Grupo de Trabajo en Red
Request for Comments: 2409
Categoría: Pila de Estándares

D. Harkins
D. Carrel
cisco Systems
Noviembre 1998

Traducción al castellano:
Hugo Adrian Francisconi <adrianfrancisconi@yahoo.com.ar>
Agosto 2005

El Intercambio de Claves en Internet (IKE)

Estado de este documento

Este documento especifica un protocolo de Internet en vías de estandarización para la comunidad de Internet y solicita debate y sugerencias para mejorarlo. Por favor, remítase a la edición actual de "Estándares de Protocolos Oficiales de Internet" (STD 1) para conocer el estado de estandarización y status de este protocolo. La distribución de este memorándum es ilimitada.

Aviso de Copyright

Lista de contenido

1. Resumen

ISAKMP ([MSST98]) proporciona un marco para la autentificación y el intercambio de claves pero no los define. ISAKMP esta diseñado para ser un intercambio de claves independiente; es decir, está diseñado para soportar una gran cantidad de intercambios de claves diferentes.

Oakley ([Orm96]) describe una serie de intercambios de claves --llamados "modos"-- y detalla los servicios proporcionados por cada uno (por ejemplo, perfect forward secrecy para claves, la protección de identidad, y autentificación).

SKEME ([SKEME]) describe una técnica de intercambio de claves que proporciona anonimato, repudiabilidad, y renovación rápida de claves.

Este documento describe un protocolo usando partes de Oakley y partes de SKEME conjuntamente con ISAKMP para obtener material clave autentificado para usarse con ISAKMP, y para otras asociaciones de seguridad tales como las de AH y ESP del DOI de IPsec de la IETF.

2. Argumento

Este documento describe un protocolo híbrido. El propósito es negociar, y proporcionar material clave autentificado para, las asociaciones de seguridad de una manera protegida.

Los procedimientos que implementa este documento pueden ser utilizados en las negociaciones de las redes privadas virtuales (VPNs), como así también para proporcionar a un usuario remoto de un sitio remoto (cuya dirección IP no necesita ser conocida de antemano) acceso a un host o red seguro.

La negociación de cliente es soportada. El modo cliente es donde las partes negociantes no son la de los extremos para los cuales la negociación de la Asociación de Seguridad (SA) se esta llevando a cabo. Cuando se usa el modo cliente, las identidades de las partes de los extremos quedan ocultas.

Esto no implementa el protocolo entero de Oakley, sino solamente un subconjunto necesario para satisfacer sus metas. No demanda conformidad con el protocolo entero de Oakley ni es dependiente de ninguna forma del protocolo Oakley.

Asimismo, esto no implementa el protocolo entero de SKEME, sino solamente el método de encriptación de claves pública para la autentificación y su concepto de rápido recambio de claves (re-keying) usando un intercambio nonce. Este protocolo no es dependiente de ninguna forma del protocolo SKEME.

3. Términos y Definiciones

3.1 Requisitos Terminológicos

Las palabras DEBE, NO DEBE, REQUERIDO, PODER, NO PODER, DEBERÍA, NO DEBERÍA, RECOMENDADO, PUEDE y OPCIONAL, cuando aparezcan en esté documento, deben interpretarse como se describe en [Bra97].

3.2 Notación

La siguiente notación se utiliza a través de todo este documento.

HDR es una cabecera de ISAKMP cuyo tipo de intercambio es el modo. Cuando está escrito como HDR* indica carga encriptada.

SA es una Carga SA de negociación con una o más propuestas. Un iniciador PODRÍA proporcionar múltiples propuestas para la negociación; un respondedor DEBE contestar solamente una.

<P>_b indica el cuerpo de la carga <P>-- la [cabecera de la] carga genérica de ISAKMP no esta incluida.

SAi_b es el cuerpo entero de la Carga SA (menos la cabecera de carga genérica de ISAKMP)-- es decir el DOI, la situación, todas las propuestas y todas las transformaciones ofrecidas por el Iniciador.

CKY-I y CKY-R son las cookies del Iniciador y del Respondedor, respectivamente, en la cabecera de ISAKMP.

g^xi y el g^xr son los valores públicos de Diffie-Hellman ([DH]) del Iniciador y del Respondedor respectivamente.

g^xy es el secreto compartido de Diffie-Hellman.

KE es la Carga de Intercambio de claves, la cual contiene la información pública intercambiada en un intercambio de Diffie-Hellman. No hay codificación particular (por ejemplo un TLV) usado para los datos de una carga KE.

Nx es la Carga Nonce; x puede ser: i o r para el iniciador y el respondedor de ISAKMP respectivamente.

IDx es la Carga de Identificación para "x". x puede ser: "ii" o "ir" para el iniciador y el respondedor de ISAKMP respectivamente durante la fase uno de la negociación; o "ui" o "ur" para el iniciador y el respondedor respectivamente durante la fase dos. El formato de la carga de identificación para el DOI de Internet se define en [Pip97].

SIG es la Carga de la Firma. Los datos para firmar son un intercambio específico.

CERT es la Carga de Certificado.

HASH (y cualquier derivado por ejemplo HASH(2) o HASH_I) es la Carga HASH. Los contenidos del Hash son específicos del método de autentificación.

prf(clave, msg) es la función de claves seudo-aleatoria-- a menudo una función de claves hash-- usada para generar una salida determinista que aparece seudo-aleatoriamente. Los prf se utilizan para derivar claves [de una clave obtener varias] y para la autentificación (es decir tal como una clave MAC). (Ver [KBC96].)

SKEYID es una cadena derivada del material secreto conocido solamente por los participantes activos en el intercambio.

SKEYID_e es el material clave usado por la SA de ISAKMP para proporcionar confidencialidad a sus mensajes.

SKEYID_a es el material clave usado por la SA de ISAKMP para autentificar sus mensajes.

SKEYID_d es el material clave usado para derivar las claves para las asociaciones de seguridad no-ISAKMP.
<x>y indica que "x" está encriptado con la clave "y".

--> significa comunicación del "iniciador al respondedor" (peticiones).

<-- significa comunicación del "respondedor al iniciador" (respuestas).

| significa concatenación de la información-- por ejemplo X | Y es la concatenación, de X con Y.

[x] indica que x es opcional.

La encriptación del mensaje (es denotado por un "*" después de la cabecera de ISAKMP) DEBE comenzar inmediatamente después de la cabecera de ISAKMP. Cuando se protege la comunicación, todas las cargas que siguen a la cabecera de ISAKMP DEBEN ser encriptadas. Las claves de encriptación son generadas por SKEYID_e en una forma que es definido por cada algoritmo.

3.3 Perfect Forward Secrecy

Cuando se usa en este documento el término Perfect Forward Secrecy (PFS) se refiere a la noción que compromete a una única clave que permitirá el acceso solamente a los datos protegidos derivados de esa única clave. Para que el PFS exista la clave usada para proteger la transmisión de datos NO DEBE ser usada para derivar claves adicionales, y si la clave usada para proteger la transmisión de los datos fue derivada de otro material clave, ese material NO DEBE ser usado para derivar más claves.

Perfect Forward Secrecy para claves e identidades es proporcionada en este protocolo. (Secciones 5.5 y 8.)

3.4 Asociación de Seguridad

Una asociación de seguridad (SA) es un conjunto de políticas y de clave(s) usadas para proteger información. La SA de ISAKMP es un conjunto de políticas y clave(s) compartidas usadas por los usuarios de la negociación en este protocolo para proteger sus comunicaciones.

4. Introducción

Oakley y SKEME definen cada uno, un método para establecer y autentificar intercambios de claves. Esto incluye la construcción de las cargas, el trasporte de cargas de información, el orden en la cual se procesan y cómo se utilizan.

Mientras que Oakley define "modos", ISAKMP define "fases". La relación entre los dos es muy simple y IKE presenta diferentes intercambios como modos que funcionan en una de las dos fases.

La fase 1 es donde los dos usuarios de ISAKMP establecen un canal seguro, autentificado en el cual comunicarse. Este se llama SA de ISAKMP (o SA ISAKMP). El "Modo Principal" y el "Modo Agresivo" se llevan a cabo en un intercambio de fase 1. El "Modo Principal" y el "Modo Agresivo" SE DEBEN utilizar SOLAMENTE en la fase 1.

La fase 2 es donde las SAs se negocian en representación de servicios tales como IPsec o cualquier otro servicio que necesite el material clave y/o negociación de parámetros. El "Modo Rápido" se lleva a cabo en un intercambio de fase 2. EL "Modo Rápido" SE DEBE utilizar SOLAMENTE en la fase 2.

El "Modo Nuevo Grupo" no está realmente ni en la fase 1 ni en la 2. Sigue a la fase 1, pero sirve para establecer un nuevo grupo que pueda ser utilizado en futuras negociaciones. El "Modo Nuevo Grupo" SE DEBE utilizar SOLAMENTE después de la fase 1.

La SA ISAKMP es bidireccional. Es decir, una vez establecida, cualquier parte puede iniciar intercambios en Modo Rápido, Informativos, y Modo Nuevo Grupo. A través del documento de ISAKMP, la SA ISAKMP es identificada por la cookie del Iniciador seguida por la cookie del Respondedor-- el papel de cada parte en el intercambio de la fase 1 dictaminará cual es la cookie del Iniciador. El orden de la cookie establecida por el intercambio de la fase 1 continúa identificando la SA de ISAKMP sin importar la dirección de los intercambios de Modo Rápido, Informativo, o Nuevo Grupo. Es decir, las cookies NO DEBEN alternar lugares cuando la dirección de la SA ISAKMP cambia.

Con el uso de las fases de ISAKMP, una implementación puede lograr claves muy rápidamente cuando sea necesario. Una simple negociación de fase 1 se puede utilizar para más de una negociación de fase 2. Una simple negociación de fase 2 puede solicitar múltiples Asociaciones de Seguridad. Con estas optimizaciones, una implementación puede perder menos de un viaje de ida y vuelta por SA, así como también menos de una exponenciación de DH (Diffie-Hellman) por SA. El "Modo Principal" para la fase 1 proporciona protección de

identidad. Cuando la protección de identidad no es necesaria, el "Modo Agresivo" se puede utilizar para reducir futuros viajes de ida y vuelta. Las indicaciones del programador para hacer estas optimizaciones son incluidas debajo. Se debe también observar que usar la encriptación de clave pública para autentificar un intercambio de Modo Agresivo proporcionará protección de identidad.

Este protocolo no define su propio DOI. La SA ISAKMP, establecida en la fase 1, PUEDE usar el DOI y la situación de un servicio no-ISAKMP (tal como el DOI de IPsec de la IETF [Pip97]). En este caso una implementación PUEDE elegir restringir el uso de la SA ISAKMP para el establecimiento de SAs para los servicios del mismo DOI. Alternativamente, la SA ISAKMP SE PUEDE establecer con el valor cero en el DOI y la situación (véase [MSST98] para una descripción de estos campos) y en este caso las implementaciones serán libres de establecer los servicios de seguridad para cualquier DOI definido usando esta SA ISAKMP. Si un DOI de valor cero se utiliza para el establecimiento de una SA de fase 1, la sintaxis de las cargas de identidad usadas en la fase 1 es la definida en [MSST98] y no la de cualquier DOI-- por ejemplo [Pip97]-- la cual pueda ampliar la sintaxis y semántica de futuras identidades.

Los siguientes atributos son utilizados por IKE y son negociados como parte de la SA ISAKMP. (Estos atributos pertenecen solamente a la SA ISAKMP y no a cualquier SA que ISAKMP pueda negociar en nombre de otros servicios.)

- algoritmo de encriptación

- algoritmo de hash

- método de autentificación

- información sobre un grupo al cual realizarle Diffie-Hellman.

Todos estos atributos son obligatorios y DEBEN ser negociados. Además, es posible negociar opcionalmente una función seudo-aleatoria ("prf"). (No hay actualmente funciones seudo-aleatorias negociables definidas en este documento. Los valores de atributo de uso privado pueden ser usados para negociar prf entre grupos concernientes.) Si un "prf" no es negociado, la versión de HMAC (véase [KBC96]) del algoritmo de hash negociado se utiliza como función seudo-aleatoria. Otros atributos no-obligatorios se describen en el Apéndice A. El algoritmo hash seleccionado DEBE soportar el modo nativo y el modo HMAC.

El grupo de Diffie-Hellman DEBE ser especificado usando una descripción definida de grupo (sección 6) o definiendo todos los atributos de un grupo (sección 5.6). Los atributos de grupo (tales como tipo o primo, vea el Apéndice A) NO SE DEBEN ofrecer conjuntamente con un grupo previamente definido (una descripción del grupo reservado o una descripción de uso privado que es establecida después de la finalización de un intercambio de Modo Nuevo Grupo).

Las implementaciones IKE DEBEN soportar los siguientes valores de atributo:

- DES [DES] en modo CBC con [claves] débiles, y semi-débiles, control de claves (las claves débiles y semi-débiles se describen en [Sch96] y se enumeran en el Apéndice A). La clave se deriva según el Apéndice B.

- MD5 [MD5] y SHA [SHA].

- Autentificación a través de claves pre-compartidas.

- Grupo MODP número uno por defecto (véase abajo).

Además, las implementaciones IKE DEBERÍAN soportar: 3DES para la encriptación; Tiger ([TIGER]) para el hash; la Firma Digital Estándar (DSS), firmas RSA [RSA] y autentificación con encriptación de clave pública RSA; y grupo MODP número 2. Las implementaciones IKE PUEDEN soportar cualquier algoritmo de encriptación definido en el Apéndice A y PUEDEN soportar los grupos ECP y EC2N.

Los modos de IKE descriptos aquí DEBEN ser implementados siempre que se implemente el DOI de IPsec de la IETF [Pip97]. Los otros DOIs PUEDEN utilizar los modos descriptos aquí.

5. Intercambios

Hay dos métodos básicos usados para establecer un intercambio de claves autentificado: Modo Principal y Modo Agresivo. Cada uno genera material clave autentificado a partir de un intercambio efímero de Diffie-Hellman. El Modo Principal DEBE ser implementado; el Modo Agresivo DEBERÍA ser implementado. Además, el Modo Rápido SE DEBE implementar como mecanismo para generar nuevo material clave y para negociar servicios de seguridad no ISAKMP. Además, el Modo Nuevo Grupo SE DEBERÍA implementar como un mecanismo para definir grupos privados para intercambios de Diffie-Hellman. Las implementaciones NO DEBEN cambiar los tipos de intercambio en el medio de un intercambio.

Los intercambios se atienen a la sintaxis de la carga de ISAKMP, codificación de los atributos, intervalo y retransmisiones de mensajes, y mensajes informativos-- por ejemplo una respuesta de notificación es enviada cuando, por ejemplo, una propuesta es inaceptable, o una verificación de firma o una desencriptación no fue exitosa, etc.

La carga SA DEBE preceder al resto de las cargas en un intercambio de Fase 1. Excepto donde se indique lo contrario, no existan requisitos para las cargas de ISAKMP en ningún mensaje cuando estén en algún orden en particular.

El valor público de Diffie-Hellman colocado en una carga KE, en un intercambio de fase 1 o fase 2, DEBE tener la longitud del grupo negociado de Diffie-Hellman, en caso de necesidad, rellenar el valor pre-pendiente con ceros [by pre-pending the value with zeros].

La longitud de la carga nonce DEBE ser entre 8 y 256 bytes inclusive.

El Modo Principal es una ejemplificación del Intercambio de Protección de Identidad de ISAKMP: los primeros dos mensajes negocian la política; los dos siguientes intercambian los valores públicos de Diffie-Hellman y datos auxiliares (por ejemplo nonces) necesarios para el intercambio; y los dos últimos mensajes autentifican el Intercambio de Diffie-Hellman. El método de autentificación negociado como parte inicial del intercambio de ISAKMP influencia en la composición de las cargas pero no en su propósito. El XCHG para el Modo Principal es el Intercambio Protección de Identidad ISAKMP.

Similarmente, el Modo Agresivo es una ejemplificación del Intercambio Agresivo de ISAKMP. Los primeros dos mensajes negocian la política, intercambian los valores públicos de Diffie-Hellman y los datos auxiliares necesarios para el intercambio, y las identidades. Además el segundo mensaje autentifica al respondedor. El tercer mensaje autentifica al iniciador y proporciona una prueba de la participación en el intercambio. El XCHG para el Modo Agresivo es el Intercambio Agresivo de ISAKMP. El mensaje final NO PUEDE ser enviado bajo la protección de la SA ISAKMP dado que cada parte pospone la exponenciación, si lo desea, hasta que la negociación de este intercambio sea completada. Las descripciones gráficas del Modo Agresivo muestran la carga final en limpio; no es necesario que esté.

Los intercambios en IKE no son abiertos y tienen un número fijo de mensajes. La Recepción de una carga de Petición de Certificado NO DEBE ampliar el número de mensajes transmitidos o esperados.

La negociación de SA está limitada en el Modo Agresivo. Debido a los requerimientos en la construcción de mensajes, el grupo en el cual el intercambio de Diffie Hellman se ejecuta no puede ser negociado. Además, métodos de autentificación diferentes pueden limitar aun más la negociación de los atributos. Por ejemplo, la autentificación con encriptación de clave pública no puede ser negociada y cuando se usa el método revisado de encriptación de clave pública para autentificar, el cifrado y el hash no pueden ser negociados. Para las situaciones donde se requiere la capacidad de negociación de numerosos atributos de IKE, el Modo Principal puede ser requerido

El Modo Rápido y el Modo Nuevo Grupo no tienen ningún análogo en ISAKMP. Los valores XCHG para el Modo Rápido y el Modo Nuevo Grupo se definen en el Apéndice A.

El Modo Principal, el Modo Agresivo, y el Modo Rápido realizan la negociación de la SA. Las ofertas SA consisten en cargas de Transformación(s) encapsulada en la Carga de la Propuesta(s) encapsulada en la carga SA. Si múltiples ofertas se están realizando para los intercambios de fase 1 (Modo Principal y Modo Agresivo) estas DEBEN consistir de múltiples Cargas de Transformación para una sola Carga propuesta en una sola carga SA. En otras palabras, para los intercambios de la fase 1 NO DEBE haber múltiples cargas de la Propuesta para una sola carga SA y NO DEBE haber múltiples cargas SA. Este documento no prohíbe tal postura en ofertas en intercambios de fase 2.

No hay límite en el número de ofertas que el iniciador puede enviar al respondedor pero las implementaciones PUEDEN elegir limitar el número de ofertas que examinará por razones de rendimiento.

Durante la negociación de la SA, los iniciadores presentan posibles ofertas de SA a los respondedores. Los respondedores NO DEBEN modificar los atributos de ninguna oferta, exceptuando la codificación de los atributos (véase el Apéndice A). Si el iniciador de un intercambio nota que los valores del atributo han cambiado o que se han agregado atributos o se han suprimido de una oferta realizada, esa respuesta DEBE ser rechazada.

Cuatro métodos diferentes de autentificación están permitidos con el Modo Principal o el Modo Agresivo-- firma digital, dos formas de autentificación con encriptación de clave pública, o clave pre-compartida. El valor del SKEYID es calculado por separado para cada método de autentificación.

Para las firmas:	SKEYID = prf(Ni_b | Nr_b, g^xy)
Para la encriptación de clave pública:	SKEYID = prf(hash(Ni_b | Nr_b), CKY-I | CKY-R)
Para las claves pre-compartidas:	SKEYID = prf(pre-shared-key, Ni_b | Nr_b)

El resultado del Modo Principal o del Modo Agresivo son tres grupos de material clave autentificado:

```
SKEYID_d = prf(SKEYID, g^xy | CKY-I | CKY-R | 0)
SKEYID_a = prf(SKEYID, SKEYID_d | g^xy | CKY-I | CKY-R | 1)
SKEYID_e = prf(SKEYID, SKEYID_a | g^xy | CKY-I | CKY-R | 2)
```

y acordando la política para proteger más comunicaciones. Los valores de 0, 1, y 2 de arriba son representados por un solo octeto. La clave usada para la encriptación es derivada a partir de SKEYID_e de un algoritmo especifico (véase el Apéndice B).

Para autentificar cualquier intercambio el iniciador del protocolo genera HASH_I y el respondedor genera HASH_R donde:

```
HASH_I = prf(SKEYID, g^xi | g^xr | CKY-I | CKY-R | SAi_b | IDii_b)
HASH_R = prf(SKEYID, g^xr | g^xi | CKY-R | CKY-I | SAi_b | IDir_b)
```

Para la autentificación con las firmas digitales, se firman y se verifican HASH_I y HASH_R; para la autentificación con encriptación de clave pública o claves pre-compartidas, HASH_I y HASH_R autentifican directamente el intercambio. La carga entera de identificación (incluyendo tipo de identificador, acceso, y protocolo pero excluyendo la cabecera genérica) es hasheada en HASH_I y HASH_R.

Según lo mencionado arriba, el método de autentificación negociado influencia el contenido y el uso de los mensajes para los modos de la fase 1, pero no su propósito. Al usar claves públicas para la autentificación, el intercambio de la fase 1 puede ser logrado usando firmas o usando la encriptación de clave pública (si el algoritmo lo soporta). Los siguientes son intercambios de fase 1 con diversas opciones de autentificación.

5.1 Fase 1 de IKE - Autentificación con Firmas Digitales

Usando firmas, la información auxiliar intercambiada durante el segundo viaje de ida y vuelta son nonces; el intercambio es autentificado firmando un hash mutuamente obtenido. El Modo Principal con autentificación de firma se describe de la siguiente forma:

```
     Iniciador                          Respondedor
    -----------                        -------------
     HDR, SA                    -->
                                <--     HDR, SA
     HDR, KE, Ni                -->
                                <--     HDR, KE, Nr
     HDR*, IDii, [ CERT, ] SIG_I -->
                                <--     HDR*, IDir, [ CERT, ] SIG_R
```

El Modo Agresivo con firmas en conjunción con ISAKMP se describe de la siguiente forma:

```
     Iniciador                          Respondedor
    -----------                        -------------
     HDR, SA, KE, Ni, IDii      -->
                                <--     HDR, SA, KE, Nr, IDir,
                                             [ CERT, ] SIG_R
     HDR, [ CERT, ] SIG_I       -->
```

En ambos modos, los datos firmados, SIG_I o SIG_R, son el resultado de la negociación del algoritmo de firma digital aplicado a HASH_I o a HASH_R respectivamente.

En general la firma será sobre HASH_I y HASH_R como arriba usando el prf negociado, o la versión de HMAC de la función negociada de hash (si no se negocia ningún prf). Sin embargo, esto se puede evitar para la construcción de la firma si el algoritmo de la firma se vincula a un determinado algoritmo de hash (por ejemplo DSS se define solamente con SHA 160 bit de salida). En este caso, la firma será sobre HASH_I y HASH_R como arriba, excepto usando la versión de HMAC del algoritmo de hash asociado con el método de la firma. El prf y la función hash negociadas continuarían siendo utilizados por el resto de las funciones seudo-aleatorias prescritas.

Puesto que el algoritmo de hash usado ya se conoce no hay necesidad de codificar su OID [Identificador de Objeto] en la firma. Además, no hay vínculos entre los OIDs usado para firmas RSA en PKCS Nº1 y los usados en este documento. Por lo tanto, las firmas RSA DEBEN estar codificadas con encriptación de clave pública en formato PKCS Nº1 y no con firma en el formato PKCS Nº1 (que incluye el OID del algoritmo hash). Las firmas del DSS SE DEBEN codificar como r seguido por s.

Una o más cargas de certificado PUEDEN ser pasadas opcionalmente.

5.2 Fase 1 de IKE - Autentificación con Encriptación de Clave Pública

Usando la encriptación de clave pública para autentificar el intercambio, la información auxiliar intercambiada son nonces encriptados. Cada parte habilitada para la reconstrucción del hash (comprobando que la otra parte desencriptó el nonce) autentifica el intercambio.

Para realizar la encriptación de la clave pública, el iniciador ya debe tener la clave pública del respondedor. En el caso de que el respondedor tenga múltiples claves públicas, un hash del certificado del iniciador es utilizado para encriptar la información auxiliar, la cual es pasada como parte del tercer mensaje. De esta manera el respondedor puede determinar la correspondiente clave privada usada para desencriptar las cargas encriptadas y la protección de identidad es mantenida.

Además del nonce, las identidades de las partes (IDii e IDir) también se encriptan con la clave pública de la otra parte. Si el método de autentificación es la encriptación de clave pública, las cargas nonce y la de identidad SE DEBEN encriptar con la clave pública de la otra parte. Solamente el cuerpo de las cargas son encriptadas, las cabeceras de la carga se dejan en limpio.

Al usar la encriptación para la autentificación, el Modo Principal se describe de la siguiente forma:

```
     Iniciador                          Respondedor
     -----------                        -----------
     HDR, SA                    -->
                                <--     HDR, SA
     HDR, KE, [ HASH(1), ]
       <IDii_b>PubKey_r,
         <Ni_b>PubKey_r         -->
                                                    HDR, KE, <IDir_b>PubKey_i,
                                <--                   <Nr_b>PubKey_i
     HDR*, HASH_I               -->
                                <--     HDR*, HASH_R
```

El Modo Agresivo autentificado con encriptación se describe de la siguiente forma:

```
      Iniciador                       Respondedor
     -----------                      -----------
      HDR, SA, [ HASH(1),] KE,
        <IDii_b>Pubkey_r,
         <Ni_b>Pubkey_r          -->
                                        HDR, SA, KE, <IDir_b>PubKey_i,
                                 <--         <Nr_b>PubKey_i, HASH_R
      HDR, HASH_I                -->
```

Donde HASH(1) es el hash (usando la función negociada de hash) del certificado que el iniciador está utilizando para encriptar el nonce y la identidad.

La encriptación RSA SE DEBE codificar en formato PKCS # 1. Mientras que solamente el cuerpo de las cargas de identificación y de nonce son encriptadas, los datos encriptados deben estar precedidos por una cabecera genérica válida de ISAKMP. La longitud de la carga es la longitud de la carga entera encriptada más la cabecera. La codificación de PKCS # 1 permite la determinación de la longitud actual de la carga de texto plano sobre la desencriptación.

Usar encriptación para la autentificación proporciona un intercambio posiblemente rechazable. No hay prueba (como con una firma digital) de que la conversación alguna vez tuvo lugar porque cada parte puede reconstruir totalmente ambos lados del intercambio. Además, a la seguridad se agrega la generación secreta puesto que un atacante tendría que quebrar exitosamente no solo el intercambio de Diffie-Hellman sino también la encriptación RSA. Este intercambio fue motivado por [SKEME].

Observe que, a diferencia de otros métodos de autentificación, la autentificación con encriptación de clave pública permite la protección de la identidad con Modo Agresivo.

5.3 Fase 1 de IKE - Autentificación con un Modo Revisado de Encriptación de Clave Pública

La autentificación con encriptación de clave pública tiene ventajas significativas sobre la autentificación con firmas (véase la Sección 5.2). Desafortunadamente, a costa de 4 operaciones de claves públicas (dos encriptaciones de clave pública y dos desencriptaciones de clave privada). Este modo de autentificación conserva las ventajas de la autentificación usando la encriptación de clave pública pero lo hace con la mitad de las operaciones de clave pública.

En este modo, el nonce todavía es encriptado usando la clave pública del usuario, no obstante la identidad del usuario (y el certificado si es enviado) es encriptado usando el algoritmo de encriptación

simétrico negociado (de la carga SA) con una clave derivada del nonce. Esta solución agrega una mínima complejidad pero esta condición economiza dos costosas operaciones de clave pública en cada uno de los extremos. Además, la carga de Intercambio de Claves también es encriptada usando la misma clave derivada. Esto proporciona protección adicional contra el análisis criptográfico en el intercambio de Diffie-Hellman.

Como con el método de autentificación con encriptación de clave pública (sección 5.2), una carga HASH puede ser enviada para identificar un certificado si el respondedor tiene múltiples certificados, los cuales contienen claves públicas utilizables (por ejemplo si el certificado no es para las firmas solamente, debido a las restricciones del certificado o a las restricciones algorítmicas). Si se envía la carga HASH esta DEBE ser la primera carga del segundo intercambio de mensajes y DEBE estar seguida por el nonce encriptado. Si la carga HASH no es enviada, la primera carga del segundo intercambio del mensaje DEBE ser el nonce encriptado. Además, el iniciador puede enviar opcionalmente una carga de certificado para proporcionar al respondedor una clave pública con la cual responder.

Al usar el modo revisado de encriptación para la autentificación, el Modo Principal es definido de la siguiente forma:

```
    Iniciador                          Respondedor
    -----------                        -----------
     HDR, SA                    -->
                                <--     HDR, SA
     HDR, [ HASH(1), ]
       <Ni_b>Pubkey_r,
       <KE_b>Ke_i,
       <IDii_b>Ke_i,
       [<<Cert-I_b>Ke_i]        -->
                                          HDR, <Nr_b>PubKey_i,
                                               <KE_b>Ke_r,
                                <--           <IDir_b>Ke_r,
     HDR*, HASH_I               -->
                                <--     HDR*, HASH_R
```

El Modo Agresivo autentificado con el método revisado de encriptación se describe de la siguiente forma:

```
      Iniciador                          Respondedor
     -----------                         -----------
      HDR, SA, [ HASH(1),]
        <Ni_b>Pubkey_r,
        <KE_b>Ke_i, <IDii_b>Ke_i
        [, <Cert-I_b>Ke_i ]        -->
                                            HDR, SA, <Nr_b>PubKey_i,
                                                 <KE_b>Ke_r, <IDir_b>Ke_r,
                                   <--           HASH_R
      HDR, HASH_I                  -->
```

donde HASH(1) es idéntico a la sección 5.2. Ke_i y Ke_r son las claves para el algoritmo de encriptación simétrico negociado en el intercambio de la carga SA. Solamente el cuerpo de las cargas son encriptados (con claves públicas y operaciones simétricas), las cabeceras de carga genérica se dejan en limpio. La longitud de la carga incluye ése agregado para realizar la encriptación.

Las claves simétricas encriptadas se derivan de los nonces desencriptados como se describe a continuación. Primero se calculan los valores Ne_i y Ne_r:

```
Ne_i = prf(Ni_b, CKY-I)
Ne_r = prf(Nr_b, CKY-R)
```

Las claves Ke_i y Ke_r se extraen de Ne_i y de Ne_r respectivamente como se describe en el Apéndice B usando las claves simétricas derivadas para usarse con el algoritmo de encriptación negociado. Si la longitud de salida del prf negociado es mayor o igual que la longitud de la clave requerida de cifrado, Ke_i y Ke_r se derivan de los bits más significativos de Ne_i y de Ne_r respectivamente. Si la longitud deseada de Ke_i y Ke_r excede la longitud de salida del prf el número necesario de bits es obtenido introduciendo repetitivamente el resultado del prf nuevamente dentro de sí mismo y concatenando el resultado hasta que se ha alcanzado el número necesario. Por ejemplo, si el algoritmo de encriptación negociado requiere 320 bits de clave y la salida del prf es de solamente 128 bits, Ke_i es los 320 bits de K más significativos, donde:

```
K = K1 | K2 | K3 y
K1 = prf(Ne_i, 0)
K2 = prf(Ne_i, K1)
K3 = prf(Ne_i, K2)
```

Por brevedad, solamente se muestra la derivación de Ke_i; Ke_r es idéntico. La longitud del valor 0 en el cálculo de K1 es de solo un octeto. Observe que Ne_i, Ne_r, Ke_i, y Ke_r son todas de corta vida (efímeras) y DEBEN ser descartadas después de usarse.

Excepto los requisitos de la localización de la carga hash (opcional) y de la carga nonce (obligatoria) no hay otros requisitos de carga. Todas las cargas-- en cualquier orden-- detrás de la del nonce encriptado SE DEBEN encriptar con Ke_i o Ke_r dependiendo de la dirección.

Si el modo CBC se utiliza para la encriptación simétrica entonces los vectores de inicialización (IVs) se determinan como sigue. El IV para la encriptación de la primera carga que sigue al nonce se fija a 0 (cero). El IV para las subsiguientes cargas encriptadas con la clave cifrada secreta efímera, Ke_i, es el último bloque de texto cifrado de las cargas previas. Las cargas encriptadas se rellenan hasta alcanzar el tamaño de bloque más cercano. Todo los bytes de relleno, con excepción del último, contienen 0x00. El último byte de relleno contiene el número de bytes de relleno usado, excluyendo el último. Observe que esto significa que siempre habrá relleno.

5.4 Fase 1 de IKE - Autentificación con Clave Pre-Compartida

Una clave derivada por un cierto mecanismo fuera de banda (out-of-band) se puede utilizar también para autentificar el intercambio. El establecimiento actual de esta clave está fuera del alcance de este documento.

Cuando se realiza una Autentificación con Clave Pre-Compartida, el Modo Principal se define de la siguiente forma:

```
          Iniciador                        Respondedor
          ----------                       -----------
          HDR, SA              -->
                               <--    HDR, SA
          HDR, KE, Ni          -->
                               <--    HDR, KE, Nr
          HDR*, IDii, HASH_I   -->
                               <--    HDR*, IDir, HASH_R
```

El Modo Agresivo con una clave pre-compartida se describe como sigue:

```
         Iniciador                         Respondedor
         -----------                       -----------
         HDR, SA, KE, Ni, IDii -->
                               <--    HDR, SA, KE, Nr, IDir, HASH_R
         HDR, HASH_I           -->
```

Al usar la autentificación con clave pre-compartida en el Modo Principal la clave solo se puede identificar por la dirección IP de los usuarios, puesto que HASH_I debe ser calculado antes de que el iniciador haya procesado el IDir. El Modo Agresivo permite una gama

más amplia de identificadores de confidencialidad pre-compartidos para utilizarse. Además, el Modo Agresivo permite que dos partes mantengan múltiples, claves pre-compartidas diferentes y seleccionar la correcta para un intercambio determinado.

5.5 Fase 2 - Modo Rápido

El Modo Rápido no es un intercambio íntegro (en cuanto a que esta limitado a un intercambio de fase 1), pero se utiliza como parte del proceso de negociación de la SA (fase 2) para derivar el material clave y negociar la política compartida para las SAs no ISAKMP. La información intercambiada en el Modo Rápido DEBE estar protegida por la SA ISAKMP-- es decir todas las cargas excepto la cabecera de ISAKMP están encriptadas. En Modo Rápido, una carga HASH DEBE seguir inmediatamente a la cabecera ISAKMP y una carga SA DEBE seguir inmediatamente a la de hash. Este hash autentifica el mensaje y también proporciona prueba de la actividad [liveliness].

El identificador del mensaje en la cabecera de ISAKMP identifica el Modo Rápido en curso para una SA ISAKMP determinada, la cual a su vez es identificada por las cookies en la cabecera de ISAKMP. Puesto que cada instancia de Modo Rápido utiliza un vector de inicialización único (véase el Apéndice B) es posible tener simultáneamente múltiples Modos Rápidos, basados solo en la SA ISAKMP, en curso en cualquier momento.

El modo rápido es esencialmente una negociación de SA y un intercambio de nonces que proporciona protección anti-replay. Los nonces se utilizan para generar el material clave actualizado y prevenir que los ataques de reenvío generen asociaciones de seguridad falsas. Una carga de Intercambio de Claves opcional puede ser intercambiada para permitir un intercambio de Diffie-Hellman adicional y una exponenciación por Modo Rápido. A pesar de que el uso de la carga de Intercambio de Claves con el Modo Rápido es opcional este DEBE ser soportado.

El Modo Rápido (sin la carga KE) actualiza el material clave derivado de la exponenciación en la fase 1. Esto no proporciona PFS. Usando la carga KE opcional, se realiza una exponenciación adicional y el PFS es proporcionado para el material clave.

Las identidades de la SAs negociadas en Modo Rápido se asume implícitamente que son las direcciones IP de los usuarios de ISAKMP, sin ninguna restricción implícita en el protocolo o en la cantidad de accesos permitidos, a menos que los identificadores del cliente se especifiquen en Modo Rápido. Si ISAKMP está actuando como un cliente

negociador en nombre de otra parte, las identidades de las partes SE DEBEN pasar como IDci y IDcr. La política local dictaminará si las propuestas son aceptables para las identidades especificadas. Si las identidades del cliente no son aceptadas por el respondedor en Modo Rápido (debido a la política o a otras razones), una carga de Notificación conteniendo el tipo de mensaje de Notificación, INFORMACIÓN-DEL-IDENTIFICADOR-NO-VÁLIDO (18), DEBERÍA ser enviado.

Las identidades del cliente se utilizan para identificar y para dirigir el tráfico al túnel apropiado en caso de que existan múltiples túneles entre dos usuarios y también para permitir SAs únicas y compartidas con diferentes niveles de modularidad.

Todas las ofertas hechas durante un Modo Rápido están lógicamente relacionadas y deben ser consistentes. Por ejemplo, si se envía una carga KE, el atributo que describe al grupo de Diffie-Hellman (véase la sección 6.1 y [Pip97]) SE DEBE incluir en cada transformación de cada propuesta de cada SA que es negociada. Semejantemente, si se utilizan las identidades del cliente, DEBEN aplicarse a cada SA en la negociación.

Se define el Modo Rápido como sigue:

```
     Iniciador                          Respondedor
     -----------                        -----------
     HDR*, HASH(1), SA, Ni
       [, KE ] [, IDci, IDcr ] -->
                               <--    HDR*, HASH(2), SA, Nr
                                            [, KE ] [, IDci, IDcr ]
     HDR*, HASH(3)             -->              -->
```

Donde: HASH(1) es el prf sobre el identificador del mensaje (M-ID) de la cabecera de ISAKMP concatenada con el mensaje entero que sigue al hash incluyendo todas las cabeceras de carga, pero excluyendo cualquier relleno agregado para la encriptación. HASH(2) es idéntico al HASH(1) excepto por el nonce del iniciador-- Ni, menos la carga de la cabecera-- que se agrega después del M-ID pero antes del mensaje completo. La suma del nonce en HASH(2) está para una prueba de actividad. HASH(3)-- para la actividad-- es el prf sobre el valor cero representado como un solo octeto, seguido por una concatenación de identificadores de mensajes y de dos nonces-- el del iniciador seguido por el del respondedor-- menos la carga de la cabecera. Es decir, los hashes para el intercambio antedicho son:

```
HASH(1) = prf(SKEYID_a, M-ID | SA | Ni [ | KE ] [ | IDci | IDcr )
HASH(2) = prf(SKEYID_a, M-ID | Ni_b | SA | Nr [ | KE ] [ | IDci |
          IDcr )
HASH(3) = prf(SKEYID_a, 0 | M-ID | Ni_b | Nr_b)
```

A excepción del HASH, SA, y de las cargas de identificación opcionales, no hay restricciones para el ordenamiento de las cargas en Modo Rápido. HASH(1) y HASH(2) pueden diferenciarse de la ilustración de arriba si el orden de las cargas en el mensaje difieren del ejemplo ilustrado o si cualquier carga opcional, por ejemplo la carga de notificación, ha sido encadenada al mensaje.

Si el PFS no es necesario, y las cargas KE no se intercambian, el nuevo material clave es definido de la siguiente forma:

```
KEYMAT = prf(SKEYID_d, protocol | SPI | Ni_b | Nr_b).
```

Si se desea el PFS y las cargas KE fueron intercambiadas, el nuevo material clave es definido de la siguiente forma:

```
KEYMAT = prf(SKEYID_d, g(qm)^xy | protocol | SPI | Ni_b | Nr_b)
```

donde g(qm)^xy es el secreto compartido del intercambio efímero de Diffie-Hellman en Modo Rápido.

En ambos casos, el "protocolo" y el "SPI" son la de la carga de la Propuesta de ISAKMP que contiene la Transformación negociada.

Una sola negociación de SA da lugar a dos SAs-- una de entrada y una de salida. Diferentes SPIs para cada SA (uno elegido por el iniciador, y el otro por el respondedor) garantizan una clave diferente para cada dirección. El SPI elegido por el destinatario de la SA se utiliza para derivar KEYMAT para esa SA.

Para las situaciones donde la cantidad de material clave deseado es mayor al proporcionado por el prf, el KEYMAT es ampliado introduciendo el resultado del prf nuevamente dentro de sí mismo y concatenando el resultado hasta que se ha alcanzado el material clave requerido. Es decir

```
KEYMAT = K1 | K2 | K3 | ...
done
  K1 = prf(SKEYID_d, [ g(qm)^xy | ] protocolo | SPI | Ni_b | Nr_b)
  K2 = prf(SKEYID_d, K1 | [ g(qm)^xy | ] protocolo | SPI | Ni_b |
       Nr_b)
  K3 = prf(SKEYID_d, K2 | [ g(qm)^xy | ] protocolo | SPI | Ni_b |
       Nr_b)
  etc.
```

Este material clave (con PFS o sin PFS, y derivado directamente o a través de la concatenación) SE DEBE utilizar con la SA negociada. Depende del servicio definir como las claves son derivadas del material clave.

En el caso de un intercambio efímero de Diffie-Hellman en Modo Rápido, el exponencial (g(qm)^xy) se quita del estado actual y el SKEYID_e y el SKEYID_a (derivados de la negociación de la fase 1) continúan protegiendo y autentificando la SA ISAKMP y el SKEYID_d se continúa utilizando para derivar las claves.

Usando el Modo Rápido, múltiples SA y claves pueden ser negociadas con el siguiente intercambio:

```
     Iniciador                        Respondedor
    -----------                       -----------
     HDR*, HASH(1), SA0, SA1, Ni,
       [, KE ] [, IDci, IDcr ]  -->
                                <--    HDR*, HASH(2), SA0, SA1, Nr,
                                          [, KE ] [, IDci, IDcr ]
     HDR*, HASH(3)              -->
```

El material clave se deriva idénticamente como en el caso de una sola SA. En este caso (negociando de dos cargas SA) el resultado sería cuatro asociaciones de seguridad-- dos para cada una de las SAs.

5.6 Modo Nuevo Grupo

El Modo Nuevo Grupo NO DEBE ser utilizado antes del establecimiento de la SA ISAKMP. La descripción de un nuevo grupo solamente DEBE seguir a la negociación de fase 1. (No es un intercambio de fase 2, sin embargo).

```
     Iniciador                        Respondedor
    -----------                       -----------
     HDR*, HASH(1), SA          -->
                                <--     HDR*, HASH(2), SA
```

donde HASH(1) es la salida del prf, usando el SKEYID_a como la clave, y el identificador de mensaje de la cabecera de ISAKMP concatenado con la propuesta SA entera, el cuerpo y la cabecera, como los datos; HASH(2) es la salida del prf, usando SKEYID_a como la clave, y el identificador del mensaje de la cabecera de ISAKMP concatenado con la contestación como los datos. Es decir, los hashes para el intercambio antedicho son:

```
   HASH(1) = prf(SKEYID_a, M-ID | SA)
   HASH(2) = prf(SKEYID_a, M-ID | SA)
```

La propuesta especificará las características del grupo (véase el Apéndice A, "Números Asignados a los Atributos"). Las descripciones de grupo para los Grupos privados DEBEN ser mayor o igual a 2^15. Si el grupo no es aceptado, el respondedor DEBE contestar con una carga de Notificación conteniendo el tipo de mensaje de Notificación ATRIBUTOS-NO-SOPORTADOS (13).

Las implementaciones de ISAKMP PUEDEN requerir que grupos privados expiren con la SA bajo la cual fueron establecidos.

Los grupos pueden ser negociados directamente en la propuesta SA con el Modo Principal. Para hacer esto las partes que lo componen-- para un grupo MODP, tipo, primo y generador; para un grupo EC2N, tipo, Polinomio Irreducible, Primer Grupo Generado, Segundo Grupo Generado, Grupo Curva A, Grupo Curva B y Orden del Grupo-- se pasan como atributos SA (véase el Apéndice A). Alternativamente, la naturaleza del grupo se puede ocultar usando el Modo Nuevo Grupo y solamente el identificador del grupo se pasa en limpio durante la negociación de la fase 1.

5.7 Intercambios Informativos de ISAKMP

Este protocolo protege a los Intercambios Informativos de ISAKMP cuando es posible. Una vez que la asociación de seguridad de ISAKMP haya sido establecida (y se han generado SKEYID_e y SKEYID_a) los Intercambios Informativos de ISAKMP, cuando se usan con este protocolo, son de la siguiente forma:

```
     Iniciador                          Respondedor
    -----------                         -----------
     HDR*, HASH(1), N/D      -->
```

donde N/D es una Carga de Notificación de ISAKMP o una Carga de Cancelación de ISAKMP y HASH(1) es la salida del prf, usando SKEYID_a como la clave, y un M-ID único para este intercambio concatenado con la carga informativa (una Notificación o Cancelación) como los datos. Es decir, el hash para el intercambio antedicho es:

```
   HASH(1) = prf(SKEYID_a, M-ID | N/D)
```

Como se observó el identificador del mensaje en la cabecera de ISAKMP-- y utilizado en el cálculo del prf-- es único para este intercambio y NO DEBE ser igual que el identificador de mensaje de otro intercambio de fase 2 que generó este intercambio informativo. La derivación del vector de inicialización, usado con SKEYID_e para encriptar este mensaje, se describe en el Apéndice B.

Si la asociación de seguridad de ISAKMP aún no se ha establecido al momento del Intercambio Informativo, el intercambio debe hacerse en "limpio" sin una carga HASH adicional.

6 Grupos de Oakley

Por medio de IKE, el grupo dentro del cual se realiza el intercambio de Diffie-Hellman es negociado. Cuatro grupos --con valores de 1 a 4-- se definen debajo. Estos grupos son originados con el protocolo Oakley y por lo tanto se llaman "Grupos de Oakley". La clasificación del atributo para el "grupo" se define en el Apéndice A. Todos los valores de 2^15 y superiores se utilizan para los identificadores de grupos privados. Para una discusión sobre la robustez de los grupos de Oakley por defecto ver la sección de Consideraciones de Seguridad debajo.

Estos grupos fueron generados por Richard Schroeppel en la universidad de Arizona. Las características de estos grupos se describen en [Orm96].

6.1 Grupo 1 de Oakley

Las implementaciones de Oakley DEBEN soportar un grupo MODP con el siguiente número primo y el generador. Este grupo es asignado al identificador 1 (uno).

El número primo es: 2^768 - 2 ^704 - 1 + 2^64 * { [2^638 pi] + 149686} Su valor hexadecimal es

FFFFFFFF FFFFFFFF C90FDAA2 2168C234 C4C6628B 80DC1CD1
29024E08 8A67CC74 020BBEA6 3B139B22 514A0879 8E3404DD
EF9519B3 CD3A431B 302B0A6D F25F1437 4FE1356D 6D51C245
E485B576 625E7EC6 F44C42E9 A63A3620 FFFFFFFF FFFFFFFF

El generador es: 2.

6.2 Grupo 2 de Oakley

Las implementaciones de IKE DEBERÍAN soportar un grupo MODP con el siguiente número primo y el generador. Este grupo es asignado al identificador 2 (dos).

El número primo es: 2^1024 - 2^960 - 1 + 2^64 * { [2^894 pi] + 129093} Su valor hexadecimal es

```
FFFFFFFF FFFFFFFF C90FDAA2 2168C234 C4C6628B 80DC1CD1
29024E08 8A67CC74 020BBEA6 3B139B22 514A0879 8E3404DD
EF9519B3 CD3A431B 302B0A6D F25F1437 4FE1356D 6D51C245
E485B576 625E7EC6 F44C42E9 A637ED6B 0BFF5CB6 F406B7ED
EE386BFB 5A899FA5 AE9F2411 7C4B1FE6 49286651 ECE65381
FFFFFFFF FFFFFFFF
```

El generador es 2 (decimal)

6.3 Grupo 3 de Oakley

Las implementaciones IKE DEBERÍAN soportar un grupo EC2N con las siguientes características. Este grupo es asignado al identificador 3 (tres). La curva se basa en el Campo de Galois GF[2^155]. El tamaño del campo es de 155. El polinomio irreducible para el campo es de:

$$u^{155} + u^{62} + 1$$

La ecuación para la curva elíptica es:

$$y^2 + xy = x^3 + ax^2 + b$$

Tamaño del campo: 155
Grupo Primo/Polinomio Irreducible:
0x080000000000000000000004000000000000001
Primer Grupo Generado: 0x7b
Grupo Curva A: 0x0
Grupo Curva B: 0x07338f

Orden del Grupo: 0X080000000000000000057db5698537193aef944

Los datos en la carga KE cuando se usa este grupo es el valor x de la solución (x, y), del punto en la curva seleccionado tomando el secreto aleatoriamente escogido ka y calculando ka*P, donde * es la repetición de la suma del grupo y de las operaciones dobles, P es el punto de la curva con coordenada X igual al generador 1 y la coordenada Y determinada de la ecuación definida. La ecuación de la curva es conocida implícitamente por el Tipo de Grupo y los coeficientes A y B. Hay dos valores posibles para la coordenada Y; cada uno puede ser utilizado exitosamente (las dos partes no necesitan convenir en la selección).

6.4 Grupo 4 de Oakley

Las implementaciones de IKE DEBERÍAN soportar un grupo de EC2N con las siguientes características. Este grupo es asignado al identificador 4 (cuatro). La curva se basa en el Campo de Galois GF[2^155]. El tamaño del campo es de 185. El polinomio irreducible

para el campo es de:

u^185 + u^69 + 1

la ecuación para la curva elíptica es:

y^2 + xy = x^3 + ax^2 + B.

Tamaño del campo: 185
Grupo Primo/Polinomio Irreducible:
0x020000000000000000000000000000200000000000000001
Primer Grupo Generado: 0x18
Grupo Curva A: 0x0
Grupo Curva B: 0x1ee9

Orden del Grupo:
0X01ffffffffffffffffffdbf2f889b73e484175f94ebc

Los datos en la carga de KE cuando se usa este grupo serán idénticos a cuando se usa el grupo 3 (tres) de Oakley.

Otros grupos pueden ser definidos usando el Modo Nuevo Grupo. Estos grupos por defecto fueron generados por Richard Schroeppel en la universidad de Arizona. Las características de éstos números primos se describen en [Orm96].

7. Incremento de la Carga para un Intercambio Completo IKE

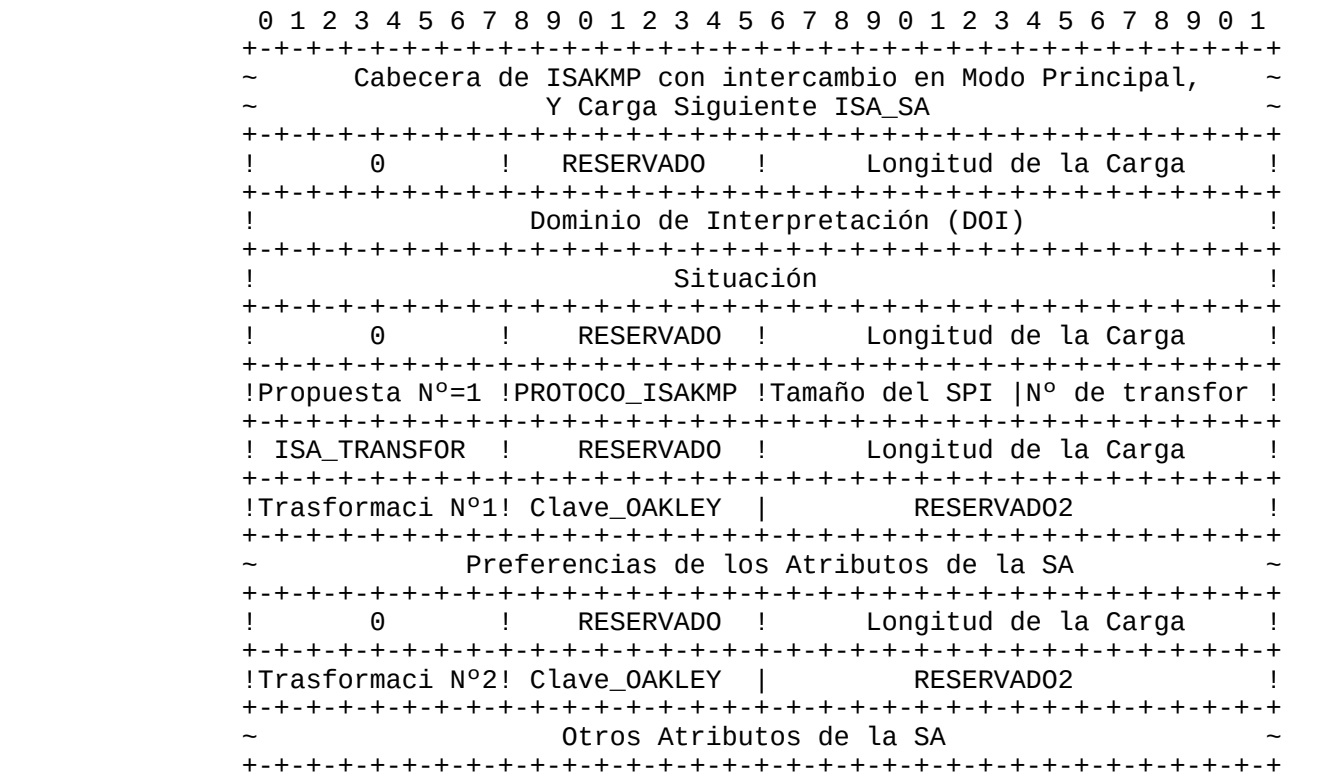
Esta sección ilustra cómo se utiliza el protocolo IKE para:

- establecer un canal seguro y autentificado entre procesos ISAKMP (fase 1); y

- generar material clave para, y negociar, una SA IPsec (fase 2).

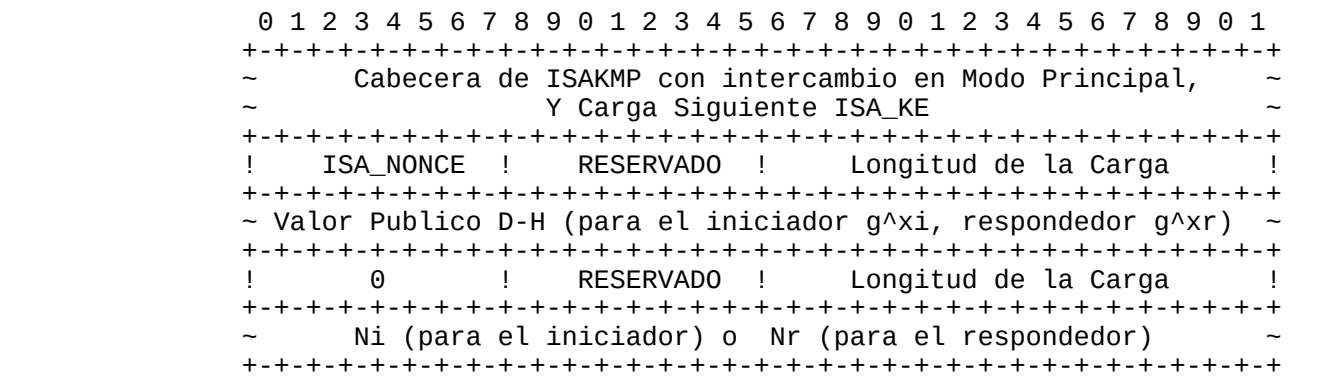
7.1 Fase 1 utilizando el Modo Principal

El diagrama siguiente ilustra las cargas intercambiadas entre dos partes en el primer intercambio de ida y vuelta. El iniciador PUEDE proponer varias propuestas; el respondedor DEBE contestar solo una.

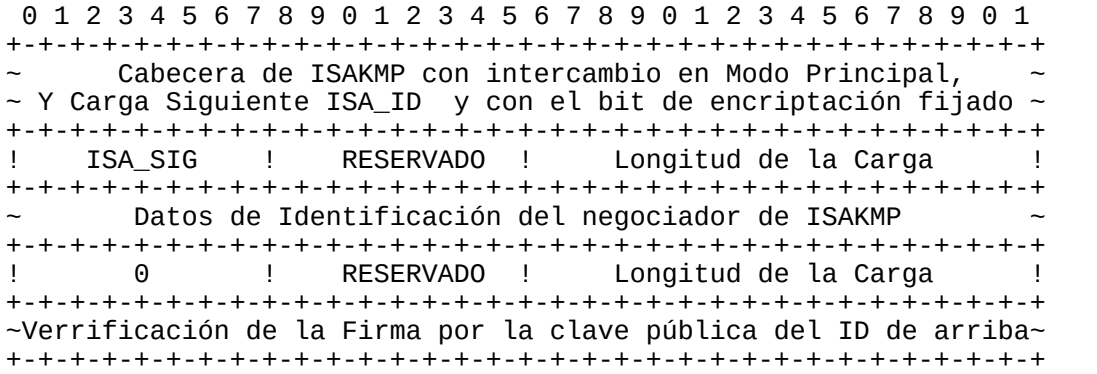


El respondedor contesta el tipo pero lo selecciona, y retorna, una propuesta de transformación (los atributos de la SA ISAKMP).

El segundo intercambio consiste en las siguientes cargas:



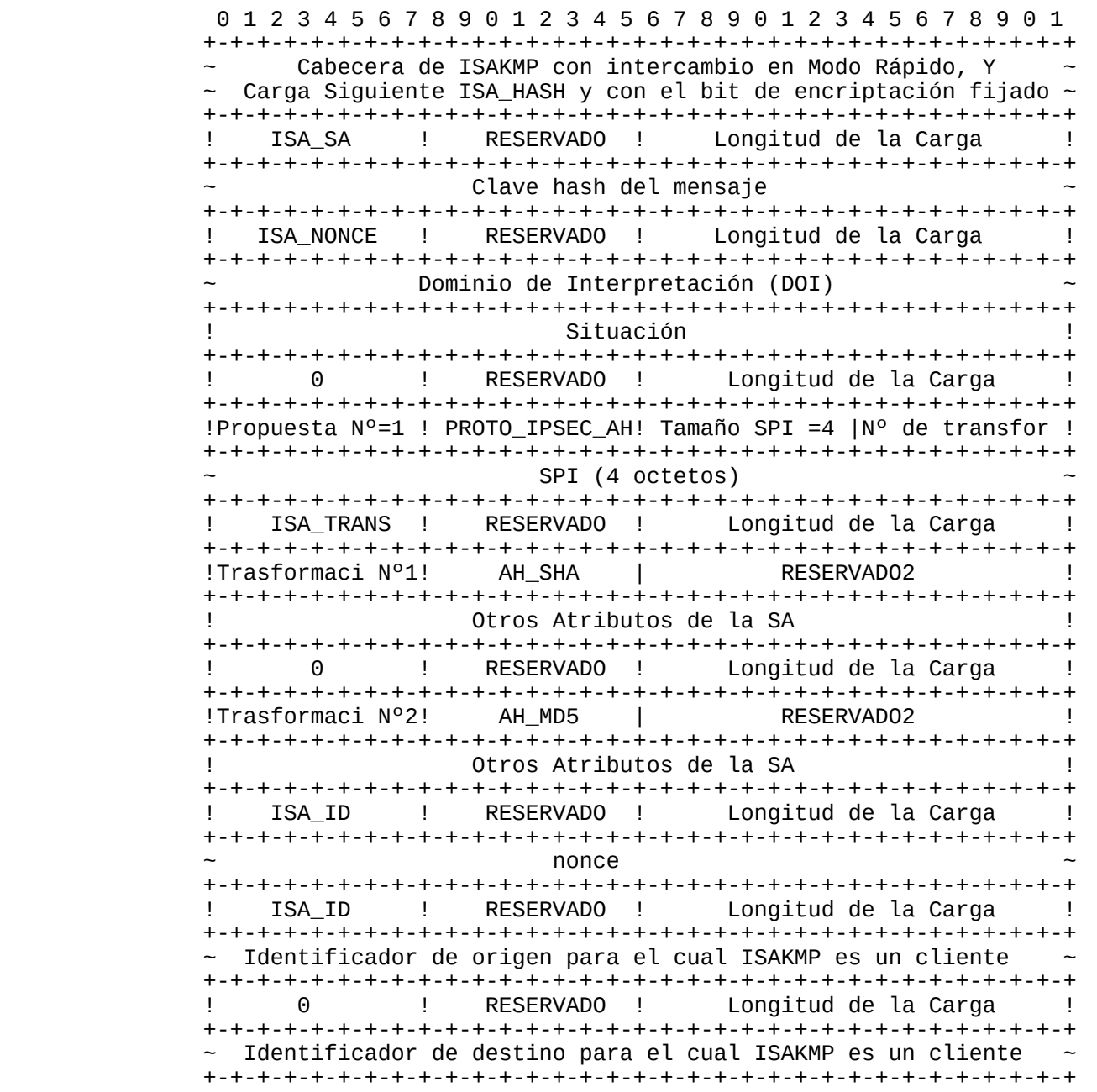
Las claves compartidas, SKEYID_e y SKEYID_a, ahora se utilizan para proteger y autentificar todas las futuras comunicaciones. Observe que SKEYID_e y SKEYID_a no están autentificados.



El intercambio de clave es autentificado por el hash firmado según lo descripto en la sección 5.1. Una vez que se haya verificado la firma usando el algoritmo de autentificación negociado como parte de la SA ISAKMP, las claves compartidas, SKEYID_e y SKEYID_a se las pueden referir como autentificadas. (Por brevedad, las cargas de certificación no fueron intercambiadas).

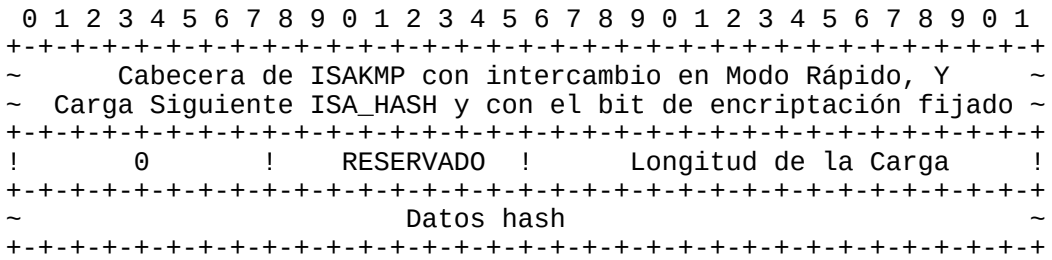
7.2 Fase 2 utilizando Modo rápido

Las cargas siguientes se intercambian en el primer ciclo del Modo Rápido con la negociación de la SA ISAKMP. En este intercambio hipotético, los negociadores de ISAKMP son representantes de otras partes que han solicitado la autentificación.



donde los contenidos del hash se describen en la sección 5.5 de arriba. El respondedor contesta con un mensaje similar que contiene

solamente una transformación-- la transformación seleccionada AH. Al recibir, el iniciador puede proporcionar el proceso clave [key engine] con la asociación de seguridad negociada y el material clave. Como una comprobación contra ataques de anti-replay, el respondedor espera hasta recibir el siguiente mensaje.



donde los contenidos del hash se describen en la sección 5.5 de arriba.

8. Ejemplos de Perfect Forward Secrecy

Este protocolo puede proporcionar PFS para las claves e identidades. Las identidades de ambos usuarios ISAKMP y, si fuera aplicable, las identidades de quienes lo están negociando se pueden proteger con PFS.

Para proporcionar PFS para las claves y todas las identidades, las dos partes realizarían lo siguiente:

- Un Intercambio en Modo Principal para proteger las identidades de los usuarios de ISAKMP.
  Esto establece una SA ISAKMP.

- Un Intercambio en Modo Rápido para negociar la protección de otro protocolo de seguridad.
  Esto establece una SA en cada extremo para este protocolo.

- Elimine la SA ISAKMP y su estado asociado.

Puesto que la clave para usarse en la SA no-ISAKMP fue derivada de solo un intercambio efímero de Diffie-Hellman, se preserva el PFS.

Para proporcionar PFS solamente a las claves de una SA no-ISAKMP, no necesitará hacer un intercambio de fase 1 si existe una SA ISAKMP entre dos usuarios. Un solo Modo Rápido en el cual se pasa la carga opcional KE, y se realiza un intercambio adicional de Diffie-Hellman, es todo lo que se requiere. En este punto el estado derivado de este

Modo Rápido debe suprimir la SA ISAKMP según lo descripto en la sección 5.5.

9. Sugerencias de Implementación

Utilizar una única negociación ISAKMP de fase 1 hace que las subsiguientes negociaciones de fase 2 sean más rápidas. Siempre y cuando el estado de la fase 1 permanezca oculto, y el PFS no sea necesario, la fase 2 puede proceder sin exponenciación. La cantidad de negociaciones de fase 2 que se pueden realizar para una única fase 1 es un asunto de la política local. La decisión dependerá de la fuerza de los algoritmos que son utilizados y de los niveles de confianza en el sistema de usuarios.

Una implementación puede desear negociar una serie de SAs cuando realiza el Modo Rápido. Haciendo esto se puede acelerar el "recambio de claves". El Modo Rápido define la forma en la que se define KEYMAT para una serie de SAs. Cuando un usuario considera que es tiempo de cambiar las SAs simplemente utiliza la siguiente dentro del rango indicado. Una serie de SAs pueden ser establecidas por múltiples SAs negociadas (idénticos atributos, diferentes SPIs) mediante un único Modo Rápido.

Una optimización que es a menudo útil es establecer Asociaciones de Seguridad con los usuarios antes de que estas sean necesarias de modo que cuando sean necesarias ya estén en su sitio. Esto asegura que no habrá retrasos debido a la administración de claves antes de la transmisión inicial de datos. Esta optimización es fácilmente implementada estableciendo más de una Asociación de Seguridad por usuario por cada Asociación de Seguridad solicitada y ocultando esas SA no usadas inmediatamente.

También, si una implementación ISAKMP es alertada que una SA pronto será necesitada (por ejemplo para sustituir una SA existente que expirará en un futuro próximo), se puede establecer una nueva SA antes de que esta sea necesaria.

La especificación de ISAKMP describe las condiciones en las cuales una parte del protocolo puede informar a la otra parte de cierta actividad-- cancelación de una asociación de seguridad o en respuesta a un cierto error en el protocolo tal como una verificación de firma fallida o una carga no desencriptada. Se sugiere encarecidamente que estos intercambios Informativos no sean contestados bajo ninguna circunstancia. Tal condición puede dar lugar a "conflicto de notificaciones" en el sentido de que no se entendería un mensaje resultante de una notificación a un usuario quien no puede entenderlo y envía su propia notificación la cual tampoco es entendida.

10. Consideraciones de seguridad

Este documento discute un protocolo híbrido, combinando partes de Oakley y partes de SKEME con ISAKMP, para negociar, y derivar el material clave para las asociaciones de seguridad de forma segura y autentificada.

La confidencialidad es asegurada con el uso de un algoritmo de encriptación negociado. La autentificación es asegurada con el uso de un método negociado: un algoritmo de firma digital; un algoritmo de clave pública que soporta encriptación; o una clave pre-compartida. La confidencialidad y la autentificación de este intercambio son solamente tan buenos como los atributos negociados como parte de la asociación de seguridad de ISAKMP.

La reiteración del recambio de claves usando el Modo Rápido puede consumir la entropía del secreto compartido de Diffie-Hellman. Los implementadores deberían observar este hecho y fijar un límite de Intercambios en Modo Rápido por medio de la exponenciación. Este documento no establece tal límite.

El Perfect Forward Secrecy (PFS) para el material clave y para las identidades es factible con este protocolo. Especificando un grupo de Diffie-Hellman, y pasando los valores públicos en cargas KE, los usuarios de ISAKMP pueden establecer el PFS para las claves-- las identidades serían protegidas por el SKEYID_e de la SA ISAKMP y por lo tanto no serían protegidas con PFS. Si se desea el PFS para el material clave y para las identidades, una ISAKMP de usuario DEBE establecer solamente una SA no-ISAKMP (por ejemplo SA IPsec) por SA ISAKMP. El PFS para las claves y para las identidades es llevado a cabo cancelando la SA ISAKMP (y opcionalmente enviando un mensaje de CANCELACIÓN) al crearse una SA no-ISAKMP. De esta forma una negociación de fase 1 esta unívocamente vinculada a una negociación de fase 2, y la SA ISAKMP establecida durante la fase 1 de la negociación nunca más es usada.

La fuerza de una clave derivada de un intercambio de Diffie-Hellman usando cualquiera de los grupos definidos aquí depende de la fuerza inherente del grupo, el tamaño del exponente usado, y de la entropía proporcionada por el generador de números aleatorios usado. Debido a estas entradas de información es difícil determinar la fuerza de una clave para cualquiera de los grupos definidos. El grupo de Diffie-Hellman por defecto (el primer grupo) cuando esta utilizado con un fuerte generador de números aleatorios y un exponente no menor de 160 bits, es suficiente para utilizar DES. Los grupos dos a cuatro proporcionan gran seguridad. Las implementaciones deberían notar estas conservadoras estimaciones al establecer la política y negociar parámetros de seguridad..

Observe que estas limitaciones están en los grupos de Diffie-Hellman. No hay nada en IKE que prohíba el uso de grupos más fuertes ni que disminuya la fuerza obtenida a partir de grupos más fuertes. De hecho, el marco extensible de IKE alienta la definición de más grupos; el uso de grupos de curvas elípticas aumentará en gran medida la fuerza usando números mucho más pequeños.

Para las situaciones donde los grupos definidos proporcionan fuerza insuficiente, el Modo Nuevo Grupo se puede utilizar para intercambiar un grupo de Diffie-Hellman que proporcione la fuerza necesaria. Es responsabilidad de las implementaciones controlar el carácter primo de los grupos que se ofrecen e independientemente llegar a estimar la fuerza.

Se asume que los exponentes de Diffie-Hellman en este intercambio son borrados de la memoria después de ser usados. En particular, estos exponentes no deben ser derivados a partir de secretos permanentes como la seed de un generador seudo-aleatorio.

Los intercambios IKE mantienen activos los vectores de inicialización (IV) donde el último bloque de texto cifrado del último mensaje es el IV para el siguiente mensaje. Para prevenir retransmisiones (o mensajes falsificados con cookies válidas) que produzcan intercambios fuera del sincronismo de IKE, las implementaciones NO DEBERÍAN actualizar sus IV hasta que el mensaje desencriptado ha pasado por una comprobación de coherencia y se determine realmente adelantar la máquina de estado de IKE-- es decir que no es una retransmisión.

Mientras que el último viaje de ida y vuelta en Modo Principal (y opcionalmente el ultimo mensaje en Modo Agresivo) está encriptado, en sentido estricto, no está autentificado. Un ataque activo de substitución contra el texto cifrado podría resultar en la corrupción de la carga. Si tal ataque corrompe las cargas obligatorias sería detectado por un error en la autentificación, pero si corrompe alguna de las cargas opcionales (por ejemplo cargas de notificación encadenadas sobre el último mensaje de un intercambio en Modo Principal) es posible que no sea perceptible.

11. Consideraciones de la IANA

Este documento contiene muchos "números mágicos" que serán mantenidos por la IANA (Internet Assigned Numbers Authority - Autoridad de Asignación de Números en Internet). Esta sección explica los criterios utilizados por la IANA para asignar números adicionales en cada una de estas listas.

11.1 Clases de Atributos

Los atributos negociados en este protocolo son identificados por su clase. Los pedidos de asignación de nuevas clases deben estar acompañados por un RFC que describa el uso de este atributo.

11.2 Clases de Algoritmos de Encriptación

Los valores de la Clase de Algoritmo de Encriptación definen un algoritmo de encriptación para utilizarse cuando sea requerido en este documento. Los pedidos de asignación de valores para nuevos algoritmos de encriptación deben estar acompañados con una referencia en vías de estandarización o un RFC informativo o una referencia a la literatura criptográfica publicada que describa este algoritmo.

11.3 Algoritmos Hash

Los valores de la Clase de Algoritmo Hash definen un algoritmo de hash para utilizarse cuando sea requerido en este documento. Los pedidos de asignación de valores para nuevos algoritmos hash deben estar acompañados con una referencia en vías de estandarización o un RFC informativo o una referencia a la literatura criptográfica publicada que describa este algoritmo. Debido a la derivación de claves y al uso de expansión de claves de tipo HMAC de los algoritmos en IKE, los pedidos de asignación de nuevos valores de algoritmo hash deben considerar las características criptográficas-- por ejemplo la resistencia a la colisión del algoritmo de hash.

11.4 Descripción de Grupo y Tipo de Grupo

Los valores de la Clase de Descripción de Grupo identifican a grupo para utilizarse en un intercambio de Diffie-Hellman. Los valores Tipo de Clase de Grupo definen el tipo de grupo. Los pedidos de asignación de nuevos grupos deben estar acompañados con una referencia en vías de estandarización o un RFC informativo que describa este grupo. Los pedidos de asignación de nuevos tipos de grupo deben estar acompañados por una referencia en vías de estandarización o un RFC informativo o por una referencia a la literatura criptográfica o matemática que describa el nuevo tipo.

11.5 Tipo de Vida

Los valores de la Clase Tipo de Vida definen un tipo de tiempo de vida el cual es aplicado a la SA ISAKMP. Los pedidos de asignación de nuevos tipos de vida deben estar acompañados por una descripción detallada de las unidades de este tipo y de su vencimiento.

12. Agradecimientos

Este documento es el resultado de consultar a Hugo Krawczyk, Douglas Maughan, Hilarie Orman, Mark Schertler, Mark Schneider, y Jeff Turner. Confía en los protocolos que fueron escritos por ellos. Sin su interés y esmero, esto no habría sido escrito.

Especiales agradecimientos a Rob Adams, Cheryl Madson, Piper de Derrell, Harry Varnis, y Elfec Weaver por la información técnica, el estímulo, y los diversos controles de coherencia a lo largo del documento.

También quisiéramos agradecer a muchos miembros del grupo de trabajo de IPsec que contribuyó al desarrollo de este protocolo en el último año.

13. Referencias

[CAST] Adams, C., "The CAST-128 Encryption Algorithm", RFC 2144, May 1997.

[BLOW] Schneier, B., "The Blowfish Encryption Algorithm", Dr. Dobb's Journal, v. 19, n. 4, April 1994.

[Bra97] Bradner, S., "Key Words for use in RFCs to indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[DES] ANSI X3.106, "American National Standard for Information Systems-Data Link Encryption", American National Standards Institute, 1983.

[DH] Diffie, W., and Hellman M., "New Directions in Cryptography", IEEE Transactions on Information Theory, V. IT-22, n. 6, June 1977.

[DSS] NIST, "Digital Signature Standard", FIPS 186, National Institute of Standards and Technology, U.S. Department of Commerce, May, 1994.

[IDEA] Lai, X., "On the Design and Security of Block Ciphers," ETH Series in Information Processing, v. 1, Konstanz: Hartung-Gorre Verlag, 1992

[KBC96] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[SKEME] Krawczyk, H., "SKEME: A Versatile Secure Key Exchange Mechanism for Internet", from IEEE Proceedings of the 1996 Symposium on Network and Distributed Systems Security.

[MD5] Rivest, R., "The MD5 Message Digest Algorithm", RFC 1321, April 1992.

[MSST98] Maughhan, D., Schertler, M., Schneider, M., and J. Turner, "Internet Security Association and Key Management Protocol (ISAKMP)", RFC 2408, November 1998.

[Orm96] Orman, H., "The Oakley Key Determination Protocol", RFC 2412, November 1998.

[PKCS1] RSA Laboratories, "PKCS #1: RSA Encryption Standard", November 1993.

[Pip98] Piper, D., "The Internet IP Security Domain Of Interpretation for ISAKMP", RFC 2407, November 1998.

[RC5] Rivest, R., "The RC5 Encryption Algorithm", Dr. Dobb's Journal, v. 20, n. 1, January 1995.

[RSA] Rivest, R., Shamir, A., and Adleman, L., "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", Communications of the ACM, v. 21, n. 2, February 1978.

[Sch96] Schneier, B., "Applied Cryptography, Protocols, Algorithms, and Source Code in C", 2nd edition.

[SHA] NIST, "Secure Hash Standard", FIPS 180-1, National Institue of Standards and Technology, U.S. Department of Commerce, May 1994.

[TIGER] Anderson, R., and Biham, E., "Fast Software Encryption", Springer LNCS v. 1039, 1996.

Apéndice A

Ésta es una lista de claves DES débiles y Semi-Débiles. Las claves provienen de [Sch96]. Todos los claves se listan en hexadecimal.

Claves DES débiles
0101 0101 0101 0101
1F1F 1F1F E0E0 E0E0
E0E0 E0E0 1F1F 1F1F
FEFE FEFE FEFE FEFE

Claves DES semi-débiles
01FE 01FE 01FE 01FE
1FE0 1FE0 0EF1 0EF1
01E0 01E0 01F1 01F1
1FFE 1FFE 0EFE 0EFE
011F 011F 010E 010E
E0FE E0FE F1FE F1FE

FE01 FE01 FE01 FE01
E01F E01F F10E F10E
E001 E001 F101 F101
FE1F FE1F FE0E FE0E
1F01 1F01 0E01 0E01
FEE0 FEE0 FEF1 FEF1

Números Asignados a los Atributos

Los Atributos negociados durante la fase 1 usan las siguientes definiciones. Los atributos de la fase 2 se definen en la especificación pertinente al DOI (por ejemplo, los atributos de IPsec se definen en el DOI de IPsec), a excepción de una descripción de grupo cuando el Modo Rápido incluye un efímero intercambio de Diffie-Hellman. Los tipos de atributo pueden ser Básico (B) o Longitud-Variable (V). La codificación de estos atributos se define en la especificación de ISAKMP como Tipo/Valor (Básico) y Tipo/Longitud/Valor (Variable).

La descripción de atributos como básico NO DEBE ser codificada como variable. El atributo longitud variable PUEDE ser codificado como atributo básico si su valor puede caber dentro de dos octetos. Si éste es el caso, un atributo ofrecido como variable (o básico) por el iniciador de este protocolo PUEDE regresar al iniciador como básico (o variable).

Clases de Atributos

Clase	Valor	Tipo
Algoritmo de Encriptación	1	B
Algoritmo Hash	2	B
Método de Autentificación	3	B
Descripción del Grupo	4	B
Tipo de Grupo	5	B
Grupo Primo/Polinomio Irreducible	6	V
Primer Grupo Generado	7	V
Segundo Grupo Generado	8	V
Grupo Curva A	9	V
Grupo Curva B	10	V
Tipo de Vida	11	B
Tiempo de Vida	12	V
PRF	13	B
Longitud de la Clave	14	B
Tamaño del Campo	15	B
Orden del Grupo	16	V

Los valores de 17 a 16383 están reservados por la IANA. Los valores de 16384 a 32767 son para usarse privadamente entre usuarios que mutuamente lo han acordado.

Clase Valores

- Algoritmos de Encriptación		Definido en el
DES-CBC	1	RFC 2405
IDEA-CBC	2	
Blowfish-CBC	3	
RC5-R16-B64-CBC	4	
3DES-CBC	5	
CAST-CBC	6	

Los valores de 7 a 65000 están reservados por la IANA. Los valores de 65001 a 65535 son para usarse privadamente entre usuarios que mutuamente lo han acordado.

- Algoritmos Hash		Definido en el
MD5	1	RFC 1321
SHA	2	FIPS 180-1
Tiger	3	Véase [TIGER]

Los valores de 4 a 65000 están reservados por la IANA. Los valores de 65001 a 65535 son para usarse privadamente entre usuarios que mutuamente lo han acordado.

- Método de Autentificación

Claves pre-compartidas	1
Firmas DSS	2
Firmas RSA	3
Encriptación con RSA	4
Encriptación revisada con RSA	5

Los valores de 6 a 65000 están reservados por la IANA. Los valores de 65001 a 65535 son para usarse privadamente entre usuarios que mutuamente lo han acordado.

- Descripción del Grupo

Grupo MODP por defecto 768-bit (sección 6.1)	1
Grupo MODP alternativo 1024-bit (sección 6.2)	2
Grupo EC2N de GP[2^155] (sección 6.3)	3
Grupo EC2N de GP[2^185] (sección 6.4)	4

Los valores de 5 a 32767 están reservados por la IANA. Los valores de 32768 a 65535 son para usarse privadamente entre usuarios que mutuamente lo han acordado.

- Tipo de Grupo

MODP (grupo exponenciación modular)	1
ECP (grupo curva elíptica sobre GF[P])	2
EC2N (grupo curva elíptica sobre GF[2^N])	3

Los valores de 4 a 65000 están reservados por la IANA. Los valores de 65001 a 65535 son para usarse privadamente entre usuarios que mutuamente lo han acordado.

- Tipo de Vida

segundos	1
kilobytes	2

Los valores de 3 a 65000 están reservados por la IANA. Los valores de 65001 a 65535 son para usarse privadamente entre usuarios que mutuamente lo han acordado. Para un "Tipo de Vida" dado el valor del atributo "Tiempo de Vida" define la duración de la SA-- un número de segundos, o un número en Kbytes protegido.

- PRF

No hay actualmente funciones seudo-aleatorias definidas.

Los valores de 1 a 65000 están reservados por la IANA. Los valores de 65001 a 65535 son para usarse privadamente entre usuarios que mutuamente lo han acordado.

- Longitud de la Clave

Al usar un algoritmo de encriptación que tiene una longitud de clave variable, este atributo especifica la longitud de la clave en bits. (Se DEBE utilizar el orden de byte de red.) Este atributo NO DEBE ser utilizado cuando el Algoritmo de Encriptación especificado use una clave de longitud fija.

- Tamaño del Campo

El tamaño del campo, en bits, de un grupo de Diffie-Hellman.

- Orden del Grupo

El orden del grupo de un grupo de curvas elípticas. Observe que la longitud de este atributo depende del tamaño del campo.

Los intercambios adicionales definidos - valores de XCHG

Modo rápido	32
Modo Nuevo Grupo	33

Apéndice B

Este apéndice describe los detalles de encriptación que se utilizarán SOLAMENTE al encriptar mensajes ISAKMP. Cuando un servicio (tal como una transformación IPsec) utiliza ISAKMP para generar material clave, todos los detalles específicos del algoritmo de cifrado (tales como generación de claves y IV, relleno, etc...) DEBEN estar definidos por ese servicio. ISAKMP no pretende generar siempre las claves que son convenientes para cualquier algoritmo de encriptación. ISAKMP produce la cantidad solicitada de material clave del cual el servicio DEBE generar una clave conveniente. Los detalles, tales como control de claves, son responsabilidad del servicio.

El uso de PRFs negociados puede requerir que la salida PRF se amplié debido al mecanismo de retroalimentación del PRF empleado por este documento. Por ejemplo, si él (ficticio) DOORAK-MAC requiere 24 bytes de clave pero produce solamente 8 bytes de salida, la salida se debe ampliar tres veces antes de que sea utilizada como clave por otra instancia de este. La salida de un PRF es ampliada retroalimentando los resultados del PRF para generar bloques sucesivos. Se concatenan estos bloques hasta que el número indispensable de bytes se ha alcanzado. Por ejemplo, para la autentificación de claves pre-compartidas con DOORAK-MAC el PRF negociado es:

```
  BLOCK1-8 = prf(clave pre-compartida, Ni_b | Nr_b)
  BLOCK9-16 = prf(clave pre-compartida, BLOCK1-8 | Ni_b | Nr_b)
  BLOCK17-24 = prf(clave pre-compartida, BLOCK9-16 | Ni_b | Nr_b)
entonces
  SKEYID = BLOCK1-8 | BLOCK9-16 | BLOCK17-24
```

por consiguiente para derivar SKEYID_d:

```
  BLOCK1-8 = prf(SKEYID, g^xy | CKY-I | CKY-R | 0)
  BLOCK9-16 = prf(SKEYID, BLOCK1-8 | g^xy | CKY-I | CKY-R | 0)
  BLOCK17-24 = prf(SKEYID, BLOCK9-16 | g^xy | CKY-I | CKY-R | 0)
y
  SKEYID_d = BLOCK1-8 | BLOCK9-16 | BLOCK17-24
```

Subsiguientes PRF son derivados de forma similar.

Las claves usadas para proteger la SA ISAKMP es derivada a partir de SKEYID_e de un algoritmo específico. Cuando SKEYID_e no es suficientemente largo para proveer todo el material clave necesario que un algoritmo requiere, la clave se deriva a partir de la concatenación de los resultados de una función seudo-aleatoria dentro de sí misma, concatenando los resultados, y tomando los bits de orden superior necesarios.

Por ejemplo, si el algoritmo AKULA (ficticio) requiere 320 bits de clave (y no tiene ningún control de clave débil) y el prf usado para generar SKEYID_e genera solamente 120 bits de material, la clave para AKULA, sería los primeros 320 bits de ka, donde:

```
    Ka = K1 | K2 | K3
y
    K1 = prf(SKEYID_e, 0)
    K2 = prf(SKEYID_e, K1)
```

Donde el prf es el prf negociado o la versión HMAC de la función hash negociada (si no se negoció ningún prf) y 0 es representado por un solo octeto. Cada resultado del prf proporciona 120 bits de material para un total de 360 bits. AKULA utilizaría los primeros 320 bits de esa cadena de 360 bits.

En la fase 1, el material para el vector de inicialización (el material IV) para el algoritmo de encriptación en modo CBC es derivado a partir de un hash de una concatenación del valor público de Diffie-Hellman del iniciador y del valor público de Diffie-Hellman del respondedor usando el algoritmo hash negociado. Esto se utiliza solamente para el primer mensaje. Cada mensaje debería ser rellenado hasta el tamaño de bloque más cercano usando bytes que contengan 0x00. La longitud del mensaje en la cabecera DEBE incluir la longitud del relleno, puesto que éste refleja el tamaño del texto cifrado. Los mensajes subsiguientes DEBEN utilizar el último bloque CBC encriptado del mensaje anterior como su vector de inicialización.

En la fase 2, el material para el vector de inicialización para la encriptación en modo CBC del primer mensaje en un intercambio en Modo Rápido es derivado a partir de un hash de una concatenación del último bloque CBC de salida de la fase 1 y del identificador del mensaje de la fase 2 usando el algoritmo hash negociado. EL IV para los subsiguientes mensajes dentro de un intercambio en Modo Rápido es el CBC del bloque de salida del mensaje anterior. El relleno y los IVs para los mensajes subsiguientes se realizan como en la fase 1.

Después de que se haya autentificado la SA ISAKMP todos los Intercambios Informativos se encriptan usando SKEYID_e. El vector de inicialización para estos intercambios se deriva exactamente de la misma manera que para el Modo Rápido-- es decir, se deriva a partir de un hash de una concatenación del último bloque CBC de salida de la fase 1 y del identificador de mensaje de la cabecera ISAKMP del Intercambio Informativo (no del identificador de mensaje que pudo haber originado el Intercambio Informativo).

Observe que el último bloque CBC de salida de la Fase 1, resultante de la encriptación/desencriptación del último mensaje de la Fase 1, se debe conservar en el estado SA ISAKMP para permitir la generación de los IVs únicos para cada Modo Rápido. Cada intercambio posterior a la fase 1 (los Modos Rápidos e Intercambios Informativos) generan IVs independientes para evitar que IVs se salga de sincronismo cuando dos intercambios diferentes comienzan simultáneamente.

En todos estos casos, hay un solo contexto bidireccional de cifrado/IV. Considerando que cada Modo Rápido e Intercambio Informativo mantienen un único contexto evitando que el IVs se salga de sincronismo.

La clave para DES-CBC se deriva de los primeros 8 bytes no-débiles y de no-semi-débiles de SKEYID_e (ver apéndice A). El IV es los primeros 8 bytes del material IV derivado anteriormente.

La clave para IDEA-CBC se deriva de los primeros 16 bytes de SKEYID_e. El IV es los primeros 8 bytes del material del IV derivado anteriormente.

La clave para el blowfish-CBC es el tamaño de la clave negociada, o los primeros 56 bytes de una clave (sino se negocia ningún tamaño de clave) derivado a partir del método seudo-aleatorio de retroalimentación de la función ya mencionada. El IV es los primeros 8 bytes del material IV derivado anteriormente.

La clave para RC5-R16-B64-CBC es el tamaño de la clave negociada, o los primeros 16 bytes de una clave (sino se negocia ningún tamaño de clave) derivado a partir del método seudo-aleatorio de retroalimentación de la función ya mencionada si fuera necesario. El IV es los primeros 8 bytes del material del IV derivado anteriormente. El número de ciclos DEBE ser 16 y el tamaño de bloque DEBE ser de 64.

La clave para 3DES-CBC es los primeros 24 bytes de una clave derivado a partir del método seudo-aleatorio de retroalimentación de la función ya mencionada. 3DES-CBC es una operación de encriptación-desencriptación-encriptación que usa la primera, mitad, y los últimos 8 bytes de la clave entera de 3DES-CBC. El IV es los primeros 8 bytes del material del IV derivado anteriormente.

La clave para CAST-CBC es el tamaño de la clave negociada, o los primeros 16 bytes de una clave derivada a partir del método seudo-aleatorio de retroalimentación de la función ya mencionada. El IV es los primeros 8 bytes del material del IV derivado anteriormente.

El soporte para otros algoritmos, a excepción del DES-CBC, es puramente opcional. Algunos algoritmos opcionales pueden depender de derechos de propiedad intelectual.

Direcciones de los Autores

Dan Harkins
cisco Systems
170 W. Tasman Dr.
San Jose, California, 95134-1706
United States of America

Phone: +1 408 526 4000
EMail: dharkins@cisco.com

Dave Carrel
76 Lippard Ave.
San Francisco, CA 94131-2947
United States of America

Phone: +1 415 337 8469
EMail: carrel@ipsec.org

Notas de los Autores

Los autores alientan implementaciones independientes, test de interoperabilidad, de este protocolo híbrido.

Declaración de Copyright Completa

Notas del Traductor

Las Siguientes palabras no han sido traducidas y su significado es el siguiente:

- Nonce: Aliatoriamente, cadena de texto única que es encriptada junto con datos y que luego es usada para detectar ataques contra el sistema que envía el dato encriptado. Un nonce es usado específicamente para la autentificación y para asegurar que el dato encriptado es diferente cada vez que es encriptado. (Definición extraída del Diccionario de IBM Corp.)

- Perfect Forward Secrecy (PFS): En criptografía, en un protocolo de establecimiento de clave, la noción que compromete a una única clave que permitirá el acceso solamente a los datos protegidos derivados de esa única clave. Para que el PFS exista la clave usada para proteger la transmisión de datos NO DEBE ser usada para derivar claves adicionales, y si la clave usada para proteger la transmisión de los datos fue derivada de otro material clave, ese material NO DEBE ser usado para derivar más claves [IKE]. Este término se lo pude encontrar traducido como: máxima confidencia en el reenvío (según la traducción de U.S. Robotics Corporation); confidencialidad directa perfecta (según la traducción de Microsoft), confidencialidad anticipada perfecta o Confidencialidad respaldada correctamente.

- Hash: resumen criptográfico (Definición extraída del Diccionario de ORCA Informática.). Un número generado a partir de una cadena de caracteres que es usado para garantizar que el mensaje trasmitido llegue intacto (Definición extraída del Diccionario de IBM Corp.).

- Seed: Un valor que añade aleatoriedad a la creación de números seudo-aleatorios (Definición extraída del Diccionario de IBM Corp.).

Los Términos que aparecen entre "[]" que no sean referencias reflejan la palabra/s en inglés de las palabra/s que se encuentran (en español) a la izquierda, debido a que NO ESTOY SEGURO de que sea la correcta traducción del término o simplemente para que no se pierda el VERDADERO sentido del texto.

Esta presente traducción fue realizada por Hugo Adrian Francisconi para mi tarjado de tesis de "Ingeniero en Electrónico" en la Facultad U.T.N. (Universidad Nacional Tecnología) Regional Mendoza - Argentina. Si le interesa IPsec y quieres saber más podes bajarte mi trabajo de tesis, "IPsec en Ambientes IPv4 e IPv6" de http://codarec6.frm.utn.edu.ar, para el cual traduje varios RFCs al español relacionados con IPsec. Cualquier sugerencia debate o comentario sobre este presente tema o traducción será bien recibida en adrianfrancisconi@yahoo.com.ar

Se a realizado el máximo esfuerzo para hacer de esta traducción sea tan completa y precisa como sea posible, pero no se ofrece ninguna garantía implícita de adecuación a un fin en particular. La información se suministra "tal como está". El traductor no será responsable ante cualquier persona o entidad con respecto a cualquier pérdida o daño que pudiera resultar emergente de la información contenida en está traducción.

Derechos de Copyright Sobre Esta Traducción

Esta traducción tiene los mismos derechos que le RFC correspondiente traducido, con el aditamento de que cualquier persona que extraiga TOTAL o PARCIALMENTE esta traducción deberá hacer mención de esta presente nota de copyright y de los datos del traductor.

Datos del Traductor

Nombre y Apellido del Traductor: Hugo Adrian Francisconi
Domicilio: Carril Godoy Cruz 2801, Villa Nueva-Guay Mallen-Mendoza-Argentina
Código Postal: 5500
Tel: 054-0261-4455427
E-mail: adrianfrancisconi@yahoo.com.ar