Fragmentación de la resolución IPv4, problemas MTU, MSS, y PMTUD con GRE e IPsec

Contenido

Introducción

Este documento describe cómo la fragmentación IPv4 y el descubrimiento del Maximum Transmission Unit de la trayectoria (PMTUD) trabajan y también discuten algunos decorados que implica el comportamiento de PMTUD cuando está combinado con diversas combinaciones de los túneles IPv4. El uso extenso actual de los túneles IPv4 en Internet ha traído los problemas que implican la fragmentación IPv4 y PMTUD a la vanguardia.

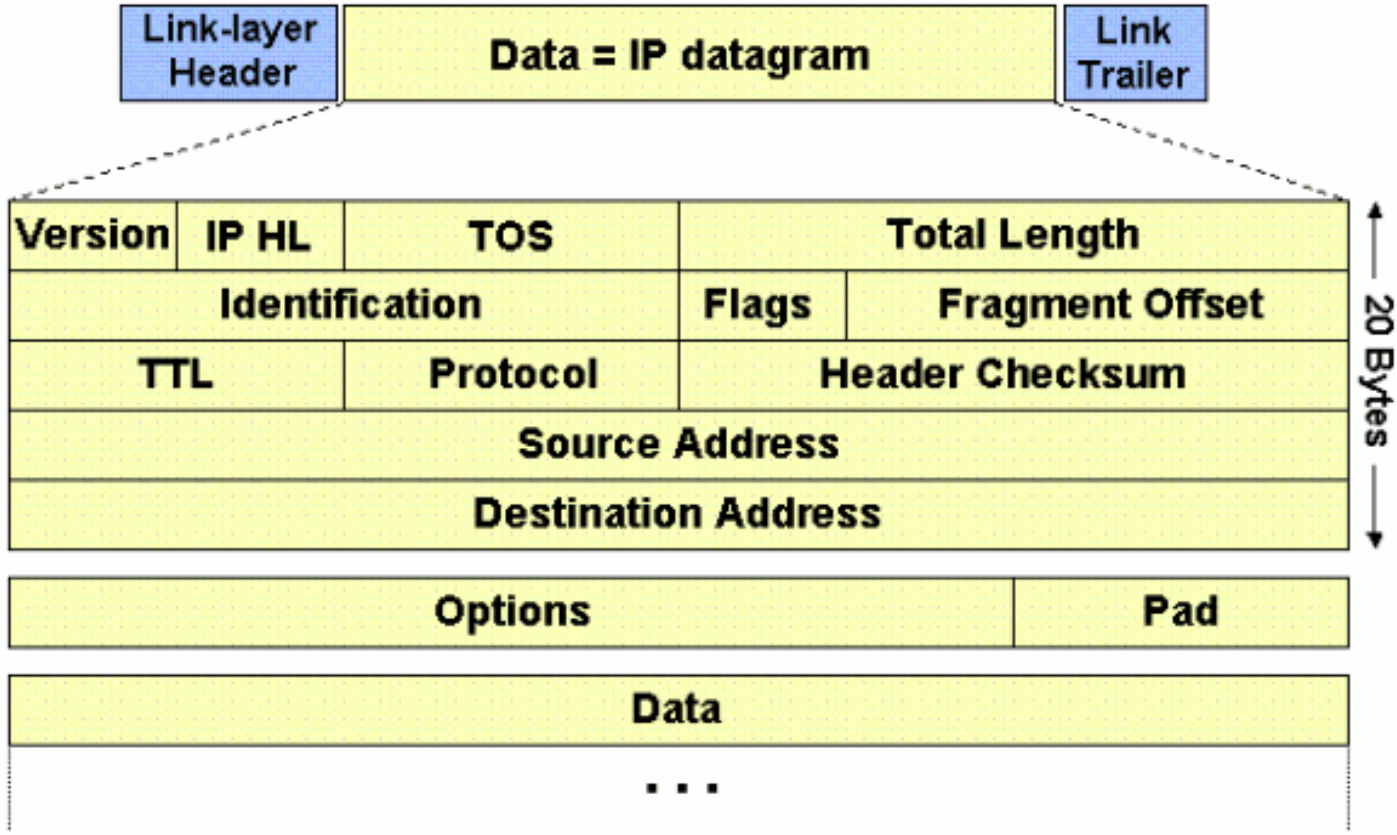
Fragmentación IPv4 y nuevo ensamble

El protocolo IPv4 fue diseñado para el uso en una amplia variedad de links de transmisión. Aunque el Largo máximo de un datagrama IPv4 sea 65535, la mayoría de los links de transmisión aplican un límite más pequeño de la longitud máxima de paquetes, llamado un MTU. El valor del MTU depende del tipo del link de transmisión. El diseño de IPv4 acomoda las diferencias de MTU

puesto que permite que el Routers haga fragmentos de los datagramas IPv4 cuanto sea necesario. La estación receptora es responsable del nuevo ensamble de los fragmentos nuevamente dentro del datagrama del mismo tamaño original IPv4.

La fragmentación IPv4 implica para romper un datagrama en varios pedazos que se puedan volver a montar más adelante. La fuente IPv4, el destino, la identificación, la longitud total, y los campos del desplazamiento del fragmento, junto con el indicadores “más fragmentos” y “no fragmentar” en la encabezado IPv4, se utilizan para la fragmentación IPv4 y el nuevo ensamble. Para más información sobre los mecánicos de la fragmentación IPv4 y del nuevo ensamble, vea el RFC 791.

Esta imagen representa la disposición de una encabezado IPv4.



La identificación es 16 bits y es un valor asignado por el remitente de un datagrama IPv4 para ayudar en el nuevo ensamble de los fragmentos de un datagrama.

El desplazamiento del fragmento es 13 bits e indica donde un fragmento pertenece en el datagrama original IPv4. Este valor es un múltiplo de ocho bytes.

En el campo de los indicadores de la encabezado IPv4, hay tres bits para los indicadores de control. Es importante observar que el bit "don't fragment" (DF) tiene una función central en la PMTUD porque determina si se permite o no que se fragmente un paquete.

0 mordido es reservado, y se fija siempre a 0. mordió 1 es el bit DF (0 = “puede hacer fragmentos”, 1 = “no hace fragmento”). El Bit 2 es el bit "more fragments" (MF) (0 = "último fragmento", 1 = "más fragmentos").

Valor	Bit 0 Reservado	Bit 1 DF	Bit 2 MF
0	0	Se puede	Último
1	0	No se puede	Más

En el siguiente gráfico, es posible ver un ejemplo de fragmentación. Si usted agrega para arriba todas las longitudes de los fragmentos IPv4, el valor excede la longitud original del datagrama IPv4 por 60. La razón que la longitud total es aumentada en 60 está porque tres encabezados adicionales IPv4 fueron creadas, una para cada fragmento después del primer fragmento.

El primer fragmento tiene un desplazamiento de 0 y la longitud de este fragmento es 1500; esto incluye 20 bytes para la encabezado original levemente modificada IPv4.

El segundo fragmento tiene un desplazamiento de 185 (185 x 8 = 1480), así que significa que la porción de datos de este fragmento comienza 1480 bytes en el datagrama original IPv4. La longitud de este fragmento es 1500; esto incluye la encabezado adicional IPv4 creada para este fragmento.

El tercer fragmento tiene un desplazamiento de 370 (370 x 8 = 2960), así que significa que la porción de datos de este fragmento comienza 2960 bytes en el datagrama original IPv4. La longitud de este fragmento es 1500; esto incluye la encabezado adicional IPv4 creada para este fragmento.

El cuarto fragmento tiene un desplazamiento de 555 (555 x 8 = 4440), así que significa que la porción de datos de este fragmento comienza 4440 bytes en el datagrama original IPv4. La longitud de este fragmento es 700 bytes; esto incluye la encabezado adicional IPv4 creada para este fragmento.

Es solamente cuando se recibe el fragmento pasado que el tamaño del datagrama original IPv4 puede ser resuelto.

El desplazamiento del fragmento en el fragmento pasado (555) da un desplazamiento de los datos de 4440 bytes en el datagrama original IPv4. Si usted entonces agrega los bytes de datos del fragmento pasado (680 = 700 - 20), ése le da 5120 bytes, que es la porción de datos del datagrama original IPv4. Entonces, la adición de 20 bytes para una encabezado IPv4 iguala el tamaño del datagrama original IPv4 (4440 + 680 + 20 = 5140) tal y como se muestra en de las imágenes.

Original IP Datagram

Sequence	Identifier	Total Length	DF May / Don't	MF Last / More	Fragment Offset
0	345	5140	0	0	0

IP Fragments (Ethernet)

Sequence	Identifier	Total Length	DF May / Don't	MF Last / More	Fragment Offset
0-0	345	1500	0	1	0
0-1	345	1500	0	1	185
0-2	345	1500	0	1	370
0-3	345	700	0	0	555

Problemas con la fragmentación IPv4

Hay varios problemas que hacen al undesirable de la fragmentación IPv4. Hay un pequeño aumento en la CPU y memoria suplementaria para hacer fragmentos de un datagrama IPv4. Esto es verdad para el remitente así como para un router en la trayectoria entre un remitente y un receptor. La creación de los fragmentos implica simplemente para crear las encabezados del fragmento y para copiar el datagrama original en los fragmentos. Esto se puede hacer bastante eficientemente porque toda la información necesaria para crear los fragmentos está inmediatamente disponible.

La fragmentación genera mayor sobrecarga para el receptor al reensamblar los fragmentos porque el receptor debe asignar memoria para los fragmentos que llegan y unirlos nuevamente en un datagrama una vez recibidos todos los fragmentos. El nuevo ensamble en un host no se considera un problema porque el host tiene el tiempo y los recursos de memoria para dedicar a esta tarea.

Pero, el reensamblado es muy ineficaz en un router cuya tarea primaria sea reenviar los paquetes tan rápido como sea posible. Un router no está diseñado para conservar los paquetes durante un período de tiempo. También, un router que hace el nuevo ensamble elige el almacenador intermediario más grande disponible (18K) con el cual para trabajar porque no tiene ninguna manera de conocer el tamaño de la original paquete IPV4 hasta que se reciba el fragmento pasado.

Otro problema de la fragmentación radica en cómo se manejan los fragmentos descartados. Si un fragmento de un datagrama IPv4 se cae, después el datagrama entero de la original IPv4 debe ser vuelto a enviar, y también se hace fragmentos. Usted ve un ejemplo de esto con el Network File System (NFS). El NFS, por abandono, tiene un tamaño del bloque de lectura y escritura de 8192, así que un datagrama NFS IPv4/UDP es aproximadamente 8500 bytes (que incluye las encabezados NFS, UDP, e IPv4). Una estación remitente conectada con un Ethernet (MTU 1500) tiene que hacer fragmentos del datagrama de bytes 8500 en seis pedazos; cinco fragmentos de 1500 bytes y un fragmento de 1100 bytes. Si es un de los seis fragmentos se caen debido a un link congestionado, el datagrama original completo tiene que ser retransmitido, así que significa que seis más fragmentos tendrán que ser creados. Si este link cae uno en seis paquetes, después las probabilidades son bajas que cualquier datos NFS se puede transferir sobre este link, puesto que por lo menos un fragmento IPv4 sería caído de cada datagrama del byte original IPv4 NFS 8500.

Los Firewall que filtran o manipulan los paquetes basados en la capa 4 (L4) con la información de la capa 7 (L7) en el paquete pudieron tener problema que procesaba los fragmentos IPv4 correctamente. Si los fragmentos IPv4 están fuera de servicio, un Firewall pudo bloquear los fragmentos no iniciales porque no llevan la información que haría juego el filtro de paquete. Esto significaría que el datagrama original IPv4 no se podría volver a montar por el host de recepción. Si el firewall está configurado para permitir que los fragmentos no iniciales con información insuficiente coincidan correctamente con el filtro, podría ocurrir un ataque de fragmentos no iniciales a través del firewall. Además, algunos dispositivos de red (como los motores de switch de contenido) dirigen paquetes basados en información de L4 a L7 y, si un paquete abarca múltiples fragmentos, es posible que el dispositivo tenga problemas para imponer sus políticas.

Evite la fragmentación IPv4: Qué TCP MSS hace y cómo trabaja

El tamaño del segmento máxima TCP (MSS) define la cantidad máxima de datos que un host está dispuesto a validar en un solo datagrama TCP/IPv4. Este datagrama TCP/IPv4 se pudo

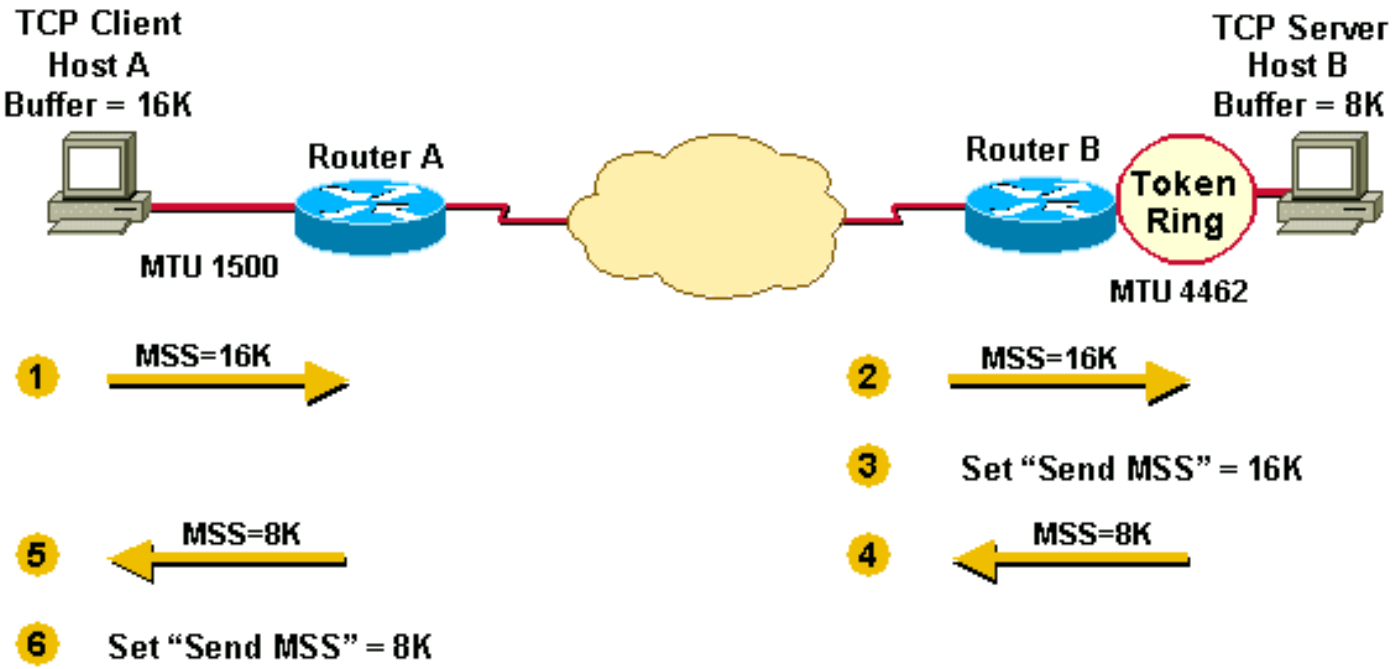
hacer fragmentos en la capa IPv4. El valor de MSS se envía como una opción de encabezado TCP solamente en los segmentos SYN de TCP. Cada lado de una conexión TCP informa su valor de MSS al otro lado. Contrariamente a la creencia popular, el valor de MSS no se negocia entre los hosts. Se requiere que el host remitente limite el tamaño de los datos en un solo segmento TCP a un valor inferior o igual al valor de MSS informado por el host receptor.

Originalmente, MSS significado cómo es grande un almacenador intermediario (mayor o igual 65496 bytes) fue afectado un aparato en una estación receptora para poder salvar los datos de TCP contenidos dentro de un solo datagrama IPv4. El valor de MSS era el segmento máxima (tramo) de datos que el receptor TCP estaba dispuesto a aceptar. Este segmento TCP podría ser tan grande como 64K (el tamaño del datagrama del máximo IPv4) y podría ser hecho fragmentos en la capa IPv4 para ser transmitido a través de la red al host de recepción. El host de recepción volvería a montar el datagrama IPv4 antes de que diera el segmento completo TCP a la capa TCP.

Aquí están un par de decorados que muestran cómo los valores MSS se fijan y se utilizan para limitar los tamaños del segmento TCP, y por lo tanto, los tamaños del datagrama IPv4.

En la situación 1, se ilustra la manera en que se implementó primero el valor de MSS. El Host A tiene un buffer de 16K y el Host B tiene un buffer de 8K. Envían y reciben sus valores MSS y ajustan su envían MSS para enviar los datos el uno al otro. Note que el host A y el host B tendrán que hacer fragmentos de los datagramas IPv4 que son más grandes que el MTU de interfaz, pero aún menos que el envío MSS porque la pila TCP podría pasar los bytes de dato 16K o 8K abajo de la pila a IPv4. En el caso del Host B, los paquetes podrían fragmentarse dos veces, una vez para llegar a la LAN Token Ring y nuevamente para llegar a la LAN Ethernet.

Escenario 1



1. El Host A envía su valor de MSS de 16K al Host B.
2. El Host B recibe el valor de MSS de 16K del Host A.
3. El Host B configura su valor de MSS de envío en 16K.
4. El Host B envía su valor de MSS de 8K al Host A.
5. El Host A recibe el valor de MSS de 8K del Host B.

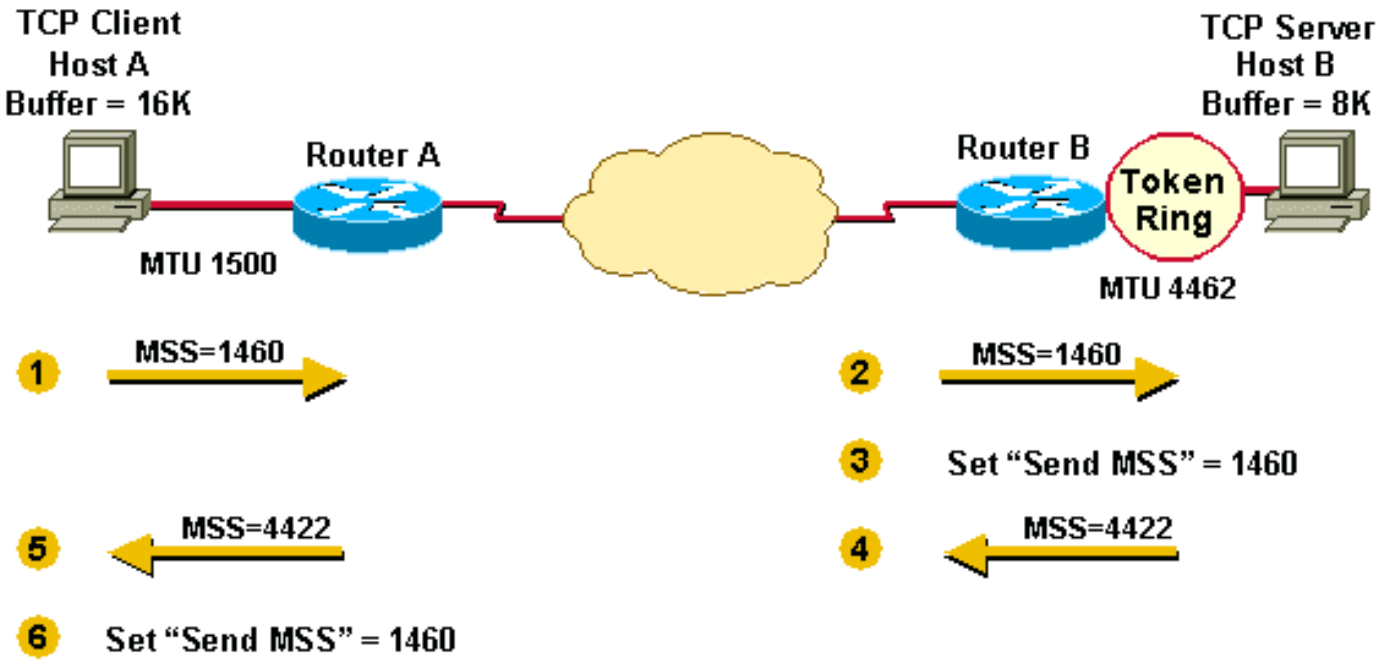
6. El Host A configura su valor de MSS de envío en 8K.

Para ayudar a evitar la fragmentación IPv4 en las puntos finales de la conexión TCP, la selección del valor MSS fue cambiada al tamaño mínimo de memoria intermedia y al MTU de la interfaz saliente (- 40). Los números MSS son 40 bytes más pequeños que los números MTU porque MSS es apenas el tamaño de los datos de TCP, que no incluye 20 la encabezado del byte IPv4 y el encabezado TCP de 20 bytes. El valor de MSS se basa en los tamaños de encabezado predeterminados; la pila del remitente debe restar los valores apropiados para la encabezado IPv4 y el dependiente del encabezado TCP en se utilizan qué opciones TCP o IPv4.

Los trabajos de la manera MSS ahora son que cada host primero comparará su MTU de la interfaz saliente con su propio almacenador intermediario y elegirá el valor más bajo como el MSS para enviar. Los host entonces compararán el tamaño MSS recibido contra su propio MTU de interfaz y elegirán otra vez el más bajo de los dos valores.

En la situación 2, se ilustra este paso adicional que realiza el emisor con el fin de evitar la fragmentación en los cables locales y remotos. Observe cómo cada host tiene en cuenta la MTU de la interfaz saliente (antes de que los hosts se envíen entre sí sus valores de MSS) y cómo esto ayuda a evitar la fragmentación.

Escenario 2



1. El Host A compara su buffer de MSS (16K) y su MTU (1500 - 40 = 1460) y utiliza el valor más bajo como el valor de MSS (1460) para enviarlo al Host B.
2. El Host B recibe el valor de MSS de envío (1460) del Host A y lo compara con el valor de su MTU de interfaz saliente - 40 (4422).
3. El host B fija el valor inferior (1460) como el MSS para enviar los datagramas IPv4 al host A.
4. El Host B compara su buffer de MSS (8K) y su MTU (4462 - 40 = 4422) y utiliza 4422 como el valor de MSS para enviarlo al Host A.
5. El Host A recibe el valor de MSS de envío (4422) del Host B y lo compara con el valor de su MTU de interfaz saliente - 40 (1460).
6. El host A fija el valor inferior (1460) como el MSS para enviar los datagramas IPv4 al host B.

El valor elegido por ambos hosts como MSS de envío recíproco es 1460. A menudo, el valor de MSS de envío será el mismo en cada extremo de una conexión TCP.

En la situación 2, la fragmentación no ocurre en los extremos de una conexión TCP porque ambas MTU de interfaz saliente son tenidas en cuenta por los hosts. Los paquetes pueden todavía hacerse fragmentos en la red entre el router A y el router B si encuentran un link con un MTU inferior que el de la interfaz de salida de cualquier host.

¿Cuál es PMTUD?

El MSS de TCP, como se describió anteriormente, se encarga de la fragmentación en los dos terminales de una conexión TCP, pero no se ocupa de la situación en la que hay un enlace de MTU más pequeño en el medio de estos dos terminales. PMTUD se desarrolló con el fin de evitar la fragmentación en la ruta entre los terminales. Se utiliza para determinar dinámicamente la MTU más baja a lo largo de la trayectoria del origen de un paquete a su destino.

> **Note**: PMTUD solo es compatible con TCP y UDP. Otros protocolos no la admiten. Si PMTUD se activa en un host, y está casi siempre, todo el TCP y paquetes UDP del host tendrán el bit DF fijado.

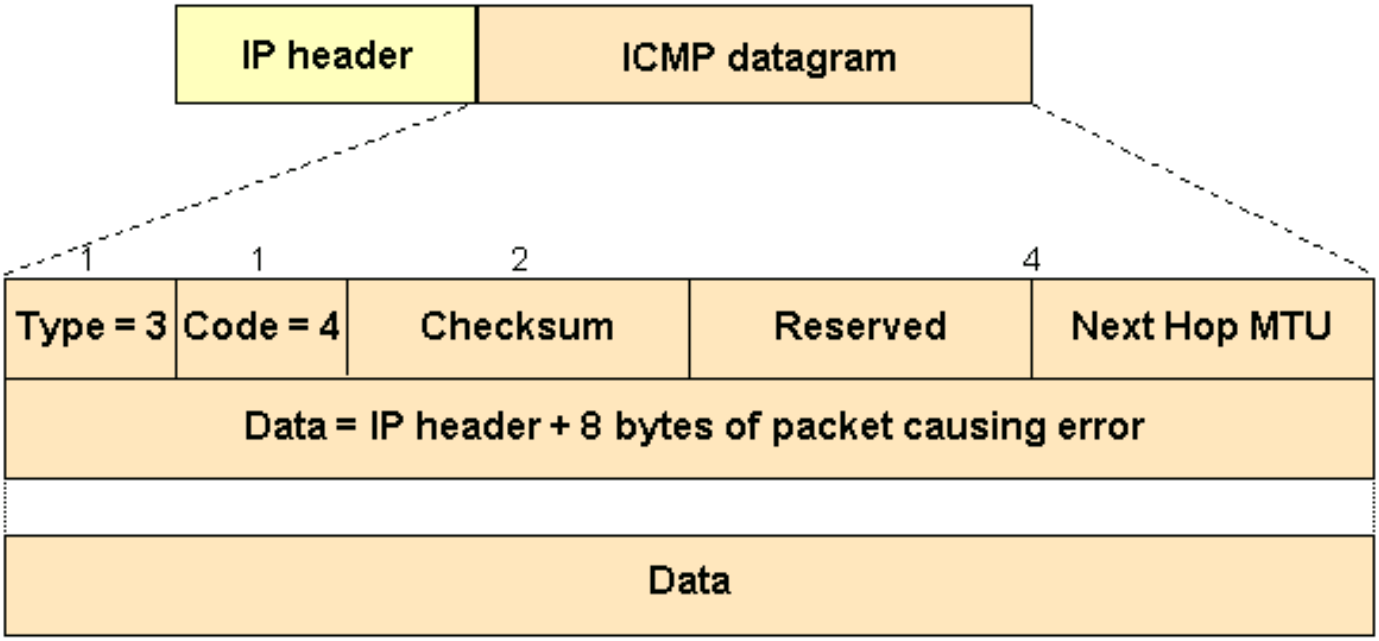
Cuando un host envía un paquete de datos de MSS completo con el bit DF configurado, PMTUD reduce el valor de MSS de envío para la conexión si recibe información que indique que el paquete requiere fragmentación. Generalmente, un host “recuerda” el valor de MTU para un destino, ya que crea una entrada de “host” (/32) en su tabla de routing con este valor de MTU.

Si un router intenta remitir un datagrama IPv4, con el conjunto del bit DF, sobre un link que tenga un MTU inferior que el tamaño del paquete, el router cae el paquete y vuelve un mensaje “Destino inalcanzable” del Internet Control Message Protocol (ICMP) a la fuente de este datagrama IPv4, con el código que indica la “fragmentación necesaria y el DF para fijar” (el tipo 3, el código 4). Cuando la estación de origen recibe el mensaje ICMP, bajará el envío MSS, y cuando el TCP retransmite el segmento, utilizará el tamaño más pequeño del segmento.

Este es un ejemplo de un mensaje ICMP “fragmentation needed and DF set" (fragmentación necesaria y DF configurado) que podría ver en un router después de activar el comando **debug ip icmp**:

```
ICMP: dst (10.10.10.10) frag. needed and DF set
unreachable sent to 10.1.1.1
```

En este diagrama, se muestra el formato del encabezado ICMP de un mensaje de “destino inalcanzable” que dice “fragmentación necesaria y DF configurado”.



Por el RFC 1191, un router que vuelve un mensaje ICMP que indica “fragmentación necesitó y el DF fijar” debe incluir el MTU de esa red del salto siguiente en los 16 bits de orden inferior del campo del encabezado adicional ICMP que se etiqueta “inusitado” en el RFC 792 de la especificación ICMP.

Las primeras implementaciones de RFC 1191 no suministraban la información de MTU de salto siguiente. Incluso cuando esta información se suministraba, algunos hosts la ignoraban. En este caso, RFC 1191 también contiene una tabla que enumera los valores sugeridos por los que se debería disminuir a la MTU durante PMTUD. Es utilizado por los host para llegar más rápidamente un valor razonable para el envío MSS y tal y como se muestra en de la imagen.

Plateau	MTU	Comments	Reference
	65535	Official maximum MTU	RFC 791
	65535	Hyperchannel	RFC 1044
65535			
32000		Just in case	
	17914	16Mb IBM Token Ring	ref. [6]
17914			
	8166	IEEE 802.4	RFC 1042
8166			
	4464	IEEE 802.5 (4Mb max)	RFC 1042
	4352	FDDI (Revised)	RFC 1188
4352 (1%)			
	2048	Wideband Network	RFC 907
	2002	IEEE 802.5 (4Mb recommended)	RFC 1042
2002 (2%)			
	1536	Exp. Ethernet Nets	RFC 895
	1500	Ethernet Networks	RFC 894
	1500	Point-to-Point (default)	RFC 1134
	1492	IEEE 802.3	RFC 1042
1492 (3%)			
	1006	SLIP	RFC 1055
	1006	ARPANET	BBN 1822
1006			
	576	X.25 Networks	RFC 877
	544	DEC IP Portal	ref. [10]
	512	NETBIOS	RFC 1088
	508	IEEE 802/Source-Rt Bridge	RFC 1042
	508	ARCNET	RFC 1051
508 (13%)			
	296	Point-to-Point (low delay)	RFC 1144
296			
68		Official minimum MTU	RFC 791

PMTUD se hace continuamente en todos los paquetes porque la trayectoria entre el remitente y el receptor puede cambiar dinámicamente. Cada vez que un remitente reciba mensajes de ICMP de "no se puede fragmentar", actualizará la información de ruteo (donde se almacena la PMTUD).

Dos cosas posibles pueden suceder durante PMTUD:

1. El paquete puede llegar finalmente al receptor sin haber sido fragmentado.

> **Note**: Para que un router proteja el CPU contra ataques de DOS, regula el número de mensajes de destino inalcanzable de ICMP que enviaría a dos por segundo. Por lo tanto, en este contexto, si tiene una situación de red en la que espera que el router responda con más de dos mensajes ICMP (tipo = 3, código = 4) por segundo (pueden ser diferentes hosts), es recomendable que deshabilite la limitación de mensajes de ICMP con el comando de interfaz **no ip icmp rate-limit unreachable [df]**.

2. El remitente puede conseguir los mensajes "Imposible realizar la fragmentación" ICMP de cualquier (o cada) salto a lo largo de la trayectoria al receptor.

PMTUD se realiza independientemente para ambas direcciones de un flujo de TCP. Pudo haber los casos donde PMTUD en una dirección de un flujo acciona una de las estaciones del extremo para bajar el envío MSS y la estación del otro extremo guarda la original para enviar MSS porque nunca envió un datagrama IPv4 bastante grande para accionar PMTUD.

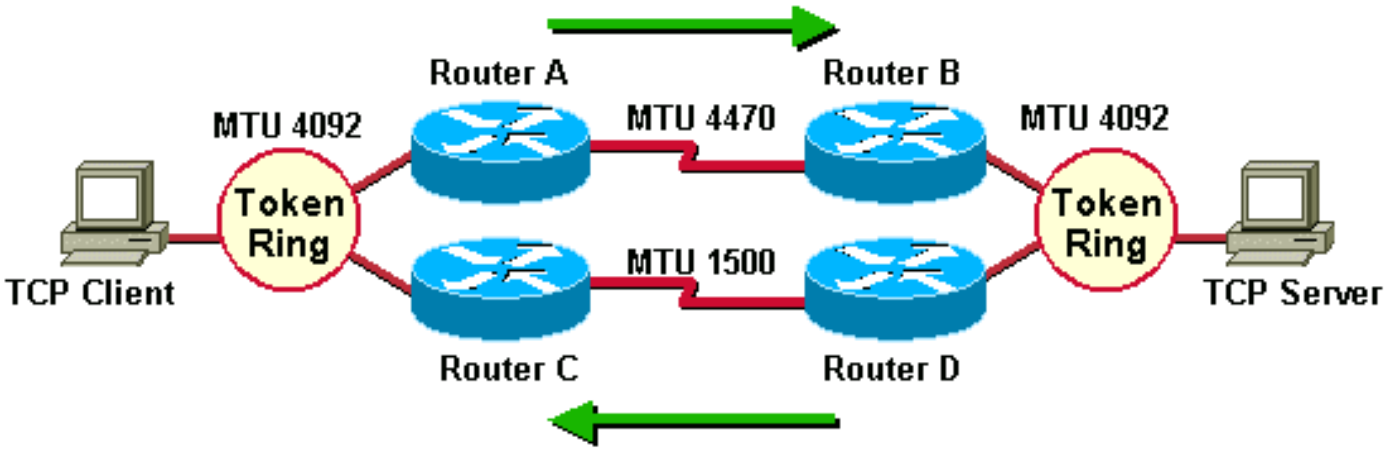
Un buen ejemplo de esto es la conexión HTTP representada a continuación en la situación 3. El cliente TCP envía paquetes pequeños y el servidor, paquetes grandes. En este caso, solo los paquetes grandes del servidor (superiores a 576 bytes) activarán PMTUD. Los paquetes del cliente son pequeños (menos de 576 bytes) y no accionarán PMTUD porque no requieren la fragmentación conseguir a través del link MTU 576.

Decorado 3

En la situación 4, se muestra un ejemplo de ruteo asimétrico donde una de las trayectorias tiene una MTU mínima más pequeña que la otra. El routing asimétrico se produce cuando se toman diferentes rutas para enviar y recibir datos entre dos terminales. En esta situación, la PMTUD accionará la disminución del valor de MSS de envío solamente en una dirección de un flujo de TCP. El tráfico del cliente TCP al servidor tiene lugar a través del router A y el router B, mientras que el tráfico de retorno proveniente del servidor y dirigido al cliente se transmite por el router D y el router C. Cuando el servidor TCP envía los paquetes al cliente, PMTUD accionará el servidor para bajar el envío MSS porque el router D debe hacer fragmentos de los 4092 paquetes de bytes antes de que pueda enviarlos al C del router.

El cliente, por el contrario, nunca recibirá un mensaje ICMP “Destination Unreachable” con el código que indica “fragmentación necesaria y DF configurado” porque el router A no tiene que fragmentar los paquetes cuando los envía al servidor por el router B.

Situación 4



Note: Utilizan al **comando ip tcp path-mtu-discovery** para activar la detección de trayecto MTU TCP para las conexiones TCP iniciadas por el Routers (BGP y Telnet por ejemplo).

Problemas con PMTUD

Hay tres cosas que pueden romper PMTUD, dos cuyo sea infrecuente y uno de los cuales es común.

- Un router puede descartar un paquete y no enviar un mensaje de ICMP. (Poco común)
- Un router puede generar y enviar un mensaje ICMP, pero un router o firewall bloquea dicho mensaje cuando se transmite entre el router y el emisor. (Común)
- Un router puede generar y enviar un mensaje de ICMP, pero el remitente ignora el mensaje. (Poco común)

El primeros y pasado de los tres puntos negros aquí son infrecuentes y son generalmente el resultado de un error, pero la viñeta del medio describe un problema común. La gente que ejecuta los filtros de paquete ICMP tiende a bloquear todos los tipos de mensaje de ICMP bastante que solamente bloqueando ciertos tipos de mensaje de ICMP. Un filtro de paquete puede bloquear todos los tipos de mensajes de ICMP, *excepto* los que indiquen "destino inalcanzable" o "tiempo excedido". El éxito o el error de PMTUD se articula sobre los mensajes del ICMP fuera de alcance que consiguen a través al remitente de un paquete TCP/IPv4. Los mensajes tiempo-excedidos ICMP son importantes para otros problemas IPv4. Aquí se muestra un ejemplo de ese filtro de paquetes implementado en un router.

```
access-list 101 permit icmp any any unreachable
access-list 101 permit icmp any any time-exceeded
access-list 101 deny icmp any any
access-list 101 permit ip any any
```

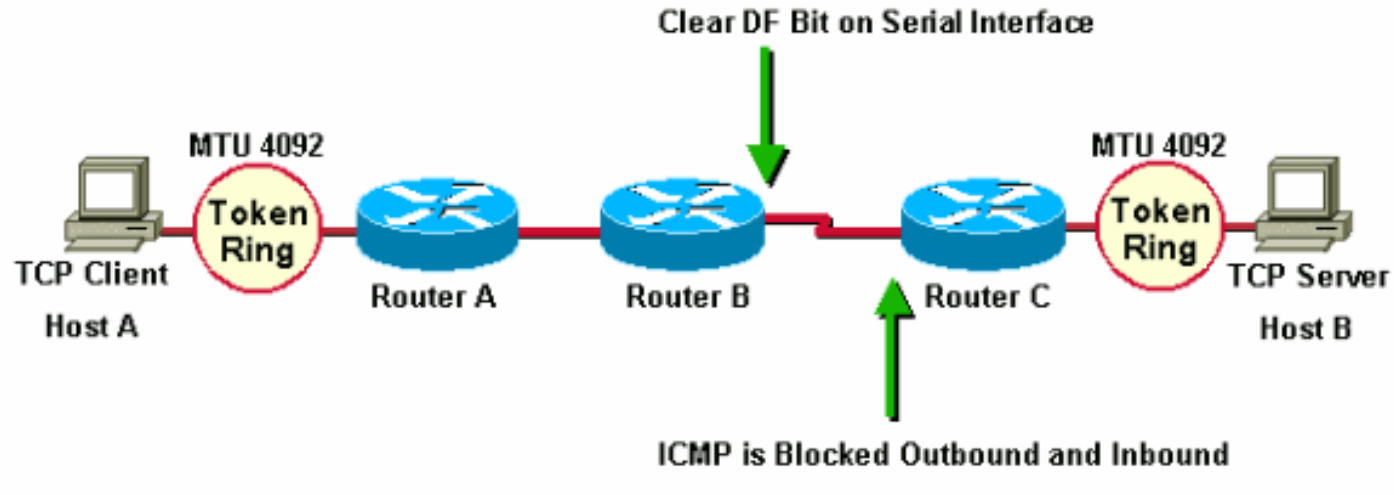
Hay otras técnicas que se pueden utilizar para ayudar a paliar el problema del ICMP que es bloqueado totalmente.

- Borre el bit DF en el router y permita la fragmentación de todas maneras (esto podría ser una mala idea. Consulte Problemas con la Fragmentación IP para obtener más información.
- Manipule el valor de la opción MSS de TCP con el comando de interfaz **ip tcp adjust-mss <500-1460>**.

En la siguiente situación, el router A y el router B están en el mismo dominio administrativo. No es posible acceder al router C y este bloquea el ICMP, por eso se interrumpe PMTUD. Una solución para esta situación es borrar el bit DF en ambas direcciones en el router B para permitir la fragmentación. Es posible hacerlo con el routing de políticas. La sintaxis para borrar el bit DF está disponible en el Cisco IOS® Software, versión 12.1(6) y posteriores.

```
interface serial0
...
ip policy route-map clear-df-bit
route-map clear-df-bit permit 10
match ip address 111
set ip df 0

access-list 111 permit tcp any any
```



Otra opción es cambiar el valor de opción MSS TCP en los paquetes SYN que atraviesan al router (disponible en Cisco IOS® 12.2(4)T y más adelante). Esto reduce el valor de la opción de MSS en el paquete SYN de TCP para que sea inferior al valor (1460) en el comando **ip tcp adjust-mss**. El resultado es que el emisor TCP enviará divide no más grande en segmentos que este valor. Paquete IPV4 el tamaño será 40 bytes más grande (1500) que el valor MSS (1460 bytes) para explicar el encabezado TCP (20 bytes) y la encabezado IPv4 (20 bytes).

Usted puede ajustar el MSS de los paquetes SYN TCP con el **comando ip tcp adjust-mss**. Esta sintaxis reducirá el valor de MSS en los segmentos de TCP a 1460. Este comando afecta el tráfico tanto entrante como saliente en la interfaz serial0.

```
int s0
ip tcp adjust-mss 1460
```

Los problemas de fragmentación IPv4 han llegado a ser más extensos puesto que los túneles IPv4 se han desplegado más extensamente. La razón por la que los túneles provocan más fragmentación es que el encapsulamiento del túnel agrega “sobrecarga” al tamaño de un paquete. Por ejemplo, la adición de la encapsulación genérica del router (GRE) agrega 24 bytes a un paquete, y después de que este aumento, el paquete pudiera necesitar ser hecho fragmentos porque es más grande que el MTU saliente. En una sección posterior de este documento, usted verá los ejemplos de las clases de problemas que puedan presentarse con los túneles y la fragmentación IPv4.

Topologías comunes de red que necesitan PMTUD

PMTUD se necesita en situaciones de red en las que los links intermedios tienen MTU más pequeñas que la MTU de los links extremos. Algunas razones comunes para la existencia de estos links más pequeños MTU son:

- Token Ring (o FDDI): hosts extremos conectados con una conexión de Ethernet entre ellos. Las MTU de Token Ring (o interfaz de datos distribuidos por fibra, FDDI) en los extremos son superiores a la MTU de Ethernet en el medio.
- El PPPoE (de uso frecuente con ADSL) necesita 8 bytes para su encabezado. Esto reduce el MTU eficaz de los Ethernetes a 1492 (1500 - 8).

Los protocolos de túneles como GRE, IPv4sec, y L2TP también necesitan el espacio para sus encabezados correspondientes y remolques. Esto también reduce el MTU eficaz de la interfaz saliente.

En las siguientes secciones, se estudia el impacto de PMTUD donde un protocolo de túnel se utiliza entre los dos hosts terminales. De los tres casos anteriores, este caso es el más complejo y abarca todos los problemas que se pueden ver en los demás casos.

Túnel

Un túnel es una interfaz lógica en un router de Cisco que proporciona una manera de encapsular los paquetes pasajeros dentro de un protocolo de transporte. Es una arquitectura diseñada para proporcionar los servicios para ejecutar un esquema de la encapsulación Point-to-Point. Los túneles tienen estos tres componentes principales:

- Protocolo pasajero (APPLETALK, Banyan VINES, CLNS, DECNet, IPv4, o IPX)
- Protocolo de portadora. Uno de estos protocolos de encapsulamiento:GRE, el protocolo de portadora multiprotocolo de Cisco. Consulte RFC 2784 y RFC 1701 para obtener más información.IPv4 en IPv4 hace un túnel - Vea el RFC 2003 para más información.
- Transport Protocol - El protocolo usado para llevar el protocolo encapsulado.

Los paquetes mostrados en esta sección ilustran los conceptos del Tunelización IPv4 donde está el Encapsulation Protocol GRE e IPv4 es el Transport Protocol. El protocolo pasajero es también IPv4. En este caso, IPv4 es el transporte y el protocolo pasajero.

Paquete Normal

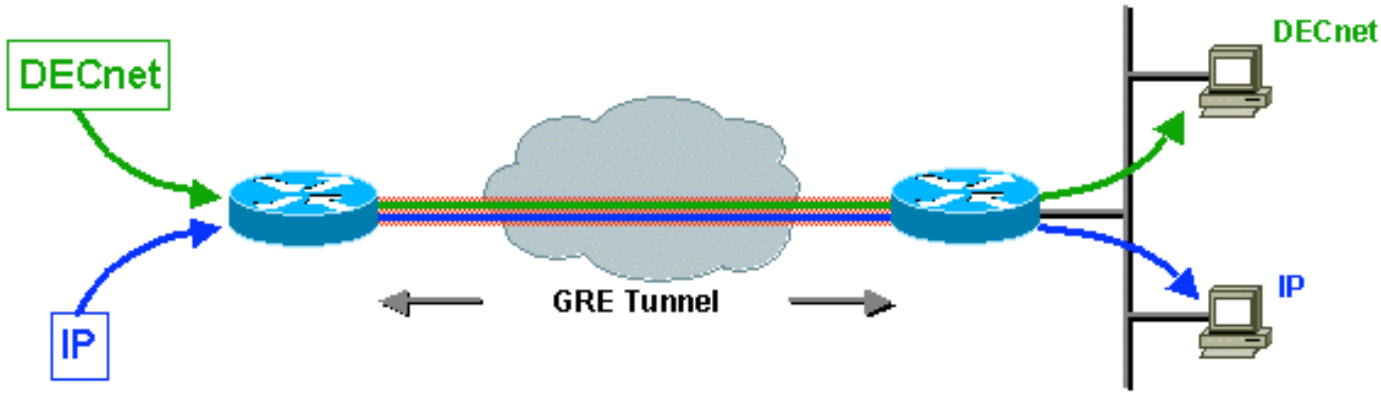
IPv4 TCP Telnet

Paquete del túnel

IPv4 GRE IPv4 TCP Telnet

- IPv4 es el Transport Protocol.
- GRE es el protocolo de encapsulación.
- IPv4 es el protocolo pasajero.

El próximo ejemplo muestra la encapsulación de IPv4 y del DECNet como protocolos pasajeros con GRE como el portador. Esto ilustra el hecho de que el protocolo de la portadora puede encapsular los protocolos pasajeros múltiples tal y como se muestra en de la imagen.



Un administrador de la red pudo considerar hacer un túnel en una situación donde hay dos redes discontiguous del non-IPv4 separadas por una estructura básica IPv4. Si las redes no contiguas ejecutan DECnet, es posible que el administrador no desee conectarlas configurando DECnet en la red troncal. El administrador no pudo querer permitir que el ruteo DECnet consuma el ancho de banda de la estructura básica porque éste podría interferir con el funcionamiento de la red IPv4.

Una alternativa viable es hacer un túnel el DECNet sobre la estructura básica IPv4. El Tunelización encapsula los paquetes del DECNet dentro de IPv4, y lo envía a través de la estructura básica al punto final del túnel donde se quita la encapsulación y los paquetes del DECNet se pueden encaminar a su destino vía el DECNet.

Encapsular el tráfico en otro protocolo ofrece estas ventajas:

- Los terminales usan direcciones privadas (RFC 1918) y la red troncal no admite el routing de estas direcciones.
- Permita redes privadas virtuales (VPN) a través de WAN o Internet.
- Una las redes discontinuas de varios protocolos a través de una backbone de un solo protocolo.
- Cifre el tráfico a través de la backbone o Internet.

Para el resto del documento, IPv4 se utiliza como el protocolo pasajero y el IPv4 como el Transport Protocol.

Consideraciones con respecto a las interfaces del túnel

Estas son las consideraciones que deben tenerse en cuenta para los túneles.

- La transferencia rápida de los túneles GRE fue introducida en la versión 11.1 del ® del Cisco IOS y la transferencia CEF fue introducida en la versión 12.0. El CEF switching para los túneles GRE multipunto se introdujo en la versión 12.2(8)T. La encapsulación y el decapsulation en los puntos finales del túnel eran operaciones lentas en las versiones anteriores del ® del Cisco IOS cuando solamente la transferencia de proceso fue utilizada.
- Hay problemas de topología y seguridad cuando se realiza la tunelización de paquetes. Los túneles pueden desviar el Listas de control de acceso (ACL) y los Firewall. Si usted usa un túnel a través de un firewall, básicamente saltea el firewall para cualquier protocolo pasajero para el que use el túnel. Por lo tanto, se recomienda para incluir funcionalidad del firewall en los puntos finales del túnel para aplicar cualquier directiva en los protocolos pasajeros.
- Los túneles podrían crear problemas con los protocolos de transporte que tengan temporizadores limitados (por ejemplo, DECnet) debido al aumento de la latencia.
- Los túneles que atraviesan entornos con enlaces de diferente velocidad, como los anillos de FDDI rápidos y a través de líneas telefónicas lentas de 9600 bps, podrían presentar problemas de reordenamiento de paquetes. Algunos protocolos pasajeros funcionan mal en redes de medios combinadas.
- Los túneles Point-to-Point pueden utilizar encima del ancho de banda en un link físico. Si ejecuta protocolos de routing en varios túneles de punto a punto, tenga en cuenta que cada interfaz de túnel tiene un ancho de banda, y que la interfaz física en la que se ejecuta el túnel también tiene un ancho de banda. Por ejemplo, desearía configurar el ancho de banda de túnel en 100 KB si hubiera 100 túneles en ejecución a través de un link de 10 MB. El ancho de banda del valor por defecto para un túnel es 9Kb.
- El encaminamiento de los protocolos pudo preferir un túnel sobre un link real porque el túnel pudo aparecer engañoso ser un link del uno-salto con la trayectoria más barata, aunque implique más saltos y sea realmente realmente más costoso que otra trayectoria. Esto se puede mitigar con una correcta configuración del protocolo de ruteo. Usted puede ser que quiera considerar funcionar con un diverso protocolo de la encaminamiento sobre la interfaz del túnel que el protocolo de la encaminamiento que se ejecutaba en la interfaz física.

- Los problemas de ruteo recurrente pueden evitarse al configurar rutas estáticas apropiadas al destino de túnel. Una ruta recurrente es cuando el mejor trayecto al destino del túnel está a través del túnel sí mismo. Debido a esta situación, se devuelve la interfaz de túnel ("rebote") una y otra vez. Usted verá este error cuando haya un problema de routing recursivo.
```
%TUN-RECURDOWN Interface Tunnel 0
temporarily disabled due to recursive routing
```

Router como Participante de PMTUD en la punto final del túnel

El router tiene dos diversas funciones de PMTUD a jugar cuando es el Punto final de un túnel.

- En el primer papel, el router es el promotor de un paquete del host. Para el procesamiento de PMTUD, el router debe verificar el bit DF y el tamaño de paquete del paquete de datos original, y realizar la acción apropiada, cuando sea necesario.
- El segundo papel entra en el juego después de que el router haya encapsulado la original paquete IPV4 dentro del paquete del túnel. En esta etapa, el router actúa más bién un host en cuanto a PMTUD y con respecto al túnel paquete IPV4.

Lets tiene una mirada en qué sucede cuando el router actúa en el primer papel, un router que adelante recibe los paquetes IPv4, en cuanto a PMTUD. Este papel entra en el juego antes de que el router encapsule el host paquete IPV4 dentro del paquete del túnel.

Si el router es el que reenvía un paquete de host, completará estas acciones:

- Controle si el bit DF está fijado
- Controle qué paquete del tamaño puede acomodar el túnel
- Fragmentar (si el paquete es demasiado grande y el bit DF no está configurado), encapsular los fragmentos y enviar. or
- Caiga el paquete (si el paquete es demasiado grande y bit se fija DF) y envíe un mensaje ICMP al remitente
- Encapsule (si el paquete no es demasiado grande) y envíe

De manera genérica, existe la opción de encapsular y fragmentar (enviar dos fragmentos de encapsulamiento) o de fragmentar y encapsular (enviar dos fragmentos encapsulados).

Algunos ejemplos que describen a los mecánicos paquete IPV4 de la encapsulación y fragmentación y dos decorados que muestran la interacción de PMTUD y los paquetes que las redes de muestra transversales se detallan en esta sección.

En el primer ejemplo, se observa lo que le sucede a un paquete cuando el router (en el origen del túnel) desempeña la función de router de reenvío. Recuerde que, para procesar PMTUD, el router debe comprobar el tamaño del paquete y bit DF del paquete de datos original, y tomar las medidas correspondientes. Este los ejemplos utilizan la encapsulación GRE para el túnel. Como puede verse, GRE realiza la fragmentación antes del encapsulamiento. En los ejemplos posteriores, se muestran situaciones donde se realiza la fragmentación después de la encapsulación.

En el ejemplo 1, el bit DF no se fija (DF = 0) y el MTU del túnel GRE IPv4 es 1476 (1500 - 24).

Ejemplo 1

1. El router de reenvío (en el origen de túnel) recibe un datagrama 1500-byte con el DF mordido claramente (DF = 0) del host de envío. Este datagrama se compone de una encabezado IP 20-

byte más un byte 1480 carga útil de TCP.

IPv4 TCP de 1480 bytes + datos

2. Porque el paquete es demasiado grande para el MTU IPv4 después de que se agregue la tara de GRE (24 bytes), el router de reenvío rompe el datagrama en dos fragmentos de 1476 (20 encabezado de los bytes IPv4 + 1456 payload de los bytes IPv4) y 44 bytes (20 bytes de la encabezado IPv4 + 24 bytes del payload IPv4) así que después de que se agregue la encapsulación GRE, el paquete no será más grande que el MTU de interfaz de la física saliente.

IP0 1456 bytes TCP + datos
IP1 24 datos de bytes

3. El router de reenvío agrega la encapsulación GRE, que incluye un encabezado GRE 4-byte más una encabezado 20-byte IPv4, a cada fragmento del datagrama original IPv4. Estos dos datagramas IPv4 ahora tienen una longitud de 1500 y 68 bytes y estos datagramas se ven como datagramas individuales IPv4, no como fragmentos.

IPv4 GRE IP0 1456 bytes TCP + datos
IPv4 GRE IP1 24 datos de bytes

4. El router de destino del túnel quita la encapsulación GRE de cada fragmento del datagrama original, que sale de dos fragmentos IPv4 de las longitudes 1476 y 24 bytes. Estos fragmentos del datagrama IPv4 son remitidos por separado por este router al host de recepción.

IP0 1456 bytes TCP + datos
IP1 24 datos de bytes

5. El host de recepción vuelve a montar estos dos fragmentos en el datagrama original.

IPv4 TCP de 1480 bytes + datos

El decorado 5 representa el papel del router de reenvío en el contexto de una topología de red.

En este ejemplo, el router actúa en el mismo papel del router de reenvío, pero este vez el bit DF se fija (DF = 1).

Ejemplo 2

1. El router de reenvío en el origen de túnel recibe un datagrama 1500-byte con DF = 1 del host de envío.

IPv4 TCP de 1480 bytes + datos

2. Puesto que se fija el bit DF, y el tamaño del datagrama (1500 bytes) es mayor que el MTU del túnel GRE IPv4 (1476), el router caerá el datagrama y enviará un mensaje "se necesita fragmentación de ICMP pero se configuró el bit DF" a la fuente del datagrama. El mensaje de ICMP alertará al remitente que la MTU es 1476.

IPv4 MTU ICMP 1476

3. El host de envío recibe el mensaje ICMP, y cuando vuelve a enviar las informaciones originales él utiliza un datagrama 1476-byte IPv4.

IPv4 1456 bytes TCP + datos

4. Esta longitud del datagrama IPv4 (1476 bytes) es igual ahora en el valor al MTU del túnel GRE IPv4 así que al router agrega la encapsulación GRE al datagrama IPv4.

IPv4 GRE IPv4 1456 bytes TCP + datos

5. El router de recepción (en el destino del túnel) quita la encapsulación GRE del datagrama IPv4 y la envía al host de recepción.

IPv4 1456 bytes TCP + datos

Ahora, usted puede mirar qué sucede cuando el router actúa en el segundo papel como host de envío en cuanto a PMTUD y con respecto al túnel paquete IPV4. Recuerde que este papel entra en el juego después de que el router haya encapsulado la original paquete IPV4 dentro del paquete del túnel.

> **Note**: Por abandono, un router no hace PMTUD en los paquetes de túnel GRE que genera. **El comando tunnel path-mtu-discovery** puede ser utilizado para girar PMTUD para los paquetes del túnel del GRE-IPv4.

El ejemplo 3 muestra qué sucede cuando el host envía los datagramas IPv4 que son bastante pequeños caber dentro del MTU IPv4 en la interfaz de túnel GRE. El DF mordido en este caso puede estar fijado o claro (1 o 0). La interfaz de túnel GRE no tiene el **comando tunnel path-mtu-discovery** configuró así que el router no hará PMTUD en el paquete del GRE-IPv4.

Ejemplo 3

1. El router de reenvío en el origen de túnel recibe un datagrama 1476-byte del host de envío.

IPv4 1456 bytes TCP + datos

2. Este router encapsula el datagrama 1476-byte IPv4 dentro de GRE para conseguir un datagrama 1500-byte GRE IPv4. El DF mordido en la encabezado GRE IPv4 estará claro (DF = 0). Este router entonces adelante este paquete al destino del túnel.

IPv4 GRE IPv4 1456 bytes TCP + datos

3. Suponga que hay un router entre el origen y el destino de túnel con una MTU de link de 1400. Este router hará fragmentos del paquete del túnel puesto que el bit DF está claro (DF = 0). Recuerde que este ejemplo hace fragmentos del IPv4 exterior, así que el GRE, el IPv4 interno, y los encabezados TCP aparecerá solamente en el primer fragmento.

IP0 GRE IP TCP de 1352 bytes + datos
IP1 Datos de 104 bytes

4. El router de destino del túnel debe volver a montar el paquete de túnel GRE.

IP GRE IP 1456 bytes TCP + datos

5. Después de que se vuelva a montar el paquete de túnel GRE, el router quita la encabezado GRE IPv4 y envía el datagrama original IPv4 en su manera.

IPv4 1456 bytes TCP + datos

El próximo ejemplo muestra qué sucede cuando el router actúa en el papel de un host de envío en cuanto a PMTUD y con respecto al túnel paquete IPV4. Esta vez el bit DF se fija (se ha configurado el DF = 1) en la encabezado original IPv4 y el **comando tunnel path-mtu-discovery** de modo que el bit DF sea copiado de la encabezado interna IPv4 (GRE + IPv4) a la encabezado externa.

Ejemplo 4

1. El router de reenvío en el origen de túnel recibe un datagrama de 1476 bytes con DF = 1 del host remitente.

IPv4 1456 bytes TCP + datos

2. Este router encapsula el datagrama 1476-byte IPv4 dentro de GRE para conseguir un datagrama 1500-byte GRE IPv4. Esta encabezado GRE IPv4 tendrá el bit DF fijado (DF = 1) puesto que el datagrama original IPv4 tenía el bit DF fijado. Este router entonces adelante este paquete al destino del túnel.

IPv4 GRE IPv4 1456 bytes TCP

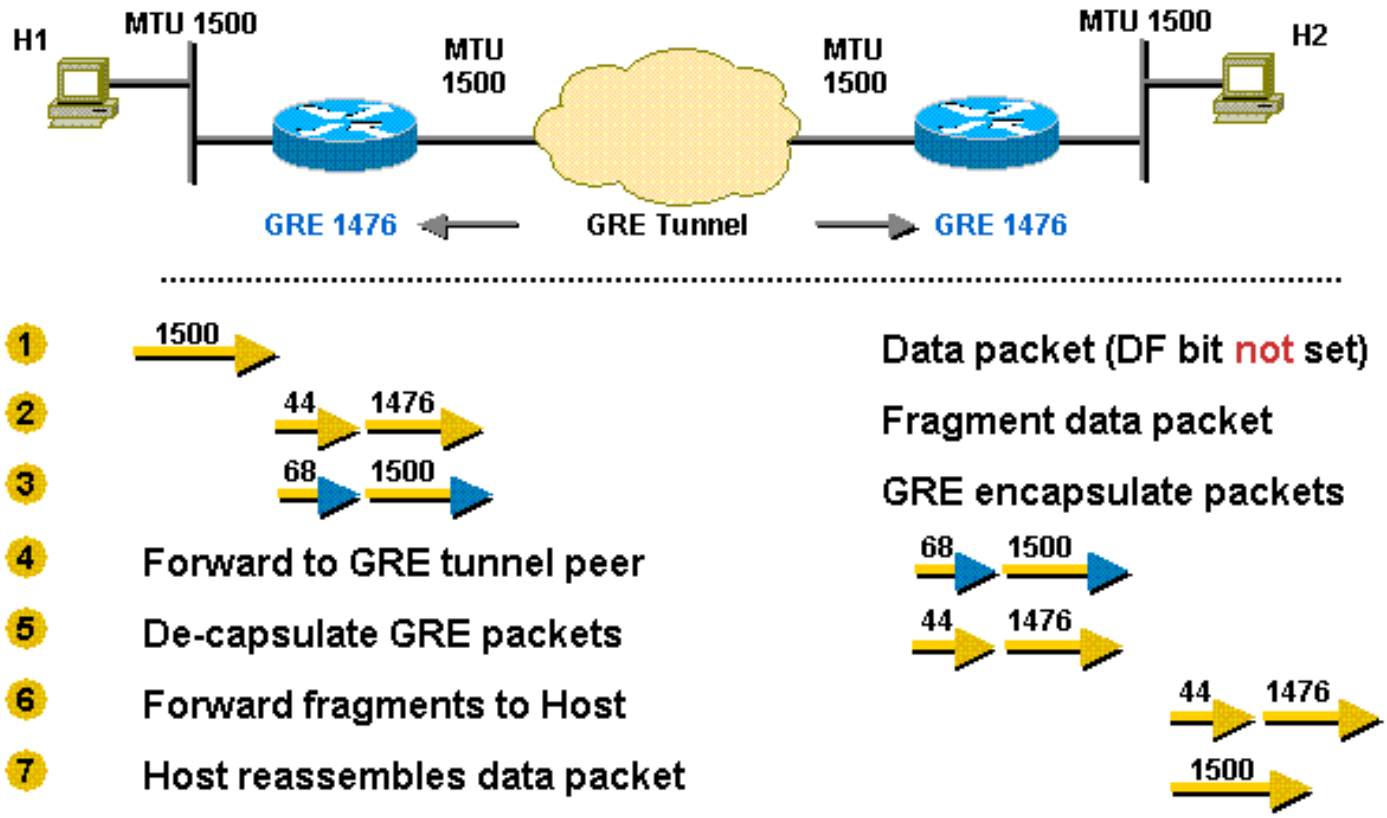
3. Una vez más, suponga que hay un router entre el origen y el destino de túnel con una MTU de link de 1400. Este router no hará fragmentos del paquete del túnel puesto que se fija el bit DF (DF=1). Este router debe caer el paquete y enviar un mensaje de error ICMP al router del origen de túnel, puesto que ése es el direccionamiento de la fuente IPv4 en el paquete.

IPv4 MTU ICMP 1400

4. El router de reenvío en el origen de túnel recibe este mensaje de error “ICMP” y bajará el MTU del túnel GRE IPv4 a 1376 (1400 - 24). La próxima vez que el host de envío retransmite los datos en un 1476-byte paquete IPV4, este paquete puede ser demasiado grande y este router enviará un mensaje de error “ICMP” al remitente con un valor MTU de 1376. Cuando el host de envío retransmite los datos, lo enviará en un 1376-byte paquete IPV4 y este paquete lo hará a través del túnel GRE al host de recepción.

Situación 5

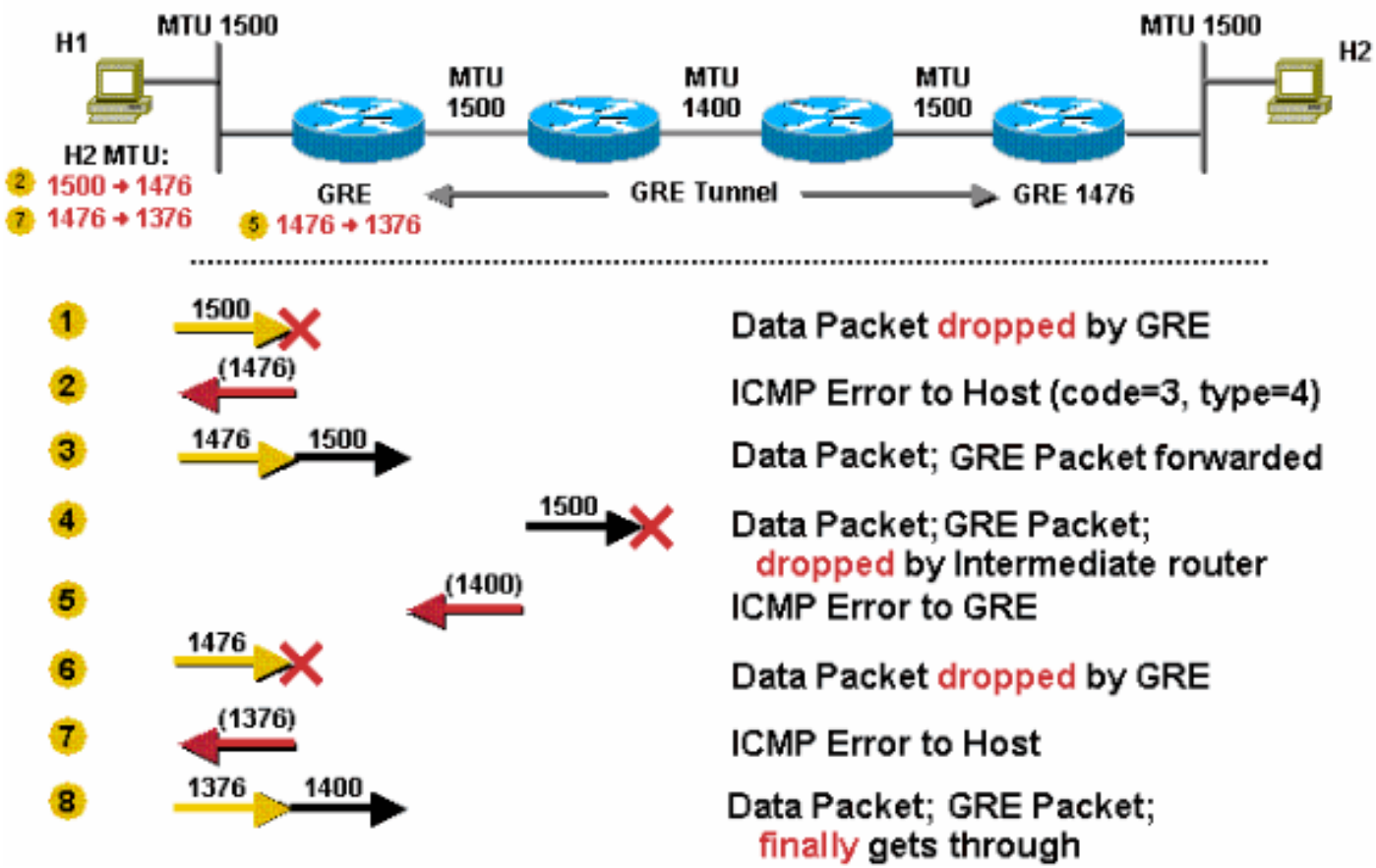
Este decorado ilustra la fragmentación GRE. Recuerde que usted hace fragmentos antes de la encapsulación para GRE, después hacen PMTUD para el paquete de datos, y el bit DF no se copia cuando paquete IPV4 es encapsulado por GRE. En esta situación, el bit DF no está configurado. El MTU de la interfaz de túnel GRE IPv4 es, por abandono, 24 bytes menos que el MTU de la interfaz física IPv4, así que el MTU del interfaz IPv4 GRE es 1476 tal y como se muestra en de la imagen.



1. El remitente envía un paquete 1500-byte (20 encabezado del byte IPv4 + 1480 bytes de carga útil de TCP).
2. Puesto que el MTU del túnel GRE es 1476, el paquete 1500-byte está roto en dos fragmentos IPv4 de 1476 y 44 bytes, cada uno antes de los 24 byes adicionales del encabezado GRE.
3. Los 24 bytes del encabezado GRE se agregan a cada fragmento IPv4. Ahora, los fragmentos son de 1500 (1476 + 24) y 68 (44 + 24) bytes cada uno.
4. Los paquetes GRE + IPv4 que contienen los dos fragmentos IPv4 se remiten al router del par del túnel GRE.
5. El router de peer de túnel GRE quita los encabezados GRE de los dos paquetes.
6. Este router adelante los dos paquetes al host del destino.
7. El host del destino vuelve a montar los fragmentos IPv4 nuevamente dentro del datagrama original IPv4.

Situación 6

Esta situación es similar a la situación 5, pero, esta vez, el bit DF está configurado. En el decorado 6, configuran al router para hacer PMTUD en los paquetes del túnel GRE + IPv4 con el **comando tunnel path-mtu-discovery**, y el bit DF se copia de la encabezado original IPv4 a la encabezado GRE IPv4. Si el router recibe un error ICMP para GRE + paquete IPV4, reduce el MTU IPv4 en la interfaz de túnel GRE. Una vez más recuerde que el MTU del túnel GRE IPv4 está fijado a 24 bytes menos que el MTU de la interfaz física por abandono, así que el MTU GRE IPv4 aquí es 1476. También note que hay un link MTU 1400 en la trayectoria del túnel GRE tal y como se muestra en de la imagen.



1. El router recibe un paquete 1500-byte (20 encabezado del byte IPv4 + 1480 carga útil de TCP), y cae el paquete. El router cae el paquete porque es más grande que el MTU IPv4 (1476) en la interfaz de túnel GRE.
2. El router lo envía un error ICMP al remitente que dice que el MTU del siguiente-salto es 1476. El host registrará esta información, generalmente pues una ruta del host para el destino en su tabla de encaminamiento.
3. El host de envío utiliza un tamaño de paquetes 1476-byte cuando vuelve a enviar los datos. El router GRE agrega 24 bytes de encapsulación GRE y envía hacia fuera un paquete 1500-byte.
4. El paquete 1500-byte no puede atravesar el link 1400-byte, así que es caído por el router intermedio.
5. El router intermedio envía un ICMP (tipo = 3, código = 4) al router GRE con una MTU de siguiente salto de 1400. El router GRE reduce esto a 1376 (1400 - 24) y fija un valor interno MTU IPv4 en el interfaz GRE. Este cambio solo se puede ver usando el **comando debug tunnel**; no se puede ver en el resultado del **comando show ip interface tunnel<#>**.
6. La próxima vez el host vuelve a enviar el paquete 1476-byte, el router GRE caerá el paquete, puesto que es más grande que el MTU actual IPv4 (1376) en la interfaz de túnel GRE.
7. El router GRE enviará otro ICMP (tipo = 3, código = 4) al emisor con una MTU de siguiente salto de 1376, y el host actualizará su información actual con el nuevo valor.
8. El host vuelve a enviar otra vez los datos, pero ahora en un paquete más pequeño 1376-byte, GRE agregará 24 bytes de encapsulación y los remitirá encendido. Esta vez, el paquete llegará al par del túnel GRE, donde se lo desencapsulará y enviará al host de destino.

Note: Si no configuraran al **comando tunnel path-mtu-discovery** en el router de reenvío en este decorado, y el bit DF fuera fijado en los paquetes remitidos a través del túnel GRE, el

host 1 todavía tendría éxito en el envío de los paquetes TCP/IPv4 para recibir 2, pero conseguirían hechos fragmentos en el centro en el link MTU 1400. Además, el peer de túnel GRE debería reensamblarlos para poder desencapsularlos y reenviarlos.

Modo túnel puro de IPsec

El protocolo de la Seguridad IPv4 (IPv4sec) es un método de estándares que proporciona a la aislamiento, a la integridad, y a la autenticidad a la información transferida a través de las redes IPv4. IPv4sec proporciona a la encripción de capa de red IPv4. IPv4sec alarga paquete IPV4 agregando por lo menos una encabezado IPv4 (modo túnel). La encabezado agregada varía de largo al dependiente en el modo de la configuración IPv4sec pero ella no excede ~58 bytes (Encapsulating Security Payload (ESP) y autenticación ESP (ESPauth)) por paquete.

IPv4sec tiene dos modos, el modo túnel y modo de transporte.

1. El modo túnel es el modo de valor por defecto. Con el modo túnel, la original entera paquete IPV4 se protege (cifrado, autenticado, o ambos) y es encapsulada por las encabezados y los remolques IPv4sec. Entonces una nueva encabezado IPv4 prepended al paquete, que especifica las puntos finales IPv4sec (pares) como la fuente y el destino. El modo túnel puede ser utilizado con cualquier tráfico del unicast IPv4 y debe ser utilizado si IPv4sec protege el tráfico contra los host detrás de los pares IPv4sec. Por ejemplo, utilizan al modo túnel con el Redes privadas virtuales (VPN) donde los host en una red protegida envían los paquetes a los host en una diversa red protegida vía un par de pares IPv4sec. Con los VPN, el IPv4sec “túnel” protege el tráfico IPv4 entre los host cifrando este tráfico entre el Routers del par IPv4sec.
2. Con el modo de transporte (configurado con el submandato, el **transporte del modo**, en la definición de transformación), solamente el payload de la original paquete IPV4 se protege (cifrado, autenticado, o ambos). El payload es encapsulado por las encabezados y los remolques IPv4sec. Las encabezados originales IPv4 permanecen intacto, salvo que el campo del protocolo IPv4 se cambia para ser ESP (50), y el valor del protocolo original se guarda en el remolque IPv4sec que se restablecerá cuando se desencripta el paquete. Se utiliza el modo de transporte solamente cuando mira el tráfico IPv4 que se protegerá está entre el IPv4sec ellos mismos, la fuente y los direccionamientos del destino IPv4 en el paquete es lo mismo que las direcciones de peer IPv4sec. El modo de transporte IPv4sec se utiliza normalmente solamente cuando otro Tunneling Protocol (como GRE) se utiliza a primero encapsula el paquete de datos IPv4, después IPv4sec se utiliza para proteger los paquetes de túnel GRE.

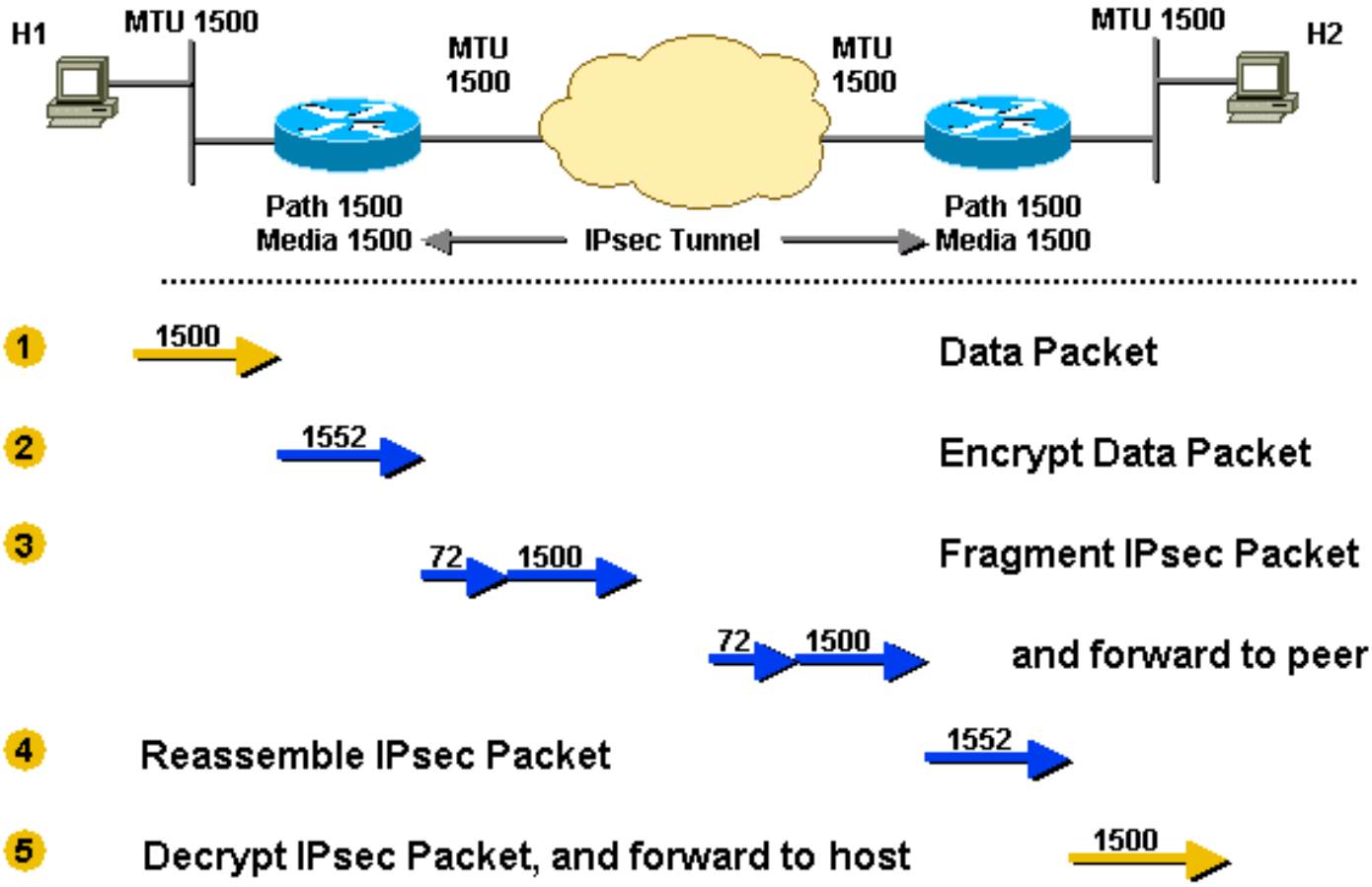
IPv4sec hace siempre PMTUD para los paquetes de datos y para sus propios paquetes. Hay comandos configuration IPv4sec de modificar el procesamiento de PMTUD para el IPv4sec paquete IPV4, IPv4sec puede borrar, fijar, o copiar el DF mordido de la encabezado del paquete de datos IPv4 a la encabezado IPv4sec IPv4. Esto se llama el función “Funcionalidad de anulación de bits DF”.

Note: Usted quiere realmente evitar la fragmentación después de la encapsulación cuando usted hace el cifrado de la dotación física con IPv4sec. El cifrado de la dotación física puede darle la producción de cerca de 50 Mbs que depende de la dotación física, pero si se hace fragmentos el paquete IPv4sec usted suelta el 50 a 90 por ciento de la producción. Esta pérdida es porque los paquetes hechos fragmentos IPv4sec proceso-se cambian para el

nuevo ensamble y después se dan al motor de encripción de la dotación física para el desciframiento. Esta pérdida de rendimiento puede bajar el rendimiento del cifrado del hardware al nivel de rendimiento del cifrado del software (2 - 10 MB).

Situación 7

Este decorado representa la fragmentación IPv4sec en la acción. En esta situación, la MTU junto con la trayectoria entera es 1500. En esta situación, el bit DF no está configurado.

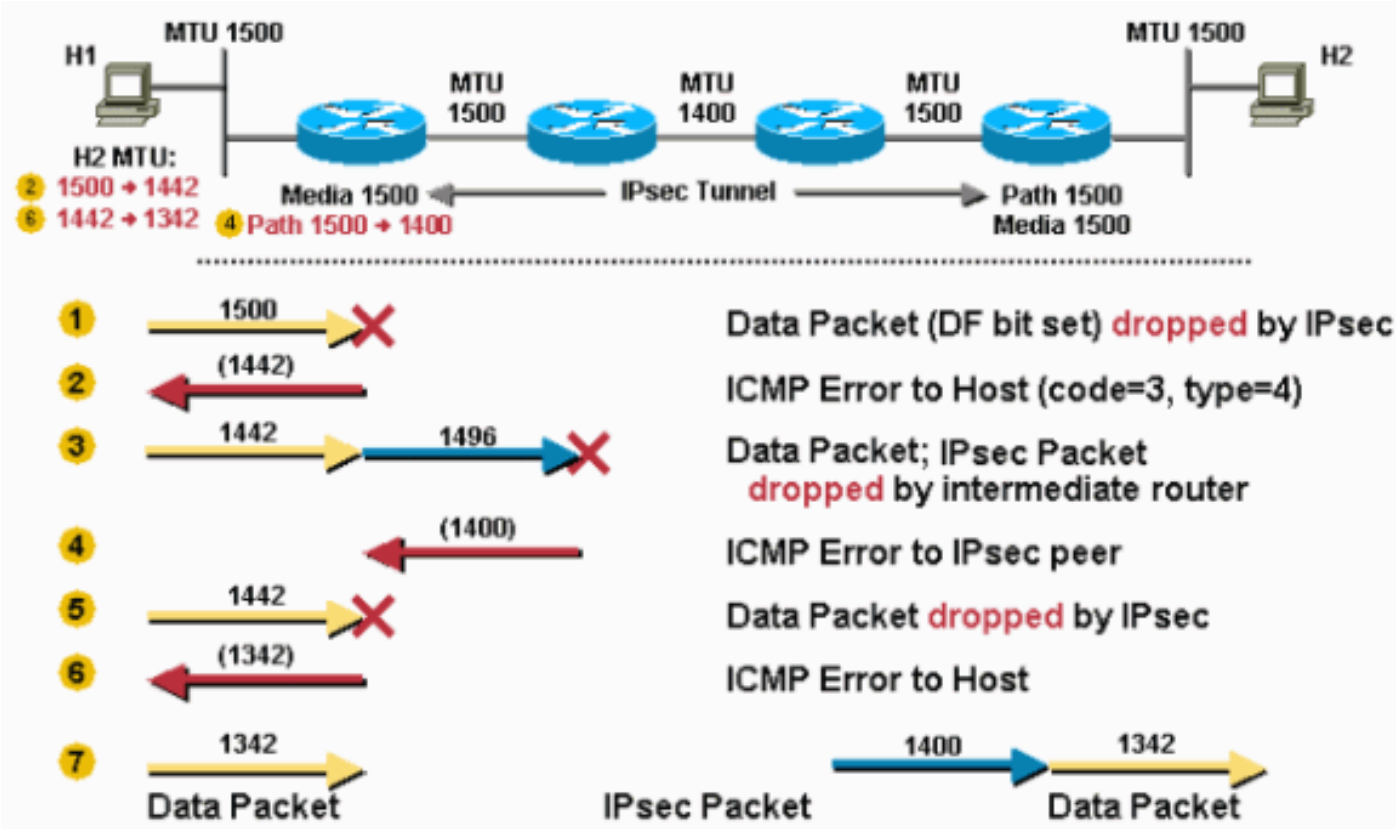


1. El router recibe un paquete 1500-byte (encabezado 20-byte IPv4 + 1480 bytes carga útil de TCP) destinado para el host 2.
2. El paquete 1500-byte es cifrado por IPv4sec y se agregan 52 bytes de tara (encabezado IPv4sec, remolque, y la encabezado adicional IPv4). Ahora IPv4sec necesita enviar un paquete 1552-byte. Puesto que el MTU saliente es 1500, este paquete tendrá que ser hecho fragmentos.
3. Dos fragmentos se crean fuera del paquete IPv4sec. Durante la fragmentación, una encabezado adicional 20-byte IPv4 se agrega para el segundo fragmento, dando por resultado un fragmento 1500-byte y un fragmento 72-byte IPv4.
4. El router del peer de túnel IPv4sec recibe los fragmentos, pela la encabezado adicional IPv4 y se une los fragmentos IPv4 nuevamente dentro del paquete original IPv4sec. Entonces IPv4sec desencripta este paquete.
5. El router entonces adelante el paquete de datos original 1500-byte para recibir 2.

Situación 8

Este decorado es similar al decorado 6 salvo que en este caso el bit DF se fija en el paquete de datos original y hay un link en la trayectoria entre los peeres de túnel IPv4sec que tiene un MTU inferior que los otros links. Este decorado demuestra cómo el router del par IPv4sec realiza ambas funciones de PMTUD, según lo descrito en el router como Participante de PMTUD en la sección del Punto final de un túnel.

Usted verá en este decorado cómo el IPv4sec PMTU cambia a un valor inferior como resultado de la necesidad de la fragmentación. Recuerde que el bit DF está copiado de la encabezado interna IPv4 a la encabezado externa IPv4 cuando IPv4sec cifra un paquete. Los valores MTU y PMTU de los media se salvan en la asociación de seguridad IPv4sec (SA). El MTU de los media se basa en el MTU del interfaz saliente del router y el PMTU se basa en el MTU del mínimo visto en la trayectoria entre los pares IPv4sec. Recuerde que IPv4sec encapsula/cifra el paquete antes de que intente hacer fragmentos de él tal y como se muestra en de la imagen.



1. El router recibe un paquete 1500-byte y lo cae porque los gastos indirectos IPv4sec, cuando están agregados, harán el paquete más grande que el PMTU (1500).
2. El router envía un mensaje ICMP para recibir 1 que le dice que el MTU del siguiente-salto es 1442 (1500 - 58 = 1442). Este 58 bytes son el máximo IPv4sec de arriba al usar IPv4sec ESP y ESPauth. Los gastos indirectos reales IPv4sec pueden ser tanto como 7 bytes menos que este valor. El Host 1 registra esta información, generalmente como una ruta de host para el destino (Host 2), en su tabla de ruteo.
3. 3. El Host 1 disminuye su PMTU para el Host 2 a 1442, así que el Host 1 enviará paquetes más pequeños (de 1442 bytes) cuando retransmita los datos al Host 2. El router recibe el paquete 1442-byte e IPv4sec agrega 52 bytes de los gastos indirectos del cifrado así que el paquete resultante IPv4sec es 1496 bytes. Porque este paquete tiene el bit DF configurado en su encabezado, es descartado por el router del medio con el link de MTU de 1400 bytes.
4. El router intermedio que cayó el paquete lo envía un mensaje ICMP al remitente del paquete IPv4sec (el primer router) que dice que el MTU del siguiente-salto es 1400 bytes. Este valor se registra en IPv4sec SA PMTU.

5. La próxima vez el host 1 retransmite el paquete 1442-byte (no recibió un acuse de recibo para él), el IPv4sec caerá el paquete. El router caerá otra vez el paquete porque los gastos indirectos IPv4sec, cuando están agregados al paquete, lo harán más grande que el PMTU (1400).
6. El router envía un mensaje de ICMP al Host 1 comunicándole que la MTU de salto siguiente ahora es 1342. (1400 - 58 = 1342). El Host 1 registrará nuevamente esta información.
7. Cuando el Host 1 retransmita otra vez los datos, utilizará el paquete con el tamaño más pequeño (1342). Este paquete no requerirá la fragmentación y la hará a través del túnel IPv4sec para recibir 2.

GRE e IPv4sec junto

Más interacciones complejas para la fragmentación y PMTUD ocurren cuando IPv4sec se utiliza para cifrar los túneles GRE. IPv4sec y GRE se combinan de este modo porque IPv4sec no utiliza los paquetes de multidifusión IPv4, así que significa que usted no puede funcionar con un protocolo de la encaminamiento dinámica sobre la red VPN IPv4sec. Los túneles GRE utilizan el Multicast, así que un túnel GRE se puede utilizar a primero encapsula el paquete de multidifusión del protocolo de la encaminamiento dinámica en un paquete de unidifusión GRE IPv4 que se pueda entonces cifrar por IPv4sec. Al hacer esto, IPv4sec se despliega a menudo en el modo de transporte encima de GRE porque los pares y las puntos finales del túnel GRE (el Routers) IPv4sec son lo mismo, y transporte-MODE salvará 20 bytes de los gastos indirectos IPv4sec.

Un caso interesante es cuando paquete IPV4 ha estado partido en dos fragmentos y encapsulado por GRE. En este caso IPv4sec verá dos independientes los paquetes GRE + IPv4. A menudo, en una configuración predeterminada, uno de estos paquetes será muy grande y necesitará ser fragmentado después de su cifrado. El par IPv4sec tendrá que volver a montar este paquete antes del desciframiento. Esta “fragmentación doble” (una vez antes de GRE y otra vez después de que IPv4sec) en el router de envío aumenta el tiempo de espera y baja la producción. Además, el reensamblado se convierte en process-switched; por lo tanto, habrá un impacto en el CPU en el router receptor siempre que suceda esto.

Esta situación puede ser evitada fijando el “MTU IP” en la interfaz de túnel GRE bajo bastante para tener en cuenta los gastos indirectos de GRE y de IPv4sec (por abandono la interfaz de túnel GRE “MTU IP” se fija al MTU saliente de la interfaz real - los bytes de la tara de GRE).

En esta tabla, se muestran los valores de MTU sugeridos para cada combinación de túnel y modo, suponiendo que la interfaz física saliente tiene una MTU de 1500.

Combinación del túnel	MTU Específica Necesaria	MTU Recomendada
GRE + IPv4sec (modo de transporte)	1440 bytes	1400 bytes
GRE + IPv4sec (modo túnel)	1420 bytes	1400 bytes

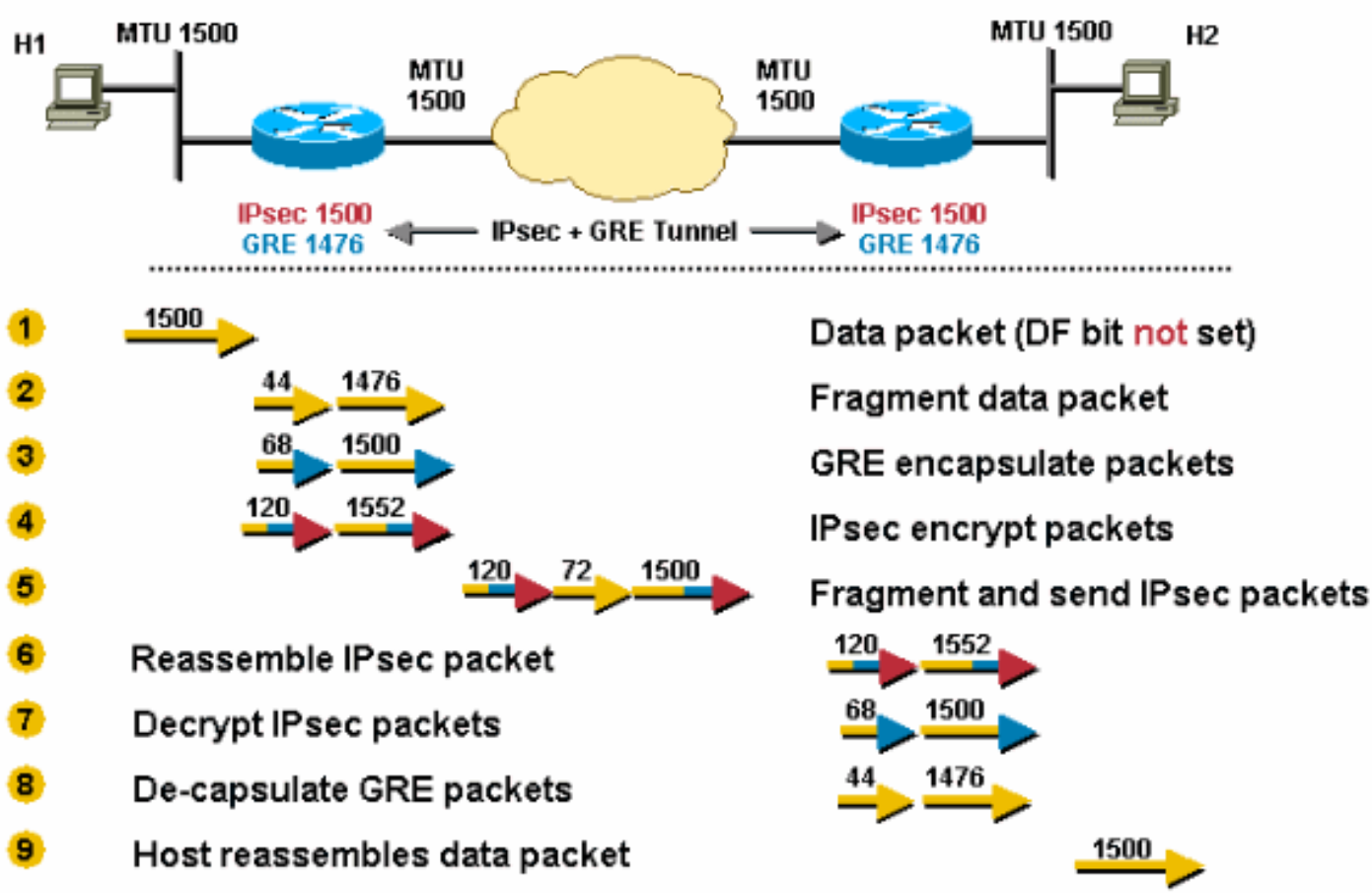
> **Note**: El valor MTU de 1400 se recomienda porque cubre el más común las combinaciones del modo GRE + IPv4sec. Además, no hay desventaja notable en permitir una sobrecarga adicional de 20 o 40 bytes. Es más fácil recordar y configurar un valor y este valor cubre casi todas las situaciones.

Decorado 9

IPv4sec se despliega encima de GRE. El MTU de la física saliente es 1500, el IPv4sec PMTU es

1500, y el MTU GRE IPv4 es 1476 (1500 - 24 = 1476). Debido a esto, los paquetes TCP/IPv4 serán hechos fragmentos dos veces, una vez antes de GRE y una vez después de IPv4sec. El paquete será hecho fragmentos antes de la encapsulación GRE y uno de estos Paquetes GRE será hecho fragmentos otra vez después del cifrado IPv4sec.

Configurar “MTU el 1440" (modo IP de transporte IPv4sec) o “MTU el 1420" (modo túnel IP IPv4sec) en el túnel GRE quitaría la posibilidad de la fragmentación doble en este decorado.



1. El router recibe un datagrama de 1500 bytes.
2. Antes de la encapsulación, GRE hace fragmentos del paquete 1500-byte en dos 1476 (1500 - 24 = 1476) y 44 (24 datos + la encabezado 20 IPv4) bytes de los pedazos.
3. GRE encapsula los fragmentos IPv4, que agrega 24 bytes a cada paquete. Esto da lugar a dos los paquetes GRE + IPv4sec de 1500 (1476 + 24 = 1500) y 68 (44 + 24) bytes cada uno.
4. IPv4sec cifra los dos paquetes, agregando 52 byes (modo túnel IPv4sec) de tara de encapsulación a cada uno, para dar un 1552-byte y un paquete del 120-byte.
5. El paquete 1552-byte IPv4sec es hecho fragmentos por el router porque es más grande que el MTU saliente (1500). El paquete 1552-byte está partido en los pedazos, un paquete 1500-byte y un paquete 72-byte (52 bytes “payload” más una encabezado adicional 20-byte IPv4 para el segundo fragmento). Los tres paquetes 1500-byte, 72-byte, y paquetes del 120-byte se remiten al par IPv4sec + GRE.
6. El router de recepción vuelve a montar los dos fragmentos IPv4sec (1500 bytes y 72 bytes) para conseguir el 1552-byte original IPv4sec + Paquete GRE. Nada necesita ser hecha al 120-byte IPv4sec + Paquete GRE.
7. IPv4sec desencripta 1552-byte y el 120-byte IPv4sec + los Paquetes GRE para conseguir los Paquetes GRE 1500-byte y 68-byte.
8. Decapsulates GRE los Paquetes GRE 1500-byte y 68-byte para conseguir paquete IPV4 los

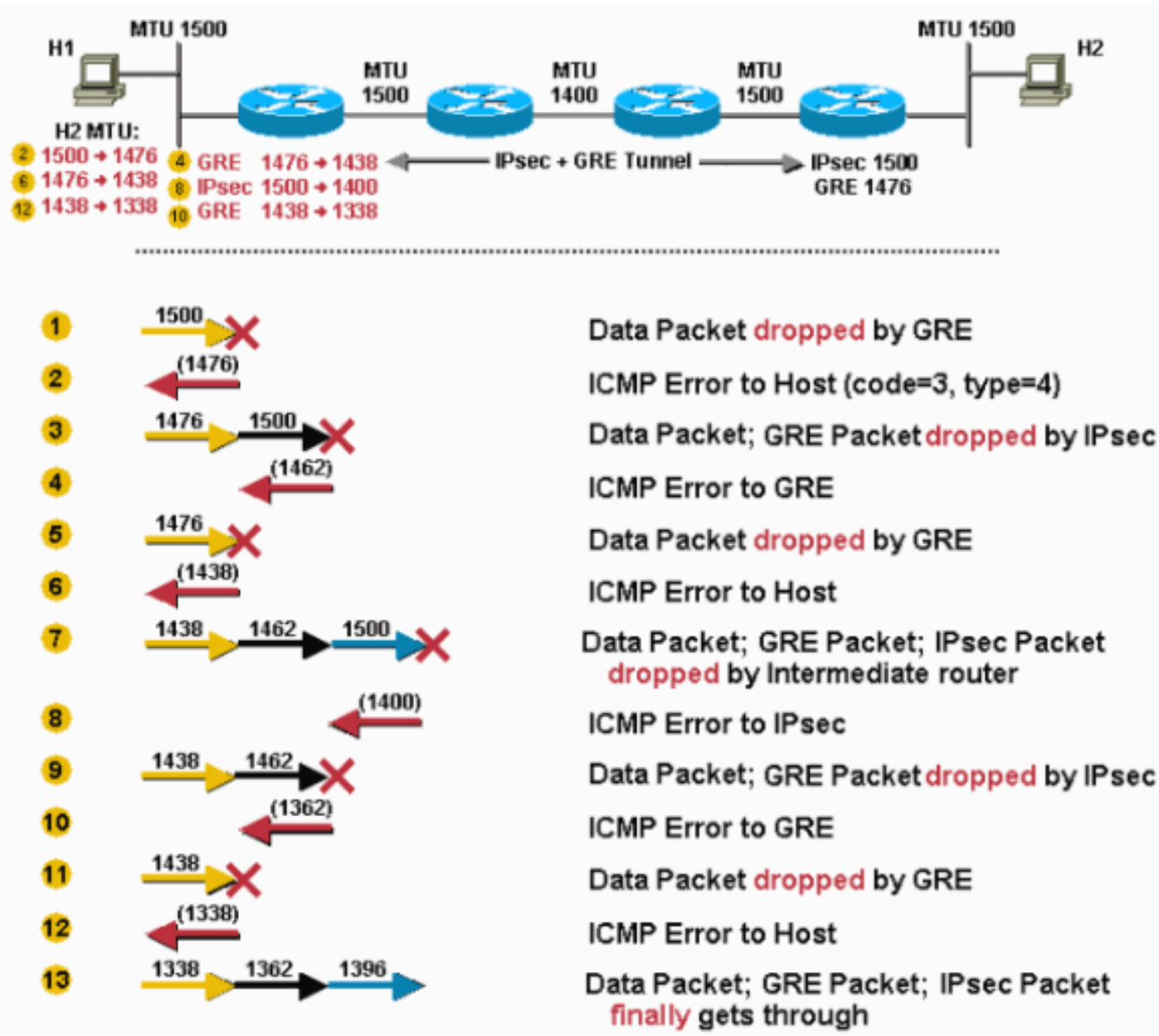
fragmentos 1476-byte y 44-byte. Éstos paquete IPV4 hacen fragmentos se remiten al host del destino.

9. El host 2 vuelve a montar estos fragmentos IPv4 para conseguir el datagrama original 1500-byte IPv4.

El decorado 10 es similar al decorado 8 a menos que haya un link del MTU inferior en la trayectoria del túnel. Éste es un peor de los casos para el primer paquete enviado del host 1 para recibir 2. Después del paso pasado en este decorado, el host 1 fija el PMTU correcto para el host 2 y todo está bien para las conexiones TCP entre el host 1 y los flujos del host 2. TCP entre el host 1 y otros host (accesibles vía IPv4sec + túnel GRE) tendrán que solamente pasar con los tres pasos pasados del decorado 10.

En este decorado, configuran al **comando tunnel path-mtu-discovery** en el túnel GRE y el bit DF se fija en los paquetes TCP/IPv4 que originan del host 1.

Situación 10



- El router recibe un paquete de 1500 bytes. GRE descarta este paquete porque no puede fragmentar ni reenviar el paquete, ya que el bit DF está configurado y el tamaño del paquete

excede la "MTU IP" de interfaz saliente una vez que se agrega la sobrecarga de GRE (24 bytes).
- El router envía un mensaje ICMP al host 1 para avisarle que la MTU del siguiente salto es de 1476 (1500 - 24 = 1476).
- El Host 1 cambia su PMTU para el Host 2 a 1476 y envía el tamaño más pequeño cuando retransmite el paquete. GRE lo encapsula y da el paquete 1500-byte a IPv4sec. IPv4sec cae el paquete porque GRE ha copiado el DF mordido (fije) de la encabezado interna IPv4, y con el IPv4sec de arriba (máximo 38 bytes), el paquete es demasiado grande remitir hacia fuera la interfaz física.
- IPv4sec envía un mensaje ICMP a GRE que indique que el MTU del siguiente-salto es 1462 bytes (puesto que un máximo 38 bytes será agregado para el cifrado e IPv4 por encima). GRE registra el valor 1438 (1462 - 24) como la "MTU IP" en la interfaz de túnel.
- **Note**: Este cambio en el valor se almacena internamente y no se puede ver en el resultado del **comando show ip interface tunnel<#>**. Usted verá solamente este cambio si usted da vuelta al uso el **comando debug tunnel**.
- La próxima vez que el Host 1 retransmita el paquete de 1476 bytes, GRE lo descartará.
- El router envía un mensaje ICMP al host 1 que indica que la MTU del siguiente salto es de 1438.
- El Host 1 disminuye la PMTU para el Host 2 y retransmite un paquete de 1438 bytes. Esta vez, GRE valida el paquete, lo encapsula, y lo da apagado a IPv4sec para el cifrado. El paquete IPv4sec se remite al router intermedio y se cae porque tiene un MTU de la interfaz de salida de 1400.
- El router intermedio envía un mensaje ICMP a IPv4sec que le diga que el MTU del siguiente-salto es 1400. Este valor es registrado por IPv4sec en el valor PMTU IPv4sec asociados SA.
- Cuando el host 1 retransmite el paquete 1438-byte, GRE lo encapsula y lo da a IPv4sec. IPv4sec cae el paquete porque ha cambiado su propio PMTU a 1400.
- IPv4sec envía un error ICMP a GRE que indique que el MTU del siguiente-salto es 1362, y GRE registra el valor 1338 internamente.
- Cuando el Host 1 retransmite el paquete original (porque no recibió reconocimiento), GRE lo descarta.
- El router envía un mensaje ICMP al host 1 que indica que la MTU del siguiente salto es de 1338 (1362 - 24 bytes). El Host 1 disminuye su PMTU para el Host 2 a 1338.
- El Host 1 retransmite un paquete de 1338 bytes y esta vez puede pasarlo finalmente a través del Host 2.

Otras recomendaciones

Configurar el **comando tunnel path-mtu-discovery** en una interfaz del túnel puede ayudar a la interacción GRE e IPv4sec cuando se configuran en el mismo router. Recuerde que sin el **comando tunnel path-mtu-discovery** configurado, el bit DF sería borrado siempre en la encabezado GRE IPv4. Esto permite el GRE paquete IPV4 sea hecho fragmentos aunque la encabezado de los datos encapsulados IPv4 tenía el bit DF fijado, que no permitiría normalmente que el paquete fuera hecho fragmentos.

Esto ocurrirá si el comando **tunnel path-mtu-discovery** está configurado en la interfaz de túnel GRE.

1. GRE copiará el DF mordido de la encabezado de los datos IPv4 a la encabezado GRE IPv4.
2. Si el bit DF se fija en la encabezado GRE IPv4 y el paquete es “demasiado grande” después del cifrado IPv4sec para el MTU IPv4 en la interfaz saliente física, después IPv4sec caerá el paquete y notificará el túnel GRE para reducir su tamaño MTU IPv4.
3. IPv4sec hace PMTUD para sus propios paquetes y si el IPv4sec PMTU cambia (si se reduce), después IPv4sec no notifica inmediatamente GRE, pero cuando viene otro paquete “demasiado grande” completo, después el proceso en el paso 2 ocurre.
4. El MTU IPv4 GRE es más pequeño ahora, así que caerá cualquier paquete de los datos IPv4 con el bit DF fijado que sea demasiado grande ahora y enviará un mensaje ICMP al host de envío.

El comando tunnel path-mtu-discovery ayuda al interfaz GRE para fijar su MTU IPv4 dinámicamente, bastante que estáticamente con el **comando ip mtu**. En realidad, se recomienda utilizar ambos comandos. Utilizan al **comando ip mtu** de proporcionar al sitio para los gastos indirectos GRE e IPv4sec en relación con el MTU de la interfaz saliente IPv4 de la física local. **El comando tunnel path-mtu-discovery** permite que el MTU del túnel GRE IPv4 sea reducido más a fondo si hay un link más bajo MTU IPv4 en la trayectoria entre los pares IPv4sec.

Aquí están algunas de las cosas que usted puede hacer si usted tiene problemas con PMTUD en una red donde hay los túneles GRE + IPv4sec configurados.

Esta lista comienza con la solución preferida.

1. Solucione el problema que impide el funcionamiento de PMTUD, que suele tener su origen en un router o firewall que bloquea ICMP.
2. Utilice el **comando ip tcp adjust-mss** en las interfaces del túnel de modo que el router reduzca el valor TCP MSS en paquete TCP Syn. Esto ayudará a los host del dos extremos (el emisor TCP y el receptor) a utilizar los paquetes bastante pequeños de modo que PMTUD no sea necesario.
3. Utilice la encaminamiento de la directiva en la interfaz de ingreso del router y configure una correspondencia de ruta para borrar el bit DF en la encabezado de los datos IPv4 antes de que consiga a la interfaz de túnel GRE. Esto permitirá que los datos paquete IPV4 sean hechos fragmentos antes de la encapsulación GRE.
4. Aumente la "MTU IP" en la interfaz de túnel GRE para que sea igual a la MTU de interfaz saliente. Esto permitirá que los datos paquete IPV4 sean GRE encapsulado sin hacer fragmentos de él primero. El Paquete GRE entonces será IPv4sec cifrado y después hecho fragmentos para salir la interfaz de salida física. En este caso, usted no configuraría el **comando tunnel path-mtu-discovery** en la interfaz de túnel GRE. Esto puede reducir dramáticamente la producción porque paquete IPV4 el nuevo ensamble en el par IPv4sec se hace en el modo de la proceso-transferencia.

Información Relacionada

- Página de Soporte de IP Routing
- Página de soporte de IPSec (protocolo de Seguridad IP)
- Calculadora de sobrecarga de IPSec (calcule el tamaño de paquetes con protocolos de encapsulamiento de IPSec)
- Detección de MTU de trayecto del RFC 1191
- Opciones de la detección de MTU IP del RFC 1063

- Protocolo de Internet del RFC 791
- Protocolo Protocolo de control de transmisión (TCP) del RFC 793
- RFC 879 el tamaño y los temas relacionados del segmento máxima TCP
- Generic Routing Encapsulation (GRE) del RFC 1701
- RFC 1241 un esquema para un protocolo de encapsulación de Internet
- Encapsulación IP del RFC 2003 dentro del IP
- Soporte Técnico y Documentación - Cisco Systems