Una Introducción al Cifrado de Seguridad IP (IPSec)

Contenido

Introducción

Este documento presenta IPsec a los usuarios de una forma rápida pero concisa. Este documento contiene configuraciones básicas del Intercambio de claves por Internet (IKE) con claves previamente compartidas, IKE con una Autoridad de certificación e IPSec. Este no es un documento exhaustivo. Pero este documento le ayuda a entender las tareas y el orden en el que se realizan.

 **Advertencia:** Hay restricciones estrictas en la exportación de la criptografía profunda. Si usted viola la ley federal E.E.U.U., después le, no Cisco, detienen responsable. Si usted tiene cualesquiera preguntas relacionadas con el control de la exportación, envíe y email a export@cisco.com.

Nota: El Multicast y el broadcast no se soportan en el LAN normal a los túneles LAN o en los

clientes VPN que terminan en cualquier dispositivo. El Multicast se puede pasar solamente en los túneles GRE. Esto se soporta solamente en el Routers y no en los concentradores VPN 3000 o los Firewall (ASA/PIX).

prerrequisitos

Requisitos

No hay requisitos específicos para este documento.

Componentes Utilizados

Este documento no tiene restricciones específicas en cuanto a versiones de software y de hardware.

Convenciones

Consulte Convenciones de Consejos TécnicosCisco para obtener más información sobre las convenciones del documento.

Antecedente

El IPSec es la plataforma cifrada de la capa de red de la última generación para las plataformas de seguridad de Cisco (Cisco IOS ® Software, PIX, y así sucesivamente). Descrito originalmente en el RFCs 1825 con 1829, que son Obsoletos ahora, el IPSec se discute actualmente en varios documentos presentados por el Grupo de trabajo de la seguridad IP IETF . El IPSec soporta actualmente versión IP 4 paquetes de unidifusión. El IPv6 y el soporte multidifusión es llegar en otro momento.

El IPSec tiene estas fuerzas sobre las ofrendas crypto actuales de Cisco:

1. **Multivendor** — Puesto que se estandardiza el marco del IPSec, los clientes no son bloqueados en ningún producto de proveedor específico. El IPSec se encuentra en el Routers, los Firewall, y los escritorios del cliente (Windows, mac, y así sucesivamente).
2. **Scalability** — El IPSec se diseña con las empresas grandes en la mente. Por lo tanto, tiene la administración de claves incorporada.

Nota: Mientras que varias Plataformas de Cisco pueden utilizar el IPSec, este documento se adapta hacia el Cisco IOS Software.

Vocabulario de encripción

Usted necesita conocer estos términos para entender el IPSec, y leer el resto de este documento. Cuando usted ve las siglas en otras porciones de este documento, refiera a esta página para las definiciones.

Advanced Encryption Standard (AES) — El AES fue concluido como Estándar de procesamiento de la información federal (FIP) - algoritmo criptográfico aprobado que se utilizará para proteger la transmisión de datos electrónicos (FIPS PUB 197). El AES se basa en el algoritmo Rijndael, que

especifica cómo utilizar las claves con una longitud del 128, 192, o los bits 256 para cifrar los bloques con una longitud de los bits 128, 192, o 256. Las nueve combinaciones de longitud de clave y de extensión de bloque son posibles.

Encabezado de autenticación — El es un Security Protocol que proporciona la autenticación y los servicios de detección de reproducción opcionales. AH se integra en los datos que se protegerán, por ejemplo, un IP datagram completo. AH puede ser utilizado en sí mismo o con el Encryption Service Payload (ESP). Consulte el RFC 2402

Autenticación — Éste es una de las funciones del marco del IPSec. La autenticación establece la integridad del datastream y se asegura de que no está tratada de forzar con adentro transita. También proporciona la confirmación sobre el origen de la secuencia de datos.

Certification Authority (CA) — Esto es una entidad de tercera persona con la responsabilidad de publicar y de revocar los Certificados. Cada dispositivo que tiene su propio certificado y clave pública de CA puede autenticar cada otro dispositivo dentro de un dominio dado de CA. Este término también se aplica al software del servidor que proporciona estos servicios.

Certificado — Un objeto criptográficamente firmado que contiene una identidad y una clave pública asociadas a esta identidad.

Criyptografía clásicao — Éste es mecanismo de la encripción propietaria de Cisco usado en el Cisco IOS Software Release 11.2. La criyptografía clásicao está disponible en el Cisco IOS Software Release 11.3. Pero, el IPSec no está adaptado al Cisco IOS Software Release 11.2. Usted puede también ver la criyptografía clásicao del nombre designada el cifrado expreso o la Tecnología de encripción de Cisco (CET) en la literatura de comercialización.

Listas de revocación de certificados (CRL) — Éste es un mensaje firmado digitalmente que enumera toda la corriente pero los Certificados revocados enumerados por CA dado. Esto es análogo a un libro de números de tarjetas de carga robada que permite a los almacenes rechazar las tarjetas de crédito.

Correspondencia de criptografía — Ésta es una entidad de configuración del Cisco IOS Software que realiza dos funciones primarias. Primero, selecciona los flujos de datos que necesitan el proceso de seguridad. En segundo lugar, define la directiva para estos flujos y el peer de criptografía a los cuales el tráfico necesite ir.

Una correspondencia de criptografía se aplica a una interfaz. El concepto de una correspondencia de criptografía fue introducido en la criyptografía clásicao pero ampliado para el IPSec.

Integridad de los datos — Éste es mecanismos de integridad de los datos, con el uso de la clave secreta basado o los algoritmos basados clave pública, que permiten que el beneficiario de un pedazo de datos protegidos para verificar que los datos no se hayan modificado adentro transitan.

Confidencialidad de los datos — Éste es el método donde se manipulan los datos protegidos de modo que ningún atacante pueda leerlos. Esto se logra comúnmente a través del encripción de datos y claves que sólo están disponibles para las partes que participan de la comunicación.

Autenticación del origen de los datos — Esto es un servicio de seguridad donde el receptor puede verificar que los datos protegidos pudieran haber originado solamente del remitente. Este servicio requiere un servicio de integridad de datos, además de un mecanismo de distribución de clave, en donde sólo el emisor y el receptor comparten una clave secreta.

Data Encryption Standard (DES) — El DES fue publicado en 1977 por la Oficina nacional de estándares y es un esquema de encripción de la clave secreta basado en el algoritmo de Lucifer de IBM. El contraste de DES es una clave pública. Cisco utiliza el DES en la criyptografía clásicao (las longitudes de clave 40-bit y 56-bit), el IPSec crypto (la clave 56-bit), y en el firewall PIX (clave 56-bit).

Diffie Hellman — Éste es un método del establecimiento de una clave compartida sobre un medio inseguro. Diffie Hellman es un componente del Oakley, que se define en esta lista de la definición.

DSS — Un Digital Signature Algorithm diseñado por el National Institute of Standards and Technology E.E.U.U. (NIST) basado en el Cifrado de clave pública. El DSS no hace el cifrado del datagrama del usuario. El DSS es un componente en la criyptografía clásicao, así como el indicador luminoso LED amarillo de la placa muestra gravedad menor del IPSec de Redcreek, pero no en el IPSec implementado en Cisco IOS Software.

Encryption Service Adapter (ESA) — Éste es un acelerador de encripción basado hardware en el cual se utiliza:

- Cisco 7204 y 7206 Router
- Procesador 2-40s de interfaz versátil (VIP2-40s) de la segunda generación en todos los routers de la serie Cisco 7500.
- VIP2-40 en los Cisco 7000 Series Router que hacen los indicadores luminosos LED amarillo de la placa muestra gravedad menor del procesador Cisco 7000 Series ruta Switch (RSP7000) y del Chassis Interface de las Cisco 7000 Series (RSP7000CI) instalar.

El IPSec no utiliza la aceleración ESA, sino que trabaja en un cuadro que tenga un indicador luminoso LED amarillo de la placa muestra gravedad menor ESA sobre una base software solamente.

Encapsulating Security Payload (ESP) — Un Security Protocol que proporciona la confidencialidad de los datos y la protección con la autenticación opcional y los servicios de detección de reproducción. El ESP encapsula totalmente los datos del usuario. El ESP se puede utilizar en sí mismo o conjuntamente con AH. Refiera al RFC 2406: Encapsulating Security Payload (ESP) IP .

Hash — Ésta es una función de una manera que toma un mensaje de entrada de la longitud arbitraria y presenta una publicación de la longitud fija. Cisco utiliza el Secure Hash Algorithm (SHA) y la publicación de mensaje 5 (MD5) desmenuza dentro de nuestra implementación del marco del IPSec. Vea la definición para el HMAC para más información.

HMAC — Esto es un mecanismo para la autenticación del mensaje que las aplicaciones criptográficas desmenuzan por ejemplo el SHA y el MD5. Refiera al RFC 2104 para una descripción completa del HMAC.

Internet Key Exchange (IKE) — Un protocolo híbrido que utiliza el Oakley y a la parte de la parte otro Conjunto de protocolos llamó el SKEME dentro del marco del Internet Security Association and Key Management Protocol (ISAKMP). El IKE se utiliza para establecer una política de seguridad compartida y las claves autenticadas para los servicios, tales como IPSec, que requieren las claves. Antes de que cualquier tráfico IPSec pueda ser pasado, cada router/Firewall/host deben poder verificar la identidad de su par. Ingrese manualmente las claves previamente compartidas en los host, por un servicio CA, o el DNS seguro próximo (DNSSec) para hacer esto. Éste es el protocolo conocido antes como ISAKMP/Oakley, y se define en el RFC 2409: El Internet Key Exchange (IKE) . Un punto potencial de confusión es que las siglas

ISAKMP y IKE ambos están utilizadas en Cisco IOS Software para referir a la misma cosa. Estos dos elementos son algo diferentes.

Internet Security Association and Key Management Protocol (ISAKMP) — Ésta es una estructura del protocolo que define a los mecánicos de la implementación de un Key Exchange Protocol y de la negociación de una política de seguridad. El ISAKMP se define en el Internet Security Association and Key Management Protocol (ISAKMP).

Transparencia NAT del IPSec — La característica de la Transparencia NAT del IPSec introduce el soporte para que el tráfico de la seguridad IP (IPSec) viaje a través del Network Address Translation (NAT) o de las puntas de la traducción de la dirección de la punta (PALMADITA) en la red dirigiendo muchas incompatibilidades sabidas entre el NAT y el IPSec. El Traversal NAT es una característica que es auto detectado por los dispositivos VPN. No hay pasos para la configuración para un router que funcione con el Cisco IOS Software Release 12.2(13)T y Posterior. Si ambos dispositivos VPN son NAT-T capaz, el Traversal NAT es auto detectado y auto negociado.

ISAKMP/Oakley — Vea el IKE.

Publicación de mensaje 5 (MD5) — Éste es un algoritmo de troceo de una manera que produce un hash del 128-bit. El MD5 y el Secure Hash Algorithm (SHA) son variaciones en el MD4, que se diseña para consolidar la Seguridad de este algoritmo de troceo. SHA es más seguro que MD4 y MD5. Las aplicaciones de Cisco desmenuzan para la autenticación dentro del marco del IPSec.

Oakley — El es un Key Exchange Protocol que define cómo adquirir el material de codificación autenticado. El mecanismo básico para Oakley es el algoritmo de intercambio de claves Diffie-Hellman. Usted puede encontrar el estándar en el RFC 2412: El Protocolo de determinación de claves OAKLEY .

Confidencialidad directa perfecta (PFS) — El PFS se asegura de que una clave dada IPSec SA no fuera derivada de ningún otro secreto, como algunas otras claves. Es decir si alguien rompe una clave, el PFS se asegura de que el atacante no pueda derivar ninguna otra clave. Si el PFS no se habilita, alguien puede potencialmente romper la clave secreta IKE SA, copia todos los datos protegidos del IPSec, y después utiliza el conocimiento del secreto IKE SA para comprometer el SA de IPSec puesto por este IKE SA. Con el PFS, la fractura del IKE no da a atacante el acceso inmediato al IPSec. El atacante necesita romper cada IPSec SA individualmente. La implementación del Cisco IOS IPsec utiliza el group1 PFS (D-H 768 mordido) por abandono.

Respuesta-detección — Esto es un servicio de seguridad donde el receptor puede rechazar viejo o los paquetes duplicados para derrotar los ataques con paquetes copiados. Los ataques con paquetes copiados confían en el atacante para enviar más viejo o los paquetes duplicados al receptor y al receptor para pensar que el tráfico falso es legítimo. la Respuesta-detección es hecha por el uso de los números de secuencia combinados con la autenticación, y es una característica estándar del IPSec.

RSA — Esto es un algoritmo criptográfico de la clave pública, nombrado después de sus inventores, Rivest, Shamir y Adleman, con una longitud de clave variable. La debilidad principal del RSA es que es perceptiblemente lenta computar comparado a los algoritmos populares de la clave secreta, tales como DES. La implementación de IKE de Cisco utiliza intercambio Diffie-Hellman para conseguir las claves secretas. Este intercambio se puede autenticar con el RSA, o las claves previamente compartidas. Con intercambio Diffie-Hellman, la clave DES nunca cruza la red, ni siquiera en la forma encriptada, que no es el caso con el RSA cifra y firma la técnica. El

RSA no es un public domain, y se debe autorizar de Rsa Data Security.

Asociación de seguridad (SA) — Éste es un caso de la política de seguridad y del material de codificación aplicados a un flujo de datos. Tanto IKE como IPsec usan SAs, aunque los SAs son independientes entre sí. El SA de IPSec es unidireccional y él es único en cada Security Protocol. Se necesita un conjunto de SAs para una canalización de datos protegida, una por dirección y por protocolo. Por ejemplo, si usted tiene un tubo que soporte el ESP entre los pares, un ESP SA se requiere para cada dirección. Los SA son identificados únicamente por el direccionamiento del destino (punto final de IPSec), el Security Protocol (AH o ESP), y el Security Parameter Index (SPI).

IKE negocia y establece los SAs en nombre de IPsec. Un usuario puede también establecer el SA de IPSec manualmente.

IKE SA es utilizado por el IKE solamente. A diferencia IPSec SA, es bidireccional.

Secure Hash Algorithm (SHA) — Esto es un hash de una manera presentado por el NIST. El SHA se modela de cerca después del MD4 y presenta una publicación del 160-bit. Porque el SHA presenta una publicación del 160-bit, es más resistente a los ataques de fuerza bruta que el 128-bit desmenuza (por ejemplo el MD5), pero es más lento.

Túnel dividido — Éste es el proceso de permitir que un usuario de VPN remoto para acceder una red pública, lo más comúnmente posible Internet, a la vez que se permite al usuario acceder los recursos en la oficina remota. Este método de acceso a la red permite al usuario para acceder los dispositivos remotos, tales como una impresora conectada y servidores al mismo tiempo que para acceder la red pública (Internet). Una ventaja del uso del Túnel dividido es que palía los embotellamientos y conserva el ancho de banda pues el tráfico de Internet no tiene que pasar a través del servidor VPN. Una desventaja de este método es que esencialmente hace el VPN vulnerable al ataque pues es accesible a través de la red pública, NON-segura.

Transforme — Una transformación describe un Security Protocol (AH o ESP) con sus algoritmos correspondientes. Por ejemplo, ESP con el algoritmo cifrado DES y HMAC-SHA para la autenticación.

Modo de transporte — Esto es un modo de encapsulación para el modo de transporte AH/ESP. encapsula el payload de la capa superior, tal como Transmission Control Protocol (TCP) o User Datagram Protocol (UDP), del datagrama IP original. Este modo sólo se puede utilizar cuando los pares son puntos finales de la comunicación. El contraste del modo de transporte es modo túnel.

Modo túnel — Ésta es la encapsulación del IP datagram completo para el IPSec. Utilizan en la orden para proteger datagramas que se originaron contra o se destinan al modo túnel al NON-IPSec los sistemas, tales como adentro un escenario del Red privada virtual (VPN).

Configuración ISAKMP

El IKE existe para establecer solamente los SA para el IPSec. Antes de que pueda hacer esto, el IKE debe negociar una relación SA (ISAKMP SA) con el par. Puesto que el IKE negocia su propia directiva, es posible configurar las declaraciones de políticas múltiples con diversas sentencias de configuración, después dejó a los dos host llegar a un acuerdo. El ISAKMP negocia:

- **Un algoritmo de encripción** — Esto se limita a 56-bit DES solamente.
- **Un algoritmo de troceo** — MD5 o SHA

- **Autenticación** — Firmas RSA., nonces encriptados RSA (números aleatorios), o claves previamente compartidas
- **Curso de la vida del SA** — En los segundos

Actualmente, hay dos métodos usados para configurar el ISAKMP:

1. Utilice las claves previamente compartidas, que son simples configurar.
2. Utilice **CA**, que es scalable en la empresa.

Nota: La negociación IKE se hace en UDP 500. Protocolos 50 y 51 IP de las aplicaciones del IPSec. Aseegurese éstos se permiten en cualquier Listas de acceso que usted tenga entre los pares.

1. Claves previamente compartidas

Éste es el método rápido y sucio usado para configurar el IKE. Mientras que la configuración IKE es simple y usted no utiliza CA, no escala muy bien.

Usted necesita hacer éstos para configurar el IKE:

- Habitaciones de la protección ISAKMP de la configuración.
- Clave de la configuración ISAKMP.

Habitaciones de la protección ISAKMP de la configuración

Este comando crea el objeto de la política isakmp. Es posible tener políticas múltiples, pero hay solamente uno en este ejemplo:

```
dt3-45a(config)#crypto isakmp policy 1 dt3-45a(config-isakmp)#
```

Con el **comando group**, usted puede declarar qué módulo del tamaño a utilizar para el cálculo de Diffie Hellman. El group1 es 768 bits de largo, y el group2 es 1024 bits de largo. ¿Por qué usted utilizaría uno sobre el otro? No todo el grupo de soporte 2. de los vendedores. También, el group2 es también significantly more uso intensivo de la CPU que el grupo uno. Por este motivo, usted no quiere utilizar el group2 en los routeres de menor capacidad como las Cisco 2500 Series o menos. Pero, el group2 es más seguro que el group1. Puesto que este ejemplo utiliza un Cisco4500, el group2 se utiliza, y se aseegura al par también se configura para utilizar el group2. El valor por defecto es group1. Si usted selecciona las propiedades predeterminadas, las líneas del group1 no aparecen cuando usted hace un **comando write terminal**.

```
dt3-45a(config-isakmp)#group 2
```

El MD5 es nuestro algoritmo de troceo en esta línea. Mientras que la implementación del SHA y el MD5 son ambo obligatorios, no todos los pares pueden ser configurados para negociar uno o el otro. El predeterminado en Cisco IOS es SHA que es más seguro que MD5.

```
dt3-45a(config-isakmp)#hash md5
```

El curso de la vida del SA, 500 segundos en este caso, se muestra en este comando. Si usted no fija un curso de la vida, omite 86400 segundos, o un día. Cuando se enciende el temporizador de vida útil, se vuelve a negociar la SA como medida de seguridad.

```
dt3-45a(config-isakmp)#lifetime 500
```

En este comando, el IKE se dice manualmente qué clave a utilizar. Por lo tanto, utilizan al **comando pre-share**. Las dos opciones aparte del comando pre-share son los comandos rsa-encr y rsa-sig. El comando rsa-encr configura nonces encripcións RSA y el comando rsa-sig configura

la firma RSA. **El RSA-encr** y los **comandos rsa-sig** se dirigen en el uso una sección de CA. Por ahora, recuerde que el RSA-**SIG** es el valor por defecto.

```
dt3-45a(config-isakmp)#authentication pre-share
```

Configure la clave ISAKMP

En estos comandos, se dice el IKE qué clave a utilizar. El par, 192.168.10.38 en este caso, debe tener la misma Slurpee-máquina dominante en su configuración.

```
dt3-45a(config-isakmp)#exit dt3-45a(config)#crypto isakmp key Slurpee-Machine address 192.168.10.38
```

Le ahora hacen con la configuración IKE. Estas líneas son la configuración IKE del par. Las configuraciones completas para ambo Routers están en la sección de configuraciones de muestra de este documento:

```
crypto isakmp policy 1
 hash md5
 group 2
 authentication pre-share
crypto isakmp key Slurpee-Machine address 192.168.10.66
```

2. Utilice CA

El uso de CA es un método complejo usado para configurar el IKE. Puesto que es muy scalable en el IPSec, usted necesita utilizar el IPSec en vez de la criyptografía clásicao. Cuando se libera el Cisco IOS Software Release 11.3(3), van solamente a ser algunos vendedores de CA que envían el producto. Inicialmente, la mayoría de las configuraciones se hacen con el uso de las **claves previamente compartidas**. Verisign, confía, Microsoft y Netscape, y probablemente un host de otros, está funcionando en los Productos de CA. Por este ejemplo, se utiliza Verisign CA.

Usted necesita hacer éstos para utilizar CA:

- Cree los pares claves RSA para el router.
- Pida el certificado de CA.
- Aliste los Certificados para el router de cliente.
- Configure las habitaciones de la protección ISAKMP.

Cree los pares de claves RSA para el router

El comando de las uso-**claves GEN rsa del crypto key** puede confundirle. Este comando crea dos pares de claves para el RSA:

- un par clave para el cifrado
- un par clave para las firmas digitales

Un par clave refiere a una clave pública y a su clave secreta correspondiente. Si usted no especifica las uso-claves en el final del comando, el router genera solamente un par clave RSA y lo utiliza para el cifrado y las firmas digitales. Como advertencia, ese este comando se puede utilizar para crear las claves DSS. Pero el DSS es una criyptografía clásicao de la parte de, no IPSec.

```
dt3-45a(config)#crypto key gen rsa usage-keys The name for the keys will be: dt3-45a.cisco.com %You
already have RSA keys defined for dt3-45a.cisco.com. %Do you really want to replace them? [yes/no] yes
```

Puesto que algunas claves RSA existen ya en este cuadro, pregunta si usted quiere librarse de las claves que existen. Puesto que la respuesta está sí, confirme el comando. Se vuelve este

prompt:

```
Choose the size of the key modulus in the range of
   360 to 2048 for your Signature keys.
Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: <return>
Generating RSA keys...
[OK]

Choose the size of the key modulus in the range of
   360 to 2048 for your Encryption keys.
Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: <return>
Generating RSA keys...
[OK]
dt3-45a(config)#
```

Los pares claves RSA con el módulo predeterminado del 512-bit ahora se crean. Salga fuera del modo de configuración y ingrese un **comando show crypto key mypubkey rsa**. Usted puede ahora ver su clave pública RSA. La porción de la clave privada del par clave nunca se considera. Incluso si usted no tiene claves preexistentes, usted ve la misma cosa de previamente.

Nota: Recuerde salvar su configuración una vez que usted ha generado sus pares de claves.

Pida un certificado de CA

Usted ahora necesita configurar al router para hablar con CA. Esto implica varios pasos. Usted necesita coordinar eventual con su administrador de CA.

En estas líneas de configuración, un Domain Name se agrega al router. Esto crea un **ciscoca-ultra del** nombre de host, y dice a router cuál es su dirección IP, y los Servidores de nombres. Usted necesita tener los nombres de host definidos para CA o un DNS que trabaje en el cuadro. Cisco recomienda que usted tiene un DNS que trabaje en el cuadro.

```
dt3-45a(config)#ip host ciscoca-ultra 171.69.54.46 dt3-45a(config)#ip domain-name cisco.com dt3-
45a(config)#ip name-server 171.692.132 dt3-45a(config)#ip name-server 198.92.30.32
```

Comience a configurar los parámetros de CA. **el Verisign-Ca** es apenas un nombre arbitrario.

```
dt3-45a(config)#crypto ca identity verisign-ca dt3-45a(ca-identity)#
```

En esta salida, el Enrollment Protocol de Cisco utiliza el HTTP para hablar con CA. El comando **URL http://ciscoca-ultra dt3-45a(ca-identity)#enrollment** dice al router ir al URL especificado para obrar recíprocamente con CA. **Los dt3-45a(ca-identity)#crypto Ca autentican el** comando Verisign-**Ca** dan instrucciones al router para traer el certificado de CA. Antes de que usted pueda alistar en CA, usted necesita aseegurarse le hablar con el CA real verifica el certificado de CA con el administrador de CA para asegurar la autenticidad.

```
dt3-45a(ca-identity)#enrollment url http://ciscoca-ultra dt3-45a(ca-identity)#exit dt3-45a(ca-
identity)#crypto ca authenticate verisign-ca
```

Aliste los Certificados para el router de cliente

Publique el **Ca crypto alistan el** comando Verisign-**Ca** para comenzar la inscripción con CA. Existen numerosos pasos para esto. Primero, usted tiene que verificar la identidad de CA, después CA tiene que verificar la identidad del router. Si usted necesita nunca revocar su certificado antes de que expire, si usted renumera las interfaces de su router o si usted cree que

su certificado está comprometido, usted necesita proporcionar una contraseña al administrador de CA. Ingrese eso, como se ilustra en esta salida. Después de que usted ingrese su contraseña, el router continúa.

```
dt3-45a(config)#crypto ca enroll verisign-ca %Start certificate enrollment .. %Create a challenge
password. You will need to verbally provide this password to the CA Administrator in order to revoke your
certificate. For security reasons your password will not be saved in the configuration. Please make a
note of it. Password: Re-enter password:
```

Usted ahora ve las huellas dactilares del CA verificar que las huellas dactilares están correctas con el administrador de CA. Además, si usted hace un **comando show crypto ca cert**, usted ve el certificado de CA, además de sus propios Certificados. Los Certificados de CA se enumeran como pendiente ahora.

```
% The subject name for the keys will be: dt3-45a.cisco.com
% Include the router serial number in the subject name? [yes/no]: yes
% The serial number in the certificate will be: 01204044
% Include an IP address in the subject name? [yes/no]: yes
Interface: Ethernet 0
Request certificate from CA? [yes/no]: yes
```

Entre en contacto al administrador de CA porque esta persona quiere confirmar la identidad de la manguera antes de que se publique un certificado. CA publica una vez el certificado, el estatus de nuestros cambios del certificado de pendiente a disponible. Con esto, finaliza la inscripción de CA. Pero, le no hacen. Usted todavía necesita configurar los objetos de la política isakmp.

Habitaciones de la protección ISAKMP de la configuración

El valor por defecto RSA-**SIG** se utiliza en esta salida. Puede tener múltiples conjuntos de protección pero en este ejemplo hay sólo uno. En caso de habitaciones del multipleprotection, las directivas se presentan al par por orden numérico y el par negocia cuál para utilizar. Usted necesita hacer esto si usted sabe que todos sus pares no soportan ciertas características. El router no intenta negociar las cosas que no tienen sentido. Por ejemplo, si usted configura su directiva para el RSA-**SIG** y usted no tiene ningún certificado, el router no negocia esto.

```
dt3-45a(config)#crypto isakmp policy 1 dt3-45a(config-isakmp)#hash md5 dt3-45a(config-isakmp)#lifetime
4000 dt3-45a(config-isakmp)#exit
```

IPSec de la configuración

Si usted utiliza las claves previamente compartidas o configura CA, una vez que usted pone el intercambio de claves de Internet IKE, usted todavía tiene que poner el IPSec. Sin importar qué método IKE usted utiliza, los pasos para la configuración para el IPSec son lo mismo.

Usted necesita hacer éstos para configurar el IPSec:

- Cree el ACL ampliado.
- Cree el IPSec
- Cree la correspondencia de criptografía.
- Aplique la correspondencia de criptografía a la interfaz.

Crear ACL ampliada

Este comando es un ACL muy simple que permite que el Routers hable con uno otro, por ejemplo, Telnet a partir de un router con el siguiente.

```
dt3-45a(config)#access-list 101 permit ip host 192.168.10.38 host 192.168.10.66
```

Un ACL más realista parece este comando. Este comando es un ACL ampliado ordinario, donde está una subred 192.168.3.0 detrás del router en la pregunta, y 10.3.2.0 es una subred en alguna parte detrás del router del par. Recuerde que el **permiso** significa cifra y **niega** significa no cifra.

```
dt3-45a(config)#access-list 101 permit ip 192.168.3.0 0.0.0.255 10.3.2.0 0.0.0.255
```

Cree el IPSec

Cree tres transforman los conjuntos. El primero utiliza sólo ESP, el segundo utiliza AH combinado con ESP y el último, sólo AH. Durante la negociación IPSec SA, los tres se ofrecen al par, que elige uno. También, para los tres transformes el conjunto, utilice al **modo túnel** predeterminado. El modo de transporte puede ser utilizado solamente cuando los puntos finales de criptografía son también los puntos finales de la comunicación. El modo transporte puede ser especificado mediante el comando mode transport en la configuración transform-set. El modo túnel se usa principalmente para el escenario VPN. También observe eso **esp-rfc1829** y **ah-rfc1828** se basan en los RFC originales para esta tecnología y son Obsoletos transforman incluido para la compatibilidad hacia atrás. No todos los vendedores soportan éstos transforman, pero los otros vendedores soportan solamente éstos transforman.

Los conjuntos de la transformación en estos comandos no son necesariamente los más prácticos. Por ejemplo, PapaBear y BabyBear tienen transformes el conjunto inferiores al nivel normal. Uso **esp-rfc1829** y **ah-rfc1828** junto en el mismo transforme el conjunto.

```
dt3-45a(config)#crypto ipsec transform-set PapaBear esp-rfc1829 dt3-45a(cfg-crypto-trans)#exit dt3-45a(config)#crypto ipsec transform-set MamaBear ah-md5-hmac esp-des dt3-45a(cfg-crypto-trans)#exit dt3-45a(config)#crypto ipsec transform-set BabyBear ah-rfc1828 dt3-45a(cfg-crypto-trans)#exit dt3-45a(config)#
```

Cree correspondencia de criptografía

La etiqueta IPSec-**ISAKMP** dice a router que esta correspondencia de criptografía es una correspondencia de criptografía del IPSec. Aunque haya solamente un par declarado en esta correspondencia de criptografía, usted puede tener los peeres múltiples dentro de una correspondencia de criptografía dada. **El curso de la vida de la clave de la sesión** se puede expresar en los kilobytes (después de la x-cantidad de tráfico, cambie la clave) o los segundos, como se muestra en estos comandos. La meta de esto es hacer los esfuerzos de un atacante potencial más difíciles. **El comando set transform-set** es donde usted asocia transforma con la correspondencia de criptografía. Además, la orden en la cual usted declara que transforma es significativa. MamaBear se prefiere más en esta configuración, y entonces el resto en el orden descendente de preferencia a través a BabyBear. Los medios del **comando match address 101** de utilizar la lista de acceso 101 para determinar qué tráfico es relevante. Usted puede tener correspondencias de criptografía múltiples con el mismo nombre, que es armadillo, en este ejemplo, y diversos números de secuencia, que es 10, en este ejemplo. Las correspondencias de criptografía de la suma de múltiples y los diversos números de secuencia permiten que usted mezcle y que haga juego la criyptografía clásicao y el IPSec. Aquí también puede modificar su configuración de PFS. El group1 PFS es el valor por defecto en este ejemplo. Usted puede cambiar el PFS al group2, o déle vuelta de todos junto, que usted no debe hacer.

```
dt3-45a(config)#crypto map armadillo 10 ipsec-isakmp dt3-45a(config-crypto-map)#set peer 192.168.10.38 dt3-45a(config-crypto-map)#set session-key lifetime seconds 4000 dt3-45a(config-crypto-map)#set transform-set MamaBear PapaBear BabyBear dt3-45a(config-crypto-map)#match address 101
```

Aplique crypto map en la interfaz

Estos comandos apply la correspondencia de criptografía a la interfaz. Usted puede asignar solamente un conjunto de la correspondencia de criptografía a una interfaz. Si las entradas de correspondencia de criptografía múltiples tienen el mismo nombre de asignación pero un diverso seq-numérico, son parte del mismo conjunto y son todas aplicadas a la interfaz. El dispositivo de seguridad evalúa la **entrada de correspondencia de criptografía** con la primera seq-numérico más bajo.

```
dt3-45a(config)#interface e0 dt3-45a(config-if)#crypto map armadillo
```

Consideraciones de memoria y CPU

Los paquetes que son procesados por el IPSec son más lentos que los paquetes que se procesan con la criyptografía clásicao. Hay varias razones de esto y puede ser que causen los problemas de rendimiento significativos:

1. El IPSec introduce la expansión de paquetes, que es más probable requerir la fragmentación y el nuevo ensamble correspondiente de los datagramas del IPSec.
2. Los paquetes encriptados se autentican probablemente, así que significa que hay dos operaciones criptográficas que se realizan para cada paquete.
3. Los algoritmos de autenticación son lentos, aunque el trabajo se haya hecho para acelerar las cosas como los cómputos de Diffie Hellman.

Además, intercambio de claves Diffie-Hellman usado en el IKE es una exponenciación mismo de los números grandes (entre 768 y 1024 bytes) y puede tomar hasta cuatro segundos en un Cisco2500. El funcionamiento del RSA es dependiente en el tamaño del número primario elegido para el par clave RSA.

Para cada router, la base de datos SA toma aproximadamente 300 bytes, más 120 bytes para cada SA en esto. En las situaciones donde hay dos SA de IPSec, un entrante y un saliente, se requieren 540 bytes, en la mayoría de los casos. Cada entrada IKE SA es aproximadamente 64 bytes cada uno. La única vez que usted tiene un IPSec SA para un flujo de datos es cuando la comunicación es unidireccional.

IPSec y IKE afecta el rendimiento cuando active. Los intercambios de la clave Diffie-Hellman, autenticación de la clave pública, y encripción/desencripción consumen una cantidad significativa de recursos. Aunque, mucho esfuerzo se haya hecho para minimizar este impacto.

Hay una pequeña disminución del funcionamiento para los paquetes no encriptados que pasan a través de una interfaz que haga crypto. Esto es porque todos los paquetes tienen que ser marcados contra la correspondencia de criptografía. No hay impacto del rendimiento en los paquetes que atraviesan al router que evita una interfaz que haga crypto. El impacto más grande está en los flujos de los datos encriptados.

Utilice el group1 para los intercambios de la clave Diffie-Hellman dentro del IKE, utilice el MD5 como su algoritmo de troceo, y utilice cursos de la vida más largos para minimizar el impacto del subsistema de criptografía en el resto del router. En el equilibrio para este ajuste del rendimiento, usted puede conseguir la criptografía débil. En última instancia, está hasta la política de seguridad del cliente para determinar que ofrece para utilizar y que al dejar solo.

Salida de los comandos show

Nota: Las capturas en estas secciones se toman de una diversa serie de pruebas que ésos

usados en las secciones anteriores de este documento. Por lo tanto, estas capturas pueden tener diversos IP Addresses y reflejar configuraciones levemente diversas. Proporcionan otra serie de **comandos show** en la sección de información del debug de este documento.

Resultado relacionado con IKE

Estudie estos comandos para marcar la inscripción de Verisign CA. Estos comandos show las claves públicas que usted utiliza para la encripción RSA y las firmas.

```
dt1-45a#show crypto key mypubkey rsa % Key pair was generated at: 11:31:59 PDT Apr 9 1998 Key name: dt1-
45a.cisco.com Usage: Signature Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241
00C11854 39A9C75C 4E34C987 B4D7F36C A058D697 13172767 192166E1 661483DD 0FDB907B F9C10B7A CB5A034F
A41DF385 23BEB6A7 C14344BE E6915A12 1C86374F 83020301 0001 % Key pair was generated at: 11:32:02 PDT Apr
9 1998 Key name: dt1-45a.cisco.com Usage: Encryption Key Key Data: 305C300D 06092A86 4886F70D 01010105
00034B00 30480241 00DCF5AC 360DD5A6 C69704CF 47B2362D 65123BD4 424B6FF6 AD10C33E 89983D08 16F1EA58
3700BCF9 1EF17E71 5931A9FC 18D60D9A E0852DDD 3F25369C F09DFB75 05020301 0001
```

Este comando muestra a Certificados que el router reconoce. Un certificado que tiene el **estado pendiente** se ha presentado a CA para la aprobación.

```
dt1-45a#show crypto ca certificates Certificate Subject Name Name: dt1-45a.cisco.com Serial Number:
01193485 Status: Available Certificate Serial Number: 650534996414E2BE701F4EF3170EDFAD Key Usage:
Signature CA Certificate Status: Available Certificate Serial Number: 3051DF7169BEE31B821DFE4B3A338E5F
Key Usage: Not Set Certificate Subject Name Name: dt1-45a.cisco.com Serial Number: 01193485 Status:
Available Certificate Serial Number: 1e621faf3b9902bc5b49d0f99dc66d14 Key Usage: Encryption
```

Esta salida muestra las claves públicas del router y donde el router aprendió sobre ellas.

```
dt1-45a#show crypto key pubkey-chain rsa Codes: M - Manually configured, C - Extracted from certificate
Code Usage IP-Address Name C Signing Cisco SystemsDevtestCISCOCA-ULTRA C General 172.21.30.71 dt1-
7ka.cisco.com
```

Esta es la tabla SA ISAKMP (IKE). Aquí usted ve que un SA existe actualmente entre 172.21.30.71 y 172.21.30.70. El par necesita tener una entrada SA en el mismo estado que la salida de este router.

```
dt1-7ka#show crypto isakmp sa dst src state conn-id slot 172.21.30.70 172.21.30.71 QM_IDLE 47 5
```

Estas líneas muestran los objetos de la directiva configurados. En este caso, las directivas **1**, **2**, y **4** se utilizan, además del valor por defecto. Las directivas se proponen al par en la orden, con **1** como preferido.

```
dt1-45a#show crypto isakmp policy Protection suite of priority 1 encryption algorithm: DES - Data
Encryption Standard (56 bit keys). hash algorithm: Message Digest 5 authentication method: Rivest-Shamir-
Adleman Signature Diffie-Hellman group: #1 (768 bit) lifetime: 180 seconds, no volume limit Protection
suite of priority 2 encryption algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm:
Secure Hash Standard authentication method: Pre-Shared Key Diffie-Hellman group: #2 (1024 bit) lifetime:
180 seconds, no volume limit Protection suite of priority 4 encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Message Digest 5 authentication method: Pre-Shared Key Diffie-
Hellman group: #2 (1024 bit) lifetime: 180 seconds, no volume limit Default protection suite encryption
algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature Diffie-Hellman group: #1 (768 bit) lifetime: 86400
seconds, no volume limit
```

Comandos show IPSec-relacionados

Este comando muestra la correspondencia de criptografía **ToOtherRouter**, los ACL, y las propuestas de transformación aplicadas a esta correspondencia de criptografía, a los pares, y a la vida útil de la clave.

```
S3-2513-2#show crypto map Crypto Map "ToOtherRouter" 10 ipsec-isakmp Peer = 192.168.1.1 Extended IP
```

```
access list 101 access-list 101 permit ip source: addr = 192.168.45.0/0.0.0.255 dest: addr =
192.168.3.0/0.0.0.255 Connection Id = UNSET (0 established, 0 failed) Current peer: 192.168.1.1 Session
key lifetime: 4608000 kilobytes/3600 seconds PFS (Y/N): N Transform proposals={ Elvis, Bubba, BarneyDino,
}
```

Esta configuración utiliza el mismo router como la salida anterior, pero diversos comandos. Usted ve todas las propuestas de transformación, que las configuraciones ellas negocian, y los valores por defecto.

```
S3-2513-2#show crypto ipsec transform-set Transform proposal Elvis: { ah-sha-hmac } supported settings =
{ Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, { esp-des } supported settings
= { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, Transform proposal Bubba: {
ah-rfc1828 } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel,
}, { esp-des esp-md5-hmac } supported settings = { Tunnel, }, default settings = { Tunnel, }, will
negotiate = { Tunnel, }, Transform proposal BarneyDino: { ah-md5-hmac } supported settings = { Tunnel, },
default settings = { Tunnel, }, will negotiate = { Tunnel, },
```

Este comando muestra las asociaciones de seguridad IPSec actuales de este router. El router tiene un AH SA para entrante y saliente.

```
S3-2513-2#show crypto ip session Session key lifetime: 4608000 kilobytes/3600 seconds S3-2513-2#show
crypto ipsec sa interface: Ethernet0 Crypto map tag: ToOtherRouter, local addr. 192.168.1.2 local ident
(addr/mask/prot/port): (192.168.45.0/255.255.255.0/0/0) remote ident (addr/mask/prot/port):
(192.168.3.0/255.255.255.0/0/0) current_peer: 192.168.1.1 PERMIT, flags={origin_is_acl,} #pkts encaps: 0,
#pkts encrypt: 0, #pkts digest 0 #pkts decaps: 0, #pkts decrypt: 0, #pkts verify 0 #send errors 5, #recv
errors 0 local crypto endpt.: 192.168.1.2, remote crypto endpt.: 192.168.1.1 path mtu 1500, media mtu
1500 current outbound spi: 25081A81 inbound esp sas: inbound ah sas: spi: 0x1EE91DDC(518594012)
transform: ah-md5-hmac , in use settings ={Tunnel, } slot: 0, conn id: 16, crypto map: ToOtherRouter sa
timing: remaining key lifetime (k/sec): (4608000/3423) replay detection support: Y outbound esp sas:
outbound ah sas: spi: 0x25081A81(621288065) transform: ah-md5-hmac , in use settings ={Tunnel, } slot: 0,
conn id: 17, crypto map: ToOtherRouter sa timing: remaining key lifetime (k/sec): (4608000/3424) replay
detection support: Y
```

Configuraciones de Ejemplo

Esta configuración utiliza las **claves previamente compartidas**. Esta configuración del router se utiliza para crear la salida de los debugs enumerada en la sección de información del debug. Esta configuración permite una red llamada X situado detrás del router de origen para hablar con una red llamada Y situada detrás del router del par. Consulte la documentación del Cisco IOS Software para su versión del Cisco IOS, o utilice la herramienta de búsqueda de comandos (clientes registrados solamente) para más información sobre un comando determinado. Esta herramienta permite que el usuario mire para arriba una descripción detallada o las pautas de configuración para un comando determinado.

Diagrama de la red

Configuraciones

- Router de origen
- Router de par

Router de origen

```
Current configuration:
!
version 11.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
```

```
!
hostname goss-e4-2513
!
enable secret 5 $1$ZuRD$YBaAh3oIv4iltIn0TMCUX1
enable password ww
!
!--- IKE configuration crypto isakmp policy 1 hash md5
authentication pre-share crypto isakmp key Slurpee-Machine
address 20.20.20.21 ! !--- IPsec configuration crypto ipsec
transform-set BearPapa esp-rfc1829 crypto ipsec transform-set
BearMama ah-md5-hmac esp-des crypto ipsec transform-set
BearBaby ah-rfc1828 ! crypto map armadillo 1 ipsec-isakmp set
peer 20.20.20.21 set security-association lifetime seconds
190 set transform-set BearPapa BearMama BearBaby !--- Traffic
to encrypt match address 101 ! interface Ethernet0 ip address
60.60.60.60 255.255.255.0 no mop enabled ! interface Serial0
ip address 20.20.20.20 255.255.255.0 no ip mroute-cache no
fair-queue crypto map armadillo ! interface Serial1 no ip
address shutdown ! interface TokenRing0 no ip address
shutdown ! ip classless ip route 0.0.0.0 0.0.0.0 20.20.20.21
!--- Traffic to encrypt access-list 101 permit ip 60.60.60.0
0.0.0.255 50.50.50.0 0.0.0.255 dialer-list 1 protocol ip
permit dialer-list 1 protocol ipx permit ! line con 0 exec-
timeout 0 0 line aux 0 line vty 0 4 password ww login ! end
```

Router de par

```
Current configuration:
!
version 11.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname goss-c2-2513
!
enable secret 5 $1$DBTl$Wtg2eS7Eb/Cw5l.nDhkEi/
enable password ww
!
ip subnet-zero
!
!--- IKE configuration crypto isakmp policy 1 hash md5
authentication pre-share crypto isakmp key Slurpee-Machine
address 20.20.20.20 ! !--- IPsec configuration crypto ipsec
transform-set PapaBear esp-rfc1829 crypto ipsec transform-set
MamaBear ah-md5-hmac esp-des crypto ipsec transform-set
BabyBear ah-rfc1828 ! ! crypto map armadillo 1 ipsec-isakmp
set peer 20.20.20.20 set security-association lifetime
seconds 190 set transform-set MamaBear PapaBear BabyBear !---
Traffic to encrypt match address 101 ! ! ! interface
Ethernet0 ip address 50.50.50.50 255.255.255.0 no ip
directed-broadcast ! interface Serial0 ip address 20.20.20.21
255.255.255.0 no ip directed-broadcast no ip mroute-cache no
fair-queue clockrate 9600 crypto map armadillo ! interface
Serial1 no ip address no ip directed-broadcast shutdown !
interface TokenRing0 no ip address no ip directed-broadcast
shutdown ! ip classless ip route 0.0.0.0 0.0.0.0 20.20.20.20
!--- Traffic to encrypt access-list 101 permit ip 50.50.50.0
0.0.0.255 60.60.60.0 0.0.0.255 dialer-list 1 protocol ip
permit dialer-list 1 protocol ipx permit ! ! line con 0 exec-
timeout 0 0 transport input none line aux 0 line aux 0 line
vty 0 4 password ww login ! end
```

Información acerca de la depuración

Esta sección tiene la salida de los debugs de una sesión normal IKE/IPsec entre dos Routers. Las configuraciones provienen de la sección de ejemplos de configuraciones de este documento. El Routers utiliza una clave previamente compartida. Ambo Routers tiene el **isakmp del debug crypto**, el **IPSec del debug crypto**, y los **comandos debug crypto engine** habilitados. Esto fue probada con un ping extendido de la interfaz de Ethernet del router de origen a la interfaz de los Ethernetes del router del par (60.60.60.60 a 50.50.50.50).

Nota: El azul, las oraciones en cursiva en este ejemplo de salida del debug es notas para ayudarle a seguir qué sucede, ellos no es parte de la salida de los debugs.

- Router de origen
- Router de origen
- Router de par con la misma secuencia de ping, tal como se observa desde el otro lado
- Comandos show de router de par

Router de origen

```
goss-e4-2513#show clock goss-e4-2513#ping Protocol [ip]:
Target IP address: 50.50.50.50 Repeat count [5]: 10 Datagram
size [100]: Timeout in seconds [2]: Extended commands [n]: y
Source address or interface: 60.60.60.60 Type of service [0]:
Set DF bit in IP header? [no]: Validate reply data? [no]:
Data pattern [0xABCD]: Loose, Strict, Record, Timestamp,
Verbose[none]: Sweep range of sizes [n]: Type escape sequence
to abort. Sending 10, 100-byte ICMP Echos to 50.50.50.50,
timeout is 2 seconds: Apr 2 12:03:55.347: IPSEC(sa_request):
, (key eng. msg.) src= 20.20.20.20, dest= 20.20.20.21,
src_proxy= 60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysize= 0, flags= 0x4004 Apr 2
12:03:55.355: IPSEC(sa_request): , (key eng. msg.) src=
20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= AH,
transform= ah-md5-hmac , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysize= 0, flags= 0x4004 Apr 2
12:03:55.363: IPSEC(sa_request): , (key eng. msg.) src=
20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-des , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysi.ze= 0, flags= 0x4004 Apr 2
12:03:55.375: IPSEC(sa_request): , (key eng. msg.) src=
20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= AH,
transform= ah-rfc1828 , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysize= 0, flags= 0x4004 !--- Note that
the router offers to the peer all of the !--- available
transforms. Apr 2 12:03:55.391: ISAKMP (14): beginning Main
Mode exchange Apr 2 12:03:57.199: ISAKMP (14): processing SA
payload. message ID = 0 Apr 2 12:03:57.203: ISAKMP (14):
Checking ISAKMP transform 1 against priority 1 policy Apr 2
12:03:57.203: ISAKMP: encryption DES-CBC Apr 2 12:03:57.207:
ISAKMP: hash MD5 Apr 2 12:03:57.207: ISAKMP: default group 1
Apr 2 12:03:57.207: ISAKMP: auth pre-share Apr 2
12:03:57.211: ISAKMP (14): atts are acceptable. Next payload
is 0 Apr 2 12:03:57.215: Crypto engine 0: generate alg param
Apr 2 12:03:5.8.867: CRYPTO_ENGINE: Dh phase 1 status: 0 Apr
```

2 12:03:58.871: ISAKMP (14): SA is doing pre-shared key authentication.. Apr 2 12:04:01.291: ISAKMP (14): processing KE payload. message ID = 0 Apr 2 12:04:01.295: Crypto engine 0: generate alg param Apr 2 12:04:03.343: ISAKMP (14): processing NONCE payload. message ID = 0 Apr 2 12:04:03.347: Crypto engine 0: create ISAKMP SKEYID for conn id 14 Apr 2 12:04:03.363: ISAKMP (14): SKEYID state generated Apr 2 12:04:03.367: ISAKMP (14): processing vendor id payload Apr 2 12:04:03.371: ISAKMP (14): speaking to another IOS box! Apr 2 12:04:03.371: generate hmac context for conn id 14 Apr 2 12:04:03.615: ISAKMP (14): processing ID payload. message ID = 0 Apr 2 12:04:03.615: ISAKMP (14): processing HASH payload. message ID = 0 Apr 2 12:04:03.619: generate hmac context for conn id 14 Apr 2 12:04:03.627: ISAKMP (14): SA has been authenticated Apr 2 12:04:03.627: ISAKMP (14): beginning Quick Mode exchange, M-ID of 1628162439 *!--- These lines represent verification that the policy !--- attributes are fine, and the final authentication of the IKE SA. !--- Once the IKE SA is authenticated, a valid IKE SA exists. !--- New IKE kicks off IPsec negotiation:* Apr 2 12:04:03.635: IPSEC(key_engine): got a queue event... Apr 2 12:04:03.635: IPSEC(spi_response): getting spi 303564824ld for SA .!!!from 20.20.20.21 to 20.20.20.20 for prot 3 Apr 2 12:04:03.639: IPSEC(spi_response): getting spi 423956280ld for SA from 20.20.20.21 to 20.20.20.20 for prot 2 Apr 2 12:04:03.643: IPSEC(spi_response): getting spi 415305621ld for SA from 20.20.20.21 to 20.20.20.20 for prot 3 Apr 2 12:04:03.647: IPSEC(spi_response): getting spi 218308976ld for SA from 20.20.20.21 to 20.20.20.20 for prot 2 Apr 2 12:04:03.891: generate hmac context for conn id 14 Apr 2 12:04:04.!! Success rate is 50 percent (5/10), round-trip min/avg/max = 264/265/268 ms goss-e4-2513#723: generate hmac context for conn id 14 Apr 2 12:04:04.731: ISAKMP (14): processing SA payload. message ID = 1628162439 Apr 2 12:04:04.731: ISAKMP (14): Checking IPSec proposal 1 Apr 2 12:04:04.735: ISAKMP: transform 1, ESP_DES_IV64 Apr 2 12:04:04.735: ISAKMP: attributes in transform: Apr 2 12:04:04.735: ISAKMP: encaps is 1 Apr 2 12:04:04.739: ISAKMP: SA life type in seconds Apr 2 12:04:04.739: ISAKMP: SA life duration (basic) of 190 Apr 2 12:04:04.739: ISAKMP: SA life type in kilobytes Apr 2 12:04:04.743: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 Apr 2 12:04:04.747: ISAKMP (14): atts are acceptable. *!--- The ISAKMP debug is listed because IKE is the !--- entity that negotiates IPsec SAs on behalf of IPsec.* Apr 2 12:04:04.747: IPSEC(validate_proposal_request): proposal part #1, (key eng. msg.) dest= 20.20.20.21, src= 20.20.20.20, dest_proxy= 50.50.50.0/255.255.255.0/0/0 (type=4), src_proxy= 60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP, transform= esp-rfc1829 , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 Apr 2 12:04:04.759: ISAKMP (14): processing NONCE payload. message ID = 1628162439 Apr 2 12:04:04.759: ISAKMP (14): processing ID payload. message ID = 1628162439 Apr 2 12:04:04.763: ISAKMP (14): processing ID payload. message ID = 1628162439 Apr 2 12:04:04.767: generate hmac context for conn id 14 Apr 2 12:04:04.799: ISAKMP (14): Creating IPSec SAs Apr 2 12:04:04.803: inbound SA from 20.20.20.21 to 20.20.20.20 (proxy 50.50.50.0 to 60.60.60.0) Apr 2 12:04:04.803: has spi 303564824 and conn_id 15 and flags 4 Apr 2 12:04:04.807: lifetime of 190 seconds Apr 2 12:04:04.807: lifetime of 4608000 kilobytes Apr 2 12:04:04.811: outbound SA from 20.20.20.20 to 20.20.20.21 (proxy 60.60.60.0 to 50.50.50.0) Apr 2 12:04:04.811: has spi 183903875 and conn_id 16 and flags 4 Apr 2 12:04:04.815: lifetime of 190 seconds Apr 2 12:04:04.815: lifetime of

```
4608000 kilobytes Apr 2 12:04:04.823: IPSEC(key_engine): got
a queue event... Apr 2 12:04:04.823: IPSEC(initialize_sas): ,
(key eng. msg.) dest= 20.20.20.20, src= 20.20.20.21,
dest_proxy= 60.60.60.0/255.255.255.0/0/0 (type=4), src_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0x12180818(303564824), conn_id= 15, keysize= 0, flags= 0x4
Apr 2 12:04:04.831: IPSEC(initialize_sas): , (key eng. msg.)
src= 20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0xAF62683(183903875), conn_id= 16, keysize= 0, flags= 0x4 Apr
2 12:04:04.839: IPSEC(create_sa): sa created, (sa) sa_dest=
20.20.20.20, sa_prot= 50, sa_spi= 0x12180818(303564824),
sa_trans= esp-rfc1829 , sa_conn_id= 15 Apr 2 12:04:04.843:
IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.21,
sa_prot= 50, sa_spi= 0xAF62683(183903875), sa_trans= esp-
rfc1829 , sa_conn_id= 16 !--- These lines show that IPsec SAs
are created and !--- encrypted traffic can now pass.
```

Salida del comando show del router de origen después de la negociación IKE/IPsec

```
goss-e4-2513#
goss-e4-2513#show crypto isakmp sa dst src state conn-id slot
20.20.20.21 20.20.20.20 QM_IDLE 14 0 goss-e4-2513#show crypto
ipsec sa interface: Serial0 Crypto map tag: armadillo, local
addr. 20.20.20.20 local ident (addr/mask/prot/port):
(60.60.60.0/255.255.255.0/0/0) remote ident
(addr/mask/prot/port): (50.50.50.0/255.255.255.0/0/0)
current_peer: 20.20.20.21 PERMIT, flags={origin_is_acl,}
#pkts encaps: 5, #pkts encrypt: 5, #pkts digest 0 #pkts
decaps: 5, #pkts decrypt: 5, #pkts verify 0 #send errors 5,
#recv errors 0 local crypto endpt.: 20.20.20.20, remote
crypto endpt.: 20.20.20.21 path mtu 1500, media mtu 1500
current outbound spi: AF62683 inbound esp sas: spi:
0x12180818(303564824) transform: esp-rfc1829 , in use
settings ={Var len IV, Tunnel, } slot: 0, conn id: 15, crypto
map: armadillo sa timing: remaining key lifetime (k/sec):
(4607999/135) IV size: 8 bytes replay detection support: N
inbound ah sas: outbound esp sas: spi: 0xAF62683(183903875)
transform: esp-rfc1829 , in use settings ={Var len IV,
Tunnel, } slot: 0, conn id: 16, crypto map: armadillo sa
timing: remaining key lifetime (k/sec): (4607999/117) IV
size: 8 bytes replay detection support: N outbound ah sas:
goss-e4-2513#show crypto isakmp policy Protection suite of
priority 1 encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Message Digest 5
authentication method: Pre-Shared Key Diffie-Hellman group:
#1 (768 bit) lifetime: 86400 seconds, no volume limit Default
protection suite encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature
Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds,
no volume limit goss-e4-2513#show crypto map Crypto Map
"armadillo" 1 ipsec-isakmp Peer = 20.20.20.21 Extended IP
access list 101 access-list 101 permit ip 60.60.60.0
0.0.0.255 50.50.50.0 0.0.0.255 Current peer: 20.20.20.21
Security association lifetime: 4608000 kilobytes/190 seconds
PFS (Y/N): N Transform sets={ BearPapa, BearMama, BearBaby, }
```

Router de par con la misma secuencia de ping, tal como se observa desde el otro lado

```
goss-c2-2513#show debug Cryptographic Subsystem: Crypto
```

```
ISAKMP debugging is on Crypto Engine debugging is on Crypto
IPSEC debugging is on goss-c2-2513# Apr 2 12:03:55.107:
ISAKMP (14): processing SA payload. message ID = 0 Apr 2
12:03:55.111: ISAKMP (14): Checking ISAKMP transform 1
against priority 1 policy Apr 2 12:03:55.111: ISAKMP:
encryption DES-CBC Apr 2 12:03:55.111: ISAKMP: hash MD5 Apr 2
12:03:55.115: ISAKMP: default group 1 Apr 2 12:03:55.115:
ISAKMP: auth pre-share Apr 2 12:03:55.115: ISAKMP (14): atts
are acceptable. Next payload is 0 !--- IKE performs its
operation, and then kicks off IPsec. Apr 2 12:03:55.119:
Crypto engine 0: generate alg param Apr 2 12:03:56.707:
CRYPTO_ENGINE: Dh phase 1 status: 0 Apr 2 12:03:56.711:
ISAKMP (14): SA is doing pre-shared key authentication Apr 2
12:03:58.667: ISAKMP (14): processing KE payload. message ID
= 0 Apr 2 12:03:58.671: Crypto engine 0: generate alg param
Apr 2 12:04:00.687: ISAKMP (14): processing NONCE payload.
message ID = 0 Apr 2 12:04:00.695: Crypto engine 0: create
ISAKMP SKEYID for conn id 14 Apr 2 12:04:00.707: ISAKMP (14):
SKEYID state generated Apr 2 12:04:00.711: ISAKMP (14):
processing vendor id payload Apr 2 12:04:00.715: ISAKMP (14):
speaking to another IOS box! Apr 2 12:04:03.095: ISAKMP (14):
processing ID payload. message ID = 0 Apr 2 12:04:03.095:
ISAKMP (14): processing HASH payload. message ID = 0 Apr 2
12:04:03.099: generate hmac context for conn id 14 Apr 2
12:04:03.107: ISAKMP (14): SA has been authenticated Apr 2
12:04:03.111: generate hmac context for conn id 14 Apr 2
12:04:03.835: generate hmac context for conn id 14 Apr 2
12:04:03.839: ISAKMP (14): processing SA payload. message ID
= 1628162439 Apr 2 12:04:03.843: ISAKMP (14): Checking IPSec
proposal 1 Apr 2 12:04:03.843: ISAKMP: transform 1,
ESP_DES_IV64 Apr 2 12:04:03.847: ISAKMP: attributes in
transform: Apr 2 12:04:03.847: ISAKMP: encaps is 1 Apr 2
12:04:03.847: ISAKMP: SA life type in seconds Apr 2
12:04:03.851: ISAKMP: SA life duration (basic) of 190 Apr 2
12:04:03.851: ISAKMP: SA life type in kilobytes Apr 2
12:04:03.855: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50
0x0 Apr 2 12:04:03.855: ISAKMP (14): atts are acceptable. Apr
2 12:04:03.859: IPSEC(validate_proposal_request): proposal
part #1, (key eng. msg.) dest= 20.20.20.21, src= 20.20.20.20,
dest_proxy= 50.50.50.0/255.255.255.0/0/0 (type=4), src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 0s and 0kb, spi= 0x0(0),
conn_id= 0, keysize= 0, flags= 0x4 Apr 2 12:04:03.867: ISAKMP
(14): processing NONCE payload. message ID = 1628162439 Apr 2
12:04:03.871: ISAKMP (14): processing ID payload. message ID
= 1628162439 Apr 2 12:04:03.871: ISAKMP (14): processing ID
payload. message ID = 1628162439 Apr 2 12:04:03.879:
IPSEC(key_engine): got a queue event... Apr 2 12:04:03.879:
IPSEC(spi_response): getting spi 183903875ld for SA from
20.20.20.20 to 20.20.20.21 for prot 3 Apr 2 12:04:04.131:
generate hmac context for conn id 14 Apr 2 12:04:04.547:
generate hmac context for conn id 14 Apr 2 12:04:04.579:
ISAKMP (14): Creating IPSec SAs Apr 2 12:04:04.579: inbound
SA from 20.20.20.20 to 20.20.20.21 (proxy 60.60.60.0 to
50.50.50.0) Apr 2 12:04:04.583: has spi 183903875 and conn_id
15 and flags 4 Apr 2 12:04:04.583: lifetime of 190 seconds
Apr 2 12:04:04.587: lifetime of 4608000 kilobytes Apr 2
12:04:04.587: outbound SA from 20.20.20.21 to 20.20.20.20
(proxy 50.50.50.0 to 60.60.60.0) Apr 2 12:04:04.591: has spi
303564824 and conn_id 16 and flags 4 Apr 2 12:04:04.591:
lifetime of 190 seconds Apr 2 12:04:04.595: lifetime of
4608000 kilobytes Apr 2 12:04:04.599: IPSEC(key_engine): got
a queue event... Apr 2 12:04:04.599: IPSEC(initialize_sas): ,
(key eng. msg.) dest= 20.20.20.21, src= 20.20.20.20,
```

```
dest_proxy= 50.50.50.0/255.255.255.0/0/0 (type=4), src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0xAF62683(183903875), conn_id= 15, keysize= 0, flags= 0x4 Apr
2 12:04:04.607: IPSEC(initialize_sas): , (key eng. msg.) src=
20.20.20.21, dest= 20.20.20.20, src_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), dest_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0x12180818(303564824), conn_id= 16, keysize= 0, flags= 0x4
Apr 2 12:04:04.615: IPSEC(create_sa): sa created, (sa)
sa_dest= 20.20.20.21, sa_prot= 50, sa_spi=
0xAF62683(183903875), sa_trans= esp-rfc1829 , sa_conn_id= 15
Apr 2 12:04:04.619: IPSEC(create_sa): sa created, (sa)
sa_dest= 20.20.20.20, sa_prot= 50, sa_spi=
0x12180818(303564824), sa_trans= esp-rfc1829 , sa_conn_id= 16
!--- The IPsec SAs are created, and ICMP traffic can flow.
```

Comandos show de router de par

```
!--- This illustrates a series of show command output after
!--- IKE/IPsec negotiation takes place. goss-c2-2513#show
crypto isakmp sa dst src state conn-id slot 20.20.20.21
20.20.20.20 QM_IDLE 14 0 goss-c2-2513#show crypto ipsec sa
interface: Serial0 Crypto map tag: armadillo, local addr.
20.20.20.21 local ident (addr/mask/prot/port):
(50.50.50.0/255.255.255.0/0/0) remote ident
(addr/mask/prot/port): (60.60.60.0/255.255.255.0/0/0)
current_peer: 20.20.20.20 PERMIT, flags={origin_is_acl,}
#pkts encaps: 5, #pkts encrypt: 5, #pkts digest 0 #pkts
decaps: 5, #pkts decrypt: 5, #pkts verify 0 #send errors 0,
#recv errors 0 local crypto endpt.: 20.20.20.21, remote
crypto endpt.: 20.20.20.20 path mtu 1500, media mtu 1500
current outbound spi: 12180818 inbound esp sas: spi:
0xAF62683(183903875) transform: esp-rfc1829 , in use settings
={Var len IV, Tunnel, } slot: 0, conn id: 15, crypto map:
armadillo sa timing: remaining key lifetime (k/sec):
(4607999/118) IV size: 8 bytes replay detection support: N
inbound ah sas: outbound esp sas: spi: 0x12180818(303564824)
transform: esp-rfc1829 , in use settings ={Var len IV,
Tunnel, } slot: 0, conn id: 16, crypto map: armadillo sa
timing: remaining key lifetime (k/sec): (4607999/109) IV
size: 8 bytes replay detection support: N outbound ah sas:
goss-c2-2513#show crypto isakmp policy Protection suite of
priority 1 encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Message Digest 5
authentication method: Pre-Shared Key Diffie-Hellman group:
#1 (768 bit) lifetime: 86400 seconds, no volume limit Default
protection suite encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature
Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds,
no volume limit goss-c2-2513#show crypto map Crypto Map
"armadillo" 1 ipsec-isakmp Peer = 20.20.20.20 Extended IP
access list 101 access-list 101 permit ip 50.50.50.0
0.0.0.255 60.60.60.0 0.0.0.255 Current peer: 20.20.20.20
Security association lifetime: 4608000 kilobytes/190 seconds
PFS (Y/N): N Transform sets={ MamaBear, PapaBear, BabyBear, }
```

Consejos sobre instrumentación para el IPSec

Éstos son algunos consejos sobre instrumentación para el IPSec:

- Asegúrese que usted tenga Conectividad entre los puntos finales de la comunicación antes de que usted configure crypto.
- Aseegurese que o el DNS trabaja en el router, o usted ha ingresado el nombre de host CA, si usted utiliza un CA.
- El IPSec utiliza los protocolos 50 y 51 IP, y el tráfico IKE pasa encendido el protocolo 17, el puerto 500 (UDP 500). Aseegurese éstos se permiten apropiadamente.
- Tenga cuidado de no utilizar la palabra en su ACL. Esto causa los problemas. Refiera a las Pautas para el uso para la **lista de acceso** en la referencia de comandos de PIX para más información.
- Las combinaciones de transformaciones recomendadas son:`esp-des and esp-sha-hmac` `ah-sha-hmac and esp-des`
- Recuerde que AH simplemente es un encabezado autenticado. La secuencia de datos del usuario real no se cifra. Usted necesita el ESP para la encripción de secuencia de datos. Si usted utiliza solamente AH y ve el texto claro ir a través de la red, no se sorprenda. También utilice el ESP si usted utiliza AH. Observe que el ESP puede también realizar la autenticación. Por lo tanto, puede utilizar una combinación de transformación como por ejemplo esp-des y esp-sha-hmac.
- **ah-rfc1828** y **esp-rfc1829** son Obsoletos transforman incluido para la compatibilidad hacia atrás con más viejas implementaciones del IPSec. Si el par no soporta más nuevo transforma, intenta éstos en lugar de otro.
- El SHA es más lento y más seguro que el MD5, mientras que el MD5 es más rápidamente y menos seguro ese SHA. En algunas comunidades, la comodidad llana con el MD5 es muy baja.
- En caso de duda, modo túnel del uso. El modo túnel es el predeterminado y se puede usar en el modo de transporte, así como para sus capacidades de VPN.
- Para los usuarios del comando cryptos clásicos que actualizan al Cisco IOS Software Release 11.3, los métodos del almacenamiento de los comandos crypto en la configuración han cambiado para permitir el IPSec. Por lo tanto, si los usuarios del comando cryptos clásicos invierten nunca al Cisco IOS Software Release 11.2, estos usuarios tienen que entrar sus configuraciones de criptografía de nuevo.
- Si usted hace una prueba de ping a través del link encriptado cuando usted acaba su configuración, el proceso de negociación puede tardar una cierta hora, cerca de seis segundos en un Cisco4500, y cerca de 20 segundos en un Cisco2500, porque SA no tiene con todo negociado. Aunque todo se configura correctamente, su ping puede fallar inicialmente. **Los comandos debug crypto ipsec y debug crypto isakmp** le muestran qué sucede. Una vez que su datastreams cifrado ha acabado su configuración, el ping trabaja muy bien.
- Si usted se ejecuta en el problema con sus negociaciones y realiza los cambios de configuración, utilice el **crypto claro es** y los **comandos clear crypto sa** para vaciar las bases de datos antes de que usted revise. Esto fuerza la negociación para comenzar de nuevo, sin ninguna negociación heredada a conseguir de la manera. **El crypto claro es** y los **comandos clear cry sa** son muy útiles de este modo.

Ayuda y links relevantes

Información de IPSec

- Página de soporte de IPSec
- Políticas de encripción y procedimientos ECRA — Envíe un email a export@cisco.com

Más configuraciones de muestra para el IPSec

- Configurando y resolviendo problemas la encripción de capa de red de Cisco: IPSec y ISAKMP
- Descripción del IPSec Network Security
- Documentación de la configuración IPSec del firewall PIXPIX 5.1PIX 5.2PIX 5.3PIX 6.0PIX 6.1PIX 6.2PIX 6,3

Entre en contacto el Soporte técnico de Cisco en el (800) 553-24HR, (408) 526-7209, o envíelo y email a tac@cisco.com si usted requiere la asistencia adicional con el IPSec.

Referencias

Harkins, *Especificación funcional de la unidad de software de la característica de* D. *ISAKMP/Oakley Protocol.* Rev A. Cisco Systems del ENG-0000.

Madson, *Rev F.* Cisco Systems de la *Especificación funcional de la unidad de software ENG-17610 de* C. *IPSec*.

Kaufman, C. Perlman R. y chaqueta de punto, *seguridad de la red* M.*: Comunicación privada en un Mundo público.* Prentice Hall, 1995.

Schneier, *Criptografía aplicada* B.*: Protocolos, algoritmos, y código fuente en el C.* En segundo lugar Ed. John Wiley & Sons, Inc.

Diversos trabajar-proyectos de la seguridad IP IETF

Información Relacionada

- Página de soporte de IPSec
- Cómo las Redes privadas virtuales funcionan
- Soluciones a los Problemas más frecuentes de IPSec VPN L2L y de Acceso Remoto
- Soporte Técnico y Documentación - Cisco Systems