UNIVERSIDAD POLITÉCNICA DE VALENCIA

Facultad de Informática

IMPLEMENTACIÓN DE UNA VPN (VIRTUAL PRIVATE NETWORK) USANDO EL ESTÁNDAR IPSEC

PROYECTO FIN DE CARRERA

REALIZADO POR:
Calixto Calderón Rodríguez

DIRIGIDO POR:
Antonio Sánchez Salmerón

Agosto 2001

0 ÍNDICE

1 Redes Privadas Virtuales (VPNs)

1.1 Introducción

El término VPN, ha llegado a ser utilizado de manera imprudente en la industria del "networking" para describir un amplio conjunto de problemas y soluciones, donde los objetivos no siempre han sido explicados claramente.

Este uso ha llevado a una situación donde todo el entorno relacionado con las tecnologías de red usan el término VPN como un referente para un conjunto diverso de tecnologías.

La idea consiste en crear una red privada –confidencial- (vía tunneling y/o encriptación) a través de la red pública Internet. Una especie de WAN "privada" con las ventajas adicionales de abaratamiento (con respecto a ISDN, Frame-Relay…) de las conexiones. Como inconveniente está la dificultad de garantizar la **privacidad** de dicha conexión.

1.2 Aproximación histórica a las VPNs

En un mundo donde la información es un bien preciado y donde los diferentes organismos (públicos y privados) así como las personas en general se encuentran inmersos en un entorno de continuo cambiamento, aparece la necesidad de contar con un medio de transmisión barato y seguro (confidencial) a través del cual poder intercambiar dicha información. Es en este contexto donde podemos situar perfectamente a las VPNs.
Con la implantación actual de Internet se alcanza el objetivo de una comunicación barata. No se debe olvidar que siempre existe la posibilidad de contratar líneas dedicadas de alta velocidad (ISDN, Frame-Relay).
Cuando se habla de comunicación segura (privada) a través de líneas públicas se encuentra la necesidad de usar protocolos seguros. Protocolos que usen bien "tunneling", bien encriptación o bien ambas técnicas a la vez.
A partir de este punto se han desarrollado múltiples soluciones comerciales que proporcionan comunicación segura.

1.3 Una definición teórica de VPN

Consideremos el caso de un conjunto de sites que están conectados a una red común que llamaremos "backbone". Apliquemos algún tipo de política para crear subconjuntos de sites y usemos la siguiente definición: dos sites pueden tener conectividad IP mediante el backbone SOLO si al menos uno de estos subconjuntos contiene a ambos sites.

Los subconjuntos que hemos creado son las VPN. Dos sites tiene conectividad IP usando el backbone común solo si hay alguna VPN que contiene a ambos. Dos sites que no tiene una VPN en común no tienen conectividad mediante el backbone.

1.3.1 Intranets-Extranets

Si todos los sites en una VPN son propiedad de la misma empresa, la VPN es una intranet corporativa. En otro caso, la VPN será una extranet. Un site puede formar parte de más de de una VPN;vg, de una intranet y de varias extranets. A ambos tipos (intranets y extranets) de configuraciones los consideraremos VPN.

Así, en todos los casos debemos asegurar el tráfico entre ambas partes ante posibles ataques externos de cualquier tipo. Para ello dispondremos de elementos físicos (gateways seguros) y lógicos (sofware gestor de los gateways así como del cliente remoto).

1.4 Seguridad

La introducción del término red privada virtual **NO** involucra la creación de nuevos paradigmas en el networking dada su completa analogía con las redes físicas.

Atendiendo a la línea física utilizada para conectar los diferentes sites que forman la VPN podemos hacer una primera división:

1) De línea dedicada
2) Conexión bajo demanda a través de un **ISP**

Generalmente, las lineas dedicadas o los circuitos dedicados (Frame Relay, ATM) entre los diferentes sites no están completamente interconectados; en su lugar usan algún tipo de topología jerárquica.

Las redes con conexión bajo demanda permiten usar la Red de Telefonía Básica (RTB) o bien la RDSI (Red Digital de Servicios Integrados). Debido al bajo coste actual de una conexión a Internet, hay un creciente interés en el desarrollo de VPNs usando Internet como soporte, lo cual no es nuevo como idea. Sin embargo, es solo ahora cuando existen los mecanismos apropiados (a nivel IP) que permiten cumplir con los requerimientos del usuario con respecto a las VPNs.
Requerimientos:

1 Transporte “oculto” de paquetes.
2 Seguridad de los datos. Existen dos opciones:

El usuario final implementa su propia política de seguridad o bien el cliente confía esta tarea a su proveedor de servicio

En el segundo caso, si los datos circulan tan solo por las instalaciones de un único proveedor es posbile que IPSec no sea necesario.
En caso contrario, es decir, si el tráfico atraviesa circuitos pertenecientes a más de un proveedor, necesitamos de un mecanismo de seguridad (vg.: IPSec)

3 Garantía de la calidad de servicio (QoS): (además de asegurar la privacidad)

Tanto las líneas alquiladas (leased) como las conexiones mediante dial-up ofrecen latencia y ancho de banda garantizados.
Las tecnologías de conexión dedicada (ATM, Frame-Relay) disponen de mecanismos para garantizar la QoS.

4 Mecanismo de tunelizado (“tunneling”):

Se trata de usar formatos de paquete y/o de direccionamiento con la VPN que sea distinto de aquel usado con la normal comunicación a través de Internet

2 Introducción a la criptografía

2.1 Conceptos

La palabra criptografía proviene de las palabras griegas kryptos -que significa *esconder*- y gráphein -que significa *escribir*-, es decir, *escritura oculta*.

La criptografía se usa principalmente para mandar información de manera que sólo el destinatario pueda acceder al contenido de la misma.

Usando la criptografía implementaremos los siguientes servicios: autenticidad, confidencialidad, integridad y no-repudiación.

Confidencialidad.- Se trata de garantizar que el acceso a una determinada información se producirá únicamente por parte del destinatario.

Integridad.- En este caso se garantiza que la información transmitida llega inalterada a su destino.

Autenticidad.- El objetivo aquí es garantizar que el origen de la información se corresponde con el esperado.

No repudiación.- Aquí lo que se logra es probar frente a terceros que la información es auténtica.

Atendiendo al tipo de clave utilizada para encriptar/desencriptar el mensaje cifrado podemos dividir los algoritmos de cifrado en dos grandes bloques:

Algoritmos de clave simétrica –*simétricos*-
Algoritmos de clave asimétrica –*asimétricos*-

2.2 Algoritmos simétricos (clave simétrica)

2.2.1 Perspectiva

Históricamente, todos los algoritmos de este tipo se han basado en usar algún tipo de desplazamiento, permutación y/o transposición del mensaje original.

El problema de estas técnicas es que mantienen una relación -en cuanto a ocurrencia de caracteres- del texto original con el texto cifrado. Así, usando técnicas de análisis estadístico es posible “atacarlos” –intentos de descubrimiento de la clave-.

La moderna criptografía se considera que empieza con los estudios de Claude Shannon. Este, introdujo la idea de usar difusión y confusión como los elementos para construir los criptosistemas.

Con la difusión lo que conseguimos es romper la relación estadística entre las ocurrencias de caracteres del texto en claro y el texto cifrado.

Con las técnicas de confusión complicamos tanto como sea posible la relación entre el texto cifrado y la clave de cifrado. Esto último se consigue usando complejos algoritmos de substitución que aseguren el que cada carácter del texto cifrado tenga el máximo número de posibles correspondencias con respecto al texto en claro.

2.2.2 Diseño de criptosistemas de clave simétrica

Actualmente, la práctica totalidad de los criptosistemas utilizados usan el método de *cifrado de bloque ("block ciphers").* Es decir, dividen el texto en claro en bloques (de longitud fija o variable) y obtienen como resultado un texto cifrado de la misma longitud del texto en claro.

Actualmente, la longitud de los bloques es de al menos 128 bits.

Otro modo plantear la encriptación es el *"stream ciphers"* en donde el algoritmo encripta uno a uno los caracteres del texto en claro. Su uso es bastante limitado.

2.2.2.1 Aspectos de seguridad

Como parámetros a tener en cuenta en el diseño de un algoritmo simétrico tenemos:

longitud de la clave –en bits-, la seguridad aumenta con el número de bits.

Longitud del bloque, sucede lo mismo que con la clave; sin embargo, debemos tener en cuenta que las prestaciones se degradan del mismo modo. Así pues, debemos llegar a una solución de compromiso.

Función de "reparto de la clave", en general, los algoritmos hacen un uso parcial de la clave en cada iteración del algoritmo, repartiendo los bits de la misma entre los diferentes bloques. Así pues, es un aspecto importantísimo el diseño de esta función para determinar la potencia del cifrado.

Número de fases ("rounds"), aquí, teóricamente un mayor número de fases desemboca en un mejor cifrado. Sin embargo, cada nueva fase origina un mayor número de operaciones que afectan a la productividad del algoritmo. Además, hay que tener en cuenta el diseño de la fase; es decir, una fase de un determinado algoritmo puede equivaler –en cuanto a potencia de cifrado- a dos de otro.

Resistencia al criptoanálisis –estudio de técnicas de "rotura" (descubrimiento de la clave) de criptosistemas-; de este modo, antes de la aparición del DES, el análisis estadístico podía ser usado con éxito. Actualmente existen un par de técnicas –de utitilidad básicamente teórica dada su complejidad- que utilizan texto en claro para realizar sus ataques. Así pues, como principio general se considera que el orden de magnitud para criptoanalizar exitosamente el algoritmo sea mayor que el requerido a través de un ataque de "fuerza bruta" (probando todas las posibilidades).

2.2.2.2 Aspectos de implementación y prestaciones

Es igual de importante el aspecto de seguridad como el que sea relativamente sencillo implementar el algoritmo a partir de software y/o hardware. Asimismo, no se puede olvidar la productividad del algoritmo en cualquier implementación del mismo.

Como aspectos principales que influyen en la implementación y prestaciones tenemos el tipo de operación básica del algoritmo. Así, cuanto más próxima sea esta al hardware, mayores prestaciones obtendremos.

Un aspecto a considerar es el uso de memoria, sobre todo actualmente, cuando la tendencia al uso de las Smart Cards –con una capacidad limitada de memoria- está aumentando extraordinariamente.

2.2.3 Ejemplos de algoritmos simétricos

Como ejemplos de algoritmos que están siendo utilizados actualmente y que cumplen todas las condiciones anteriormente expuestas tenemos los siguientes.

La lista está centrada sobre todo en los finalistas del concurso organizado por el NIST (National Institute for the Standards and Technology) americano para la obtención de un algoritmo utilizado por el gobierno americano durante los próximos 20 años.

El vencedor resultó el algoritmo de Rijndael –que pasó a denominarse AES (Advanced Encryption System)-.

2.2.3.1 DES

Algoritmo actualmente en desuso ya que con la potencia actual de los computadores es factible el descubrimiento de la clave en un tiempo razonable.

El problema con DES no se refiere a su diseño, sino a la longitud de su clave (56 bits para encriptar bloques de texto en claro de 64 bits). Así, dado un texto en claro de 64 bits y el correspondiente texto cifrado (64 bits = 56 + 8 paridad) resultado de cifrar el

primero con DES, la clave DES de 56 bits se puede encontrar en un máximo de 2^55 operaciones.

2.2.3.2 3DES

Es una variante más segura de DES (longitud de clave de 168 bits). Sin embargo, su funcionamiento resulta mucho más lento que el del DES.

Al igual que el DES, usa cifrado de bloques de 64bits.

2.2.3.3 MARS

Fue desarrollado por un grupo de trabajo de IBM. Es de libre disposición –royalty-free- en todo el mundo.

Usa un tamaño de bloque de 128 bits y claves de longitud variable en un rango de 128 a 448 bits.

Las operaciones básicas son sumas, restas, OR-EX, búsquedas en tablas y rotación de bits fija y dependiente de la cantidad. Además, usa 16 operaciones de multiplicación.

2.2.3.4 RC6

Desarrollado por Ronald Rivest y un grupo de los laboratorios RSA. Está protegido por copyright.

Usa un tamaño de palabra variable, un número variable de fases y tamaño de clave de hasta 2040 bits.

Como operaciones básicas usa la summa, resta, OR-EX y rotación de bits además de 32 operaciones de multiplicación.

2.2.3.5 AES

Diseñado por Joan Daemen y Vincent Rijmen (Bélgica).

Sopporta tallas de clave y bloque de 128, 192 6 256 bits. El número de fases depende de los tamaños de bloque y clave.

Las operaciones básicas que usa son: sustitución de byte –permutación no lineal-, desplazamiento de fila –desplazamiento cíclico de bytes-, mezcla de columna –transformación lineal- y suma de clave –en cada ronda una clave es derivada de la principal por medio de la función de reparto de la clave. La longitud de esta clave coincide con la del bloque de encriptación.-

2.2.3.6 SERPENT

Es un cifrado de bloque en 32 fases. Como tamaño de bloque usa 128 bits de texto en claro y produce una salida de bloques de 128 bits de texto cifrado.

Emplea claves de talla variable de hasta 256 bits.

Las operaciones básicas son: OR-EX, rotación de bit y desplazamiento de bits. También usa tablas de permutación con enteros entre 0 y 127 (ambos incluidos).

2.2.3.7 TWOFISH

Fue diseñado por Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall y Niels Ferguson. Es de libre uso.

Se compone de un cifrado en 16 fases que usa bloques de 128 bits y longitud de clave variable con un máximo de 256 bits.

Operaciones primitivas que utiliza son: OR-EX, suma, rotación de bits y multiplicación entera.

2.3 Algoritmos asimétricos (clave secreta)

Hasta ahora hemos visto algoritmos donde la clave para cifrar y descifrar era única. Es decir, ambos –emisor y receptor- necesitan conocer la misma clave. Ahora bien, la clave debe tener un origen único y, a partir de aquí, compartida con la otra parte; para compartirla necesitamos utilizar un medio seguro, el cual, no siempre está disponible...

Los algoritmos asimétricos nacieron para obviar este aspecto. De tal modo que se pensó (Diffie & Hellman 1976) en usar dos claves diferentes: una para encriptar los datos y la otra para desencriptarlos.

Así, a partir de ahora hablaremos de dos tipos de claves: pública y privada. La primera puede ser dada a conocer sin restricciones mientras que la segunda debe permanecer oculta.

Matemáticamente, estos algoritmos se basan en funciones con una alta dificultad de reversibilidad. Sin embargo, esta dificultad disminuye drásticamente conociendo un determinado valor. En inglés: “trapdoor one-way functions”, donde el valor que nos permite revertir la función es el “trapdoor”.

2.3.1 RSA

Los creadores de este algoritmo fueron Ronald Rivest, Adi Shamir y Len Adlemar (RSA) en 1978.

El problema matemático en el que está basado es la descomposición de un número muy grande en números primos.

2.3.2 El Gamal

Este criptosistema fue desarrollado por ElGamal y publicado en 1985.

Está basado en el problema del Logaritmo Discreto en un campo finito.

2.3.3 Criptografía de curva elíptica (ECC)

En 1985 se propusieron las curvas elípticas como un problema típico de aplicación en los criptosistemas.

Han despertado un gran interés porque para la misma longitud de clave que los anteriores, son mucho más difíciles de computar –es decir, de “romper”-

3 IPSec

3.1 Conceptos y definiciones acerca de IPSec

El propósito de la arquitectura IPSec es proporcionar diversos servicios de seguridad para el tráfico de red a nivel IP, tanto en entornos Ipv4 como Ipv6.

Los siguientes son los componentes fundamentales de la arquitectura de seguridad IPSec:

a. Protocolos de Seguridad – Authentication Header (AH) y Encapsulating Security Payload (ESP)
b. Asociaciones Seguras (SA) – qué son y cómo trabajan, cómo se gestionan y su procesamiento asociado.
c. Gestión de claves – manual y automatica (IKE)
d. Algoritmos para autenticación y encriptación

No conviene olvidar que cuando hablamos de seguridad mediante Ipsec siempre nos estamos refiriendo a la seguridad solo a nivel IP, proporcionada a través del uso de una combinación de mecanismos criptográficos y protocolos de seguridad.

3.1.1 Objetivos de IPsec

IPSec ha sido diseñado para proveer seguridad basada en la criptografía, de alta calidad y con características de interoperabilidad para Ipv4 y Ipv6.

El conjunto de servicios de seguridad ofrecidos incluye control de acceso, integridad en sistemas orientados “sin conexión” (datagramas), autenticación del origen de los datos, protección contra replays (una forma de integridad de secuencia parcial), confidencialidad (encriptación), y una confidencialidad limitada sobre el flujo del tráfico.

Estos servicios son proporcionados a nivel IP, ofreciendo protección para el protocolo IP y de niveles superiores.

Estos objetivos se consiguen a través del uso de dos protocolos de seguridad de tráfico y uno de gestión de claves criptográficas:

el Authentication Header (AH),
y el Encapsulating Security Payload (ESP),
y usando protocolos y procedimientos de gestión de claves criptograficas (IKE).

Así, el conjunto de protocolos IPSec empleados en cualquier contexto, y los modos en los que son empleados serán determinados por los requerimientos de seguridad de los: sistemas, usuarios, aplicaciones, y/o sites/organizaciones.

Cuando estos mecanismos son correctamente implementados y llevados a la práctica, no debe advertirse su presencia ni afectar a usuarios, hosts, y, en general, a cualesquiera otros componentes de Internet que no empleen estos mecanismos de seguridad para la protección de su tráfico. Estos mecanismos también son diseñados para ser independientes del algoritmo empleado. Esta modularidad permite la selección de diferentes conjuntos de algoritmos sin que esto afecte al resto de la implementación. Por ejemplo, comunidades de usuarios diferentes pueden seleccionar diferentes conjuntos de algoritmos (a través del mouse) si así lo requieren.

3.1.2 Beneficios y ventajas

3.1.2.1 Beneficios

Como principales beneficios podemos citar:

Servicio	**Mecanismo**
Integridad	Encriptación
Privacidad	Encriptación
Autenticidad	PKI (Public Key Internet)
Protección “anti-replay” (“eavesdropping”)	Uso del número de secuencia

Como beneficios adicionales tenemos:

Una mejora en la seguridad de nuestra conexión a Internet. Así:

Permite un uso en la LAN de direcciones pertenecientes a RFC1918
El dispositivo que gestiona la VPN esconde las direcciones internas.

3.1.2.2 Ventajas

Ventajas del estándar IPSec:

Interoperabilidad- Cualquier dispositivo que cumpla con el estándar IPSec puede conectar con cualquier otro que también lo haga sin importar fabricante, modelo...

Adaptabilidad- El estándar puede ser expandido para incluir nuevos desarrollos sin que lo mismo suponga ningún contratiempo en las implementaciones ya existentes.

3.1.2.3 Recomendación

Así pues, se requerirá el uso de IPSec siempre que tengamos necesidad de una alta encriptación y/o autenticación.

3.1.3 Seguridad del sistema

Esta suite de protocolos y algoritmos asociados por defecto (IPSec) están diseñados para proporcionar seguridad de alta calidad para el tráfico que fluye por Internet.

Muchas veces, la seguridad ofrecida por el uso de estos protocolos depende en última instancia de la calidad de su implementación. Además, hay que tener en cuenta que la seguridad de un sistema computerizado o red depende de otros muchos factores incluyendo las prácticas habituales mantenidas por el personal, los compromisos establecidos, procedimientos, el uso “físico” del sistema.

Resumiendo: IPSec es sólo una parte a tener en cuenta en el aspecto de seguridad de un sistema computerizado.

Por último, la seguridad proporcionada por IPSec es críticamente dependiente del propio sistema operativo en el cual IPSec se ejecuta (seguridad del SO, calidad de la semilla para la obtención de números aleatorios..,) todos ellos pueden degradar el nivel de seguridad proporcionado por IPSec.

3.1.4 Funcionamiento general

Descripción a alto nivel de cómo trabaja IPSec, de los componentes del sistema, y cómo se involucran todos ellos para proporcionar los servicios de seguridad descritos en el punto anterior. Así se consigue tener una idea general del proceso.

Una implementación IPSec opera en un entorno host o gateway de seguridad, ofreciendo seguridad al tráfico de red.

La protección está basada a partir de los requerimientos definidos por la Base de Datos que contiene la política de Seguridad (SPD), establecida y gestionada por un usuario o administrador del sistema o por una aplicación que opera a partir de aquello que ha programado el usuario.

En general, a cada paquete se le aplica, en función de la información contenida en la cabecera del nivel de red y de transporte, y comparada con las entradas de la SPD, una de las tres siguientes opciones: se le aplica el servicio correspondiente IPSec: borrado, bypass o aplicación del protocolo.

3.2 Qué hace IPSec

Provee servicios de seguridad a nivel IP mediante un sistema que selecciona los protocolos de seguridad requeridos, determina qué algoritmos usar para el servicio requerido, y pone en juego las claves criptográficas requeridas para proveer los servicios requeridos. Así, IPSec puede ser usado para proteger uno o más "canales" entre un par de hosts, entre un par de gateways de seguridad (aquellos que implementan los protocolos IPSec de modo que un router o firewall que implementen IPSec pueden ser denominados como "security gateway").

El conjunto de servicios de seguridad que IPSec provee incluye control de acceso, integridad orientada a servicios "sin conexión", autenticación del origen de los datos, rechazo de los paquetes "retransmitidos" (replayed), confidencialidad (encriptación) y confidencialidad limitada del flujo de tráfico. Como estos servicios son provistos a nivel IP, pueden ser usados por cualquier protocolo de nivel superior: TCP, UDP, ICMP, BGP, etc…

El DOI (Domain of Interpretation) de IPSec también soporta la negociación de la compresión IP.

3.3 Cómo trabaja IPSec

IPSec usa dos protocolos para proveer la seguridad del tráfico:

El protocolo AH que provee integridad en comunicaciones "sin conexión", autenticación del origen de los datos y un servicio opcional anti-replay.

El protocolo ESP puede proveer confidencialidad (encriptación), y confidencialidad limitada del flujo de tráfico. Asímismo puede proveer integridad en comunicaciones "sin conexión", autenticación del origen de los datos y servicio anti-replay.

Ambos, AH y ESP, proporcionan: control de acceso basado en la distribución de las claves criptográficas y la gestión del tráfico en relación a estos protocolos de seguridad.

Estos protocolos pueden ser aplicados de forma aislada o conjuntamente para proveer los servicios de seguridad deseados en Ipv4 e Ipv6.

Cada protocolo soporta dos modos de uso:

Modo transporte- En el primero, los protocolos proporcionan protección principalmente para los protocolos de nivel superior.

Modo túnel- Los protocolos son aplicados a paquetes completamente transformados (tunneled).

IPSec permite al usuario (o administrador del sistema) controlar la granularidad a la que se ofrece el servicio de seguridad. Así, se puede crear un sencillo túnel encriptado para gestionar **todo el tráfico** entre dos "security gateways"o un túnel encriptado puede ser creado de forma separada para **cada conexión TCP** entre un par de hosts que se comuniquen a través de los mencionados gateways.

Una gestión adecuada de IPSec puede incorporar los servicios que permitan especificar:

Servicios de seguridad a usar y en qué combinaciones,
la granularidad a la que la protección de seguridad debe ser aplicada y
los algoritmos usados para efectuar la seguridad basada en la criptografía.

Debido a que los servicios de seguridad usados comparten valores secretos (claves criptográficas), IPSec confía en un conjunto separado de mecanismos para manejar estas claves (que son usadas para servicios de autenticación / integridad y de encriptación). Así, se requiere el soporte para una distribución de claves manual y automática. De este modo, se especifica una aproximación basada en clave pública (IKE) para el intercambio de clave automático; sin embargo, otras técnicas automáticas de distribución de claves pueden ser usadas. Por ejemplo, Kerberos, SKIP…

3.4 Entorno de implementación de IPSec

Existen diferentes implementaciones de IPSec en un host o en conjunción con un router o firewall (para crear un gateway de seguridad). Algunos ejemplos comunes:

a. Integración de IPSec en una implementación nativa de IP. Se requiere acceso al código fuente IP y es aplicable a ambos extremos: hosts y gateways de seguridad.

b. BITS ("Bump-in-the-stack").- donde IPSec es implementado "en medio"de una implementación de la pila de protocolos IP ya existente; entre los drivers IP nativos y los de la red local. El acceso al código fuente para la pila IP no es requerido en este contexto, haciendo de esta implementación una aproximación apropiada para

usar con sistemas "legacy" (propietarios??). Cuando se adopta esta aproximación, normalmente ocurre en los hosts.

c. BITW ("Bump-in-the-wire").- Cuando se utiliza un procesador externo encargado de realizar las tareas de encriptación (entornos militares). Esta implementación sirve tanto para un host como para un gateway (o ambos). Normalmente el dispositivo BITW es IP direccionable. Cuando sirve de soporte a un único host es bastante similar a la implementación BITS, pero cuando lo hace para un router o firewall, debe operar como un gateway de seguridad.

3.5 Asociaciones Seguras (SAs)

Este concepto es fundamental en IPSec. Ambos protocolos, AH y ESP, hacen uso de las SAs y una función principal de IKE es su establecimiento y gestión. Cualquier implementación de AH o ESP DEBE soportar el concepto de SA.

3.5.1 Definición y objetivos

Es una conexión unidireccional, necesaria para el establecimiento de los servicios de seguridad en el tráfico que gestiona. Los servicios "seguros" son afrontados por un SA mediante el uso de AH o ESP, pero no ambos. Si ambas protecciones (AH y ESP) son aplicadas al canal de tráfico, entonces dos (o más) SAs se necesitarán para poder afrontar la protección del mencionado canal. Para asegurar la típica comunicación bi-direccional entre dos hosts, o entre dos gateways "seguros", serán necesarias dos SAs (una en cada dirección).

Una SA es unívocamente identificada por una tripla consistente en: un parámetro de seguridad SPI (Security Parameter Index), una dirección destino IP y un identificador del protocolo de seguridad (AH o ESP). En principio, la dirección destino puede ser una "unicast", una dirección IP "broadcast" o un grupo de direcciones "multicast". Sin embargo, los mecanismos de gestión de SA actualmente están definidos solo para unicast.

Como ya se ha comentado, se han definido dos tipos de SAs: modo transporte y modo túnel. Un SA en modo transporte lo es entre dos hosts. En Ipv4, en el protocolo de seguridad en modo transporte una cabecera (header) aparece justo después de la cabecera IP (con sus posibles opciones) y antes de cualquier protocolo de nivel superior (p. ej. TCP o UDP). En Ipv6, la cabecera del protocolo de seguridad aparece después de la cabecera IP (con sus extensiones), pero puede aparecer antes o después de las opciones "destino" y antes de los protocolos de nivel superior. En el caso de ESP, el modo transporte de SA provee servicios seguros solo para aquellos protocolos de nivel superior, no para la cabecera IP o cualquier otra precedente a la del ESP. En el caso de AH, la protección se extiende a las porciones seleccionadas de la cabecera

IP, porciones seleccionadas de la cabecera extendida, y opciones seleccionadas (contenidas en la cabecera Ipv4, cabecera de extensión Hop-by-Hop de Ipv6 o cabeceras de extensión del destino Ipv6).

El modo túnel SA es esencialmente un SA aplicado a un túnel IP. Siempre que un "final" de la SA es un gateway seguro, el SA DEBE estar en modo túnel. Así, un SA entre dos gateways seguros es siempre un SA en modo túnel, como sucede si tenemos un SA entre un host y un gateway seguro. Conviene tener en cuenta el caso en el que el tráfico es explícitamente destinado a un gateway seguro, pej. Comandos SNMP, en este caso el gateway seguro está actuando como un host y el modo transporte está, por lo tanto, permitido. Conviene notar que en este caso, el gateway seguro no está actuando como un verdadero gateway, no está actuando como un elemento de transición. Dos hosts pueden establecer un SA, modo túnel, entre ellos. El requerimiento, para cualquier (tráfico de tránsito) SA involucrado en un gateway seguro, para utilizar el modo túnel surge debido a la necesidad de evitar potenciales problemas respecto a la fragmentación y reensamble de los paquetes IPSec, y en aquellas circunstancias donde múltiples posibles caminos (pej., vía diferentes gateways seguros) existan para el mismo destino tras los gateways seguros.

Para un SA modo túnel, hay una cabecera IP "outer" (externo) que especifica el destino IPSec (siguiente) más una cabecera IP "inner" (interna) que especifica el (aparentemente) destino final para el paquete. La cabecera del protocolo de seguridad aparece después de la cabecera "outer y antes de la "inner". Si AH ha empleado el modo túnel, partes de la cabecera "outer" IP son "protegidas" al igual que todo el paquete IP que ha sido "tunelizado" (pej., todo el "inner" está protegido así como los protocolos de más alto nivel). Si se emplea ESP, la protección es proporcionada solo al paquete "tunelizado, no a la cabecera externa.

Resumiendo:

a)Un host DEBE soportar ambos modos: transporte y túnel.

b)Un gateway seguro basta con que soporte el modo túnel. Si además soporte el modo transporte, sólo debería ser usado cuando actúa como un host (pej., para la gestión de la red

3.5.2 Funcionalidad de las SAs

El conjunto de servicios de seguridad ofrecidos por un SA depende de: el protocolo de seguridad seleccionado, el modo SA, los elementos del SA que se comunican (host y/o gateway), y de los servicios opcionales utilizados con el protocolo. Por ejemplo, AH provee servicio de integridad orientado a "sin conexión" y autenticación del origen para datagramas IP (resumiendo: autenticación). La precisión de este servicio de autenticación está en función de la granularidad de la SA con la que AH se haya empleado, tal y como se comentó en el capítulo dedicado a los *Selectores*.

AH también ofrece un servicio “anti-replay” (integridad de secuencia “parcial”) dependiendo de si se demanda por parte del receptor para ayudar al “contador” que se encarga de evitar los ataques por “denial of service”. AH es un protocolo inapropiado para emplear cuando la confidencialidad no es un MUST (o no es permitida, pej., debido a restricciones del gobierno en el uso de la encriptación). AH también provee autenticación para las porciones seleccionadas de la cabecera IP, lo que puede ser útil y/o necesario en algunos contextos. Así, si la integridad de una opción en Ipv4 o una cabecera de extensión en Ipv6 debe ser protegida entre el emisor y el receptor, AH puede proveer este servicio (salvo para las partes no predecibles pero cambiantes de la cabecera IP).

Opcionalmente, ESP provee confidencialidad para el tráfico. (La bondad del servicio de confidencialidad depende en parte del algoritmo de encriptación empleado). ESP también puede de forma opcional proveer autenticación tal y como se ha mencionado. Si la autenticación es negociada para un ESP SA, el receptor también puede elegir reforzar con el servicio anti-replay que tenga las mismas características que el servicio AH anti-replay. El alcance de la autenticación ofrecida por ESP es menor que el proporcionado por AH, por ejemplo, la cabecera IP “fuera”de la cabecera ESP no está protegida. Si únicamente los protocolos de nivel superior son los que deben ser autenticados, entonces, la autenticación ESP es una elección apropiada y más eficiente que el uso de AH encapsulando ESP. Debe tenerse en cuenta que aunque ambos –confidencialidad y autenticación- son opcionales, ambos no pueden ser omitidos. Al menos uno de ellos debe ser seleccionado.

Si el servicio de confidencialidad es seleccionado, entonces un ESP (modo túnel) SA entre dos gateways seguros puede ofrecer confidencialidad parcial de flujo en el tráfico. El uso del modo túnel permite a las cabeceras IP “inner” ser encriptadas, escondiendo las identidades de (último) el origen y el destino últimos. Además, el contenido del payload puede ser utilizado para esconder la longitud de los paquetes, escondiendo las características externas del tráfico. Servicios similares de control de flujo pueden ser ofrecidos cuando a un usuario móvil –IP móvil- le es asignada una dirección IP dinámica en un contexto dial-up y establece un (modo túnel) ESP SA con un firewall corporativo –que actúa como un gateway seguro-. Debe notarse que una granularidad “fina” en SAs generalmente es más vulnerable al análisis del tráfico que una granularidad “no fina” que lleva tráfico para muchos destinos.

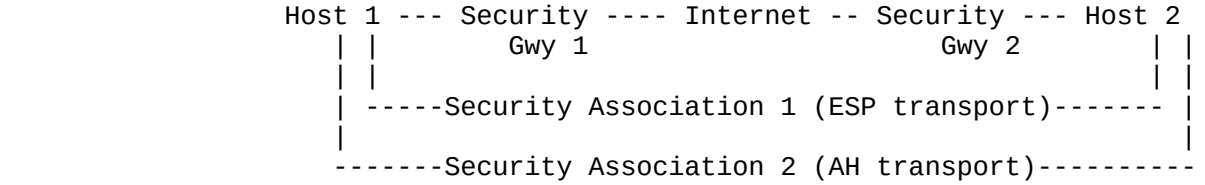
3.5.3 Combinando SAs

Los datagramas transmitidos a través de un SA son protegidos por un único protocolo de seguridad, bien AH bien ESP, pero no ambos. A veces, una política de seguridad puede invocar una combinación de servicios para un flujo particular que no es posible con un único SA. En tales casos será necesario emplear múltiple SAs para implementar la política de seguridad requerida. El término “SA bundle” es aplicado a

una secuencia de SAs a través de las cuales el tráfico debe ser procesado para satisfacer una política de seguridad. El orden de la secuencia es definido por la política. (Debe tenerse en cuenta que los SAs comprendidos en un bundle pueden terminar en diferentes destinos. Por ejemplo, un SA puede comunicar un host dinámico (IP dinámica) con un gateway seguro y un segundo, anidado SA, puede comunicar con un host a través del gateway seguro.

SAs pueden ser combinados en bundles de dos modos: adyacencia de transporte y tunelización iterada.

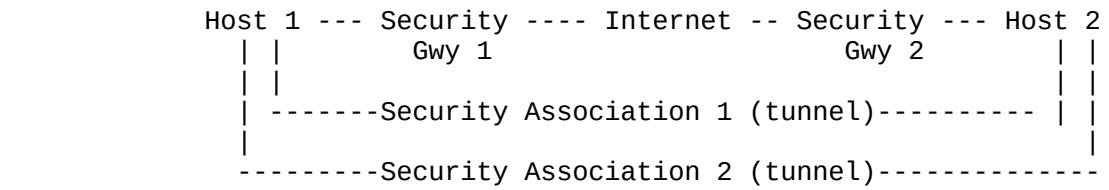
Transport adjacency se refiere a aplicar más de un protocolo de seguridad al mismo datagrama IP, sin invocar tunelizado. Esta aproximación a combinar AH y ESP permite un único nivel de combinación; un mayor nivel de anidamiento no añade mayor beneficio (asumiendo el uso adecuado de algoritmos seguros en cada protocolo) dado que el procesamiento es realizado en una instancia IPSec en el (último) destino.



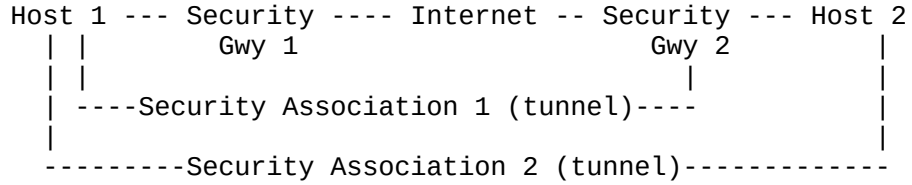
Iterated tunneling se refiere a la aplicación de múltiples niveles de protocolos de seguridad efectuado a través del tunelizado IP. Esta aproximación permite para múltiples niveles de anidamiento, dado que cada túnel puede originar o terminar en un site IPSec diferente durante el camino. No se espera un tratamiento especial para el tráfico ISAKMP en los gateways seguros que se encuentran de manera intermedia además de aquel que sea especificado a través de las entradas SPD apropiadas.

Hay 3 casos básicos de iterated tunneling –se requiere el soporte para los casos 2 y 3:

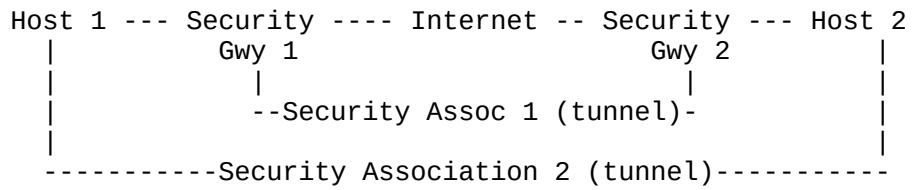
1. Ambas partes del SA son la misma, es decir, o ambos SAs van de host a host o ambos van de GW a GW, pero ninguno va de host a GW –Los túneles inner y outer pueden ser bien AH, bien ESP, si bien es improbable que el Host1 especificara el mismo dos veces, pej., AH dentro de otro AH o ESP dentro de otro ESP.



2. Una parte del SA es la misma; es decir, un SA va de host a – Los túneles inner y outer pueden ser AH o ESP



3. Ninguno de los “endpoints” de ambos SAs coincide –Los túneles inner y outer pueden entonces ser cualquiera AH o ESP



Estas dos aproximaciones pueden también ser combinadas, pej., un bundle SA puede ser construido desde un SA modo túnel y uno o dos SAs modo transporte, aplicados en secuencia. Debe notarse que los túneles anidados pueden también presentarse donde ninguno ni la fuente ni el destino de ninguno de los túneles sea el mismo. En este caso, no habría ni host ni GW seguro con un bundle correspondiente a los túneles anidados.

Para el modo transporte de los SAs, solo parece apropiada una opción en cuanto a los protocolos de seguridad. AH se aplica a ambos: protocolos de nivel superior y (partes de) la cabecera IP. Entonces si AH se usa en modo transporte, in conjunción con ESP, AH debe aparecer como la primera cabecera después de la IP, antes de la aparición de ESP. En este contexto, AH se aplica a la salida cifrada de ESP. Al contrario, para los SAs en modo túnel, se pueden imaginar diferentes opciones de AH y ESP. El conjunto completo de tipos de bundles SA que deben ser soportados por un sistema IPSec compliant se describe en la sección 4.5

3.5.4 Bases de datos con información de las Asociaciones Seguras

Muchos de los detalles asociados con el procesamiento del tráfico IP en una implementación IPSec son casi siempre una cuestión local, no sujeta a estandarización. Sin embargo, algunos aspectos externos del procesamiento deben ser estandarizados para asegurar interoperabilidad y proveer una mínima capacidad de gestión esencial para un uso productivo de IPSec. Esta sección describe un modelo general para procesar el tráfico relativo a las SA de manera que se obtengan las metas de interoperabilidad y funcionalidad. El modelo que se describirá a continuación es nominal; implementaciones IPSec no necesitan seguir cada detalle de este modelo,

pero si el comportamiento externo de tales implementaciones debe ser equivalente a las observables en el modelo.

Hay dos bases de datos nominales en este modelo: la base de datos de la política de seguridad (SPD) y la base de datos de la Asociación de Seguridad (SA). La primera especifica las políticas que determinan la disposición de todo el tráfico de entrada y salida en una implementación IPSec desde un host, GW seguro o BITS o BITW. La segunda base de datos contiene los parámetros asociados con cada SA (activa). Esta sección también define el concepto de: Selector, conjunto de valores de campos del protocolo IP y superiores que es usado por la SPD para mapear el tráfico a una política, pej., una SA (o SA bundle).

Cada interfase para el que IPSec esta habilitado requiere de forma nominal bases de datos separadas de entrada y salida (SAD y SPD), debido a la direccionalidad de muchos de los campos que son usados como selectores.

Típicamente hay un único interfase, para un host o GW seguro SG. Se debe tener en cuenta que un SG debería tener siempre al menos 2 interfaces, pero el “interno“ a la red corporativa normalmente no tendría habilitado el IPSec por lo tanto se necesitarían un par de SADs y un par de SPDs. Por otro lado, si un host tuviera múltiples interfaces o un SG tuviera mas de una interfase externa, pudiera ser necesario tener pares separados de SAD y SPD para cada interfase.

3.6 La base de datos de la política de seguridad (SPD)

Así pues, una SA es un objeto de gestión usado para reforzar una política de seguridad en un entorno IPSec. De este modo, un elemento esencial del procesamiento SA es un inherente SPD que especifica que servicios son ofrecidos a los datagramas IP y en que modo. La forma de la base de datos y su interfase caen fuera del objetivo de esta especificación. Sin embargo, esta sección especifica ciertos aspectos mínimos de funcionalidad que deben ser proporcionados para permitir a un usuario o administrador del sistema controlar como IPSec se aplica al tráfico transmitido o recibido por un host o retransmitido por un SG.

El SPD debe ser consultado durante el procesamiento de todo el tráfico (ENTRADA y SALIDA), incluyendo el tráfico no-IPSec. Para poder realizar lo anterior, el SPD requiere distintas entradas para el tráfico de entrada y salida. Se puede pensar en esto como SPDs separadas entrada vs. Salida.) Además, un SPD separado debe ser proporcionado para cada interfase IPSec-habilitado.

Un SPD debe discriminar entre el tráfico al que se le proporciona protección IPSec y el tráfico al que se le aplica el bypass IPSec. Esto sirve tanto para la protección IPSec que es aplicado por el emisor como para la protección IPSec que debe estar presente en el receptor. Para cualquier datagrama de entrada o salida se presentan tres posibles

actuaciones: borrado, bypass IPSec o aplicar IPSec. La primera se refiere al tráfico que no se permite de ningún modo salir del host, atravesar el SG o ser entregada a una aplicación. La segunda trata con el tráfico al que se permite pasar sin ninguna protección adicional por parte de IPSec. La tercera y ultima se refiere al tráfico que esta bajo la protección IPSec y para el mismo el SPD debe especificar los servicios de seguridad que deben ser proporcionados, protocolos a emplear, algoritmos a utilizar, etc.

Para cada implementación IPSec DEBE existir una interfase administrativo que permita a un usuario o administrador del sistema gestionar el SPD. Específicamente, cada paquete de entrada o salida esta sujeto al procesamiento por IPSec y el SPD debe especificar que acción Serra tomada en cada caso. Así la interfase administrativo debe permitir al usuario (o administrador del sistema) especificar el procesamiento de seguridad a aplicar a cualquier paquete que entre o salga del sistema, y esto debe hacerse para cada paquete. (En una implementación de IPSec para un host que haga uso de un interfase socket, puede que el SPD no necesite ser consultado para cada paquete, pero el efecto debe ser el mismo. La interfase de gestión para el SPD DEBE permitir la creación de entradas consistente con los selectores y DEBE soportar de forma total la petición de tales entradas. Se espera que a través del uso de wildcards en diversos campos selectores y a causa de que todos los paquetes en un a conexión UDP o TCP simple tendera a coincidir con una única entrada del SPD, este requerimiento no impone un excesivo nivel de detalle en la especificación del SPD. Los selectores son análogos a lo que se encuentra en un firewall sin gestión de estados (stateless) o un router de filtrado y que actualmente se controlan de este modo.

En los hosts, se DEBE permitir que las aplicaciones seleccionen que procesamiento de seguridad se va a aplicar al tráfico que generan y consumen. (El porque de especificar tales requerimientos en la implementación IPSec esta fuera del objetivo de este estándar. Sin embargo, el administrador del sistema DEBE ser capaz de especificar si un usuario o aplicación puede “override” (default) las políticas del sistema. Nótese que las políticas especificadas para una aplicación pueden satisfacer los requerimientos del sistema, de modo que el sistema no necesitaría en este caso un procesamiento adicional a nivel IPSec mas allá del que la propia aplicación necesita para cumplir con sus requerimientos. La forma de la interfase de gestión en este documento y puede diferir para hosts vs. SGs y, dentro de los hosts, puede diferir entre las implementaciónes basadas en socket vs. Implementaciónes BITS. Sin embargo, este documento especifica un conjunto estándar de elementos SPD que todas las implementaciónes IPSec DEBEN soportar.

El SPD contiene una lista ordenada de políticas. Cada entrada es una política “indexada” o apuntada por uno o más selectores que definen el conjunto de tráfico IP al que le afecta esta política. Estos definen la granularidad de las políticas o SAs. Cada entrada incluye una indicación de si el tráfico al que se le aplica esta política será ignorado (bypassed), borrado o sujeto al procesamiento IPSec. Si se aplica el procesamiento IPSec, la entrada incluye una especificación SA (o SA bundle),

detallando protocolos IPSec, modos y algoritmos a emplear incluyendo cualquier requerimiento de anidado. Por ejemplo, una entrada puede decir que todo el tráfico al que se le aplique la misma de be ser protegido por ESP en modo transporte usando 3DES-CBC con un anidado explicito IV dentro de AH en modo túnel usando HMAC/SHA-1. Para cada selector, la entrada especifica como derivar los valores correspondientes para una nueva entrada SAD de aquellas que hay en el SPD y el paquete (Nótese que en este momento, los rangos son solamente soportados para direcciones IP; pero "wildcarding" puede ser expresado para todos los selectores):

a. Usar el propio valor del paquete – De este modo se limitara el uso del SA a aquellos paquetes que tienen este valor de paquete para el selector incluso si el selector para la política tiene un rango de valores validos o existe un "wildcard" para este selector.
b. Usar el valor asociado con la política a aplicar – Si debiésemos obtener únicamente un valor entonces no habría diferencia entre (b) y (a. Sin embargo, si los valores permitidos para el selector están en un rango (para direcciones IP) o wildcard, entonces en el caso de un rango, (b) habilitaria el uso del SA para cualquier paquete con un valor de selector del rango y no habilitaria el uso de este SA para aquellos paquetes que únicamente tienen el valor del selector del paquete coincidente con la creación del SA. En el caso de un "wildcard", (b) permitiría el uso del SA para paquetes con cualquier valor para este selector.

Por ejemplo, supongamos que hay una entrada PSD donde el valor permitido como dirección fuente es cualquier en el rango de hosts 192.168.2.1 a 192.68.2.10. Y supongamos que un paquete que se va a enviar tiene una dirección fuente de 192.168.2.3. El valor usado por el SA podría ser cualquier de los valores que se muestran a continuación dependiendo de que la política a aplicar para este selector diga si se usa el valor de selector o el valor del origen IP:

Origen para el ejemplo del valor para la nueva SAD usado en el valor de selector SA:

```
            source for the  example of
            value to be     new SAD
            used in the SA  selector value
            --------------- ------------
            a. packet       192.168.2.3 (one host)
            b. SPD entry    192.168.2.1 to 192.168.2.10 (range of
hosts)
```

Nótese que si el SPD tuviera un valor valido de "wildcard" para la dirección fuente, entonces el valor para el selector SAD seria "wildcard" (cualquier host.)El caso (a) puede ser usado para prohibir comparticion, incluso entre paquetes que usen el mismo SPD.

Como se describirá posteriormente, los selectores pueden incluir una entrada "wildcard" y entonces puede que los selectores para dos entradas sean el mismo. (Es análogo al solapamiento que se encuentra en los ACLs o entradas del filtro en router o filtrado de paquetes en firewalls). Entonces, para asegurar consistencia, procesamiento predecible, las entradas del SPD DEBEN estar ordenadas y el SPD DEBE siempre ser analizado en el mismo orden, de forma que la primera entrada coincidente sea seleccionada de forma consistente. (Este requerimiento es necesario como efecto de que el procesamiento de tráfico contra las entradas SPD debe ser determinístico, pero no hay modo de situar específicamente las entradas SPD debido al uso de "wildcards" para algunos selectores. Un mayor detalle acerca del matching de paquetes contra entradas SPD será proporcionado más adelante.

Nótese que si se especifica ESP, autenticación o encriptación (pero no ambos) puede ser omitido. Así, DEBE ser posible configurar el valor SPD "NULL" para los algoritmos de encriptación o autenticación. Sin embargo, al menos uno de estos servicios DEBE ser seleccionado, pej., NO DEBE ser posible configurar ambos como "NULL".

El SPD puede ser usado para mapear tráfico para especificar SAs o SA bundles. De este modo puede funcionar de dos maneras a la vez: como la base de datos de referencia para la política de seguridad y como el mapa para las SAs existentes (o SA bundles). (Para acomodar las políticas de borrado y bypass citadas anteriormente, el SPD también DEBE proporcionar un medio de mapear el tráfico a estas funciones, incluso aunque no pertenecen, per se, al procesamiento IPSec). El modo en el que el SPD opera es diferente para el tráfico de entrada que para el de salida y puede diferir para una implementación en un host vs. SG, BITS, y BITW.

Debido a que una política de seguridad puede requerir de mas de un SA para un conjunto especificado de tráfico, en un orden especifico, la política en el SPD debe preservar este requerimiento de orden, cuando se encuentren presentes. Así, debe ser posible para una implementación IPSec el determinar que un paquete de entrada o salida deba ser procesado a través de una secuencia de SAs. Conceptualmente, para el procesamiento de salida, nos podemos imaginar links (al SAD) desde una entrada en el SPD para la que hay SAs activas, y cada entrada consistiría de ambas: una unica SA o una lista ordenada de SAs que comprende un SA bundle. Cuando un paquete es coincidente con una entrada SPD y hay una SA o SA bundle existente que puede ser usada para llevar el tráfico, el procesamiento del paquete es controlado por la entrada SA o SA bundle en la lista. Para un paquete de entrada IPSec para el que múltiples SAs IPSEC serán aplicadas, el lookup basado en la dirección destino, el protocolo IPSec y el SPI identificaran un único SA.

El SPD es usado para controlar el flujo de TODO el tráfico a través de un sistema IPSec, incluyendo tráfico de seguridad y de gestión de claves (pej. , ISAKMP) de/a entidades detrás de un SG. Esto significa que el tráfico ISAKMP debe ser explícitamente tenido en cuenta en el SPD, sino será borrado. Nótese que un SG

puede prohibir el paso de paquetes encriptado de varias formas, pej., teniendo una entrada DISCARD en el SPD para paquetes ESP o proporcionando intercambio de claves proxy. En el ultimo caso, el tráfico seria internamente enrutado al modulo de gestión de claves en el SG.

3.7 Selectores

Un SA (o SA bundle) PUEDE SER fine-grained o coarse-grained, dependiendo de los selectores usados para definir el conjunto del tráfico para el SA. Por ejemplo, todo el tráfico entre dos hosts puede ser transportado a través de un único SA, y establecido un conjunto uniforme de servicios de seguridad. Alternativamente, el tráfico entre un par de hosts puede ser aplicado a múltiples SAs, dependiendo de las aplicaciones usadas (tal y como se define en el Siguiente Protocolo y campos de Port), con diferentes servicios de seguridad ofrecidos por diferentes SAs. De forma similar, todo el tráfico entre un par de SG puede ser llevado a cabo con un único SA, o un SA puede ser asígnado para cada par de hosts comunicando. Los siguientes parámetros de selector DEBEN ser soportados para la gestión del SA de forma que se facilite el control de la granularidad del SA. Nótese que en el caso de recibir un paquete con una cabecera ESP, pej., en una implementación SG encapsulado o BITW, el protocolo del nivel de transporte, puertos origen/destino, y Nombre (si se menciona) puede ser "OPACO", pej., inaccesible debido a la encriptación o fragmentación. Nótese también que ambas direcciones: Fuente y Destino deberían ser o Ipv4 o Ipv6.

- Dirección IP destino (Ipv4 o Ipv6): puese ser una unica dirección IP (unicast, anycast, broadcast (Ipv4 solo), o grupo multicast), un rango de direcciones (valores altos y bajos (inclusive), dirección + mask, o una dirección wildcard. Las ultimas tres son usadas para soportar mas de un sistema destino compartiendo el mismo SA (pej., detrás de un SG). Notese que este selector es conceptualmente diferente del campo "Destination IP Address" en la tupla (Destination IP Address, IPSec Protocol, SPI) usada para identificar un SA de forma unica. Cuando un paquete "tunelizado" llega al extremo del tunel, su SPI/Destino dirección/protocolo son usados para mirar el SA para este paquete en el SAD. Esta dirección destino viene de la cabecera IP encapsulada. Una vez que el paquete ha sido procesado de acuerdo al tunel SA y ha salido del tunel, sus selectores son "looked up" en el SPD de entrada. El SPD de entrada tiene un selector llamado dirección destino. Esta dirección destino IP es justo la de la cabecera IP inner (encapsulada). En el caso de un paquete transmitido en modo "transporte", existira solo una cabecera IP y, por lo tanto, desaparece la ambigüedad. (REQUERIDA para todas las implementaciónes)

- Dirección-es IP fuente (Ipv4 o Ipv6): esta puede ser una dirección unica (unicast, anycast, broadcast (Ipv4 solo), o grupo multicast), rango de direcciones (valores altos y bajos inclusive), dirección + mask, o una

dirección wildcard. Las ultimas tres sonusadas para soportar mas de un sistema origen compartiendo la misma SA (pej., detrás de un SG o en un host multihomed). (REQUERIDA para todas las implementaciónes)

- Name: Hay dos casos (Notese que estas formas de nombres están soportadas en el DOI IPSec.)

    - ID de usuario
        - Una cadena de nombre de usuario completamente calificado (DNS), pej., mozartarrobafoo.bar.com
        - Nombre distinguido X.500, pej., C = US, SP = MA, O = GTE Internetworking, CN = Stephen T. Kent.
    - Nombre del sistema (host, SG, etc.)
        - Un nombre DN completamente calificado, pej., foo.bar.com
        - Nombre distinguido X.500
        - Nombre general X.500

      NOTA: Uno de los posibles valores de este selector es “OPACO”.
      (REQUERIDA para los siguientes casos. Nótese que el soporte para nombres diferentes de direcciones no es requerido para Sas con clave gestionada manualmente.
    - ID de usuario
        - Implementaciones host nativas
        - Implementaciones BITW y BITS actuando como HOSTS con solo un usuario
        - Implementaciones SG para procesamiento de ENTRADA
    - Nombres de sistema – todas las implementaciones)

- Nivel de sensitividad de los datos: (etiquetas IPSO/CIPSO)
  (REQUERIDA para todos los sistemas que proporcionan seguridad en el flujo de la información como se cuenta en la sección 8, OPCIONAL para el resto de sistemas.)

- Protocolo de Transporte: obtenido del campo “Protocol” (Ipv4) o del campo “Next Header” (Ipv6). Puede ser un numero de protocolo individual. Estos campos de paquete pueden no contener el Protocolo de Transporte debido a la presencia de las cabeceras IP, pej., una cabecera de enrutamiento, opciones Hop-by-hop, etc. Nótese que el Protocolo de Transporte puede no estar disponible en el caso de recepción de un paquete con una cabecera ESP, entonces DEBERIA soportarse la existencia del valor “OPACO”.
  (REQUERIDA para todas las implementaciones)

  NOTA: Para localizar el protocolo de transporte, un sistema tiene que buscar a través de las cabeceras de paquete chequeando el campo

"Protocol" o "Next Header" hasta que encuentre uno que reconozca como un protocolo de transporte o hasta que encuentre uno que no esta en su lista de cabeceras de extensión, o hasta que encuentre una cabecera ESP lo que significa que tenemos un protocolo de transporte opaco.

- Puertos origen y destino (p.ej., TCP/UDP): Pueden ser valores de puerto UDP o TCP individuales o un puerto wildcard. (El uso del campo Next Protocol y los campos Puerto origen y/o destino (en conjunción con los campos de dirección origen y/o destino), como un selector SA es a veces documentado como un "session-oriented keying".) Notese que los puertos origen y destino pueden no estar disponibles en el caso de recepción de un paquete con una cabecera ESP, entonces DEBERIA soportarse la existencia del valor "OPACO".

  La siguiente tabla resume la relacion entre el valor "Next Header" en el paquete y SPD y el valor selector de puerto derivado para el SPD y SAD.

Next Hdr in Packet	Transport Layer Protocol in SPD	Derived Port Selector Field Value in SPD and SAD
ESP	ESP or ANY	ANY (i.e., don't look at it)
-don't care-	ANY	ANY (i.e., don't look at it)
specific value fragment	specific value	NOT ANY (i.e., drop packet)
specific value not fragment	specific value	actual port selector field

Si el paquete ha sido fragmentado, entonces la información del puerto puede no estar disponible en el fragmento actual. Si esto ocurre, borrar el fragmento. Un PMTU ICMP deberia ser enviado por el primer fragmento, que tendra la información del puerto. (PUEDE ser soportado).

El contexto de la implementación IPSec determina como se usan los selectores. Por ejemplo, una implementación host integrada en la pila puede hacer uso de una interfase socket. Cuando una nueva conexión es establecida el SPD puede ser consultado y un SA (o SA bundle) lleva al socket. Entonces, el tráfico enviado via dicho socket no necesita obtener resultados en adicionales búsquedas en el SPD/SAD. En contraste, una implementación BITS, BITW, o SG necesita mirar cada paquete y realizar una búsqueda SPD/SAD basada en los selectores. Los valores permisibles para los campos del selector difieren entre el flujo de tráfico, la SA y la política de seguridad.

La siguiente tabla resume los tipos de entradas que uno necesita para poder enlazar con el SPD y el SAD. Esta tabla muestra como están relacionados con los campos en el tráfico de datos que están sujetos al muestreo IPSec. (Nota: la entrada "wild" o "wildcard" para direcciones src y dst incluye una mask, rango, etc.)

Field	Traffic Value	SAD Entry	SPD Entry

src addr	single IP addr	single,range,wild	single,range,wildcard
dst addr	single IP addr	single,range,wild	single,range,wildcard
xpt protocol*	xpt protocol	single,wildcard	single,wildcard
src port*	single src port	single,wildcard	single,wildcard
dst port*	single dst port	single,wildcard	single,wildcard
user id*	single user id	single,wildcard	single,wildcard
sec. labels	single value	single,wildcard	single,wildcard

Las entradas SAD y SPD para estos campos pueden ser “OPACO” ya que el valor en el tráfico esta encriptado.

NOTA: En principio, se pueden tener selectores y/o valores de selectores en el SPD que no pueden ser negociados para un SA o SA bundle.

Ejemplos pueden incluir valores de selector usados para seleccionar tráfico para ser borrado o listas enumeradas que originan la creación de un SA diferente para cada item en la lista. Es aceptable tener una interfase administrativa que soporte el uso de valores de selector que no puedan ser negociados para que no confunda al usuario en la creencia de que esta creando un SA con estos valores de selector. Por ejemplo, la interfase puede permitir al usuario especificar una lista enumerada de valores pero resultaría en la creación de distintas políticas y Sas para cada item en la lista.

3.8 Base de datos de la Asociación de Seguridad (SAD)

En cada implementación hay una SAD nominal en la que cada entrada define los parámetros asociados con un SA. Cada SA tiene una entrada en el SAD. Para el procesamiento de salida, las entradas están apuntadas por entradas en el SPD. Nótese que si una entrada en el SPD, en un momento dado, no apunta a un SA apropiado para el paquete, la implementación crea un SA (o SA bundle) ad-hoc y enlaza la entrada del SPD con la entrada en el SAD. Para el procesamiento de entrada, cad entrada en el SAD esta indexada por: una dirección IP dst, el tipo de protocolo IPSec y el SPI. Los siguientes parámetros están asociados con todas las entradas en el SAD. Esta descripción no intenta ser un MIB, sino una especificación de los mínimos items de datos requeridos para implementar un SA en un entorno IPSec.

Para procesamiento de entrada: Los siguientes campos de paquete son usados para buscar la SA en el SAD:

> Dirección IP destino de la cabecera outer: la dirección dst Ipv4 o Ipv6.
> (REQUERIDO para todas las implementaciónes)
> Protocolo IPSec: AH o ESP, usado comoun indice para localizar el SA en esta base de datos. Especifica el protocolo IPSec que debe ser aplicado al tráfico en esta SA.
> (REQUERIDO para todas las implementaciónes)

SPI: el valor de 32 bits usado para distinguir entre diferentes Sas que llevan al mismo destino y usan el mismo protocolo IPSec.
(REQUERIDO para todas las implementaciones)

Para cada uno de los selectores, la entrada SA en el SAD DEBE contener el valor o valores que fueron negociados en el momento de la creación del SA. Para el emisor, estos valores son usados para decidir si, dado un SA, es apropiado para usar con un paquete de salida. Esto forma parte del chequeo para comprobar si existe un SA que pueda ser usado. Para el receptor, estos valores son usados para chequear que los valores del selector en un paquete de entrada match aquellos para el SA (y, por lo tanto, indirectamente aquellos para la política de match). Para el receptor, esto forma parte de la verificación de si el SA era apropiado para este paquete. Estos campos pueden tener la forma de valores específicos, rangos, wildcards o "OPACO" como se describió con anterioridad en la sección *Selectores.* Nótese que para un SA ESP, el algoritmo de encriptación o el algoritmo de autenticación puede ser "NULL". Sin embargo no pueden ser ambos "NULL".

Los siguientes campos SAD se en el procesamiento IPSec:

Sequence Number Counter: un valor de 32 bits usado para generar el campo Sequence Number en las cabeceras AH o ESP.
(REQUERIDO para todas las implementaciónes, pero usado solo para el tráfico de salida).

Séquense Counter Overflow: un flag que indica si existe overflow del contador del numero de secuencia. En ese caso debe generar un evento auditable y prevenir la transmisión de subsecuentes paquetes en dicho SA.
(REQUERIDO para todas las implementaciónes, pero usado solo el tráfico de salida).

Anti-Replay Window: un contador de 32 bits y un bit-map (o equivalente) usado para determinar si un paquete AH o ESP es un replay.
(REQUERIDO para todas las implementaciónes pero usado solo para el tráfico de entrada. NOTA: si se ha deshabilitado el servicio anti-replay por el receptor, p. Ej., en el caso de una gestión manual de la clave SA, entonces la ventana anti-replay no es usada.)

AH algoritmo de autenticación, claves, etc.
ESP, algoritmo de encriptación, claves, IV modo, IV, etc.
(REQUERIDO para implementaciónes ESP)

ESP, algoritomo de autenticación, claves, etc. Si el servicio de autenticación no ha sido seleccionado, este campo será nulo (null).
(REQUERIDO para implementaciónes ESP)

El tiempo de vida de una SA: intervalo de tiempo desde que un SA debe ser reemplazado por un nuevo SA (y un nuevo SPI) o bien debe ser terminado, más una indicación de cual de estas acciones debe ocurrir. Puede ser expresado como una cuenta en bytes o en tiempo o un uso simultáneo de ambos, en función del primer tiempo de vida que expire. Una implementación compatible con IPSec DEBE permitir ambos tipos de tiempos de vida, y debe permitir un uso simultáneo de ambos. Si el tiempo es empleado y IKE usa certificados X.509 para el establecimiento del SA, el tiempo de vida del SA está constrenyido por los tiempos de validez de los certificados, y el NextIssueDate de los CRLs usados en el intercambio IKE para el SA. Ambos, el iniciador y el que responde son responsables para elegir el tiempo de vida del SA teniendo en cuenta el factor certificados.
(REQUERIDO para todas las implementaciónes)

NOTA: Los detalles de cómo manejar el refresco de las claves cuando expiran los Sas es una cuestión de implementación local. Sin embargo, una aproximación razonable sería:

a) Si se usa la cuenta en bytes, entonces la implementación DEBE contar el número de bytes a los que se aplica el algoritmo IPSec. Para ESP, este es el algoritmo de encriptación (incluyendo encriptación Null) y para AH, este es el algoritmo de autenticación. Esto incluye los pad bytes, etc. Nótese que las implementaciónes DEBEN ser capaces de manejar los contadores cuando llegan al final de la cuenta de una SA que está fuera de sincronismo, pej., a causa de un paquete perdido o porque las implementaciónes en ambos lados de la comunicación del SA no están haciendo las cosas del mismo modo.
b) DEBERÍAN existir dos tipos de tiempo de vida – un tiempo de vida soft que avisa a la implementación el comienzo de una acción tal como establecer el reemplazo de un SA y un tiempo de vida hard cuando el SA actual finalice.
c) Si el paquete no es entregado después de que finalice el tiempo de vida del SA entoces DEBERÍA ser borrado.

Modos del protocolo IPSec: túnel, transporte o wildcard.

Indica qué modo de AH o ESP se aplica al tráfico en el SA. Nótese que si este campo es “wildcard” en la parte emisora del SA entoces la aplicación tienes que especificar el modo de la implementación IPSec. Este uso de wildcard permite al miso SA ser usado con ambos modos de tráfico: túnel o transporte con un tratamiento paquete a paquete, pej., por diferentes sockets. El receptor no necesita conocer el modo para poder tratar correctamente las cabeceras IPSec de los paquetes.
(REQUERIDO del siguiente modo: a menos que sea implícitamente definido por el contexto:

-implementaciones host deben permitir ambos modos
-implementaciones GW deben permitir modo túnel

NOTA: El uso de wildcard para el modo de protocolo de un SA de entrada puede añadir complejidad a la situación en la parte receptora (sólo en el caso de host). Dado que los paquetes en un SA de este tipo pueden ser entregados en cualquier modo (túnel o transporte), la seguridad de un paquete de entrada puede depender en parte del modo que haya sido utilizado para su entrega. Si, como resultado de lo anterior, una aplicación necesita conocer el modo SA de un paquete, entonces la aplicación necesitaría un mecanismo para obtener la información de qué modo se está utilizando.

Path MTU: cualquier camino MTU observado y variables de tiempo.
(REQUERIDO para todas las implementaciones pero usado solo para tráfico de salida)

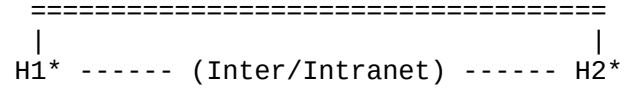
3.9 Combinaciones Básicas de las Asociaciones Seguras

Esta sección describe cuatro ejemplos de combinaciones de SA que DEBEN ser permitidas por cualquier entorno IPSec (host o SG). Combinaciones adicionales de AH y/o ESP en modo túnel y/o transporte PUEDEN ser soportadas a discreción del desarrollador. Las implementaciones compatibles IPSec DEBEN ser capaces de generar estas cuatro combinaciones y en cuanto al receptor, DEBERÍA ser capaz de recibir y procesar cualquier combinación. Los diagramas y texto que siguen a continuación describen los casos base. La leyenda para los diagramas es:

==== = una o más SA (AH o ESP, transporte o túnel)
--- = conectividad (o si es así etiquetada, administrative boundary)
Hx = host x
SGx = security gateway x
X* = X soporta IPSec

NOTA: Las SA que siguen a continuación pueden utilizar bien AH bien ESP. El modo (túnel vs. Transporte) está determinado por la naturaleza de ambos extremos. Para Sas host-to-host, el modo puede ser cualquiera: transporte o túnel.

Caso1. El caso de proporcionar seguridad end-to-end entre 2 hosts a través de Internet (o una Intranet).

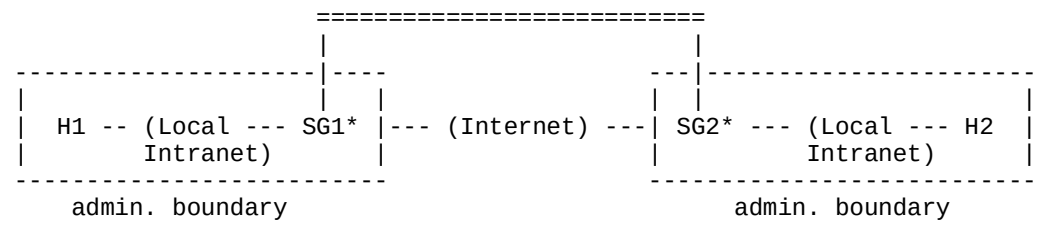


Nótese que ambos modos (túnel o transporte) pueden ser seleccionados por los hosts. Así pues, las cabeceras en un paquete entre H1 y H2 pueden realizar las siguientes propuestas para establecer el SA:

```
Transport                          Tunnel
-----------------                  --------------------
1. [IP1][AH][upper]                4. [IP2][AH][IP1][upper]
2. [IP1][ESP][upper]               5. [IP2][ESP][IP1][upper]
3. [IP1][AH][ESP][upper]
```

Nótese que no existe ningún requerimiento para soporta el anidamiento de forma general, pero en el modo transporte, ambos (AH y ESP) pueden ser aplicados al paquete. En este caso, el procedimiento de establecimiento del SA DEBE asegurar que primero ESP y después AH son aplicados al paquete.

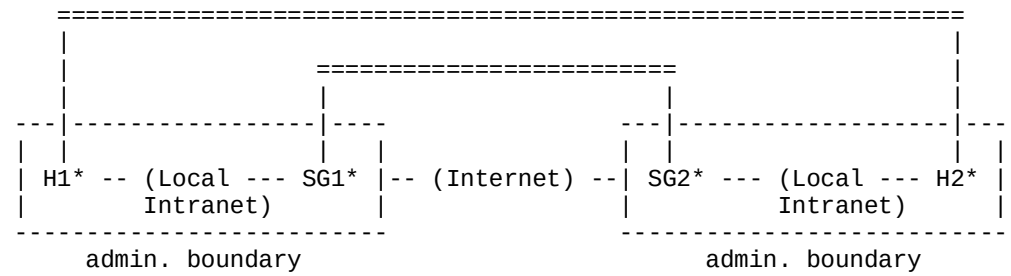
Caso2. Este caso muestra un soporte sencillo de redes privadas virtuales.



Aquí, solo el modo túnel es requerido. Así, las cabeceras de un paquete entre SG1 y SG2 pueden ser como cualquiera de las que siguen a continuación:

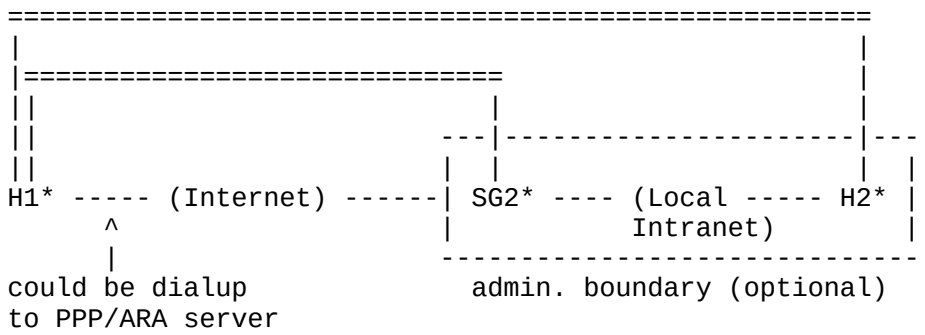
Tunnel
---
4. [IP2][AH][IP1][upper]
4. [IP2][ESP][IP1][upper]

Caso 3. Este caso combina los casos 1 y 2, añadiendo seguridad end-to-end entre los hosts emisor y receptor. No impone ningún nuevo requerimiento ni a los hosts ni a los SGs distinto del requerimiento para un SG que debe ser configurable para que permita el paso del tráfico IPSec (incluyendo el tráfico ISAKMP) hacia los hosts que están detrás de él (que protege).



Caso 4. Este caso cubre la situación donde un host remoto (H1) usa Internet para alcanzar un firewall de una organización (SG2) para poder acceder a algún servidor o cualquier otra máquina tras el firewall (H2). El host remoto puede ser un host móvil

(H1) que conecta mediante dial up a un servidor local PPP/ARA (no se muestra en el gráfico) en Internet y después cruza Internet hasta llegar al firewall de la organización (SG2), etc.



En este caso solo el modo túnel es requerido entre H1 y SG2. Así, las opciones para el SA entre H1 y SG2 podría ser alguna de las del caso 2. Las opciones para el SA entre H1 y H2 podría ser cualquiera de las del caso 1.

Nótese que en este caso, el emisor DEBE aplicar la cabecera transporte antes de la cabecera túnel. Asimismo, la interfase que gestiona la implementación IPSec DEBE soportar la configuración del SPD y del SAD para asegurar este orden de aplicación de la cabecera IPSec.

Como se mencionó anteriormente, el soporte para combinaciones adicionales de AH y ESP es opcional. El uso de cualquier otra combinación opcional puede, según casos, afectar a la interoperabilidad.

3.10 Gestión de claves y de SAs

IPSec establece como mandato el soportar establecimiento del SA de modo manual y automático así como la gestión de claves criptográficas. Los protocolos IPSec, AH y ESP, son independientes en gran medida de las técnicas de gestión asociadas al SA, aunque dichas técnicas afectan a algunos de los servicios ofrecidos por los protocolos. Por ejemplo, los servicios opcionales anti-replay disponibles para AH y ESP requieren una gestión automática del SA. Además, la granularidad empleada en la distribución de las claves con IPSec determina la granularidad de la autenticación. En general, la autenticación del origen de los datos en AH y en ESP está limitada por cómo el tipo de secreto utilizado con los algoritmos de autenticación (o con el protocolo de gestión de la clave que crea tales secretos) es compartido entre múltiples posibles orígenes.

El siguiente texto describe los requerimientos mínimos para ambos tipos de gestión del SA.

3.10.1 Técnicas manuales

La forma más sencilla de gestión es la gestión manual en la que una persona configura manualmente cada sistema con datos referentes a la clave y a la gestión del SA relevantes para obtener una comunicación segura con otros sistemas. Las técnicas manuales son prácticas en entornos pequeños, estáticos donde la escalabilidad no es un factor demasiado importante. Por ejemplo, una compañía puede crear una VPN usando IPSec en SG en algunos sites. Si el número de sites es pequeño y dado que todos los sites están bajo el control de un único dominio administrativo, este modo se ajusta para un uso de las técnicas manuales. En este caso, el SG debe proteger el tráfico de manera selectiva a/desde otros sites que pertenezcan a la organización usando una clave configurada manualmente y, al mismo tiempo, no debe proteger tráfico para otros destinos. También puede ser adecuado cuando solo un determinado tipo de comunicaciones necesitan ser protegidas. Un argumento similar se puede aplicar al uso de IPSec dentro de una organización para un determinado (pequeño) número de hosts y/o gateways. Estas técnicas a menudo emplean claves simétricas, configuradas de forma estática, a pesar de existir otras alternativas.

3.10.2 Gestión automática de claves y SAs

Dada la amplia difusión y uso de IPSec se requiere un protocolo de gestión de SA automatizado, escalable y estándar (soporte de Internet). Tal soporte es necesario para facilitar el uso de características anti-replay de AH y ESP, y para acomodar la creación bajo demanda de Sas, pej., para gestión de claves orientada a sesión y orientada a usuario. (Nótese que la noción de “rekeying” un SA normalmente implica la creación de un nuevo SA con un nuevo SPI, un proceso que generalmente implica a su vez el uso de un protocolo automático de gestión SA/clave.)

El protocolo de gestión de claves automático por defecto para su uso con IPSec es IKE bajo el dominio de interpretación IPSec. Otros protocolos de gestión de SA automáticos PUEDEN ser empleados.

Cuando un protocolo automático de gestión SA/clave es empleado, la salida de este protocolo puede ser usada para generar múltiples claves, pej., para un único ESP SA. Esto puede cobrar sentido debido a:

El algoritmo de encriptación usa múltiples claves (pej., triple DES)
El algoritmo de autenticación usa múltiples claves
Ambos algoritmos (autenticación y encriptación) son empleados

El sistema de gestión de claves provee una cadena separada de bits para cada clave o puede genera una cadena de bits de los cuales se deben extraer las mismas. Si se proporciona una única cadena de bits, se debe tener en cuenta el asegurar que las partes del sistema que mapean la cadena de bits para las calves requeridas lo hacen del mismo modo en ambos extremos del SA. Para asegurar que las implementaciones

IPSec en cada extremo del SA usan los mismos bits para las mismas claves, y no importa qué parte del sistema divide la cadena de bits en claves individuales, la-s claves de encriptación DEBEN ser tomadas desde los primeros bits (más-a-la-izquierda, mayor-importancia – left-most, high-order) y la-s claves de autenticación DEBEN ser tomadas desde los bits restantes. El número de bits para cada clave se define en la especificación RFC del algoritmo utilizado. En el caso de múltiples claves de encriptación o múltiples calves de autenticación, la especificación para el algoritmo debe especificar el orden en el que son seleccionadas desde una única cadena de bits proporcionada al algoritmo.

3.10.3 Localizando un Security Gateway

Esta sección discute temas acerca de cómo un host conoce la existencia de relevantes SG y una vez que un host ha contactado estos SG, cómo sabe que son los SG correctos. Los detalles de donde se almacena la información requerida es un asunto local.

Consideremos una situación en la que un host remoto (H1) está usando Internet para acceder a un servidor u otra máquina (H2) y existe un SG (SG2), pej., un firewall, a través del cual el tráfico de H1 debe pasar. Un ejemplo de esta situación sería un host móvil (Road Warrior) cruzando Internet para llegar al firewall de la organización de trabajo (SG2). Esta situación nos lleva a estudiar las siguientes situaciones:

1. Como H1 sabe/aprende acerca de la existencia del SG2?
2. Como autentifica SG2, y una vez que lo hace, como confirma que SG2 ha sido autorizado para representar a H2?
3. Cómo SG2 autentifica H1 y verifica que H1 está autorizado para contactar con H2?
4. Cómo H1 sabe/aprende acerca de otros gateways (backup gateways) que proporcionan caminos alternativos para llegar a H2?

Para poder solucionar estos problemas, un host o SG DEBE tener una interfase administrativa que permita al usuario/administrador configurar la dirección de un SG para cualquier conjunto de direcciones destino que requieran su uso. Esto incluye la capacidad para configurar:

La información necesaria para localizar y autentificar al SG y verificar su autorización para representar al host destino.

La información necesaria para localizar y autenticar cualquier gateway alternativo y verificar su autorización para representar al host destino.

Se asume que la información contenida en la política del SPD cubre cualquier otro requerimiento IPSec para el camino hacia el SG y el host destino.

3.11 Asociaciones Seguras y Multicast

El que la SA esté orientada al receptor implica que, en el caso de tráfico unicast, el sistema destino normalmente selecciona el valor SPI. Siendo esto así, no hay forma de que el modo de configuración manual del SA origine un conflicto con el SA configurado automáticamente (pej., vía protocolo de gestión de claves IKE) o de que para las SA desde múltiples orígenes origine un conflicto con cualquier otra. Para tráfico multicast, hay sistemas que permiten enviar a grupo "multicast". Algún sistem a o persona necesitará coordinar entre grupos multicast para seleccionar un SPI o SPIs en representación del grupo y después comunicar la información IPSec del grupo a todos los miembros legítimos del grupo multicast a través de mecanismos no definidos aquí.

Múltiples emisarios a un grupo multicast DEBERÍAN usar un único SA (y, por lo tanto SPI) para todo el tráfico referente a dicho grupo cuando se emplee un algoritmo simétrico de encriptación o autenticación. En tales circunstancias, el receptor sabe solo que el mensaje vino desde un sistema que tenía la clave para el grupo multicast. En este entorno, un receptor generalmente no será capaz de autenticar qué sistema envió el tráfico multicast. Especificaciones para otros casos multicast más generales se postergan a nuevos documentos IPSec.

Inicialmente, los protocolos automatizados existentes para la distribución de claves multicast no fueron considerados lo suficientemente maduros para ser estandarizados. Para grupos multicast que tengan relativamente pocos miembros, la distribución de claves manual o el uso múltiple de algoritmos de distribución de claves unicast ya existentes como el Diffie-Hellman modificado parecen ser suficientes. Para grupos grandes o muy grandes, se necesitarán nuevas técnicas escalables. Un ejemplo de trabajo de investigación en este campo es el Group Key Managemente Protocol.

3.12 Procesamiento del tráfico IP

Como ya se ha mencionado, el SPD debe ser consultado durante el procesamiento de TODO el tráfico (ENTRADA y SALIDA), incluyendo tráfico no IPSec. Si no se encuentra ninguna política en el SPD que pueda ser aplicada al paquete (para tráfico de entrada y salida), el paquete DEBE ser borrado.

NOTA: Todos los algoritmos criptográficos usados en IPSec esperan su entrada ordenada byte a byte (RFC 791) y generan su salida también de forma ordenada byte a byte. Los paquetes IP son también transmitidos ordenadamente byte a byte (network byte order).

3.12.1 Procesamiento del tráfico IP de salida

3.12.1.1 Seleccionando y usando un SA o SA Bundle

En una implementación SG o BITW (y en muchas BITS), cada paquete de salida es comparado contra el SPD para determinar el procesamiento requerido para el paquete. Si el paquete debe ser borrado, este es un evento auditable. Si al tráfico le es permitido omitir el procesamiento IPSec, el paquete continua el procesamiento "normal" para el entorno en el que se ha implementado IPSec. Si se debe aplicar el procesamiento IPSec, el paquete es bien mapeado a un SA existente (o SA bundle), o un nuevo SA (SA bundle) se crea para el paquete. Dado que un selector de paquete puede coincidir con múltiples políticas o múltiples SAs y dado que el SPD sigue un orden, pero el SAD no, IPSec DEBE:

1. Match los campos del selector de paquete contra las políticas de salida en el SPD para localizar la primera política apropiada, que apuntará a cero o más SA bundles en el SAD.

2. Match los campos de selector del paquete contra aquellos en el SA bundle encontrados en (1) para localizar el primer SA bundle que matches. Si no se encuentra ningún SA o ninguno match, se debe crear un SA bundle apropiado y enlazar la entrada SPD a la entrada SAD. Si no se encuentra ninguna entidad de gestión de claves, se debe ignorar (drop) el paquete.

3. Usar el SA bundle encontrado/creado in (2) para hacer el procesamiento IPSec requerido, pej., autenticación y encriptación.

En una implementación IPSec de host basada en sockets, el SPD será consultado siempre que un nuevo socket sea creado para determinar cual, si hay alguno, procesamiento IPSec será aplicado al tráfico que fluirá en dicho socket.

NOTA: Una implementación acorde IPSec NO DEBE permitir la instanciación de un ESP SA que emplee ambos algoritmos de: encriptación NULL y autenticación NULL. Un intento de negociar un SA tal es un evento auditable.

Construcción de la cabecera en el Modo Túnel

Esta sección describe el manejo de las cabeceras IP inner y outer, cabeceras de extensión, y opciones para túneles AH y ESP. Esto incluye cómo construir la cabecera IP encapsulada (outer), como manejar los campos en la cabecera IP inner, y qué otras acciones deben ser tomadas. La idea general está modelada a partir de la descrita en RFC 2003 "IP Encapsulation with IP":

> Las direcciones fuente y destino de la cabecera IP outer identifican los "endpoints" (endpoint: hosts o GW) del túnel (el encapsulador y el

desencapsulador). Las direcciones fuente y destino de la cabecera IP inner identifican al emisor original y al recipiente del datagrama (desde la perspectiva de este túnel), respectivamente.

La cabecera IP inner no es cambiada excepto para decrementar el TTL como se mencionará, y permanece sin cambiar durante su entrega al punto de salida del túnel.

Ningún cambio a las cabeceras de extensión u opciones IP en la cabecera inner debe ocurrir durante la entrega del datagrama encapsulado a través del túnel.

Si es necesario, otras cabeceras de protocolo tal como la cabecera de autenticación IP puede ser insertada entre la cabecera IP outer y la cabecera IP inner.

Las tablas en las siguientes sub-secciones muestran el manejo para los diferentes campos cabecera/opción (construcción= el valor en el campo outer es construido independientemente del valor en el inner)

< Correspondencia entre Outer Hdr e Inner Hdr >		
	Outer Hdr at	Inner Hdr at
IPv4	Encapsulator	Decapsulator
Header fields:	----------	----------
Version	4 (1)	no change
header length	constructed	no change
TOS	copied from inner hdr (5)	no change
total length	constructed	no change
ID	constructed	no change
flags (DF,MF)	constructed, DF (4)	no change
fragmt offset	constructed	no change
TTL	constructed (2)	decrement (2)
Protocol	AH, ESP, routing hdr	no change
Checksum	constructed	constructed (2)
src address	constructed (3)	no change
dest address	constructed (3)	no change
Options	never copied	no change

1.- La versión IP en la cabecera encapsulada puede ser diferente del valor en la cabecera inner.

2.- El TTL en la cabecera inner es decrementado por el encapsulador antes de realizar el envío y por el desencapsulador si a su vez reenvía el paquete. (El checksum cambia cuando el TTL cambia). NOTA: El decremento del TTL es una de las acciones normales que se realizan al reenviar un paquete. Los paquetes originados

desde el mismo nodo, como el encapsulador, no tienen su TTL decrementado, dado que el nodo emisor está creando el paquete en lugar de enviarlo. En el nodo origen no se decrementa.

3.-Las direcciones src y dest dependen del SA, que es usado para determinar la dirección destino que en su lugar determina qué dirección src (interfase de red) es usado para enviar el paquete.

NOTA: En principio, la dirección fuente IP encapsulada puede ser cualquiera de las direcciones de la interfase del encapsulador e incluso una dirección diferente de cualquiera de las direcciones IP del encapsulador, (pej., si está actuando como un NAT box) en tanto en cuanto la dirección sea alcanzable a través del encapsulador desde el entorno en el que el paquete es enviado. Esto no debe ser un problema porque actualmente IPSec no tiene ningún requerimiento de procesamiento INBOUND que involucra la dirección fuente de la cabecera IP encapsulada. Así, mientras el endpoint de recepción del túnel busca la dirección destino en la cabecera IP encapsulada, solamente busca la dirección fuente en la cabecera IP inner (encapsulada).

4.- La configuración determina si copiar, borrar o set el DF desde la cabecera IP inner (solo para IPV4).

5.- Si la cabecera inner es Ipv4 (Protocol=4), copia el TOS. Si la cabecera es Ipv6 (Protocol=41), mapear la Clase a TOS.

Construcción de la cabecera en el modo túnel para Ipv6

< Correspondencia entre Outer Hdr e Inner Hdr >		
IPv4	Outer Hdr at Encapsulator	Inner Hdr at Decapsulator
Header fields:	----------	----------
Version	6 (1)	no change
Class	Copied or constructed (6)	no change
Flow id	copied or configured	no change
Len	constructed	no change
Next header	AH,ESP,routing hdr	no change
Hop limit	constructed, (2)	decrement (2)
Src address	Constructed (3)	no change
Dest address	constructed (3)	no change
Extension headers	Never copied	no change

6.- Si la cabecera IP inner es Ipv6 (Next Header = 41), copiar la clase. Si la cabecera inner es Ipv4 (Next Header = 4), mapear el TOS a class.

3.12.2 Procesamiento del tráfico IP de entrada

Antes de realizar el procesamiento AH o ESP, todos los fragmentos IP son reensamblados. Todo datagrama IP de entrada al que se le aplica el procesamiento IPSec es identificado por la aparición de los valores AH o ESP en el campo IP Next Protocol (o de una cabecera de extensión AH o ESP en el contexto Ipv6).

3.12.2.1 Selección y uso de SA o SA Bundle

Mapear el datagrama IP al SA apropiado está simplificado debido a la presencia del SPI en la cabecera AH o ESP. Conviene notar que el chequeo del selector se hace sobre las cabeceras inner y no las outer (túnel). Los pasos seguidos son:

1.- Usar la dirección destino del paquete (cabecera IP outer), protocolo IPSec, y el SPI para buscar el SA en la SAD. Si la búsqueda del SA falla, ignorar (drop) el paquete y reportar, incluir en el log dicho fallo/error.

2.- Úsese el SA encontrado en (1) para el procesamiento IPSec, pej., autenticación y desencriptación. Este paso incluye matching los selectores del paquete (si la cabecera inner está tunneled) contra los selectores del SA. La política local determina la especificidad de los selectores SA (valor simple, lista, rango, wildcard). En general, la dirección src de un paquete DEBE match el valor del selector en SA. Sin embargo, un paquete ICMP recibido en SA modo túnel puede tener una dirección src diferente de aquella que originó el enlace al SA y, de este modo, estos paquetes deberían ser permitidos como excepciones al chequeo. Para un paquete ICMP, los selectores del paquete “problema”(las direcciones y puertos src y dst deberían ser intercambiadas) deberían ser chequeados contra los selectores del SA. Nótese que alguno o todos los selectores pueden ser inaccesibles a causa de limitaciones acerca de cuántos bits del paquete “problema” al paquete ICMP le es permitido incluir (carry) o debido a la encriptación.

Hacer los pasos (1) y (2) para cada cabecera IPSec hasta que una cabecera del protocolo de transporte o una cabecera IP que NO es para este sistema sea encontrada. Mantener un record de que SAs han sido usadas y su orden de aplicación.

3.- Encuéntrese una política de llegada in el SPD que matches el paquete. Esto puede ser llevado a cabo, por ejemplo, usando punteros-atrás desde los SAs al SPD o mediante el matching los selectores de paquete (cabecera inner si túnel) contra aquellos de las entradas de política en el SPD.

4.- Se debe chequear si el procesamiento IPSec requerido ha sido aplicado, pej., verificar que el/las SAs encontrados en (1) y (2) match el tipo y orden de SAs requeridos por la política encontrada en (3).

NOTA: La política correcta elegida (que haga match) no será necesariamente la primera política de entrada que se encuentre. Si el chequeo en (4) falla, los pasos (3) y (4) serán repetidos hasta que todas las entradas hayan sido chequeadas o hasta que el chequeo tenga éxito.

Al final de estos pasos, el paqueta resultante se pasa a la capa de Transporte o bien se reenvía el paquete. Nótese que cualquier cabecera IPSec procesada en estos pasos puede haber sido borrada, pero esta información, pej., qué SAs fueron aplicados y en qué orden, puede necesitarse para subsiguientes procesamientos IPSec o del firewall.

Nótese que en el caso de un SG, si el reenvío origina un paquete para salir vía interfase IPSec-enabled, entonces procesamiento IPSec adicional puede que deba ser aplicado.

3.13 Manejo de túneles AH y ESP

El manejo de las cabeceras IP inner y outer, cabeceras de extensión y opciones para los túneles AH y ESP debería ser realizado como se describió anteriormente.

3.14 Procesamiento ICMP (en relación con IPSec)

El enfoque de esta sección se realizará sobre el manejo de los mensajes de error ICMP. Otro tipo de tráfico ICMP, pej., Echo/Reply, debería ser tratado como cualquier otro tipo de tráfico tal y como se ha descrito hasta ahora.

Un mensaje de error ICMP protegido por AH o ESP ;y generado por un router DEBERÍA ser procesado y reenvíado en un SA modo túnel. La política local determina si está sujeto o no a los chequeos de dirección fuente por el router en la parte dst del túnel. Nótese que si el router en la parte origen del túnel está reenviando un mensaje de error ICMP desde otro router, el chequeo de la dirección src fallaría. Un mensaje ICMP protegido por AH o ESP y generado por un router NO DEBE ser reenviado en un SA modo transporte (a menos que el SA haya sido establecido con el router actuando como un host, pej., una conexión Telnet usada para gestionar-administrar el router). Un mensaje ICMP generado por un host DEBERÍA ser chequeado contra los selectores dirección IP src

3.15 Internet Engineering Task Force(IETF)-Request For Comments(RFC)

La IETF se encarga de decidir cuáles documentos deben ser incluidos en la lista de RFCs.

Es tan importante cuánto haya sido probado, implementado y testado la propuesta en cuestión como la importancia de su cometido para que llegue a ser considerado como un nuevo RFC

Muchos RFCs definen los estándards a partir de los cuales se fundamenta Internet.

Virtual Private Network Consortium (VPNC)
Este consorcio está formado por las principales compañías del sector (Checkpoint, Nokia, Alcatel...)

El VPNC declara como los tres principales protocolos de comunicación para asegurar VPNs:

IPsec
L2TP (bajo IPSec)
PPTP

De los cuales, se espera que IPSec sea el que se impondrá en un futuro próximo. L2TP ejecutado bajo IPSec aún está en fase de desarrollo y PPTP es un protocolo propietario (Microsoft y otros)

El protocolo IPSec se compone de una serie de estándar y recomendaciones especificados por la Internet Engeneering Task Force (IETF)

RFCs e Internet Drafts

Con el propósito de definir IPSec, cualquier protocolo o nota que llegue a ser un Request For Comments (RFC) puede ser tratado “más o menos” como un estándar.

Antes de que dicha propuesta desemboque en un RFC debe ser nominada como un Internet Draft (I-D) y discutida en diversos forums, a menudo Grupos de Trabajo (WGs) del IETF. Si finalmente se demuestra su utilidad, pasan a ser un RFC de modo que el nombre I-D desaparece y se asigna un número RFC.

4 CONSTRUCCIÓN DE UNA VPN BAJO IPSEC

4.1 Situación inicial

Tenemos una intranet en la cual están involucrados diferentes sites con su propia LAN interna.

4.2 Objetivo

Queremos conseguir una comunicación segura –encriptada- a través de Internet usando el estándard IPSec.

4.3 Material:

4.3.1Hardware:

- 2 firewalls BIGfire asimismo como un medio de soporte para simular la red insegura (Internet),
- 2 firewalls NOKIA CC-500 Gateway VPN
- 2 lineas ISDN,
- 3 diferentes PC'S,
- 2 direcciones IP válidas,
- cables de par cruzado de categoría 5 (100Mbps) *transparentes* & *crossover (cruzados para conectar dos NIC directamentes).*
- 1 hubX5 (10Mbps).

4.3.2 Software

- OS: W98 & W2000.
- BALI –proporcionado por BIODATA-
- sniffers y proxys -free y shareware-
- Nokia VPN Manager, software proporcionado por Nokia.

4.4 Descripción del firewall BIGfire+

Tenemos al firewall como elemento clave en el desarrollo del proyecto. Así, a continuación pasaremos a describir sus componentes y características.

Características

Incorpora tres interfaces de red, llamados *admin.*, *intern* y *extern* con propósitos administrativos, de conexión de la red interna y de conexión del router respectivamente.

Asimismo dispone de una entrada RS-232 para realizar la configuración inicial.

Encriptación.- Disponemos de la posibilidad de encriptar la información enviada y, por supuesto de desencriptar la que llega (usando los algoritmos DES & 3DES).

NAT (Network Address Translation), es decir, si solo disponemos de una dirección IP válida pero estamos usando múltiples ordenadores en nuestra red privada con posibilidad de acceder al exterior, este dispositivo se encarga de realizar la traducción a la única dirección válida de aquellas que previamente hemos asignado.

Filtrado.- Como una característica adicional de seguridad disponemos de la opciónde filtrar paquetes transmitidos a través de TCP-IP como de aquellos que son enviados a través de protocolos orientados ¨sin conexión¨ (UDP, ICMP).

Herramientas de Seguridad.- Como una caracterísitca avanzada de seguridad tenemos mecanismos: anti-spoofing, bloqueo de paquetes ICMP, filtrado de direcciones IP no permitidas en la subred así como de evitación del ataque DoS (Denial of Service).

Así pues, este dispositivo dispone de las características que nos interesan: permite crear una conexión segura entre dos localizaciones usando –tal y como queremos- la –barata- conexión a Internet en lugar de usar líneas dedicadas (pej. ISDN) –mucho más caras en comparación- pero con un nivel de seguridad similar.

4.5 Configuración de los firewalls

Primero de todo, se trata de configurar la dirección del interface de administración –admin-; lo haremos a través de la conexión RS-232.

A continuación, realizaremos la configuración completa del firewall a través de la aplicación BALI utilizando un cable cruzado conectado entre el firewall y el computador de administración del mismo. Resumiendo, hasta ahora tenemos lo siguiente:

- La clave usada para asegurar nuestra configuración (de utilidad también para clientes remotos, VPN, etc.) encriptada usando el algoritmo 3DES.
- Dirección del loghost.
- Subredes válidad que usaremos para filtrar los servicios que nos interesen.
- Los filtros correspondientes para definir las rutas que necesitemos.

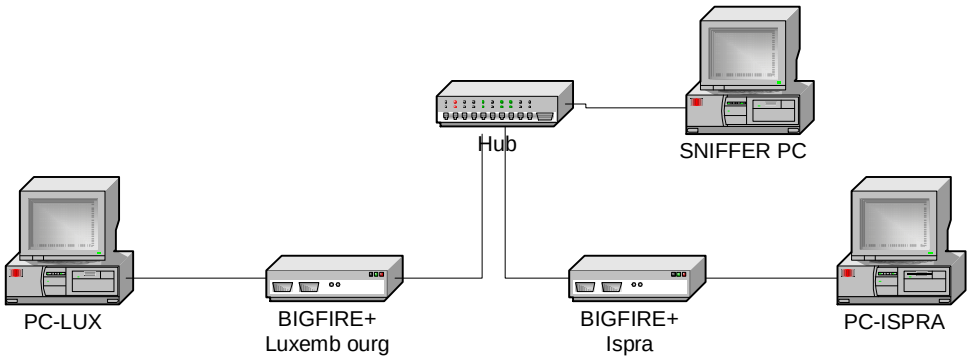
NAT
No usado, ya que en la prueba disponemos de un único ordenador como simulación de la red privada.

Tunneling

Construimos el túnel entre dos subredes escogidas libremente por nosotros.

4.6 TESTS

4.6.1 BIGFIRE CON BIGFIRE:



Tipo de conexión: crossover, es decir, un cable de par cruzado entre los dos interfaces de red externos de cada Bigfire.

Se ha programado un túnel ESP (3DES) en cada uno de los firewall.

Colocaremos una máchina ¨espía¨ con un software ¨sniffer¨ instalado entre ambos interfaces. También instalamos el “sniffer” en el PC destino. A continuación realizamos la transferencia de un fichero de texto entre ambas localizaciones; la máquina “espía” captura los paquetes enviados, pero su contenido es ilegible debido a la encriptación utilizada. Por supuesto, el sniffer instalado en la máquina destino es capaz de obtener todos los paquetes desencriptados.

Por último, hemos realizado la transferencia del fichero de texto deshabilitando previamente la encriptación. En este caso, la máquina “espía” es capaz de leer el texto transmitido sin ningún problema.

4.6.2 TESTS USANDO COMO ROUTER *fli4l*

En primer lugar, hemos habilitado dos LAN para poder simular dos localizaciones distintas.

A continuación, hemos configurado un router (*fli4l –router software implementado con LINUX)* entre la LAN remota e Internet para asegurarnos de que tenemos acceso a Internet a través del router.

Sin embargo, no es posible un funcionamiento correcto entre ambas localizaciones debido a que no disponemos de una dirección IP válida para poder acceder a Internet.

A continuación se intenta la conexión a través de un cable cruzado (crossover) entre ambas localizaciones; pero debido a problemas de enrutamiento tampoco llega a funcionar de este modo.

4.6.3 TEST ENTRE CLIENTES IPSEC: (PGPNet)-BIGFire+

Se ha configurado el cliente de la forma más similar posible a cómo está definido el Ipsec en el Bigfire+ (algoritmo ESP con el algoritmo de encriptación 3DES).

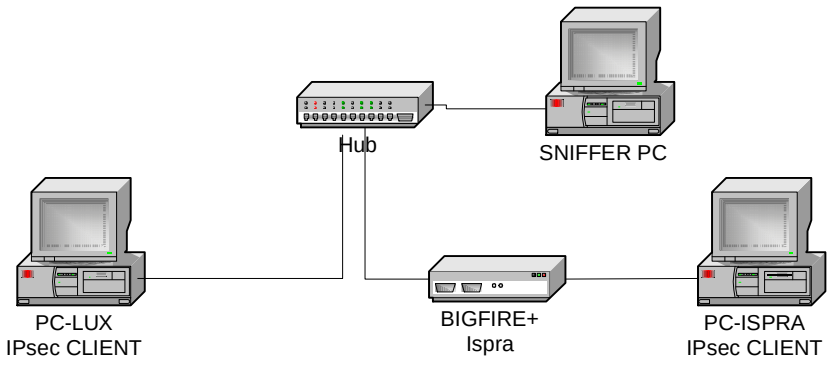
Así, se ha creado un túnel entre la subred de una localización y la máquina con el cliente Ipsec.

Antes de completar la programción del túnel se comprueba la conectividad física a los recursos de ambas máquinas. El resultado es correcto.

A continuación y después de configurar el túnel no es posible completar la conexión. El problema estriba en que nuestro firewall –BIGfire+- no tiene un mecanismo de autenticación –solo de encriptación-, el cual es un requisito del protocolo Ipsec.

Así pues, no hay modo de completar una conexión segura a través de un cliente que use el estándar Ipsec (pej. PGPNet) y el BIGfire+.

4.6.4 TEST ENTRE DOS CLIENTES IPSEC USANDO EL BIGFIRE PARA ACCEDER A LA RED INSEGURA (INTERNET)



Hemos procedido a instalar el cliente Ipsec *SSH Sentinel* en una máquina conectada al interfaz interno del BIGFire+ y otro cliente en otra máquina conectada al interface externo.

También en este caso y como ayuda en la comprobación del funcionamiento de la encriptación, se instala un sniffer (Iris2.0 thread) en la máquina externa.

Previamente a la conexión Ipsec hemos comprobado la conectividad entre ambos clientes.

A continuación se procede a la instalación del software en ambas máquinas. Como autenticación usaremos el método *preshared key*.

Para conectar hemos utilizado el Sistema Operativo, accediendo a los recursos compartidos –por el SO- de ambas máquinas. Como información transmitida se utiliza un fichero de texto fácilmente reconocible.

Cuando hemos habilitado ambos clientes Ipsec resulta imposible el descifrar toda la información transmitida. Ni tan siquiera el protocolo usado por los paquetes!

Sin embargo, cuando no se usa ningún método para asegurar la información, el sniffer es capaz de descifrar el tráfico sin ningún problema.

4.6.5 TEST ENTRE DOS CLIENTES IPSEC CONECTADOS A TRAVÉS DE UN CABLE CRUZADO (CROSSOVER)

Este test es muy similar al anterior. Pero esta vez hemos desconectado el BIGfire+ de la red y conectado ambos clientes IPSec directamente a través de un cable cruzado (crossover).

Los resultados son, al igual que en el caso anterior, satisfactorios.

4.6.6 TEST IPSEC (PGPNet)- IPSEC(SSH SENTINEL)

Ahora comprobaremos el funcionamiento de dos clientes diferentes que implementan Ipsec. (PGP y SSH)

Es decir, en este caso tendremos un test de interoperabilidad entre soluciones Ipsec de distintos fabricantes.

En este caso la configuración es más complicada que en el caso de usar dos clientes del mismo fabricante.

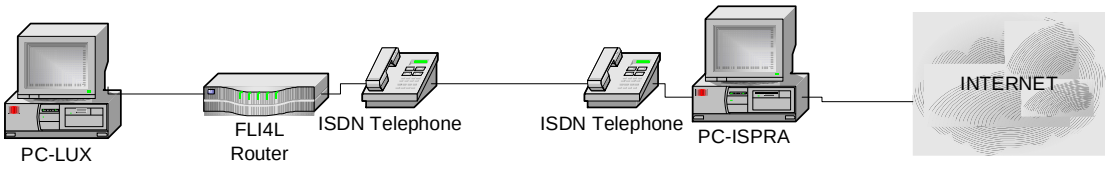
Pero, una vez configurados ambos clientes de modo que ambos utilicen los mismos protocolos con opciones compatibles los resultados son de nuevo satisfactorios.

NOTA: En todos estos tests –usando clientes IPsec -, el aspecto más importante es el relacionado con la fase de autenticación. El modo más sencillo de llevarlo a cabo es usar el método de *preshared key*, con una clave creada al efecto. Una vez lograda la autenticación, el funcionamiento del túnel es el esperado.

4.6.7 TEST SSH-FLI4L(ROUTER)-SSH A TRAVÉS DE UNA LÍNEA RDSI

Las direcciones IP usadas son:
192.168.10.20 PC-LUX
192.168.10.10 FLI4L
139.191.71.29 PC-ISPRA

ISDN-Line 1 (Ispra) 0332782460
ISDN-Line 2 (Lux) 0332782488
El SO del PC-LUX OS es el W2K
El SO del PC-ISPRA es el WNT4 Server con el SP5

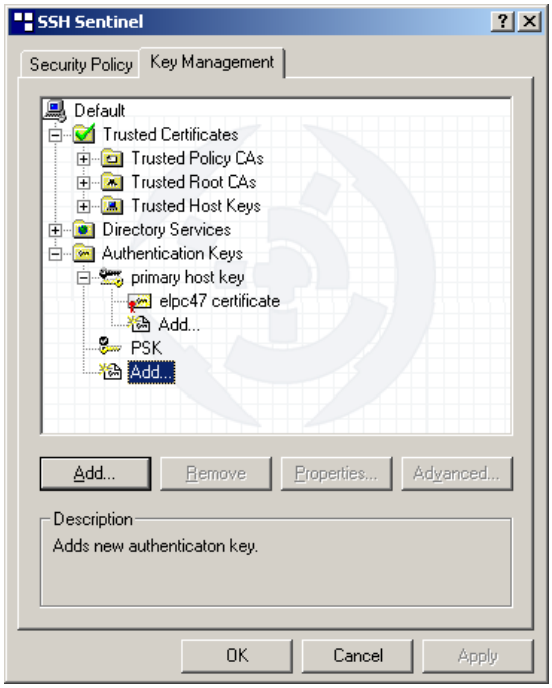


El ordenador PC-LUX usa como método para conectar con el PC-ISPRA el *dial-in.* Este último usa el servicio RAS para asignar una dirección IP al ordenador llamante (PC-LUX), en este caso: 139.191.71.108

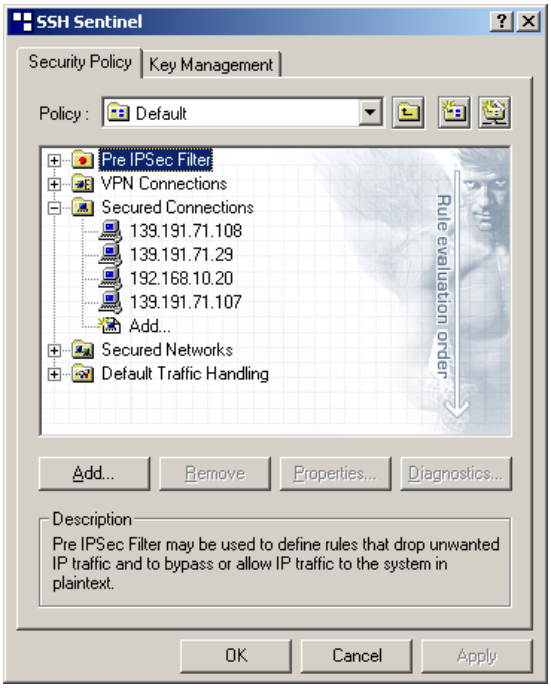
En este caso no podemos alcanzar el ordenador PC-LUX más allá del router debido a 1) su dirección IP no es una dirección válida en Internet y 2) el router utilizado *fli4l* no ofrece servicio NAT, luego no tenemos la posibilidad de adquirir una para el PC. Sin embargo, desde el ordenador PC-LUX sí que es posible alcanzar máquinas externas, dado que el resto sí que usa direcciones IP válidas.

SSH Client Configuration:

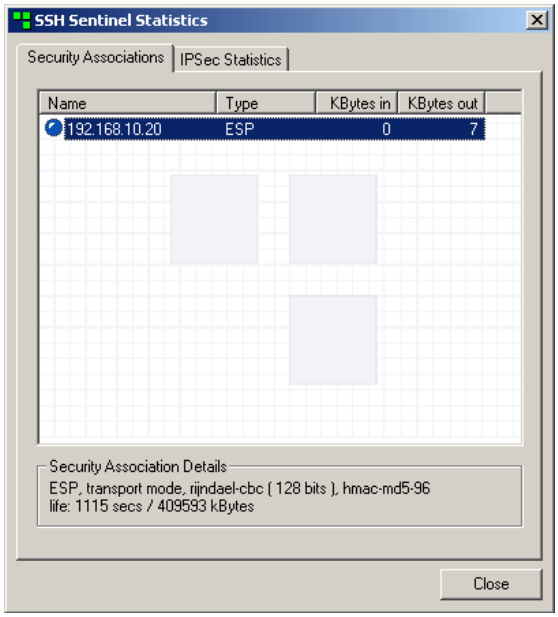
En primer lugar, hemos de elegir el método de autenticación a usar. Así será: *shared password*. Es el modo más sencillo y, dado que no tenemos ningún certificado aceptado por ambas partes, el único posible. La siguiente imagen muestra el método elegido.



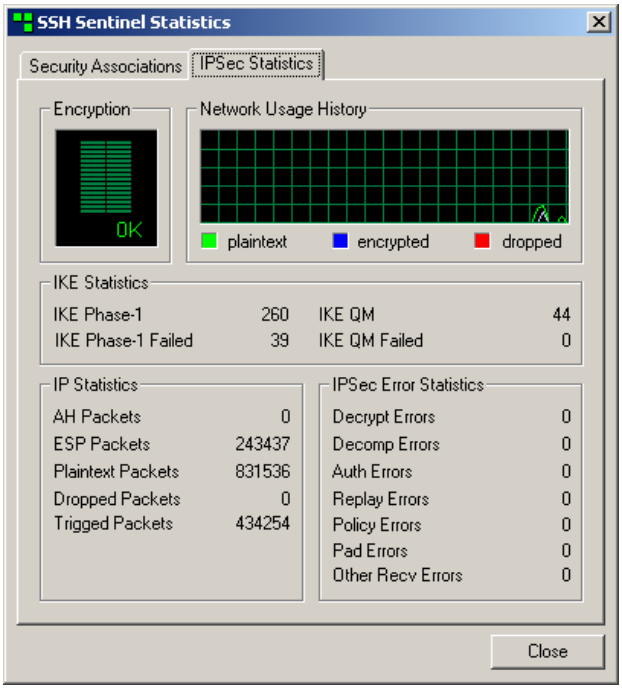
Una vez que tenemos el método de autenticación, debemos construir nuestra conexión segura. Elegimos la opción *Secured Connections* ya que estamos intentado conectar dos hosts sin ningún gateway seguro –que gestione el túnel- entre ambos.



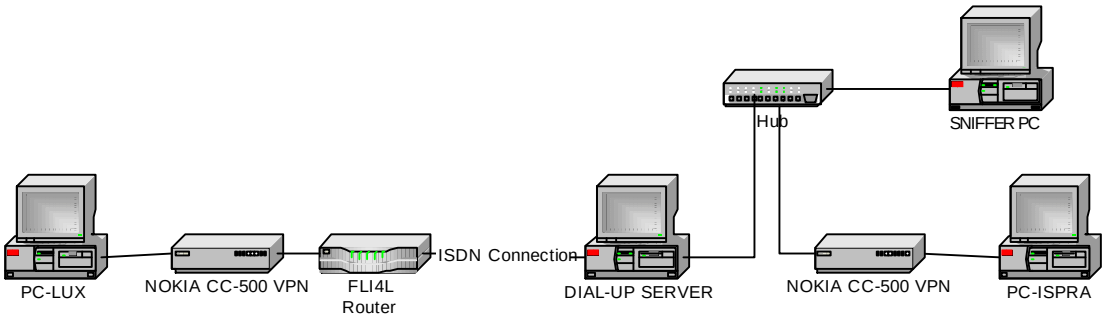
Cuanto tenemos la conexión segura establecida, aparece una linea indicando la dirección IP del host remoto, el tipo de encriptación utilizado así como la cantidad de bytes transferidos.


SSH Sentinel Statistics
Security Associations
IPSec Statistics
Name
Type
KBytes in
KBytes out
192.168.10.20
ESP
0
7
Security Association Details
ESP, transport mode, rijndael-cbc (128 bits), hmac-md5-96
life: 1115 secs / 409593 kBytes
Close

La aplicación nos muestra las estadísticas y otros detalles relacionados con TODAS las transferencias realizadas por el host (encriptadas y no encriptadas) solo con elegir la opción *Ipsec Statistics*:



4.6.8 TEST NOKIA CC-500 VPN - NOKIA CC-500 VPN A TRAVÉS DE UNA LÍNEA RDSI



Configuración

VPN Manager:

Routing GW Ispra:

192.168.10.0 mask 255.255.255.0 139.191.71.29

192.168.40.0 mask 255.255.255.0 139.191.71.29

Routing GW Lux:

192.168.20.0 mask 255.255.255.0 192.168.10.2

139.191.71.0 mask 255.255.255.0 192.168.10.2

Máquina servidora del DIAL-UP:

Añadimos las siguientes rutas una vez conectado el cliente mediante dial-in:

route add 192.168.40.0 mask 255.255.255.0 192.168.20.2 if 1

route add 192.168.10.0 mask 255.255.255.0 192.168.20.2 if 1

Direcciones IP:

PC-LUX: 192.168.40.2
NOKIA-LUX-INT: 192.168.40.1
NOKIA-LUX-EXT: 192.168.10.1
FLI4L-LUX:192.168.10.2
FLI4L-RAS:192.168.20.2
DIAL-UP SERVER RAS:192.168.20.1
DIAL-UP SERVER ETH:192.168.20.4
SNIFFER-PC:192.168.20.5
NOKIA-ISPRA-EXT:192.168.20.3
NOKIA-ISPRA-INT:139.191.71.29
ISPRA-PC:139.191.71.47

Notas acerca de la configuración:

Problemas de configuración cno el router fli4l. Aleatoriamente, el router fi4l no permite la comunicación entre Lux-LAN y el servidor del DIAL-UP (Ispra side).

Cuando esto sucede, podemos hacer un telnet al fli4l, a continuación hacer el dial-up desde la máquina con el gestor *imond*; después de lo cual usaremos el comando linux *route* sin parámetros –tan solo para mostrar la tabla de enrutamiento actual- y a continuación todo vuelve a la normalidad en la mayoría de los casos. Si no sucede así, podemos repetir la operación. En mi opinión, el problema se encuentra en la implementación del fli4l.

Usando el Nokia VPN Manager.- Aplicar siempre los cambios en la configuración a la parte remota del VPN y, a continuación, a la parte local.

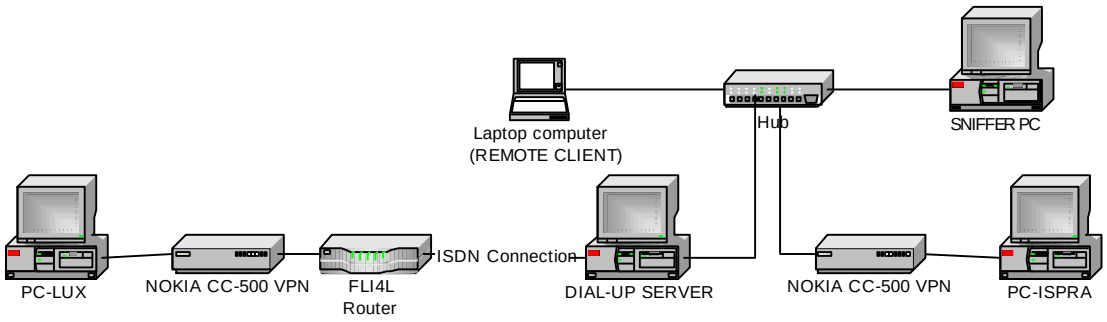
En nuestro caso, debemos siempre tener en cuenta que la conexión entre la máquina con el imond y el fli4l debe ser mantenida sin encriptar; de otro modo, no podremos -dado que el fli4l no está configurado para usar Ipsec- comunicar con el fli4l ni, por tanto, efectuar el dial-up!

Se ha chequeado la fase 1 IKE (main mode) con las opciones *preshared password* y *digital certificate*; en el último caso, hemos usado como autoridad certificadora la que proporciona el Nokia.

En ambos casos el funcionamiento ha sido satisfactorio.

Por otra parte, si queremos activar ambos canales RDSI, debemos marcar la opción *Multilink* en las *properties routing and remote service windows- network*. De otro modo, solo se activará el primer canal cada vez que se efectúe una llamada.

4.6.9 TEST NOKIA CC-500 VPN – CLIENTE REMOTO NOKIA VPN A TRAVÉS DE CABLE CRUZADO *"CROSSOVER"*



Configuración

VPN Manager:

Routing GW Ispra:
192.168.10.0 mask 255.255.255.0 139.191.71.29
192.168.40.0 mask 255.255.255.0 139.191.71.29
Routing GW Lux:
192.168.20.0 mask 255.255.255.0 192.168.10.2
139.191.71.0 mask 255.255.255.0 192.168.10.2

DIAL-UP Server Machine:
Se añaden las siguientes rutas tras establecer la llamada:
route add 192.168.40.0 mask 255.255.255.0 192.168.20.2 if 1
route add 192.168.10.0 mask 255.255.255.0 192.168.20.2 if 1

Direcciones IP:

PC-LUX: 192.168.40.2
REMOTE CLIENT: 192.168.20.6
NOKIA-LUX-INT: 192.168.40.1
NOKIA-LUX-EXT: 192.168.10.1
FLI4L-LUX:192.168.10.2
FLI4L-RAS:192.168.20.2
DIAL-UP SERVER RAS:192.168.20.1
DIAL-UP SERVER ETH:192.168.20.4
SNIFFER-PC:192.168.20.5
NOKIA-ISPRA-EXT:192.168.20.3
NOKIA-ISPRA-INT:139.191.71.29
ISPRA-PC:139.191.71.47

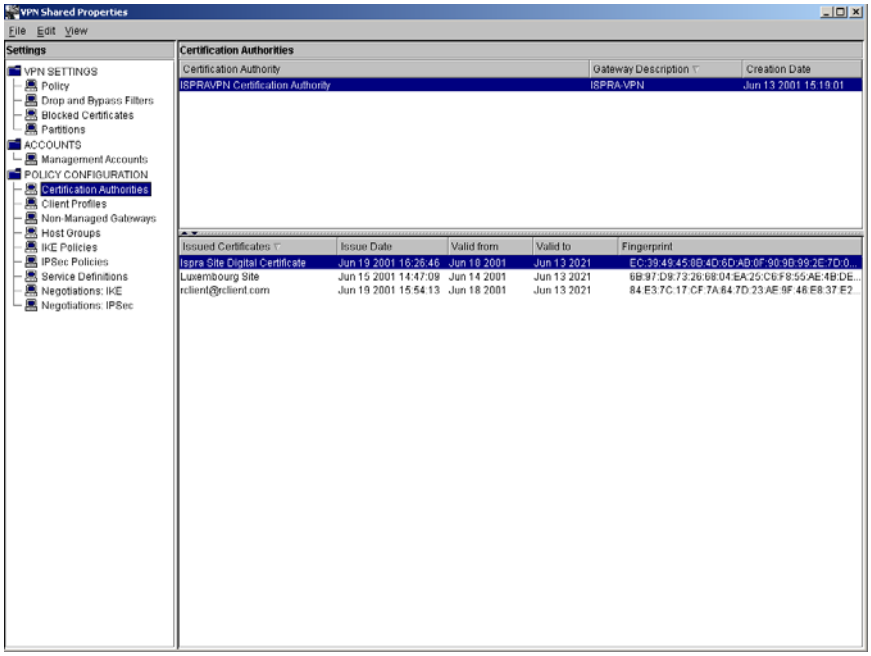
Establecimiento de la política del cliente Ipsec:

1. Creación de la Autoridad Certificadora

2. Creación del Certificado Digital para el Nokia VPN
3. Configuración de la política de acceso del cliente para el Nokia VPN
4. Creación del perfil de cliente
5. Configuración del cliente remoto

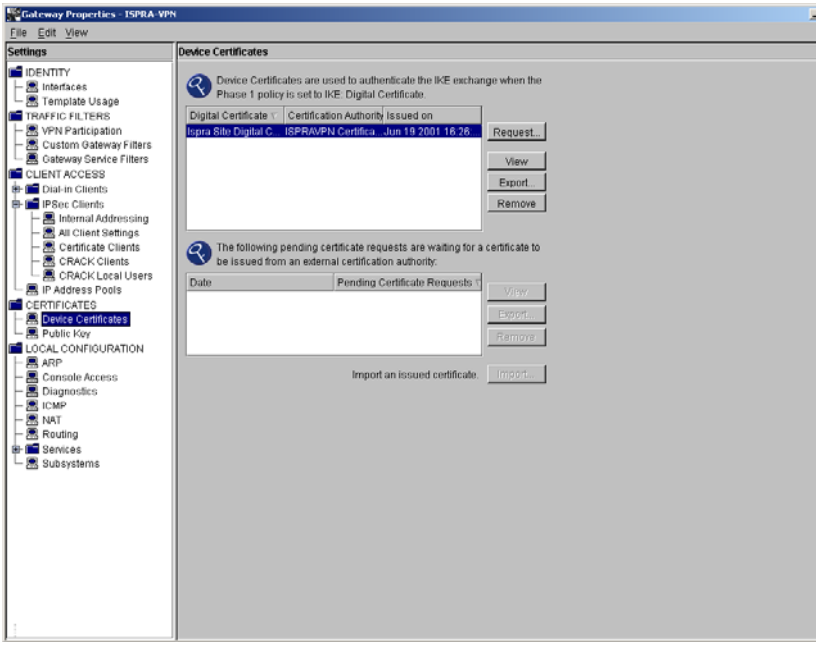
1 Creación de la Autoridad Certificadora

El Nokia CC-500 VPN provee una Autoridad de Certificación Interna que cumple con nuestros requerimientos para usar Certificados Digitales de modo que podamos autenticar los sites (y/o clientes remotos).



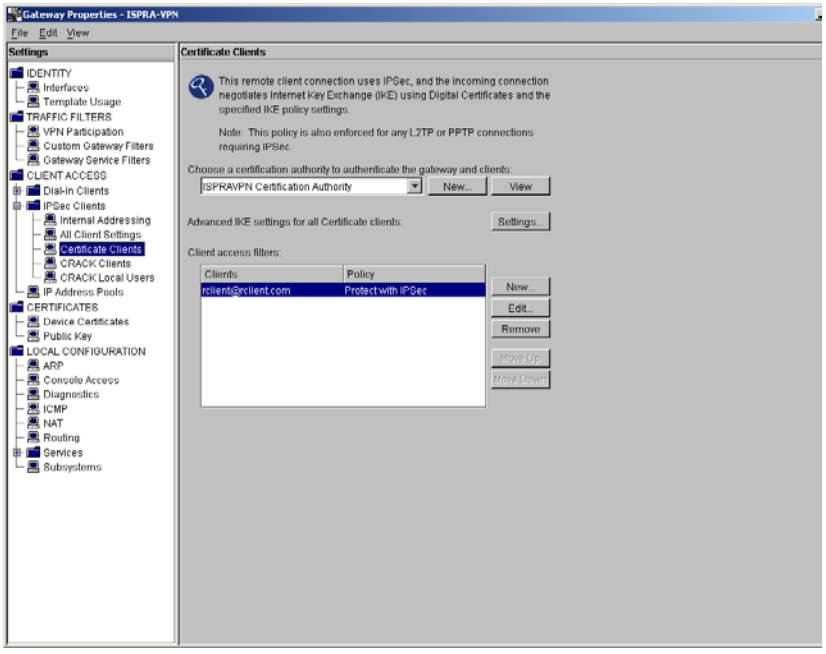
2 Creación del Certificado Digital para el Nokia VPN

La información que se incluye en los certificados tiene que ver con el usuario, la compañía donde trabaja…



3 Configuración de la política de acceso del cliente para el Nokia VPN

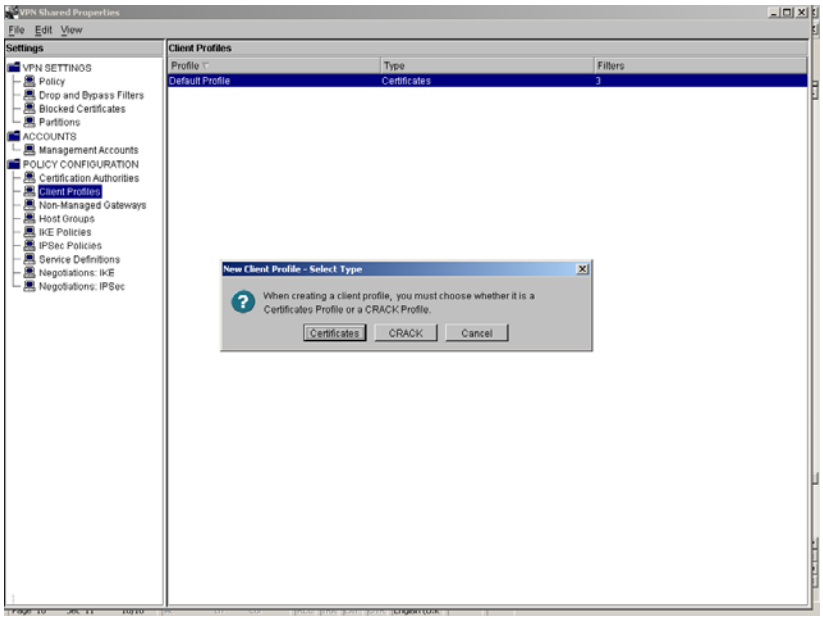
Una vez que tenemos un Certificado Válido emitido por nuestra Autoridad Certificadora, lo siguiente es especificar el tipo de autenticación usada en nuestras conexiones; elegiremos el método *digital certificate*. De este modo, especificaremos al Nokia VPN como la Autoridad de Certificación en la que confiamos.



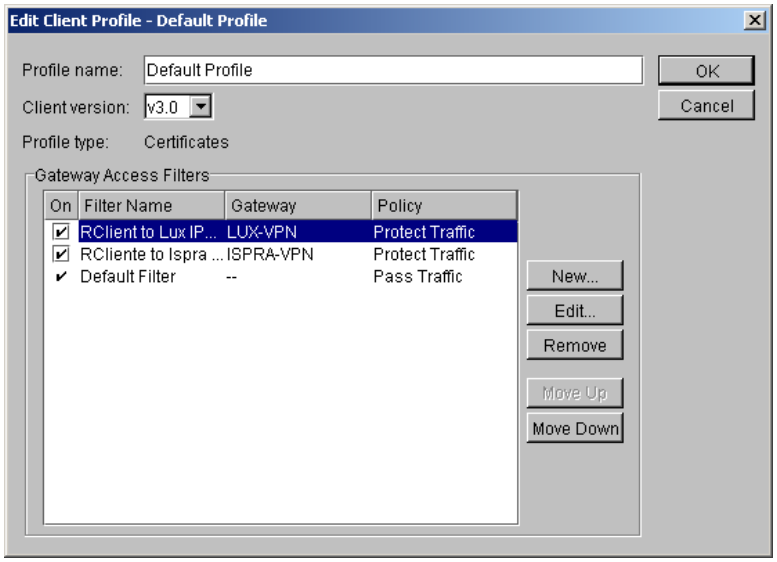
4 Creación del perfil de cliente

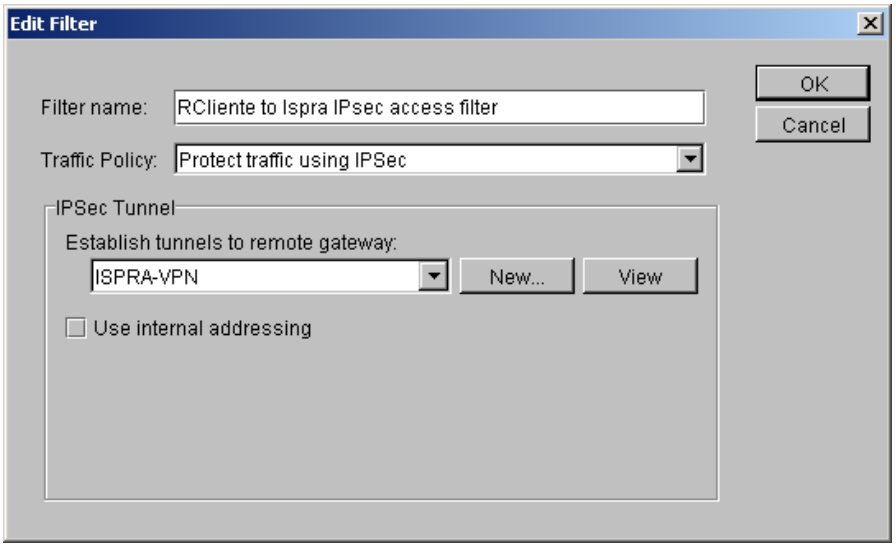
Primero debemos elegir el método de autenticación:

Con el Nokia VPN, los únicos métodos de autenticación válidos entre un cliente remoto y el gateway son dos: usando certificado digital y usando el método *crack.*

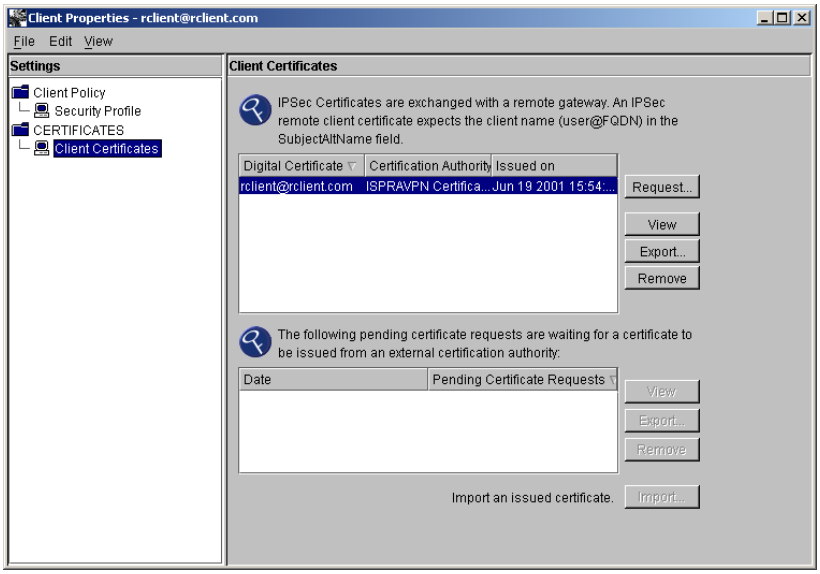


A continuación programamos los filtros adecuados para cumplir con nuestros requerimientos. Se pueden crear diferentes filtros dependiendo de los sites afectados.

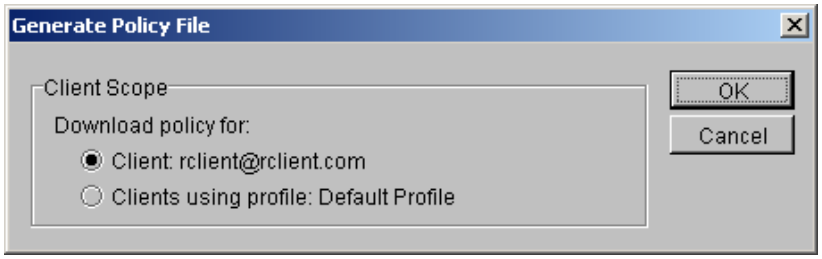




Para una correcta configuración del perfil del cliente remoto, debemos especificar un certificado válido para el mismo (normalmente, este certificado será emitido por la propia Autoridad Certificador Interna del Nokia Gateway)



Una vez que el perfil ha sido creado, obtenemos un fichero especial que contiene la política para el cliente remoto. Este fichero tiene como única finalidad el ser usado por el software del cliente remoto “Remote Client Software from Nokia”.



5 Configuración del cliente remoto

Aquí debemos cargar la política que habíamos creado con el Nokia VPN Policy Manager. Para ello, nos pide la clave que autentifica nuestro acceso.

4.6.10 TEST NOKIA CC-500 VPN - NOKIA <-> CC-500 VPN - NOKIA A TRAVÉS DE INTERNET

La intención es conectar dos LAN a través de Internet protegiendo el canal con los Gateways NOKIA mediante IPSec.

Para autentificar ambas partes utilizaremos los tres métodos descritos por el estándar IPSec. Además, como medida de seguridad adicional, ambas LAN tendrán direcciones IP no permitidas en Internet (RFC1908). Así, deberemos programar el NAT (Network Address Translation) en ambos gateways para poder acceder a las mismas.

1 Preshared key.
2 Public Key.
3 Digital Certificate.

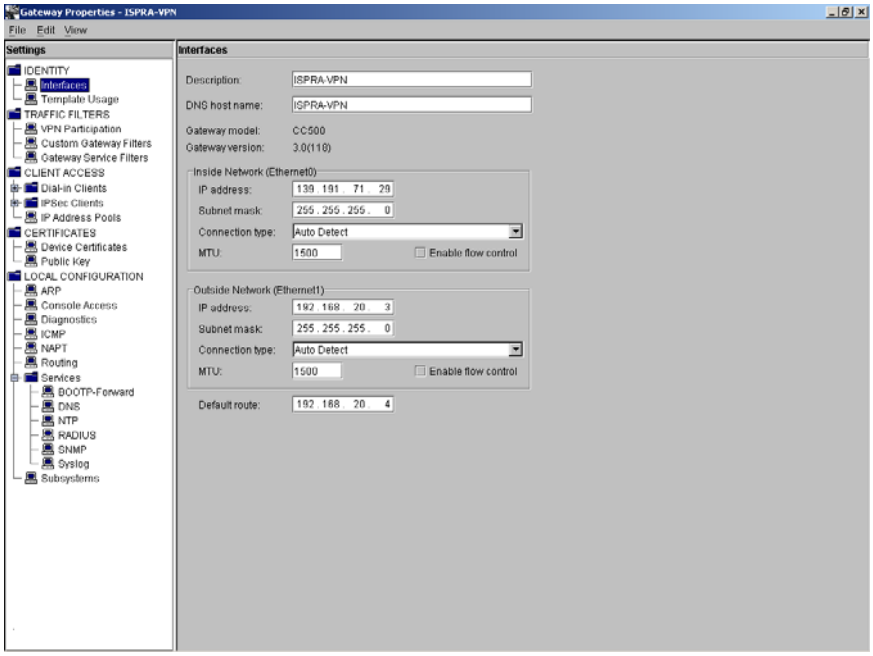
La topología y descripción de la conexión se encuentran ampliamente descritas en la página siguiente:

CONFIGURACIÓN GATEWAY LOCAL:

Pasemos a describir los pasos necesarios para programar los dispositivos Nokia utilizando el Nokia VPN Policy Manager. Siempre dentro de la ventana de propiedades del gateway local, entramos en las diferentes opciones:

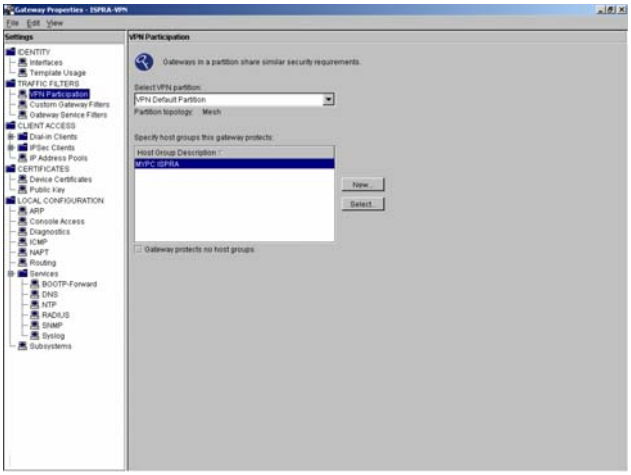
IDENTITY-Interfaces:

En primer lugar se definen los interfaces del gateway (direcciones IP, máscaras de la subred, dirección del router, nombre del gateway, tipo de interface y máxima unidad de transmisión –tamaño máximo de los paquetes-)



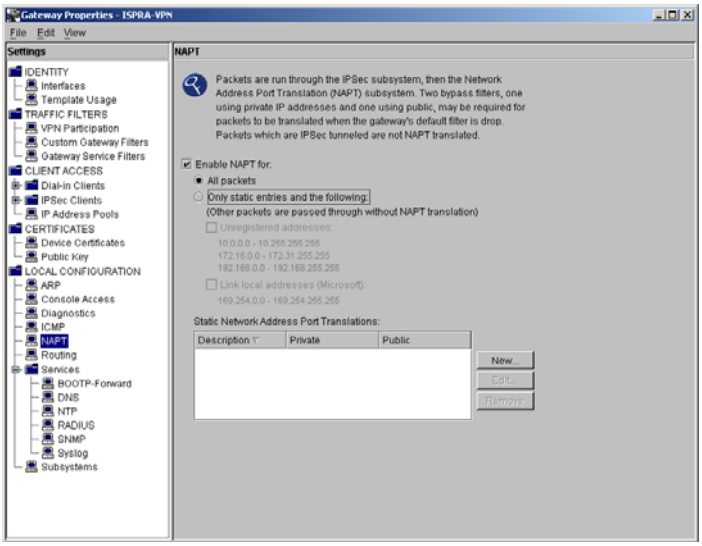
TRAFFIC FILTERS-VPN Participation:

Aquí se definen los hosts locales –conectados a través del interface Ethernet0- que serán protegidos por la VPN con respecto al exterior –hosts accedidos a través del interface Ethernet1-



LOCAL CONFIGURATION-Services-NAPT:

Editamos las propiedades del gateway y, dentro del grupo de opciones LOCAL CONFIGURATION, accedemos al NAPT (Network Address and Port Translation). Allí podemos habilitar la traducción de direcciones para un determinado subconjunto de la LAN, o para todas. En nuestro caso, como no disponemos de ninguna dirección permitida en Internet seleccionamos la traducción para todo el tráfico.

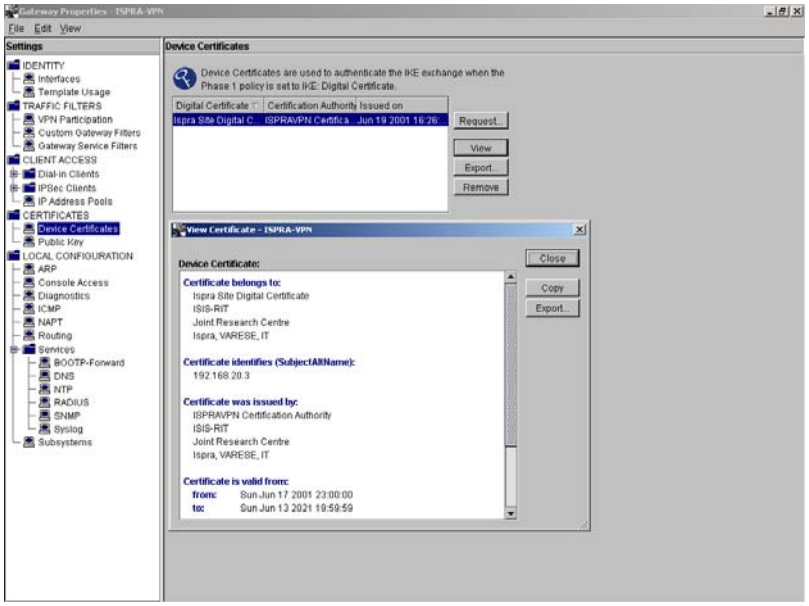


CERTIFICATES

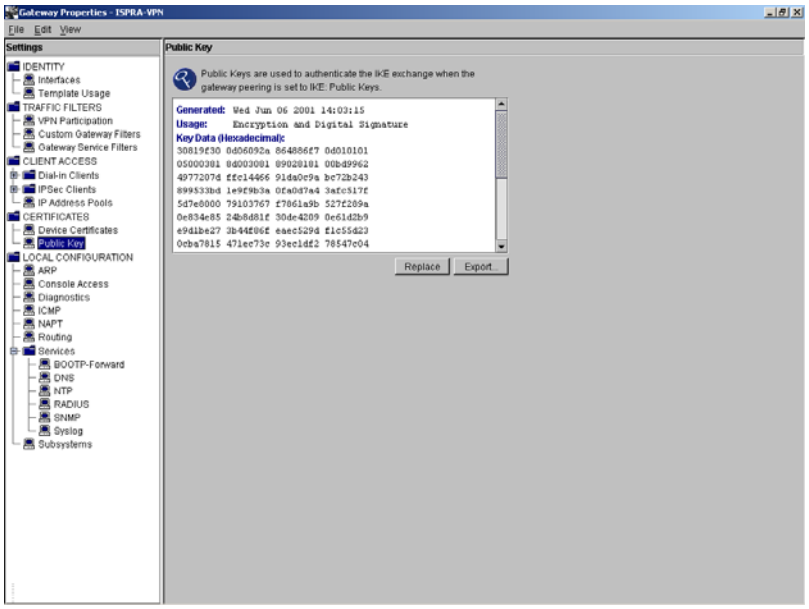
Para utilizar los métodos de autentificación con clave pública y /o certificado digital utilizaremos aquellos (clave y certificado) que nos proporciona el dispositivo. Dentro de las propiedades del gateway, en el grupo de opciones CERTIFICATES, accedemos a ambos:

Device Certificates

Certificado generado por la CA interna:



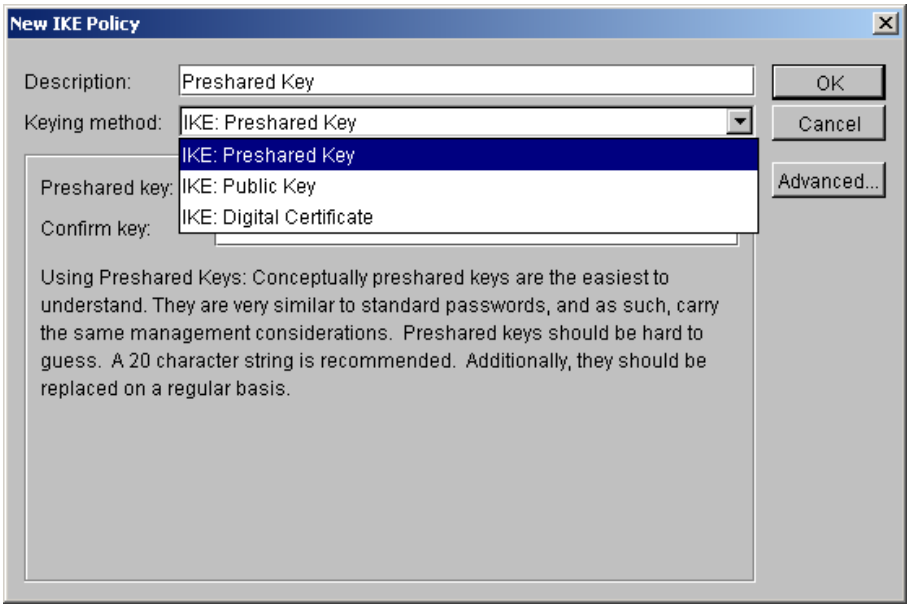
Public Key
Clave pública de 2048 bytes:



Una vez establecidas las propiedades de nuestro gateway y dado que el gateway de la otra parte no será gestionado por nosotros, podemos pasar a configurar las propiedades de nuestra VPN.

Primero procederemos a programar la política que será usadas en nuestra VPN:
IKE Policy:
POLICY CONFIGURATION-IKE Policies:

Las opciones son tres: Preshared Key, Public Key y Digital Certificate. Dependiendo de cuál escojamos, tendremos que asegurarnos de disponer de todos los objetos (claves y/o certificados) para poder llevarla a cabo.

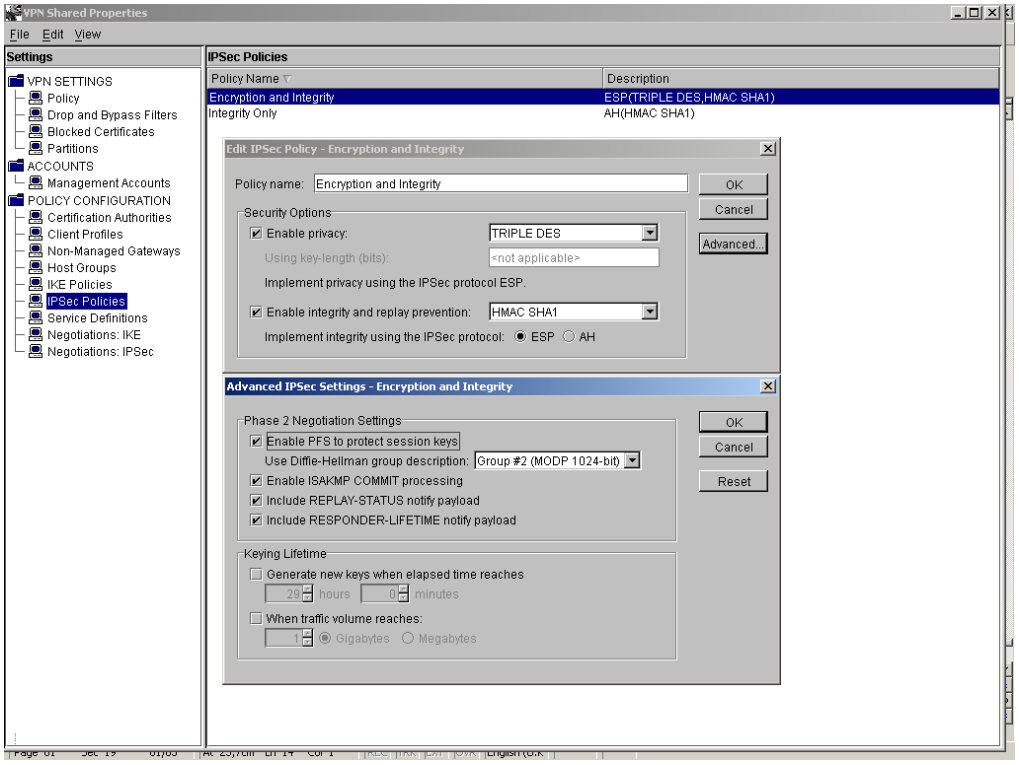


POLICY CONFIGURATION-Ipsec Policicies:

En cuanto a las opciones disponibles con el tráfico IPSec. Primero debemos elegir los servicios que utilizaremos (Encryption y/o Integrity –al menos uno de los dos!!-).

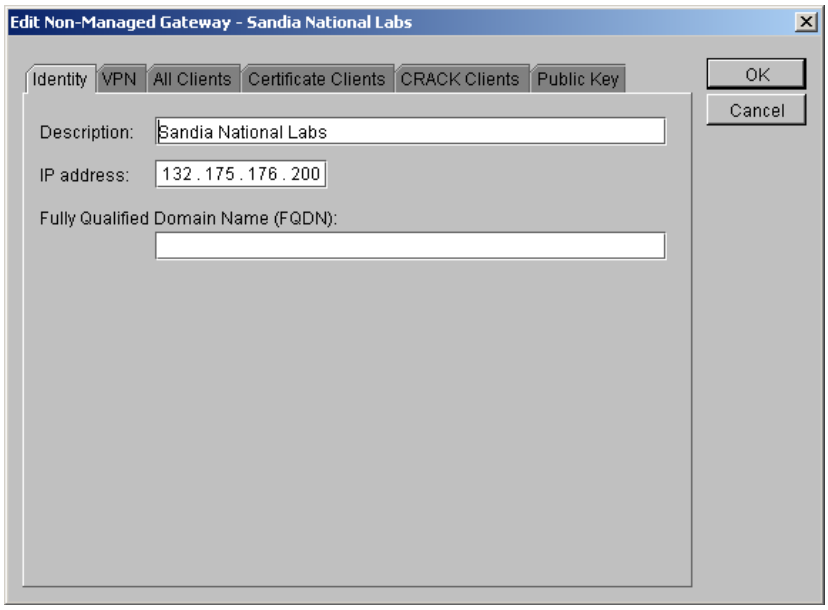
En el caso de la encriptación podemos escoger entre los siguientes algoritmos: 3DES, DES, CAST RC5 y Blowfish; con aquellos algoritmos que lo permiten –CAST, RC5 y Blowfish- podemos programar la longitud de la clave.

Para la integridad de la información primero escogemos el protocolo (ESP o AH). Y, a continuación, el algoritmo de HASH que será utilizado –SHA1 o MD5-. (El más usado a nivel práctico es el SHA1).



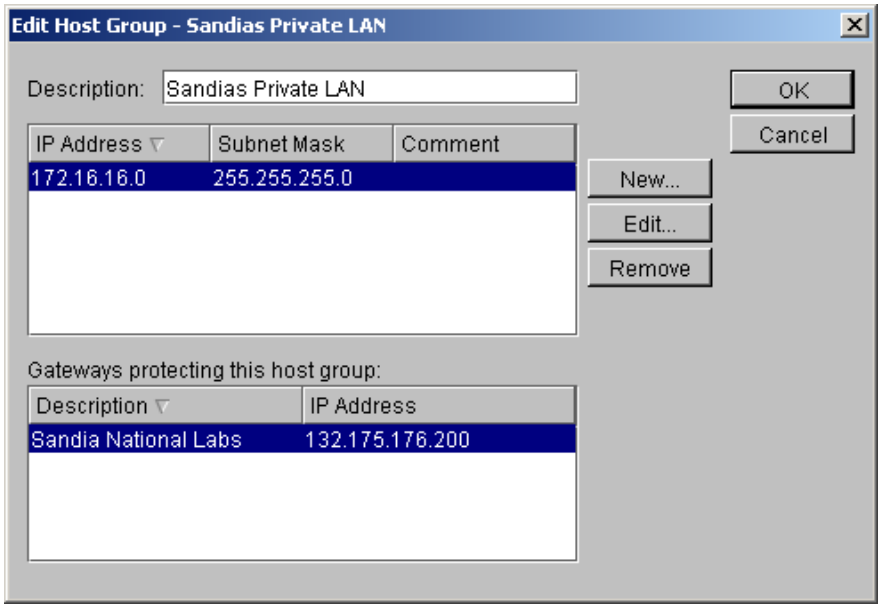
Ahora podemos programar las características del gateway no gestionado por nosotros. Deberemos explicitar: descripción, dirección IP –interface externa-, el grupo de hosts que protege dicho gateway, la política IPSec que usaremos –de entre aquellas que hayamos definido-, la Certification Authority (CA) que usaremos para identificar los certificados, así como su llave pública –que deberemos conseguir intercambiándola por la nuestra de forma segura (entrega en mano o usando una conexión segura ya establecida)-

POLICY CONFIGURATION-Non-Managed Gateways



POLICY CONFIGURATION-Host Groups

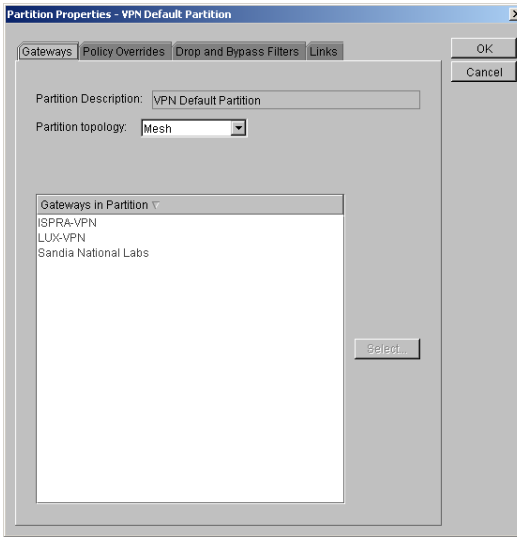
Aquí configuramos –si no lo hemos hecho ya desde las propiedades del "Non-managed Gateways"- aquellos hosts con los que nos comunicaremos de forma segura a través del VPN.

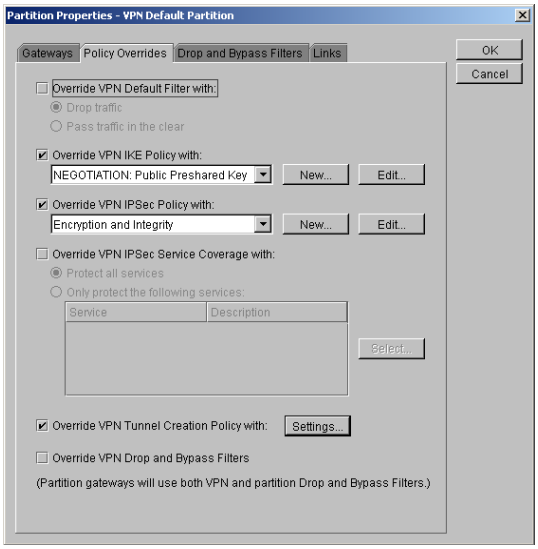


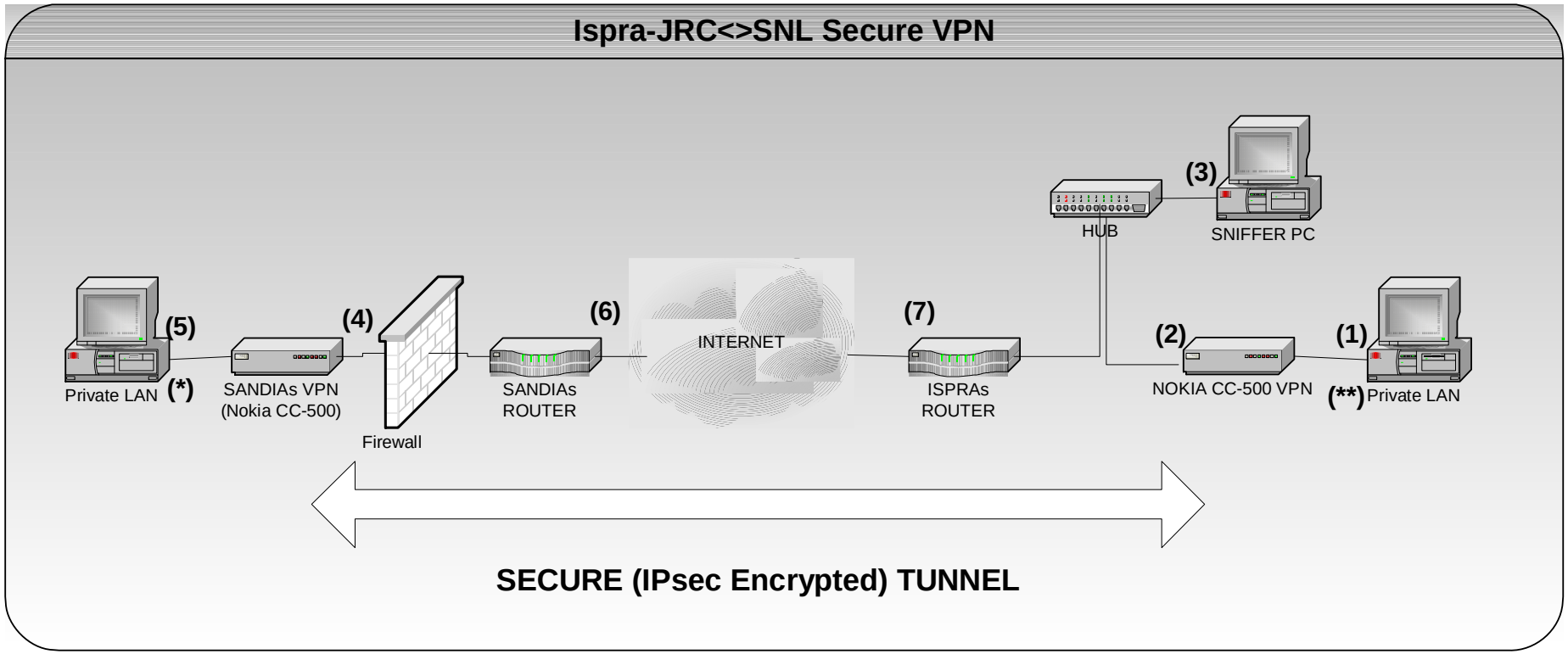
Nos falta por definir el tipo de conexión entre los diferentes Gateways de nuestra VPN, que puede ser: Mesh (interconexión entre todos los gateways de la partición),

Hub-and-Spoke (existe un gateway master) o personalizada (se pueden distinguir políticas de conexión diferentes entre diferentes Gateways), además, desde las propiedades de la partición podemos sobreescribir los filtros por defecto así como las políticas de conexión ya establecidas.

VPN SETTINGS-Partitions







SNL addresses:

(5) Private Subnet: 172.16.16.0/16 gw 172.16.16.2

(4) VPN External Interface: 132.175.176.142 subnet mask 255.255.255.192

(6) Router SNL: 132.175.176.136

(*) Server SNL: 172.16.16.3

Ispra addresses:

(1) Private Subnet: 192.168.20.0/16 gw 192.168.20.1

(2) VPN External Interface: 139.191.7.140

(3) Sniffer machine: 139.191.7.141

(7) Router Ispra: 139.191.7.1

(**) Server Ispra: 192.168.20.2

Ipsec parameters proposed:

Authentication methods: 1) Shared passphrase, 2) Public keys and 3) Digital certificates
Integrity algorithm: SHA-1 (DH Group #2)
Encryption protocol (algorithm): ESP (3DES)

5 CONCLUSIONES

Los tests llevados a cabo para la implementación de la red privada virtual han permitido conocer lo siguiente:

El estándar Ipsec no está siendo todavía utilizado de forma masiva, pero todo indica –esfuerzo de fabricantes- que lo será en un corto plazo de tiempo.

La interoperabilidad entre productos de diferentes fabricantes no se cumple en todos los casos. Peor aún, hay un número importante de fabricantes que utilizan el "full compliant Ipsec" de cualquier modo. Sin embargo, luego resulta que no cumplen todos los requisitos del protocolo Ipsec.

La gestión de las diferentes soluciones está mejorando conforme pasa el tiempo. Un buen ejemplo es el Nokia CC-500 VPN con un interfaz claro y sencillo de utilizar. Un ejemplo negativo sería en este caso la versión probada del Bigfire de la empresa BIODATA.

En lo que respecta a las soluciones software: tanto la solución de NAI como la de SSH son muy completas, proporcionando estadísticas y un número suficiente de algoritmos de encriptación. Además, el interface resulta bastante sencillo.

En cuanto al rendimiento en el uso de la transmisión encriptada cabe decir que no ha habido un decremento significativo de la velocidad en ningún caso. (Tan solo si se activan las auditorías el rendimiento cae drásticamente).

Por último, indicar que dado el auge creciente de Internet, con una población de usuarios que crece sin parar, el tema de seguridad a través de la misma cobra una especial relevancia y, sin duda, la utilización de Ipsec permite afrontarlo exitosamente.

6 BIBLIOGRAFÍA

[CDA01] Carlton R. Davis, “IPSec: Securing VPNs,” Osborne McGraw-Hill, 3 May 2001.

[KAT98] Kent, S., Atkinson, R. “Security Architecture for the Internet Protocol” Request for Comments (RFC 2401), November 1998.

[GLE00] Gleeson, B., Lin, A., Heinanen, J., Armitage, G. Malis, A. “A Framework for IP Based Virtual Private Networks”, Request for Comments (RFC 2764) February 2000.

[FEH98] Ferguson, P., Huston, G. “What is a VPN?”, Cisco Systems, Revision 1, April 1998.

[MOV01] movian VPN White Paper, “Integrating Wireless Devices into VPN Infrastructures”, http://www.certicom.com, 2001.

[MEN99] Mendivil, I.“El ABC de los documentos electrónicos seguros”, SeguriData, borrador-documento en proceso, 7 octubre 1999.

[NA99] Network Associates, Inc.,“An introduction to Cryptography”, http://www.nai.com, 1999.

[NOKA01] “Nokia VPN Gateway Technology Overview”, Version 3.0, Nokia Inc., 2001

[NOKB01] “Nokia VPN Client Administration and User Guide”, Version 3.0, Nokia Inc., January 2001

[NOKC01] “Nokia VPN Gateway Configuration Guide”, Version 3.0, Nokia Inc., 2001.

[NOKD01] “Nokia VPN Gateway Quick Start Guide”, Version 3.0, Nokia Inc., 2001.

[NOKE01] “Nokia VPN Gateway Command-Line Summary”, Version 3.0, Nokia Incl, 2001.

[BIG99] “BIGfire +: User Manual”, V9908+, Biodata, 1999.

[THA98] Thayer, R., Doraswamy, N., Glenn, R. “IP Security Document Roadmap”, Request for Comments, (RFC 2411), November 1998.\

[KRY01] Krywaniuk, A. “Security Properties of the Ipsec Protocol Suite,” INTERNET-DRAFT <draft-krywaniuk-ipsec-properties-00.txt>, July 9 2001.

[SCH95] Schneier, B. “Applied Cryptography”, John Wiley and Sons, Paperback, March 1995

[SCH00] Schneier, B. “Secrets and Lies”, John Wiley and Sons, Hardcover, 1 September, 2000

[CHI01] Chirillo, J. “Hack Attacks Revealed” John Wiley and Sons, Paperback, 3 May, 2001