyecto dedicado al desarrollo de IPv6 por medio de sus distribuciones FreeS/WAN, OpenS/WAN, y StrongS/WAN.

SA (**S**ecurity **A**ssociations / Asociaciones de Seguridad)**:** Asociaciones de seguridad de IPSec, *véase pág. 92.*

SAD (**S**ecurity **A**ssociations **D**atabase / Base de datos de las Asociaciones de Seguridad)**:** Almacenamiento de las SA de IPSec, *véase pág. 92.*

Servidor web: Servidor dedicado a la publicación de páginas web por medio del protocolo HTTP.

SET (**S**ecure **E**lectronic **T**ransaction / Transacciones electrónicas seguras): Protocolo de seguridad, *véase pág. 21*.

SHA (**S**ecure **H**ash **A**lgorithm / Algoritmo hash seguro)**:** Algoritmo de autenticación, *véase pág. 102.*

S-HTTP (**S**ecure **- H**yper **T**ext **T**ransfer **P**rotocol / Protocolo de Transferencia de Hipertexto Seguro)**:** Protocolo de seguridad, *véase pág. 19.*

SIIT (**S**tateless **I**P/**I**CMP **T**ranslator / Traductor IP/ICMP sin estado): Tipo de traductor, *véase pág. 84.*

Sistema operativo: Conjunto de programas que sirve como interfaz entre el usuario y el dispositivo permitiendo la administración eficaz de los recursos de éste.

SKEME: Protocolo de seguridad, *véase pág. 136.*

SMTP (**S**imple **M**ail **T**ransfer **P**rotocol / Protocolo de Transferencia de Correo Sencillo): Protocolo para la transferencia de correo electrónico entre servidores.

Sniffing: Acción que consiste en espiar y obtener la información que circula por una red.

Socket: Concepto abstracto por el cual dos programas (posiblemente situados en computadoras distintas) pueden intercambiar cualquier flujo de datos, generalmente de manera fiable y ordenada.

SOCKS: Tipo de traductor, *véase pág. 88.*

SP (**S**ecurity **P**olicies / Políticas de Seguridad)**:** Políticas de seguridad de IPSec, *véase pág. 92.*

SP (**S**ervice **P**ack / Paquete de servicio): Grupo de parches que actualizan, corrigen y mejoran aplicaciones y sistemas operativos. Esta denominación fue popularizada por Microsoft cuando comenzó a empaquetar grupos de parches que actualizaban su sistema operativo Windows.

SPD (**S**ecurity **P**olicies **D**atabase / Base de datos de las Políticas de Seguridad)**:** Almacenamiento de las SP de IPSec, *véase pág. 92.*

SPI (**S**ecurity **P**arameter **I**ndex / Índice de Parámetros de Seguridad): Identificador de seguridad utilizado por IPSec *véase pág. 92.*

Spoofing: Acción que consisten en la suplantación de identidad por parte del atacante para hacerse pasar como una persona autorizada.

SRI-NIC (**S**tanford **R**esearch **I**nstitute - **N**etwork **I**nformation **C**enter / Instituto de Investigación de Stanford – Centro de información de red): Instituto que

desempeñaba funciones de administración y supervisión de algunos recursos de Internet como lo eran ARPANET y NSFNET.

SSH (**S**ecure **Sh**ell / Intérprete de comandos seguro): Protocolo de seguridad, véase *pág. 14.*

SSL (**S**ecure **S**ockets **L**ayer / Seguridad a nivel capa Socket): Protocolo de seguridad, *véase pág. 22.*

ST2+ (Internet **St**ream Protocol version **2**): Es una extensión experimental del IP, sin embargo no se concluyó en nada quedándose en desuso

Switch: Es dispositivo electrónico de interconexión de redes de dispositivos para la conmutación de paquetes.

TACACS (**T**erminal **A**ccess **C**ontroller **A**ccess **C**ontrol **S**ystem / Sistema de control de acceso para controlar de acceso a terminales): es un protocolo de autenticación remota que se usa para comunicarse con un servidor de autenticación.

TAHI: Proyecto para el desarrollo de IPv6 (http://www.tahi.org).

TCP (**T**ransmission **C**ontrol **P**rotocol / Protocolo de Control de Transmisión): Protocolo de transporte orientado a conexión utilizado en Internet y utilizado por muchas aplicaciones como: Telnet, FTP, SMTP y HTTP para establecer comunicaciones confiables.

Telnet: Servicio que nos permite la comunicación con otros dispositivos de la red vía remota.

Teredo: Tipo de túnel automático, *véase pág. 78.*

TightVNC: Es un software que permite acceder de forma remota a una máquina desde otra conectada a Internet, además de disponer de un cliente Java que se puede utilizar desde un navegador.

TLS (**T**ransport **L**ayer **S**ecure / Seguridad para capa de Transporte): Protocolo de seguridad, *véase pág. 25.*

Traductor: Dispositivo que convierte direcciones IPv4 en IPv6 y viceversa.

TRT (**T**ransport **R**elay **T**ranslator / Traductor que retransmite en la capa de Transporte): Tipo de traductor, *véase pág. 88.*

Túnel: Comunicación entre dos dispositivos para intercambio de información.

UDP (**U**ser **D**atagram **P**rotocol / Protocolo de datagrama a nivel de usuario)**:** Protocolo de nivel de transporte basado en el intercambio de datagramas a través de la red sin que se haya establecido previamente una conexión, ya que el propio datagrama contiene suficiente información de direccionamiento en su encabezado.

Unicast: Envío de información de un único emisor a un único receptor.

USAGI: Proyecto para el desarrollo de IPv6 (http://www.linux-ipv6.org).

VLC (**V**ideo **L**AN **C**lient)**:** Reproductor multimedia multiplataforma y de código libre que puede ser usado como servidor en unicast o multicast, en IPv4 o IPv6, en una red de banda ancha.

VPN (**V**irtual **P**rivate **N**etworks / Redes Privadas Virtuales): Se refiere a una red en la cual algunas partes se conectan usando Internet público, pero los datos enviados por Internet se cifran de manera que toda la red es "virtualmente" privada

WEP (**W**ired **E**quivalency **P**rivacy / Privacidad equivalente a redes cableadas): Protocolo de seguridad, *véase pág. 30.*

WIDE: Proyecto para el desarrollo de IPv6 (http://www.wide.ad.jp).

Wi-Fi: Conjunto de estándares para redes inalámbricas basados en la especificación IEEE 802.11.

Wireless: Término denominado a la tecnología inalámbrica de comunicaciones.

WPA (**W**i-Fi **P**rotected **A**ccess / Acceso protegido Wi-Fi): Protocolo de seguridad, *véase pág. 32.*

X.509: Estándar para infraestructuras de llaves públicas para la emisión de certificados

X11: El sistema de ventanas X fue desarrollado en el MIT para otorgar de una interfaz gráfica a los sistemas Unix. Este protocolo permite la interacción gráfica en red entre un usuario y una o más computadoras haciendo transparente la red para éste. Generalmente se refiere a la versión 11 de este protocolo (usado actualmente).

BIBLIOGRAFÍA

LIBROS CONSULTADOS

Doraswamy Naganand, Harkins Dan, *IPSec The New Security Standard for the Internet, Intranets, and Virtual Private Networks*, ed. Prentice Hall, E.U.A., 1999.

European Commission, *IPv6 and Broadband (IPv6 Cluster)*, Information Society Technologies, 2002

Frankel Sheila, *Demystifying the IPSec Puzzle*, ed. Artech House Inc., Boston-London, 2001.

Hagen Silvia, *IPv6 Essentials*, ed. O'Reilly Media Inc., EUA, 2002.

Iljitsch van Beijnum, *Running IPv6*, ed. Apress, E.U.A., 2005

Johnson Kevin, *Internet Email Protocols (A Developer's guide)*, ed. Addison-Wesley, Massachusetts, 2000.

Mostafa Hashem Sherif, *Protocols for secure electronic commerce*, the CRC Press advanced and emerging communications technologies series, Florida E.U.A., 2000.

Oppliger Rolf, *Security Technologies for the World Wide Web*, ed. Artech House, 2da edición, Boston-London, 2003.

Parenti Edgar Jr., Browne Brian, Knipp Eric, *Configuring IPv6 for Cisco IOS* ed. Syngress, E.U.A., 2002

Seberry Jennifer, Hardjono Thomas, Pieprzyk Josef, *Fundamentals of Computer Security*, ed. Springer-Verlag Berlin and Heidelberg GmbH & Co. KG, Alemania, 2003

Welsh Matt, Kalle Matthias, Kaufman Lar, *Running Linux*, ed. O'Reilly, 3a edición, California EUA, 1999.

PAGINAS DE INTERNET

6BONE: Tested for deployment of IPv6. http://www.6bone.net.

6sos: Servicio de información y soporte para IPv6 http://www.6sos.org.

BSD: Distribuciones para el soporte de IPv6 sobre BSD http://www.freebsd.org, http://www.openbsd.org.

CERT: Reporte Anual 2003 http://www.cert.org/annual_rpts/cert_rpt_03.html.

Cisco: Empresa con soporte IPv6 e IPSec http://www.cisco.com.

CUDI: Corporación Universitaria para el Desarrollo de Internet, http://www.cudi.edu.mx.

IETF: Grupo de Trabajo. http://www.ietf.org, http://www.rfc-editor.org.

IPv6 Ready: IPv6 Ready Logo Committee http://www.ipv6ready.org/frames.html.

IPv6: Pagina de información sobre IPv6 http://www.ipv6.org.

KAME: http://www.kame.net.

Kernels Linux: http://www.kernel.org/pub/linux.

Microsoft: Soporte para IPv6 e IPSec http://www.microsoft.com/ipv6.

NAv6TF: North American IPv6 Task Force http://www.nav6tf.org. "*IPv6 Security Technology Paper*", 2006.

S/WAN: Proyecto Linux para el desarrollo de IPv6. http://www.freeswan.org, http://www.openswan.org, http://www.strongswan.org.

TAHI: http://www.tahi.org.

TheIPv6PORTAL: Guide, news, projects, developments, events, faqs, etc. for IPv6 http://www.ist-ipv6.org

TLDP: The Linux Documentation Project http://tldp.org. The official IPSec How-to for Linux http://www.ipsec-howto.org

USAGI: UniverSAl playGround for Ipv6 http://www.linux-ipv6.org

UNAM: Proyecto de IPv6 en la UNAM http://www.ipv6.unam.mx

WIDE: Widely Integrated Distributed Environment http://www.wide.ad.jp

RFCs CONSULTADOS

RFC1334 *PPP Authentication Protocols*, Octubre 1992.

RFC1421 *Privacy Enhancement for Internet Electronic Mail Part I: Message Encryption and Authentication Procedures*, Febrero 1993.

RFC1510 *The Kerberos Network Authentication Service*, Septiembre 1993

RFC1848 *MIME Object Security Services*, Octubre 1995.

RFC1991 *PGP Message Exchange Formats*, Agosto 1996.

RFC1994 W. Simpson. *PPP Challenge Handshake Authentication Protocol (CHAP)*, Agosto 1996.

RFC2015 *MIME Security with Pretty Good Privacy (PGP*), Octubre 1996.

RFC2246 *The TLS Protocol Version 1.0*, Enero 1999.

RFC2284 *PPP Extensible Authentication Protocol (EAP)*, Marzo 1998.

RFC2375 *IPv6 Multicast Address Assignments*, Julio 1998.

RFC2433 *Microsoft PPP CHAP Extensions*, Octubre 1998.

RFC2460 *Internet Protocol, Version 6 (IPv6) Specification*, Diciembre 1998.

RFC2633 *S/MIME Version 3 Message Specification*, Junio1999.

RFC2660 *The Secure HyperText Transfer Protocol*, Agosto 1999.

RFC3031 *Multiprotocol Label Switching Architecture*, Enero 2001.

RFC3513 *IP Version 6 Addressing Architecture*, Febrero 2006.

RFC3587 *IPv6 Global Unicast Address Format*, Agosto 2003.

RFC3596 *DNS Extensions to Support IP Version 6*, Octubre 2003.

RFC3775 *Mobility Support in IPv6*, Junio 2004.

RFC3971 *SEcure Neighbor Discovery (SEND)*, Marzo 2005.

RFC4193 *Unique Local IPv6 Unicast Addresses*, Octubre 2005.

RFC4213 *Basic Transition Mechanisms for IPv6 Hosts and Routers*, Octubre 2205.

RFC4251 *The Secure Shell (SSH) Protocol Architecture*, Septiembre 1993.

RFC4301 *Security Architecture for the Internet Protocol*, Diciembre 2005.

RFC4302 *IP Authentication Header*, Diciembre 2005.

RFC4303 *IP Encapsulating Security Payload (ESP)*, Diciembre 2005.

RFC4306 *Internet Key Exchange (IKEv2) Protocol*, Diciembre 2005.

RFC4581 *Cryptographically Generated Addresses (CGA)*, Octubre 2006.

RFC4835 *Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)*, Abril 2007.

RFC4861 *Neighbor Discovery for IP version 6 (IPv6),* Septiembre 2007.

RFC4864 *Local Network Protection for IPv6*, Mayo 2007.

RFC4869 *Suite B Cryptographic Suites for IPSec*, Mayo 2007.

RFC4891 *Using IPSec to Secure IPv6-in-IPv4 Tunnels*, Mayo 2007.