Fragmentação de IP da resolução, edições MTU, MSS, e PMTUD com GRE e IPSEC

Índice

Introdução

O documento descreve como a fragmentação de IP e o Path Maximum Transmission Unit Discovery (PMTUD) trabalham e igualmente discutem algumas encenações que envolvem o comportamento do PMTUD quando combinadas com as combinações diferentes de túneis IP. O uso difundido atual dos túneis IP no Internet trouxe os problemas que envolvem a fragmentação de IP e o PMTUD ao pelotão da frente.

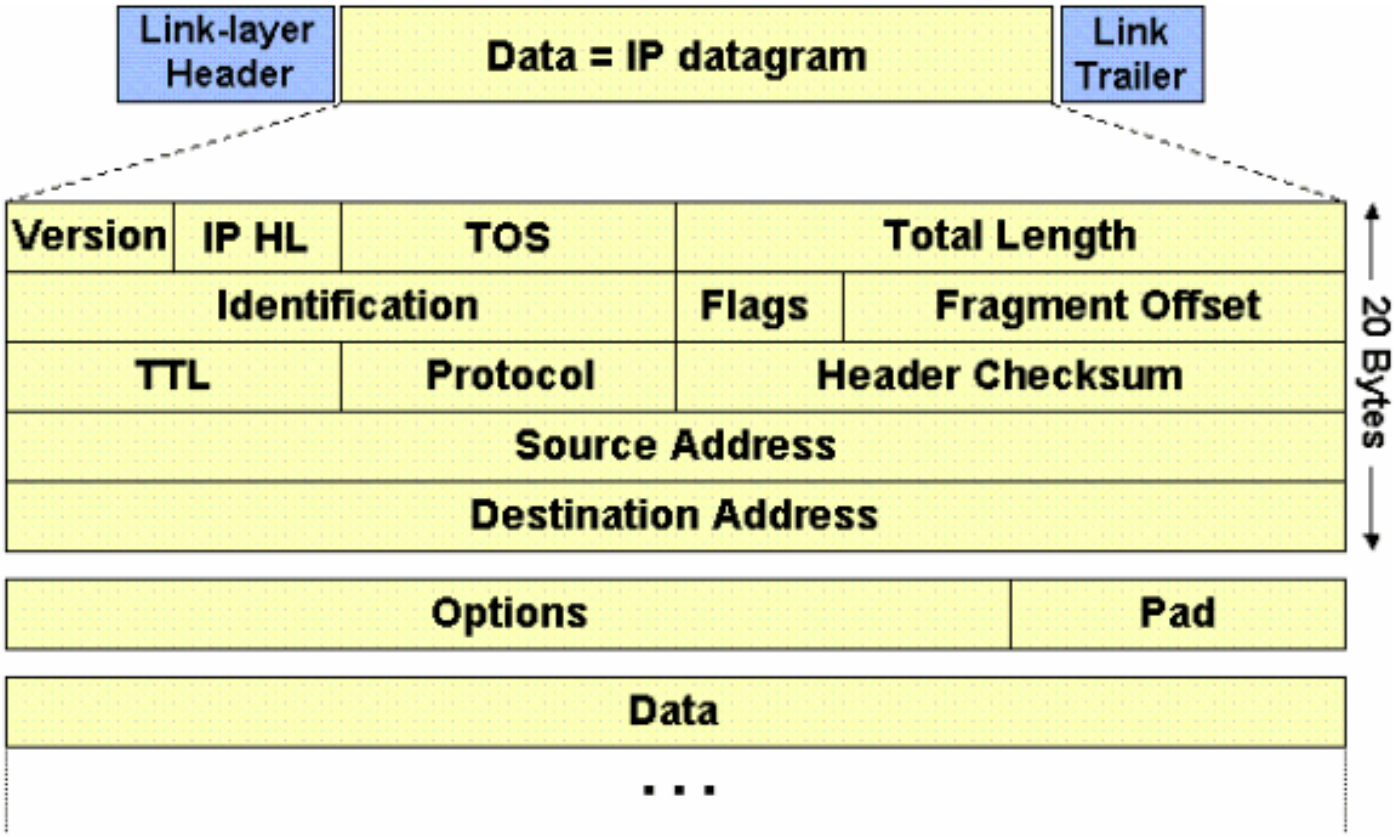
Fragmentação e remontagem IP

O protocolo IP foi projetado para o uso em uma ampla variedade de enlaces de transmissão. Embora o comprimento máximo de um IP datagram seja 65535, a maioria de enlaces de transmissão reforçam um limite menor do comprimento máximo do pacote, chamado um MTU. O valor da MTU depende do tipo do link de transmissão. O projeto do IP acomoda diferenças de

MTU desde que permite que o Roteadores fragmente datagramas IP como necessário. A estação de recepção é responsável para a remontagem dos fragmentos de novo no IP datagram sem redução original.

A fragmentação de IP envolve quebrar uma datagrama em um número de partes que podem ser remontadas mais tarde. A fonte de IP, o destino, a identificação, o comprimento total, os campos de compensação de fragmento, junto com os flags mais fragmentos e não fragmente no cabeçalho do IP, são utilizados para fragmentação e remontagem de IP. Para obter mais informações sobre dos mecânicos da fragmentação e reinstalação de IP, veja o RFC 791 .

Esta imagem descreve a disposição de um cabeçalho IP.



A identificação é 16 bit e é um valor atribuído pelo remetente de um IP datagram para ajudar na remontagem dos fragmentos de uma datagrama.

O deslocamento do fragmento é de 13 bits e indica a posição à qual o fragmento pertence no datagrama de IP original. Esse valor é um múltiplo de oito bytes.

No campo das bandeiras do cabeçalho IP, há três bit para flags de controle. É importante notar que o “Don't Fragment” (DF) mordeu jogos um papel central no PMTUD porque determina mesmo se um pacote está permitido ser fragmentado.

0 mordido são reservados, e ajustados sempre a 0. morderam 1 são o bit DF (0 = “podem fragmentar,” 1 = “não fazem fragmento”). 2 mordidos são o bit MF (0 = “último fragmento,” 1 = “mais fragmentos”).

Valor	0 mordido reservado	Mordido 1 DF	2 mordidos MF
0	0	maio	Último
1	0	Não faça	Mais

O gráfico seguinte mostra um exemplo de fragmentação. Se você adiciona acima todos os comprimentos dos fragmentos IP, o valor excede o comprimento da datagrama de IP original por 60. O motivo do comprimento geral ter aumentado de 60 é devido à criação de três cabeçalhos de IP adicionais, um para cada fragmento após o primeiro fragmento.

O primeiro fragmento tem um offset de 0, o comprimento deste fragmento é 1500; isto inclui 20 bytes para o cabeçalho de IP original levemente alterado.

O segundo fragmento foi um deslocamento de 185 (185 x 8 = 1480), o que significa que a porção de dados desse fragmento começa com 1480 bytes no datagrama de IP original. O comprimento deste fragmento é 1500; isto inclui o cabeçalho IP adicional criado para este fragmento.

O terceiro fragmento tem um deslocamento de 370 (370 x 8 = 2960), o que significa que a parte de dados desse fragmento começa com 2960 bytes no datagrama IP original. O comprimento deste fragmento é 1500; isto inclui o cabeçalho IP adicional criado para este fragmento.

O quarto fragmento tem uma compensação de 555 (555 x 8 = 4440), que significa que a parte de dados deste fragmento começa com 4440 bytes no datagrama de IP original. O comprimento deste fragmento é 700 bytes; isto inclui o cabeçalho IP adicional criado para este fragmento.

Ésomente quando o último fragmento é recebido que o tamanho da datagrama de IP original pode ser determinado.

O deslocamento de fragmento no último fragmento (555) dá um offset dos dados de 4440 bytes na datagrama de IP original. Se você adiciona então os bytes de dados do último fragmento (680 = 700 - 20), aquele dá-lhe 5120 bytes, que é a porção de dados da datagrama de IP original. Em seguida, adicionando 20 bytes para um cabeçalho de IP obtém-se o tamanho do datagrama de IP original (4.440 + 680 + 20 = 5.140).

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 com fragmentação de IP

Há diversas edições que fazem o undesirable da fragmentação de IP. Há um pequeno aumento

na sobrecarga de CPU e de memória para fragmentar um datagrama IP. Isso continua verdadeiro para o remetente e para um roteador no caminho entre o remetente e o receptor. Criar fragmentos envolve simplesmente criar encabeçamentos e copi do fragmento da datagrama original nos fragmentos. Isso pode ser feito com eficientemente porque todas as informações necessárias para criar os fragmentos estão imediatamente disponíveis.

A fragmentação causa mais despesas gerais para o receptor ao remontar os fragmentos porque o receptor deve atribuir a memória para os fragmentos de chegada e coalescer eles de novo em uma datagrama depois que todos os fragmentos são recebidos. A remontagem em um host não é considerada um problema porque o host tem o momento e os recursos de memória de devotar a esta tarefa.

Mas, a remontagem é muito incapaz em um roteador cujo o trabalho principal seja enviar o mais rapidamente possível pacotes. Um roteador não é projetado aferrar-se aos pacotes para nenhum intervalo de tempo. Igualmente um roteador que faça a remontagem escolhe o buffer o maior disponível (18K) com que trabalhar porque não tem nenhuma maneira de conhecer o tamanho do pacote IP original até que o último fragmento estiver recebido.

Uma outra questão de fragmentação envolve como os fragmentos deixados cair são segurados. Se um fragmento de um IP datagram é deixado cair, a seguir a datagrama de IP original inteira deve ser enviada novamente, e será fragmentada igualmente. Você verá um exemplo disso com o NFS (Network File System). O NFS, à revelia, tem um tamanho de bloco de leitura e de gravação de 8192, assim que uma datagrama NFS IP/UDP será aproximadamente 8500 bytes (que inclui o NFS, o UDP, e os cabeçalhos IP). Uma estação de envio conectada a um Ethernet (MTU 1500) terá que fragmentar a datagrama de byte 8500 em seis partes; cinco 1500 fragmentos do byte e um fragmento de 1100 bytes. Se alguns dos seis fragmentos são deixados cair devido a um link congestionado, a datagrama original completa terá que ser retransmitida, assim que significa que seis mais fragmentos terão que ser criados. Se este link deixa cair um em seis pacotes, a seguir as probabilidades são baixas que todos os dados NFS podem ser transferidos sobre este link, desde que pelo menos um fragmento IP seria deixado cair de cada IP datagram do byte original NFS 8500.

Os Firewall que filtram ou manipulam os pacotes baseados na camada 4 (L4) com a informação da camada 7 (L7) no pacote puderam ter o problema que processa fragmentos IP corretamente. Se os fragmentos IP são foras de serviço, um Firewall pôde obstruir os fragmentos não iniciais porque não levam a informação que combinaria o filtro de pacote. Isto significaria que a datagrama de IP original não poderia ser remontada pelo host de recepção. Se o Firewall é configurado para permitir que os fragmentos não iniciais com informação insuficiente combinem corretamente o filtro, a seguir um ataque do fragmento não inicial com o Firewall poderia ocorrer. Também, os pacotes diretos de alguns dispositivos de rede (tais como os motores de switch de conteúdo) baseados no L4 com a informação L7, e se um pacote mede fragmentos múltiplos, a seguir o dispositivo puderam ter o problema que reforça suas políticas.

Evite a fragmentação de IP: O que o MSS TCP faz e como ele funciona

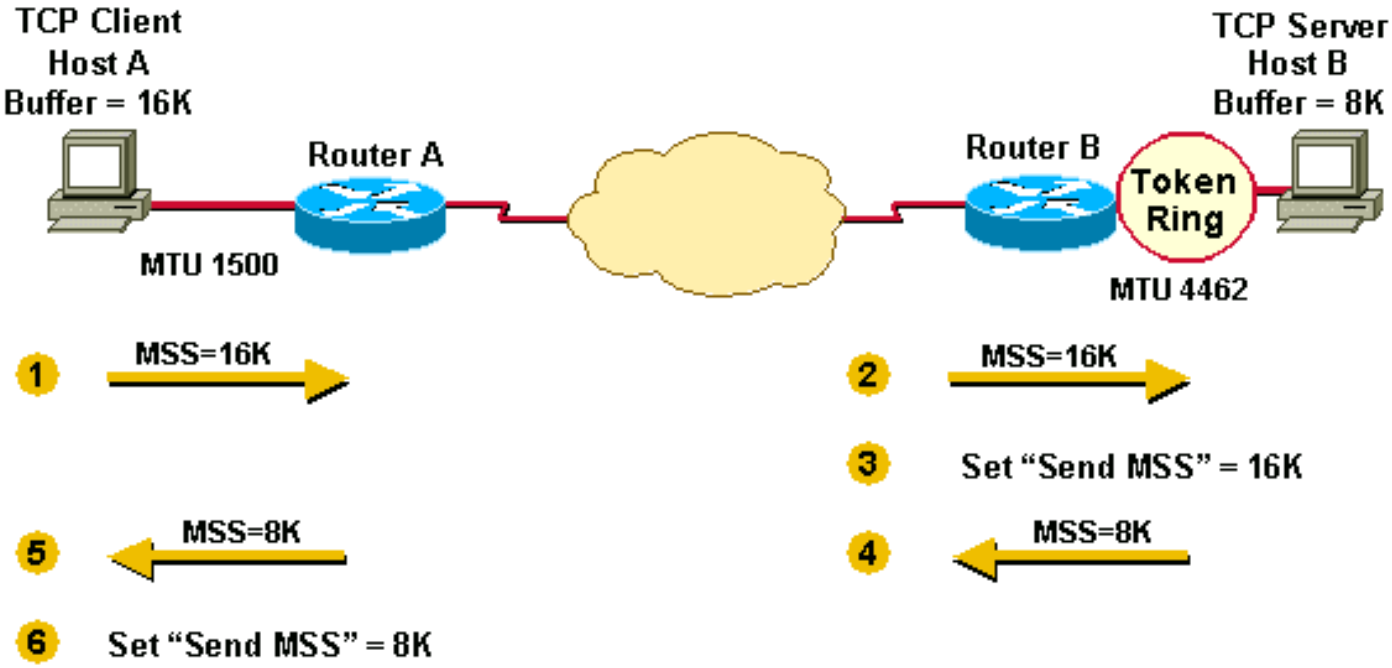
O MSS do TCP define a quantidade máxima de dados que um host pode aceitar em um datagrama de TCP/IP simples. Esta datagrama TCP/IP pôde ser fragmentada na camada IP. O valor MSS é enviado como uma opção de cabeçalho de TCP somente em segmentos TCP SYN. Cada lado de uma conexão de TCP relata seu valor MSS ao outro lado. O contrário à crença popular, o valor MSS não é negociado entre anfitriões. O host de emissão é exigido limitar o tamanho dos dados em um único segmento TCP a um valor inferior ou igual ao MSS relatado pelo host de recepção.

Originalmente, o MSS significava o tamanho do buffer (maior ou igual a 65496 K) que era alocado em uma estação de recebimento para poder armazenar os dados TCP incluídos em um único datagrama IP. O MSS era o segmento máximo (pedaço) dos dados esses o receptor de TCP era disposto aceitar. Este segmento TCP poderia ter 64K (o tamanho máximo de datagrama de IP) e poderia estar fragmentado na camada de IP a fim de ser transmitido na rede para o host de recebimento. O host receptor remontava o datagrama de IP antes de entregar o segmento completo de TCP à camada de TCP.

Estão abaixo um par encenações que mostram como os valores MSS são ajustados e usados para limitar tamanhos do segmento TCP, e consequentemente, tamanhos do IP datagram.

A encenação 1 ilustra a maneira que o MSS foi executado primeiramente. Hospede A tem um buffer de 16K e de Host B um buffer de 8K. Eles enviam e recebem seus valores de MSS e ajustam o MSS de envio para enviar dados um ao outro. Observe que que hospeda A e Host B terá que fragmentar as datagramas IP que são maiores do que a interface MTU, mas ainda menos do que a emissão MSS porque a pilha TCP poderia passar os bytes de dados 16K ou 8K abaixo da pilha ao IP. No caso do host b, os pacotes podiam ser fragmentados duas vezes, uma vez para obter no LAN de token ring e para obter outra vez no LAN de Ethernet.

Cenário 1



1. Hospede A envia seu valor MSS de 16K ao Host B.
2. O Host B recebe o valor 16K MSS do host A.
3. O Host B ajusta-se seu envia o valor MSS a 16K.
4. O Host B envia seu valor MSS de 8K para hospedar o A.
5. Hospede A recebe o valor 8K MSS do Host B.
6. Hospede os grupos seus enviam o valor MSS a 8K.

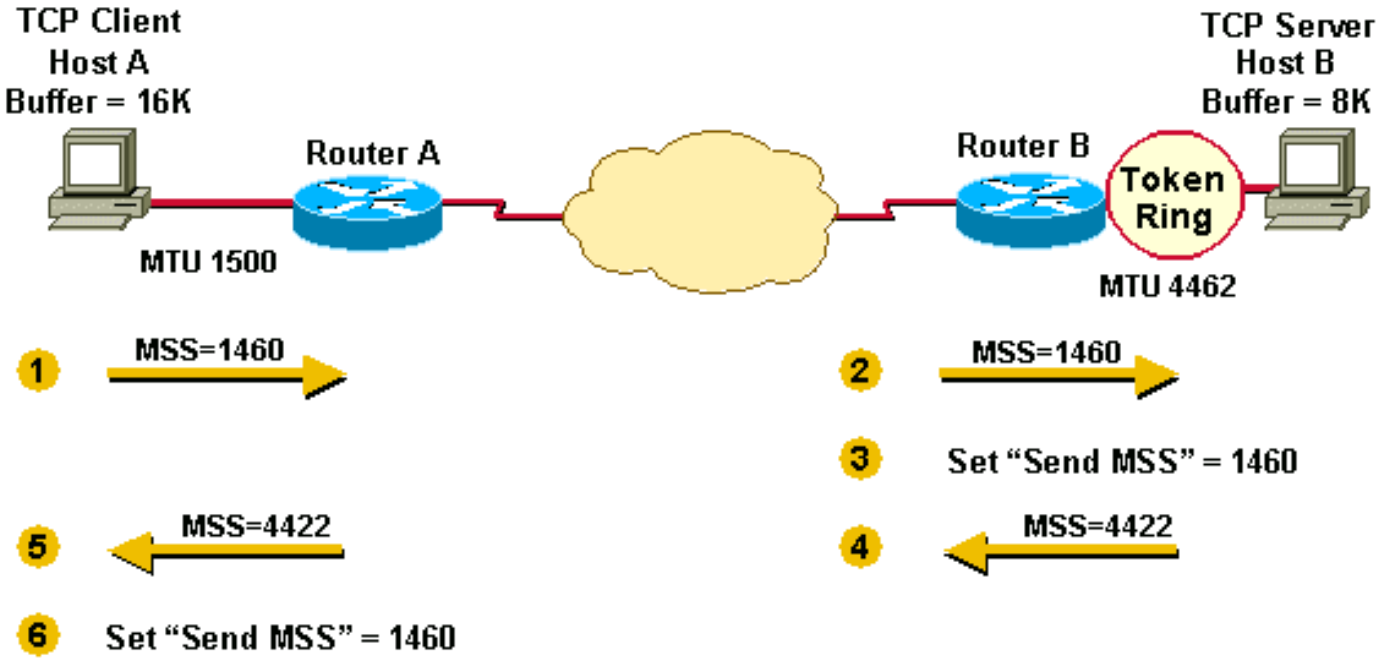
A fim ajudar em evitar a fragmentação de IP nos valores-limite da conexão de TCP, a seleção do valor MSS foi mudada ao tamanho mínimo de buffer e ao MTU da interface enviada (- 40). Os números MSS são 40 bytes menores do que números MTU porque o MSS é apenas o tamanho de dados TCP, que não inclui o encabeçamento do IP de byte 20 e o cabeçalho TCP com XX bytes 20. O MSS é baseado em tamanhos de cabeçalho padrão; a pilha do remetente deve subtrair os valores apropriados para o cabeçalho IP e o dependente do cabeçalho de TCP em

que TCP ou opções IP são usados.

Agora, o MSS faz com que cada host compare primeiro a interface MTU de saída com o próprio buffer e escolha o menor valor como o MSS a ser enviado. Os hosts compararão o tamanho MSS recebido com base em sua própria MTU de interface e escolherão novamente o menor dos dois valores.

A encenação 2 ilustra esta etapa adicional tomada pelo remetente a fim evitar a fragmentação nos fios locais e remotos. Observação como o MTU da interface enviada está levado em consideração por cada host (antes que os anfitriões se enviam seus valores MSS) e como este ajuda a evitar a fragmentação.

Cenário 2



1. Hospede A compara seu buffer MSS (16K) e seu MTU (1500 - 40 = 1460) e usa o valor mais baixo como o MSS (1460) para enviar ao Host B.
2. O Host B recebe o host que os a enviam MSS (1460) e compara-o ao valor de sua interface externa MTU - 40 (4422).
3. O Host B ajusta o valor mais baixo (1460) como o MSS para enviar datagramas IP para hospedar o A.
4. O Host B compara seu buffer MSS (8K) e seu MTU (4462-40 = 4422) e usos 4422 como o MSS enviar para hospedar o A.
5. Hospede A recebe o host que os b enviam MSS (4422) e compara-o ao valor de sua interface externa MTU -40 (1460).
6. Hospede A ajusta o valor mais baixo (1460) como o MSS para enviar datagramas IP ao Host B.

1460 é o valor escolhido por ambos os hosts como o MSS de envio de um para o outro. Frequentemente o valor da emissão MSS será o mesmo em cada extremidade de uma conexão de TCP.

Na encenação 2, a fragmentação não ocorre nos valores-limite de uma conexão de TCP porque ambo a interface enviada MTU é levada em consideração pelos anfitriões. É possível que os pacotes ainda sejam fragmentados na rede, entre os roteadores A e B, se encontrarem um enlace

com uma MTU inferior à das interfaces de saída dos hosts.

O que é PMTUD?

O TCP MSS como mais adiantado descrito toma da fragmentação nos dois valores-limite de uma conexão de TCP, mas não segura o caso onde há um link menor MTU no meio entre estes dois valores-limite. O PMTUD foi desenvolvido a fim evitar a fragmentação no trajeto entre os valores-limite. É usado para determinar dinamicamente o mais baixo MTU ao longo do trajeto da fonte de um pacote a seu destino.

> Nota: O PMTUD é apoiado somente pelo TCP e pelo UDP. Outros protocolos não o apoiam. Se o PMTUD está permitido em um host, e é quase sempre, todo o TCP/IP ou pacotes de UDP do host terão o jogo do bit DF.

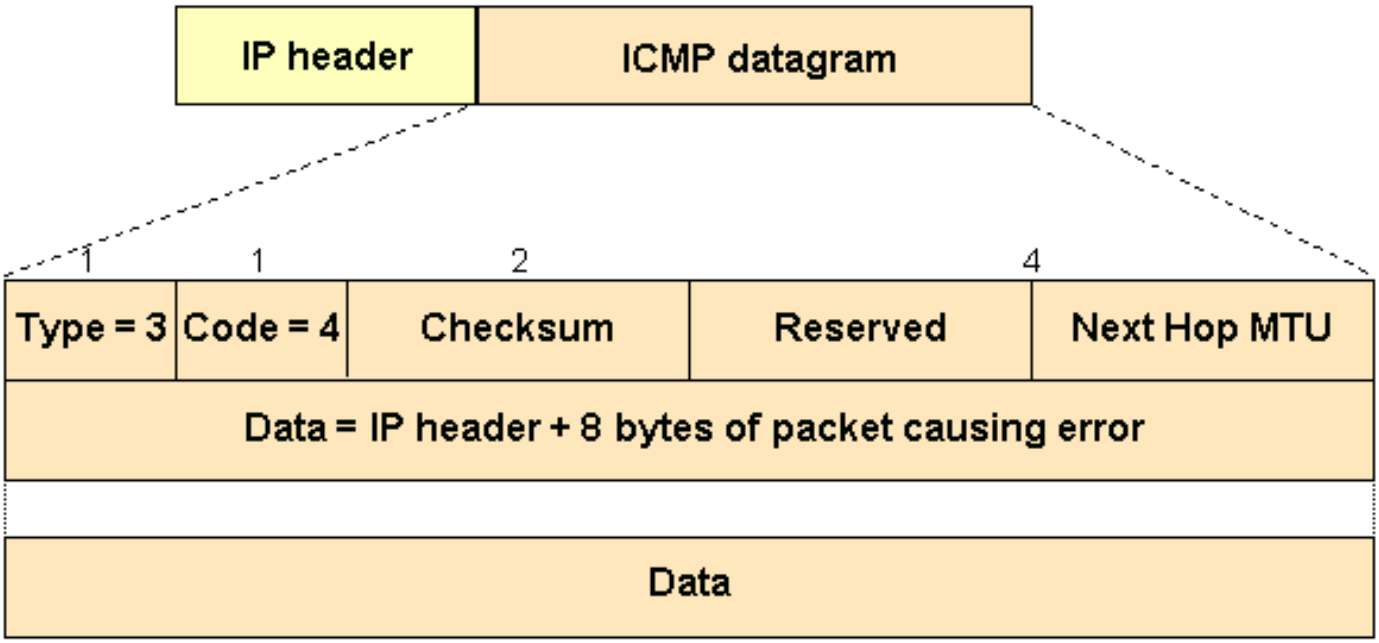
Quando um host envia um pacote de dados completo MSS com o jogo do bit DF, o PMTUD reduz o valor da emissão MSS para a conexão se recebe a informação que o pacote exigiria a fragmentação. Um host geralmente “recorda” o valor MTU para um destino desde que cria uma /32) de entrada do “host” (em sua tabela de roteamento com este valor MTU.

Se um roteador tenta encaminhar um IP datagram, com o jogo do bit DF, em um link que tenha um MTU inferior do que o tamanho do pacote, o roteador deixará cair o pacote e retornará um mensagem " destino inalcançável " do Internet Control Message Protocol (ICMP) à fonte deste IP datagram, com o código que indica a “fragmentação necessária e o DF para ajustar-se” (tipo 3, código 4). Quando a estação de origem recebe a mensagem do ICMP, diminuirá o MSS de envio e, quando o TCP retransmitir o segmento, usará o menor tamanho de segmento.

Está aqui um exemplo de um ICMP “fragmentação necessária e do DF ajustar” a mensagem que você pôde ver em um roteador depois que o **comando debug ip icmp** é girado sobre:

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

Este diagrama mostra o formato do cabeçalho ICMP de uma “fragmentação necessária e do DF ajustar” o mensagem " destino inalcançável ".



Pelo RFC 1191 , um roteador que retorne um mensagem ICMP que indique “fragmentação necessária e o DF ajustar-se” devem incluir o MTU dessa rede de próximo salto nos bit do ordem baixa 16 do campo de cabeçalho adicional ICMP que é etiquetado “não utilizado” no RFC 792 da especificação de ICMP .

As implementações precoces do RFC 1191 não forneceram a informação de MTU do salto seguinte. Mesmo quando esta informação foi fornecida, alguns anfitriões ignoram-na. Para este caso, o RFC 1191 igualmente contém uma tabela que aliste os valores sugeridos por que o MTU deve ser abaixado durante o PMTUD. É usado por anfitriões a fim chegar mais rapidamente em um valor razoável para a emissão MSS.

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

O PMTUD é executado continuamente em todos os pacotes porque o caminho entre o remetente e o receptor pode ser alterado dinamicamente. Cada vez que um remetente recebe “não pode fragmentar” mensagens ICMP que atualizará a informação de roteamento (onde armazena o PMTUD).

Duas coisas podem acontecer durante o PMTUD:

- O pacote pode obter toda a maneira ao receptor sem ser fragmentada.Nota: Para que um roteador proteja o CPU contra ataques DoS, estrangula o número de mensagens que não chega a seu destino do ICMP que enviaria, a dois por segundo. Consequentemente, nestes contexto, se você tem um cenário de rede em que você espera que o roteador precisaria de responder com mais de dois mensagens ICMP (tipo = 3, código = 4) por segundo (podem ser os anfitriões diferentes), você quereria desabilitar o estrangulamento dos mensagens ICMP com **nenhum comando interface inacessível do [df] do taxa-limite ICMP IP**.
- O remetente pode obter mensagens ICMP "Cant Fragment" a partir de quaisquer (ou de todos) os nós junto com o caminho para o receptor.

O PMTUD é efetuado independentemente de ambas as direções de um fluxo de TCP. Pôde haver os casos onde o PMTUD em um sentido de um fluxo provoca uma das estações final para abaixar a emissão MSS e a estação da outra extremidade mantém o original para enviar o MSS porque nunca enviou um IP datagram grande bastante para provocar o PMTUD.

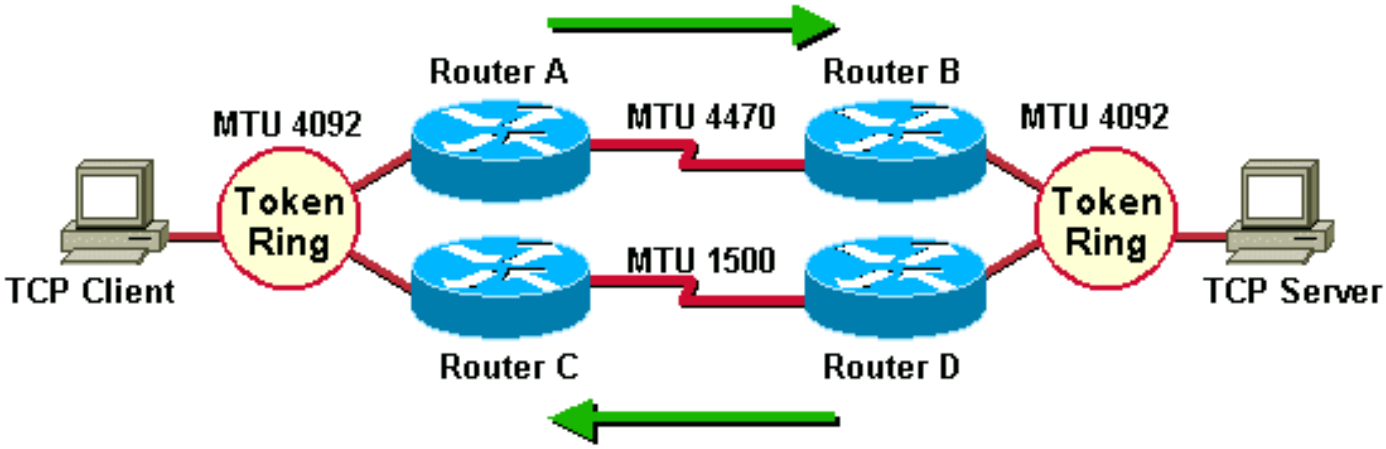
Um bom exemplo deste é a conexão de HTTP descrita abaixo na encenação 3. O cliente TCP envia pacotes pequenos e o server envia grandes pacotes. Neste caso, somente os grandes pacotes do server (maior de 576 bytes) provocarão o PMTUD. Os pacotes do cliente são pequenos (menores que 576 bytes) e não dispararão PMTUD porque não requerem fragmentação para chegar ao link de 576 MTU.

Cenário 3

A encenação 4 mostra a um exemplo do roteamento assimétrico onde um dos trajetos tem um mínimo menor MTU do que o outro. O roteamento assimétrico ocorre quando os trajetos diferentes são tomados para enviar e receber dados entre dois valores-limite. Nesta encenação, o PMTUD provocará a redução da emissão MSS somente em um sentido de um fluxo de TCP. O tráfego do cliente TCP ao server corre através do roteador A e do roteador B, visto que o tráfego de retorno que vem do server ao cliente corre através do roteador D e do C do roteador. Quando o servidor de TCP envia pacotes para o cliente, o PMTUD disparará o servidor para reduzir o envio de MSS porque o roteador D deve fragmentar os pacotes de 4092 bytes antes que ele possa enviá-los para o roteador C.

O cliente, por outro lado, nunca receberá um mensagem " destino inalcançável " ICMP com o código que indica a “fragmentação necessária e o DF se ajustar” porque o roteador A não faz tem que pacotes de fragmento quando os envia ao server através do roteador B.

Encenação 4



Nota: O comando ip tcp path-mtu-discovery é usado para ativar a descoberta do caminho do MTU TCP para conexões TCP iniciadas por roteadores (BGP e Telnet, por exemplo).

Problemas com o PMTUD

Há três coisas que podem romper o PMTUD, duas que são incomuns e uma que é comum.

- Um roteador pode deixar cair um pacote e não enviar um mensagem ICMP. (Incomum)
- Um roteador pode gerar e enviar um mensagem ICMP, mas o mensagem ICMP obtém obstruído por um roteador ou por um Firewall entre este roteador e o remetente. (Terra comum)
- Um roteador pode gerar e enviar um mensagem ICMP, mas o remetente ignora a mensagem. (Incomum)

O primeiro e o último dos três marcadores acima são incomuns e normalmente são resultantes de um erro, mas o marcador central descreve um problema comum. As pessoas que implementam filtros de pacotes de ICMP tendem a bloquear todos os tipos de mensagens de ICMP em vez de bloquear somente determinados tipos de mensagens de ICMP. Um filtro de pacote pode obstruir todos os tipos de mensagem ICMP *exceto* aqueles que são “inacessíveis” ou “tempo excedido.” O sucesso ou êxito de PMTUD depende de mensagens ICMP inacessíveis que passam até o remetente de um pacote TCP/IP. As mensagens do tempo excedido ICMP são importantes para outras edições IP. Um exemplo de tal filtro de pacote, executado em um roteador é mostrado aqui.

```
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
```

Há outras técnicas que podem ser úteis para amenizar o problema de o ICMP estar completamente bloqueado.

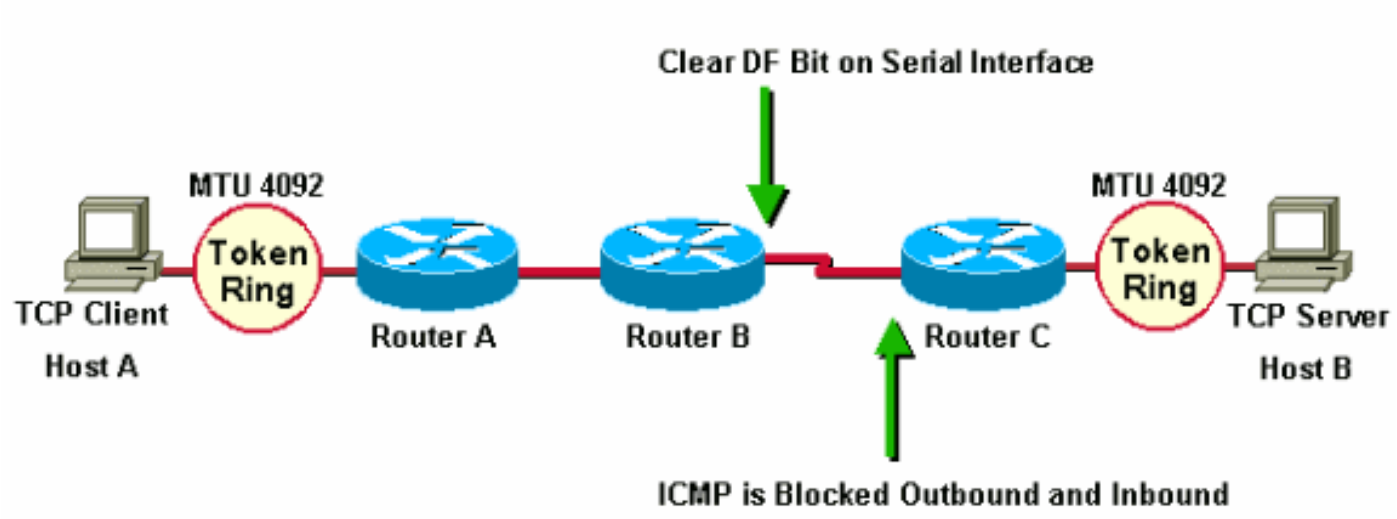
- Cancele o bit DF no roteador e permita a fragmentação de qualquer maneira (esta não pôde ser uma boa ideia, embora. Veja edições com fragmentação de IP para mais informação).
- Manipule o valor de opção MSS MSS TCP com o comando interface que o **IP tcp ajusta-mss <500-1460>**.

Na encenação seguinte, o roteador A e o roteador B estão no mesmo campo administrativo. O C do roteador é inacessível e obstrui o ICMP, assim que o PMTUD é quebrado. Uma ação

alternativa para esta situação é cancelar o bit DF nos ambos sentidos no roteador B a fim permitir a fragmentação. Isto pode ser feito com roteamento de política. A sintaxe para cancelar o bit DF está disponível no Software Release 12.1(6) e Mais Recente de Cisco IOS®.

```
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
```



Uma outra opção é mudar o valor de opção MSS TCP nos pacotes SYN que atravessam o roteador (disponível no Cisco IOS 12.2(4)T e mais tarde). Isto reduz o valor de opção MSS no pacote SYN de TCP de modo que seja menor do que o valor (1460) no **comando ip tcp adjust-mss**. O resultado é que o emissor do TCP enviará segmentos não maiores que esse valor. O tamanho de pacote IP será 40 bytes maior (1500) do que o valor MSS (1460 bytes) a fim esclarecer o cabeçalho de TCP (20 bytes) e o cabeçalho IP (20 bytes).

Épossível ajustar o MSS dos pacotes TCP SYN com o comando ip tcp adjust-mss. Esta sintaxe reduzirá o valor MSS em segmentos TCP a 1460. Este comando efetua o tráfego de entrada e de partida no serial0 da relação.

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

As edições da fragmentação de IP tornaram-se mais difundidas desde que os túneis IP se tornaram distribuídos mais extensamente. A razão que os túneis causam mais fragmentação é porque o encapsulamento de túnel adiciona “despesas gerais” ao tamanho de um pacote. Por exemplo, a adição do Generic Router Encapsulation (GRE) adiciona 24 bytes a um pacote, e depois que este aumento que o pacote pôde precisar de ser fragmentado porque é maior do que o MTU de partida. Em uma seção mais recente deste documento, você verá exemplos dos tipos dos problemas que podem elevarar com túneis e fragmentação de IP.

Topologias de rede comuns que necessitam de PMTUD

O PMTUD é necessário nas situações de rede em que os links intermediários têm MTUs menores que o MTU dos links finais. Algumas razões comuns para a existência desses enlaces de MTU menores são:

- Token Ring (ou FDDI) - host finais conectados com uma conexão Ethernet entre eles. O Token Ring (ou o FDDI) MTU nas extremidades são maior do que os Ethernet MTU no meio.
- O PPPoE (geralmente utilizado com ADSL) precisa de 8 bytes para seu cabeçalho. Isso reduz o MTU efetivo de Ethernet para 1492 (1500 - 8).

Os protocolos de tunelamento como o GRE, o IPsec, e o L2TP igualmente precisam o espaço para seus cabeçalhos respectivos e reboques. Isso também reduz a MTU efetiva da interface de saída.

Nas próximas seções, o impacto do PMTUD onde um protocolo de tunelamento é usado em algum lugar entre os dois host finais é estudado. Dos três casos precedentes, este caso é o mais complexo e cobre todas as edições que você pôde ver nos outros casos.

O que é um túnel?

Um túnel é uma interface lógica em um roteador Cisco que forneça uma maneira de encapsular pacotes de passageiro dentro de um protocolo de transporte. É uma arquitetura projetada proporcionar os serviços para executar um esquema do encapsulamento de Point-to-Point. O Tunelamento tem estes três componentes principais:

- Protocolo de passageiro (APPLETALK, Banyan VINES, CLNS, DECNet, IP, ou IPX)
- Protocolo do portador - Um destes protocolos de encapsulamento:GRE - O protocolo de portador multiprotocolo de Cisco. Veja o RFC 2784 e o RFC 1701 para mais informação.Túneis do IP in IP - Veja o RFC 2003 para mais informação.
- Protocolo de transporte - O protocolo usado para transportar o protocolo encapsulado

Os pacotes mostrados nesta seção ilustram os conceitos de Tunelamento IP onde o GRE é o protocolo de encapsulamento e o IP é o protocolo de transporte. O protocolo de passageiro também é IP. Neste caso, o IP é o transporte e o protocolo de passageiro.

Pacote normal

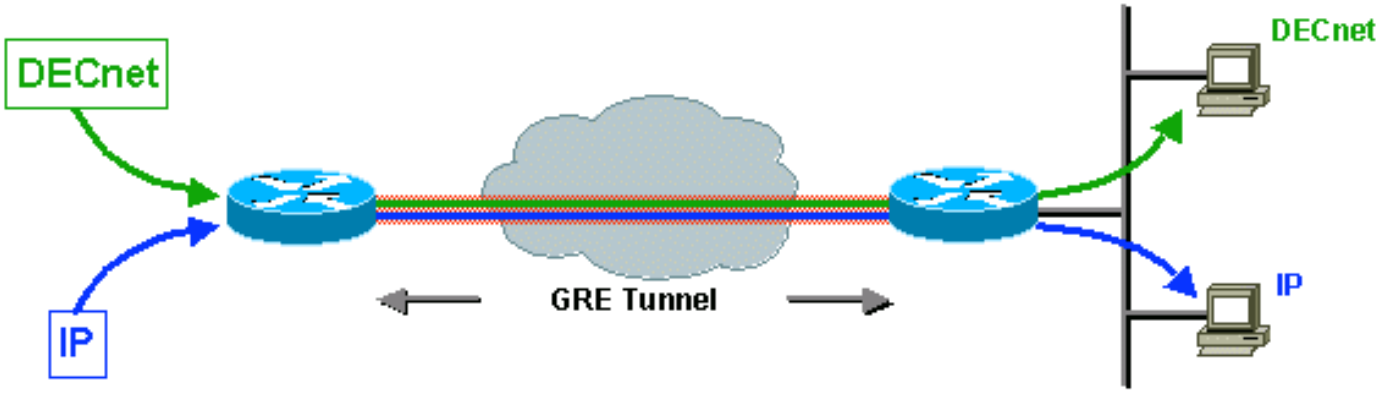
IP TCP Telnet

Pacote de túneis

IP GRE IP TCP Telnet

- O IP é o protocolo de transporte.
- O GRE é o protocolo de encapsulamento.
- O IP é o protocolo de passageiro.

O próximo exemplo mostra o encapsulamento de IP e DECnet como protocolos de passageiro com GRE como portador. Isso ilustra o fato de que o protocolo da portadora pode encapsular vários protocolos de passageiro.



Um administrador de rede pôde considerar escavar um túnel em uma situação onde houvesse duas redes não-IP discontiguous separadas por um backbone IP. Se as redes descontígua executam o DECNet, o administrador não pôde querer conectá-las junto configurando o DECNet no backbone. O administrador não pôde querer permitir o roteamento decnet consumir a largura de banda de backbone porque esta poderia interferir com o desempenho da rede IP.

Uma alternativa viável é escavar um túnel o DECNet sobre o backbone IP. O Tunelamento encapsula os pacotes decnet dentro do IP, e envia-os através do backbone ao ponto final de túnel onde o encapsulamento é removido e os pacotes decnet podem ser distribuídos a seu destino através do DECNet.

Encapsular o tráfego dentro de um outro protocolo fornece estas vantagens:

- Os valores-limite usam endereços privados (RFC 1918) e o backbone não apoia a distribuição destes endereços.
- Permita o Virtual Private Networks (VPNs) através dos WAN ou do Internet.
- Junte-se a redes multiprotocolo do juntar redes descontíguas sobre um backbone de protocolo único.
- Cifre o tráfego sobre o backbone ou o Internet.

Para o resto do documento, o IP é usado como o protocolo de passageiro e o IP como o protocolo de transporte.

Considerações com relação às interfaces de túnel

Estas são considerações ao escavar um túnel.

- O interruptor rápido dos túneis GRE foi introduzido no Cisco IOS Release 11.1 e o CEF switching foi introduzido na versão 12.0. O CEF switching para túneis GRE multipontos foi introduzido na versão 12.2(8)T. O encapsulamento e o decapsulation em pontos finais de túnel eram operações lentas nas versões anterior do Cisco IOS quando somente a comutação do processo foi apoiada.
- Há uns problemas de segurança e de topologia ao escavar um túnel pacotes. Os túneis podem desviar listas de controle de acesso (ACLs) e firewalls. Se você escava um túnel com um Firewall, você contorneia basicamente o Firewall para o que protocolo de passageiro você está escavando um túnel. Portanto, é recomendado incluir a funcionalidade de firewall nos pontos finais do túnel para aplicar qualquer política nos protocolos de passageiro.
- O Tunelamento pôde criar problemas com os protocolos de transporte que limitaram temporizadores (por exemplo, DECNet) devido à latência aumentada.
- Escavando um túnel através dos ambientes com links diferentes da velocidade, como os

aneis FDDI rápidos e através das linhas telefônica 9600-bps lentas, pôde introduzir o pacote que requisita novamente problemas. Alguns protocolos de passageiro funcionam deficientemente em redes da mídia mista.
- Os túneis ponto a ponto podem usar a largura de banda em um enlace físico. Se você executa protocolos de roteamento sobre o ponto múltiplo para apontar túneis, mantenha na mente que cada interface de túnel tem uma largura de banda e que a interface física sobre que o túnel é executado tem uma largura de banda. Por exemplo, você quereria ajustar a largura de banda de túnel ao Kb 100 se havia 100 túneis que são executado sobre um link do 10 Mb. A largura de banda padrão para um túnel é 9Kb.
- Os protocolos de roteamento puderam preferir um túnel sobre um link “real” porque o túnel pôde deceptively parecer ser um link do um-salto com o trajeto o mais barato, embora realmente envolvesse mais saltos e fosse realmente mais caro do que um outro trajeto. Isso pode ser mitigado com a configuração correta do Routing Protocol. É possível considerar a execução como um Routing Protocol diferente sobre a interface do túnel e não o roteamento de uma execução de protocolo na interface física.
- Problemas com roteamento recursivo podem ser evitados configurando rotas estáticas apropriadas para o destino do túnel. Uma rota recursiva é quando o melhor caminho ao “destino de túnel” é através do túnel próprio. Esta situação faz com que a interface de túnel salte para cima e para baixo. Você verá este erro quando há um problema de roteamento recursivo.`%TUN-RECURDOWN Interface Tunnel 0`
`temporarily disabled due to recursive routing`

O roteador como um PMTUD participante no ponto final de um túnel

O roteador tem duas funções PMTUD diferentes quando é o ponto final de um túnel.

- No primeiro papel o roteador é o remetente de um pacote do host. Para o processamento de pmtud, o roteador precisa de verificar o bit DF e o tamanho do pacote do pacote de dados originais e de tomar a ação apropriada quando necessário.
- A segunda função é exercida depois que o roteador encapsulou o pacote IP original dentro do pacote de túnel. Nesta fase, o roteador atua mais como um host no que diz respeito ao PMTUD e com respeito ao pacote IP do túnel.

Deixa o começo olhando o que acontece quando o roteador atua no primeiro papel, um roteador que para a frente pacotes do IP de host, no que diz respeito ao PMTUD. Este papel entra o jogo antes que o roteador encapsule o pacote do IP de host dentro do pacote de túnel.

Se o roteador participa porque o remetente de um pacote do host ele terminará estas ações:

- Verifique se o bit DF esteja ajustado.
- Verifique que pacote do tamanho o túnel pode acomodar.
- O fragmento (se o pacote é demasiado grande e bit DF não está ajustado), encapsula fragmentos e envia-os; ou
- Deixe cair o pacote (se o pacote é demasiado grande e bit DF está ajustado) e envie um mensagem ICMP ao remetente.
- Encapsular (se o pacote não é demasiado grande) e envie.

Genericamente, há uma opção de encapsulamento e então uma fragmentação (envie dois fragmentos do encapsulamento) ou uma fragmentação e então um encapsulamento (envie dois fragmentos encapsulados).

Alguns exemplos que descrevem os mecânicos do encapsulamento e fragmentação do pacote IP e duas encenações que mostram a interação do PMTUD e os pacotes que as redes de exemplo transversais são detalhadas nesta seção.

O primeiro exemplo mostra o que acontece a um pacote quando o roteador (no origem de túnel) atua no papel do roteador de encaminhamento. Recorde isso processar o PMTUD, o roteador precisa de verificar o bit DF e o tamanho do pacote do pacote de dados originais e de tomar a ação apropriada. Este exemplo usa encapsulamento GRE para o túnel. Como pode ser visto, o GRE faz a fragmentação antes do encapsulamento. Exemplos mostram cenários mais atrasados em que a fragmentação é feita após o encapsulamento.

No exemplo 1, o bit DF não é ajustado (DF = 0) e o IP de túnel GRE MTU é 1476 (1500 - 24).

Exemplo 1

1. O roteador de encaminhamento (na origem de túnel) recebe um datagrama de 1500 bytes com o bit DF limpo (DF = 0) do host de envio. Esse datagrama é composto por um cabeçalho de IP de 20 bytes e um payload de TCP de 1480 bytes.
2. Porque o pacote será demasiado grande para o IP MTU depois que a carga adicional de GRE (24 bytes) é adicionada, o roteador de encaminhamento quebra a datagrama em dois fragmentos de 1476 (cabeçalho IP de 20 bytes + virulência IP de 1456 bytes) e 44 bytes (20 bytes do cabeçalho IP + 24 bytes da virulência IP) assim que depois que o encapsulamento de GRE é adicionado, o pacote não será maior do que a interface MTU da física de saída.
3. O roteador de encaminhamento adiciona o encapsulamento de GRE, que inclui um cabeçalho de GRE 4-byte mais um cabeçalho IP 20-byte, a cada fragmento da datagrama de IP original. Estas duas datagramas IP têm agora um comprimento de 1500 e 68 bytes e estas datagramas são consideradas como datagramas do IP individual, não como fragmentos.
4. O roteador de destino de túnel remove o encapsulamento de GRE de cada fragmento da datagrama original, que sae de dois fragmentos IP dos comprimentos 1476 e 24 bytes. Estes fragmentos de datagrama de IP serão encaminhados separadamente por este roteador para o host de recepção.
5. O host de recepção remontará estes dois fragmentos na datagrama original.

O cenário 5 descreve a função do roteador de encaminhamento no contexto de uma topologia de rede.

Neste exemplo o roteador atua no mesmo papel do roteador de encaminhamento, mas esta vez o bit DF é ajustado (DF = 1).

Exemplo 2

1. O roteador de encaminhamento na origem do túnel recebe um datagrama de 1500 bytes com DF = 1 do host de envio.
2. Como o bit DF está definido e o tamanho do datagrama (1500 bytes) é maior do que o túnel GRE IP MTU (1476), o roteador encerra o datagrama e envia uma mensagem "fragmentação ICMP necessária, menos o conjunto de bits DF" para a origem do datagrama. O mensagem ICMP alertará o remetente que o MTU é 1476.
3. O host de emissão recebe o mensagem ICMP, e quando envia novamente os dados originais ele usará um IP datagram 1476-byte.
4. O comprimento desse datagrama IP (1476 bytes) está igual em termos de valor à MTU IP do

túnel GRE e, portanto, o roteador adiciona o encapsulamento GRE ao datagrama IP.
5. O roteador de recebimento (no destino do túnel) remove o encapsulamento do GRE do datagrama IP e envia-o para o host recebedor.

Agora nós podemos olhar o que acontece quando o roteador atua no segundo papel como um host de emissão no que diz respeito ao PMTUD e com respeito ao pacote IP do túnel. Com essa rechamada, a função entra em ação após o roteador ter encapsulado o pacote IP original no pacote de túnel.

> Nota: À revelia um roteador não faz o PMTUD nos pacotes de túnel GRE que gerencie. O comando tunnel path-mtu-discovery pode ser utilizado para ativar PMTUD para pacotes de túnel GRE-IP.

O exemplo 3 mostra o que acontece quando o host envia as datagramas IP que são pequenas bastante caber dentro do IP MTU na interface do túnel GRE. O bit DF, nesse caso, pode ser configurado ou limpo (1 ou 0). A interface do túnel GRE não tem o **comando tunnel path-mtu-discovery** configurou assim que o roteador não fará o PMTUD no pacote GRE-IP.

Exemplo 3

1. O roteador de encaminhamento na origem do túnel recebe um datagrama de 1476 bytes do host de envio.
2. Esse roteador encapsula o datagrama de IP de 1.476 bytes dentro do GRE para obter um datagrama de IP GRE de 1.500 bytes. O bit DF no cabeçalho IP GRE será limpo (DF = 0). Esse roteador encaminha, em seguida, esse pacote ao destino do túnel.
3. Supõe que há um roteador entre o origem e destino do túnel com um link MTU de 1400. Esse roteador fragmentará o pacote de túnel porque o bit de está limpo (DF = 0). Recorde que este exemplo fragmenta o IP ultraperiférico, assim que o GRE, o IP interno, e os cabeçalhos de TCP aparecerá somente no primeiro fragmento.
4. O roteador de destino do túnel deve realizar a remontagem dos pacotes GRE do túnel.
5. Depois que o pacote de túnel GRE é remontado, o roteador remove o cabeçalho IP GRE e envia a datagrama de IP original em sua maneira.

O exemplo seguinte mostra o que acontece quando o roteador atua no papel de um host de emissão no que diz respeito ao PMTUD e com respeito ao pacote IP do túnel. Esta vez o bit DF é ajustado (DF = 1) no cabeçalho de IP original e o **comando tunnel path-mtu-discovery** foi configurado de modo que o bit DF fosse copiado do cabeçalho IP interno (GRE +IP) ao encabeçamento exterior.

Exemplo 4

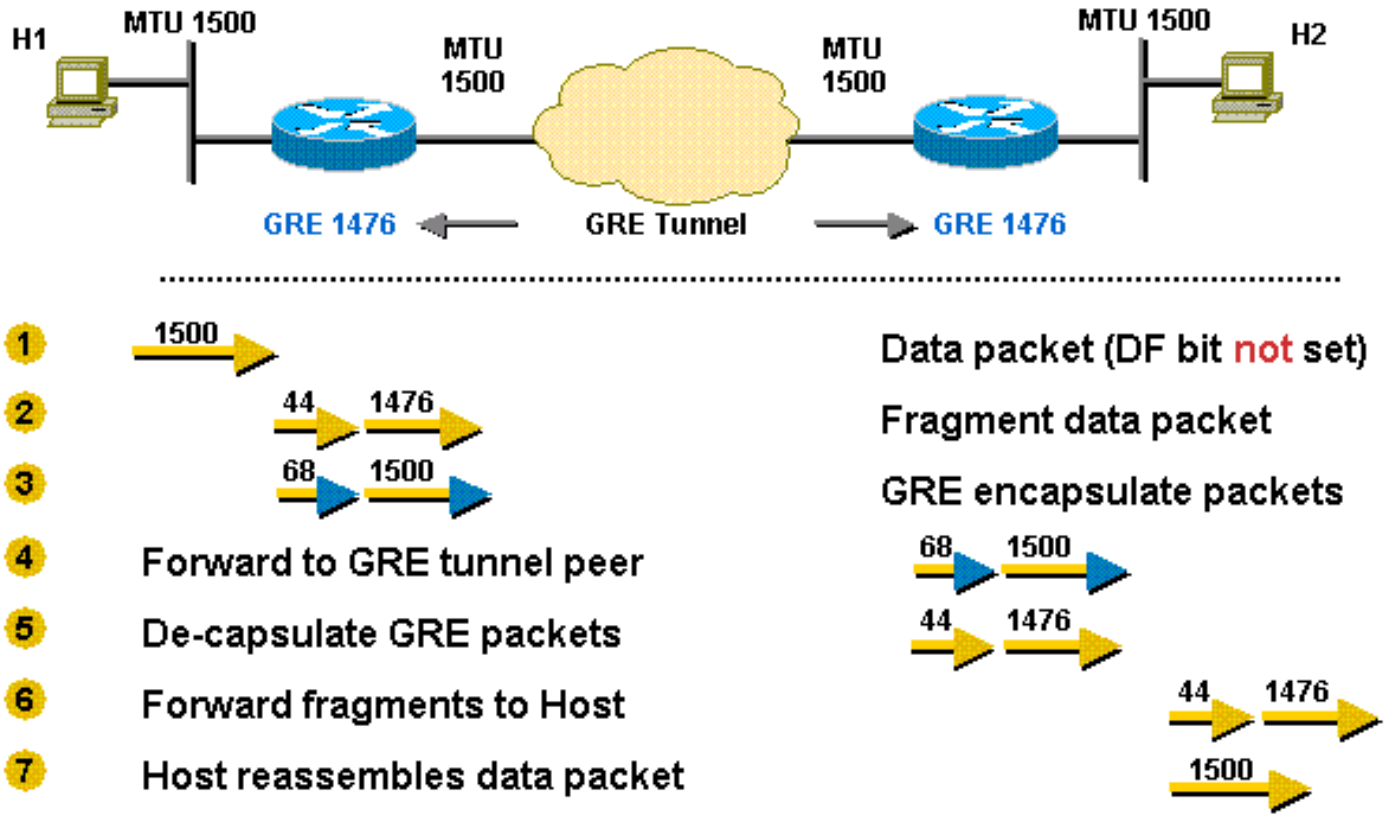
1. O roteador de encaminhamento no origem de túnel recebe uma datagrama 1476-byte com DF = 1 do host de emissão.
2. Esse roteador encapsula o datagrama de IP de 1.476 bytes dentro do GRE para obter um datagrama de IP GRE de 1.500 bytes. Este cabeçalho IP GRE terá o jogo do bit DF (DF = 1) desde que a datagrama de IP original teve o jogo do bit DF. Esse roteador encaminha, em seguida, esse pacote ao destino do túnel.
3. Além disso, supõe que há um roteador entre o origem e destino do túnel com um link MTU de 1400. Esse roteador não fragmentará o pacote de túnel, uma vez que o bit DF está definido (DF = 1). Este roteador deve deixar cair o pacote e enviar um mensagem de erro ICMP ao roteador do origem de túnel, desde que aquele é o endereço IP de origem no

pacote.

4. O roteador de encaminhamento na origem do túnel recebe esta mensagem de erro de ICMP e diminuirá o MTU IP de túnel GRE para 1376 (1400 - 24). A próxima vez que o host de envio retransmitir os dados em um pacote IP de 1476 bytes, este será considerado muito grande, e o roteador enviará uma mensagem de erro ICMP ao emissor com valor MTU de 1376. Quando o host de emissão retransmite os dados, enviá-lo-á em um pacote IP 1376-byte e este pacote fá-lo-á através do túnel GRE ao host de recepção.

Encenação 5

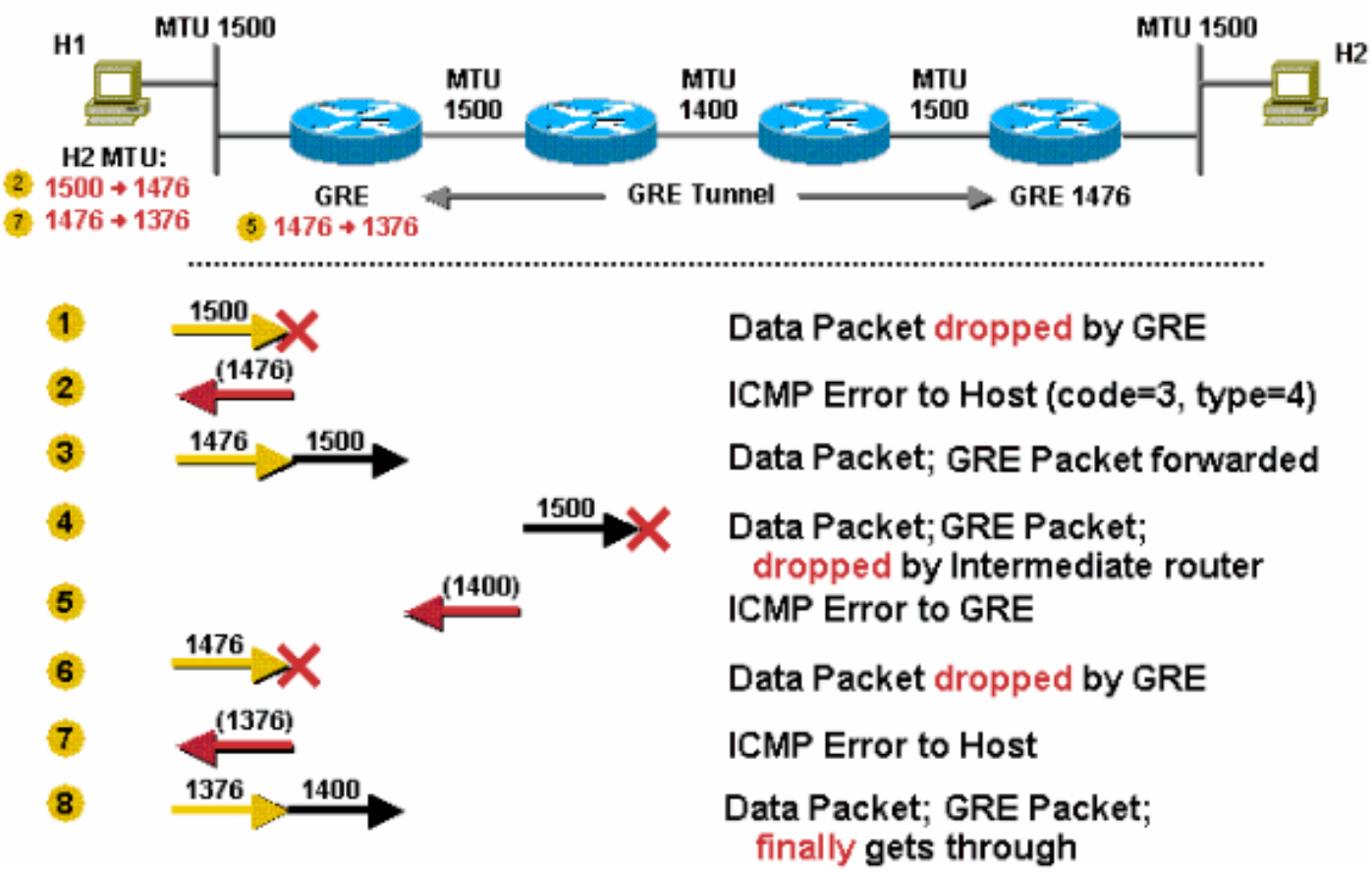
Esse cenário ilustra a fragmentação GRE. Recorde que você fragmenta antes do encapsulamento para o GRE, a seguir fazem o PMTUD para o pacote de dados, e o bit DF não é copiado quando o pacote IP é encapsulado pelo GRE. Nesta encenação, o bit DF não é ajustado. O MTU IP da interface do túnel GRE é, por padrão, 24 bytes menor do que o MTU IP da interface física, por isso o MTU IP da interface GRE é 1476.



1. O remetente envia um pacote 1500-byte (encabeçamento do IP de byte 20 + 1480 bytes do payload de TCP).
2. Como a MTU do túnel GRE é 1476, o pacote de 1500 bytes está dividido em dois fragmentos IP de 1476 e 44 bytes, cada em antecipação dos 24 bytes adicionais do cabeçalho GRE.
3. Os 24 bytes do cabeçalho GRE são adicionados a cada fragmento IP. Agora os fragmentos são 1500 (1476 + 24) e 68 (44 + 24) bytes cada um.
4. Os pacotes GRE +IP que contêm os dois fragmentos IP são enviados ao roteador de peer do túnel GRE.
5. O roteador de peer do túnel GRE remove os cabeçalhos de GRE dos dois pacotes.
6. Esse roteador encaminha os dois pacotes para o host de destino.
7. O host de destino remonta os fragmentos IP de novo na datagrama de IP original.

Encenação 6

Esta encenação é similar à encenação 5, mas esta vez o bit DF é ajustado. Na encenação 6, o roteador é configurado para fazer o PMTUD em pacotes de túnel GRE +IP com o **comando tunnel path-mtu-discovery**, e o bit DF é copiado do cabeçalho de IP original ao cabeçalho IP GRE. Se o roteador recebe um erro ICMP para o pacote GRE +IP, reduz o IP MTU na interface do túnel GRE. Além disso, recorde que o IP de túnel GRE MTU está ajustado a 24 bytes menos do que a interface física MTU à revelia, assim que o IP MTU GRE aqui é 1476. Igualmente observe que há um link de 1400 MTU no trajeto do túnel GRE.



1. O roteador recebe um pacote de 1500 bytes (cabeçalho de IP de 20 bytes + payload de TCP 1480) e descarta o pacote. O roteador deixa cair o pacote porque é maior do que o IP MTU (1476) na interface do túnel GRE.
2. O roteador envia um erro ICMP ao remetente informando que o próximo MTU de nó é 1476. O host registrará essas informações, normalmente como uma rota de host para o destino em sua tabela de roteamento.
3. O host de envio usa um tamanho de pacote de 1.476 bytes quando reenvia os dados. O roteador GRE acrescenta 24 bytes de encapsulamento de GRE e envia um pacote de 1500 bytes.
4. O pacote de 1500 bytes não pode atravessar o enlace de 1400 bytes; portanto, será descartado pelo roteador intermediário.
5. O roteador intermediário envia um ICMP (tipo = 3, código = 4) ao roteador de GRE com um salto seguinte MTU de 1400. O roteador GRE reduz isso para 1376 (1400 - 24) e define um valor de MTU IP interno na interface GRE. Esta mudança pode somente ser considerada ao usar o **comando debug tunnel**; não se pode ver na saída do **comando interface tunnel<-> da mostra IP**.
6. A próxima vez o host envia novamente o pacote 1476-byte, o roteador de GRE deixará cair

o pacote, desde que é maior do que o IP atual MTU (1376) na interface do túnel GRE.
7. O roteador de GRE enviará um outro ICMP (tipo = 3, código = 4) ao remetente com um salto seguinte MTU de 1376 e o host atualizará sua informação atual com valor novo.
8. O host reenvia novamente os dados, mas agora num pacote menor de 1376 bytes; o GRE incluirá 24 bytes de encapsulamento e o encaminhará. Esta vez o pacote fá-lo-á ao par do túnel GRE, onde o pacote será descapsulado e enviado ao host de destino.Nota: Se o **comando tunnel path-mtu-discovery** não foi configurado no roteador de encaminhamento nesta encenação, e o bit DF foi ajustado nos pacotes enviados através do túnel GRE, o host 1 ainda sucederia em enviar pacotes TCP/IP para hospedar 2, mas obteriam fragmentados no meio no link de 1400 MTU. Igualmente o par do túnel GRE teve que remontá-los antes que poderia decapsulate e enviá-los sobre.

Modo de túnel IPsec “puro”

O protocolo da Segurança IP (IPsec) é um método baseado em padrões que forneça a privacidade, a integridade, e a autenticidade à informação transferida através das redes IP. O IPsec fornece a criptografia de camada de rede IP. O IPsec alonga o pacote IP adicionando pelo menos um cabeçalho IP (modo de túnel). O encabeçamento adicionado varia de comprimento o dependente no modo da configuração IPSec mas não excede ~58 bytes (Encapsulating Security Payload (ESP) e autenticação ESP (ESPauth)) pelo pacote.

O IPsec tem dois modos, modos de túnel e modos de transporte.

- Modo de túnel é o modo padrão. Com modo de túnel, o pacote IP original inteiro é protegido (cifrado, autenticado, ou ambos) e encapsulado pelos cabeçalhos IPSec e pelos reboques. Então um cabeçalho IP novo prepended ao pacote, que specifes os pontos finais de IPSec (pares) como a fonte e o destino. O modo de túnel pode ser usado com qualquer tipo de tráfego de IP de unicast e deve ser usado se o IPSec estiver protegendo o tráfego de hosts por atrás dos peers do IPSec. Por exemplo, o modo de túnel é usado com Virtual Private Networks (VPNs) onde os anfitriões em uma rede protegida enviam pacotes aos anfitriões em uma rede protegida diferente através de um par de ipsec peer. Com VPNs, o túnel" IPSec protege o tráfego IP entre os hosts ao criptografar esse tráfego entre os roteadores peer do IPSec.
- Com o modo de transporte (configurado com o subcomando mode transport, na definição de transformação), somente a payload do pacote IP original será protegida (criptografada, autenticada ou ambas). O payload é encapsulado pelos cabeçalhos e trailers de IPsec. Os cabeçalhos de IP original permanecem intactos, salvo que o campo do protocolo IP é mudado para ser ESP (50 pés), e o valor do protocolo original salvar no trailer IPsec a ser restaurado quando o pacote é decifrado. O modo de transporte é usado somente quando o tráfego IP a ser protegido está entre os próprios peers do IPsec e quando os endereços IP de origem e destino no pacote são os mesmos que os endereços do peer do IPsec. O modo transporte de IPpsec está usado normalmente somente quando um outro protocolo de tunelamento (como o GRE) está usado a primeiramente encapsula o pacote de dados IP, a seguir o IPsec está usado para proteger os pacotes de túnel GRE.

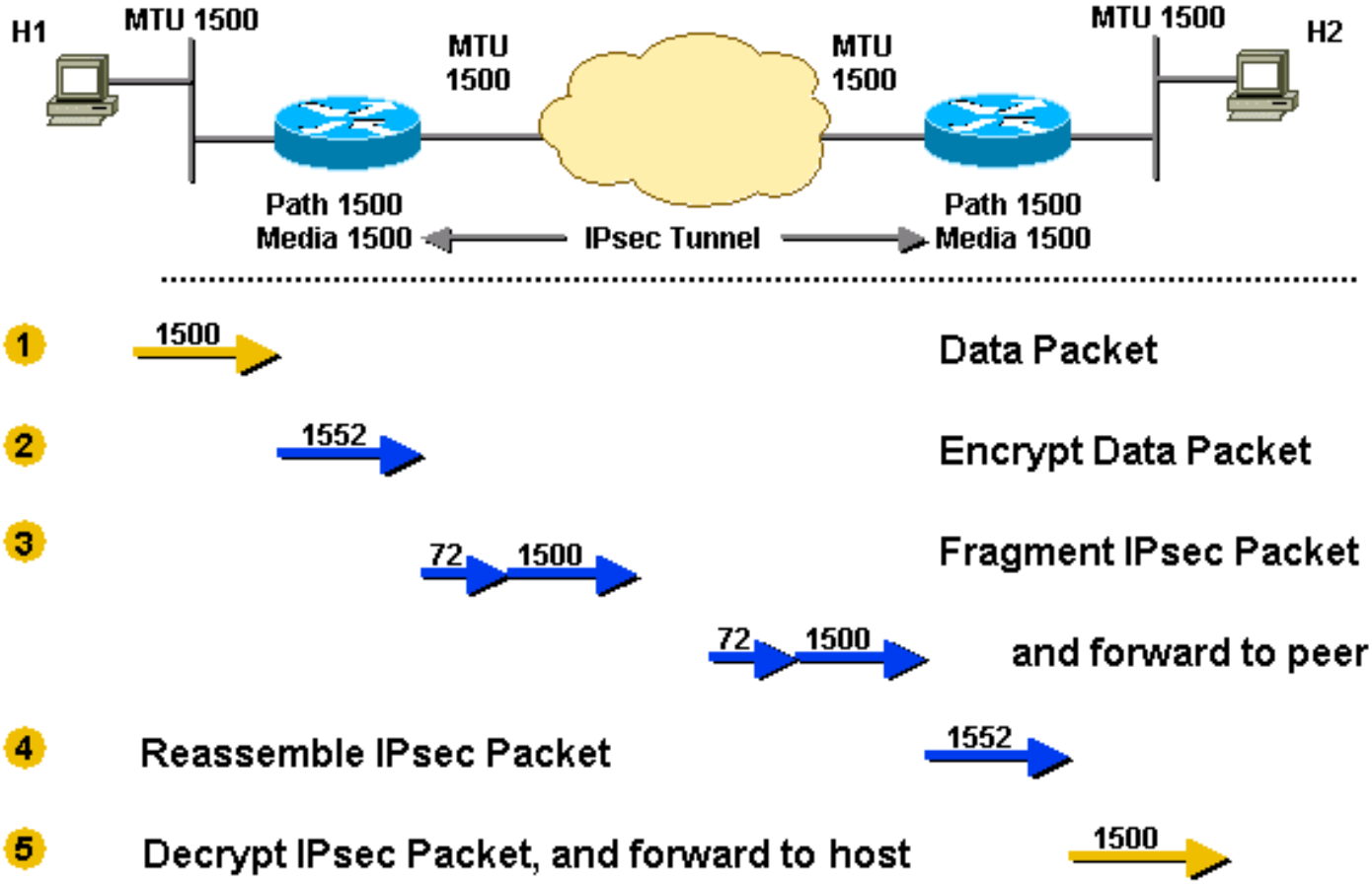
O IPsec faz sempre o PMTUD para pacotes de dados e para seus próprios pacotes. Há comandos de configuração Ipsec para modificar o processamento de PMTUD de maneira que o pacote de IP IPsec, IPsec possa limpar, definir ou copiar o bit DF do cabeçalho IP do pacote de dados para o cabeçalho IP IPsec. Esse recurso é denominado "Funcionalidade de Anulação de

Bit DF".

> Nota: Você realmente deseja evitar a fragmentação após o encapsulamento quando executa criptografia de hardware com IPsec. A criptografia de hardware pode dar-lhe a taxa de transferência aproximadamente de Mbs dos 50 pés segundo o hardware, mas se o pacote de IPsec o é fragmentado frouxamente 50 pés a 90 por cento da taxa de transferência. Essa perda ocorre porque os pacotes IPsec fragmentados são comutados por processo para remontagem e, em seguida, são enviados ao mecanismo de criptografia do hardware para decodificação. Esta perda de throughput pode derrubar a taxa de transferência da criptografia de hardware ao nível de desempenho da criptografia de software (2-10 Mbs).

Encenação 7

Esta encenação descreve a fragmentação de IPsec na ação. Nesta encenação, o MTU ao longo do trajeto inteiro é 1500. Nesta encenação, o bit DF não é ajustado.



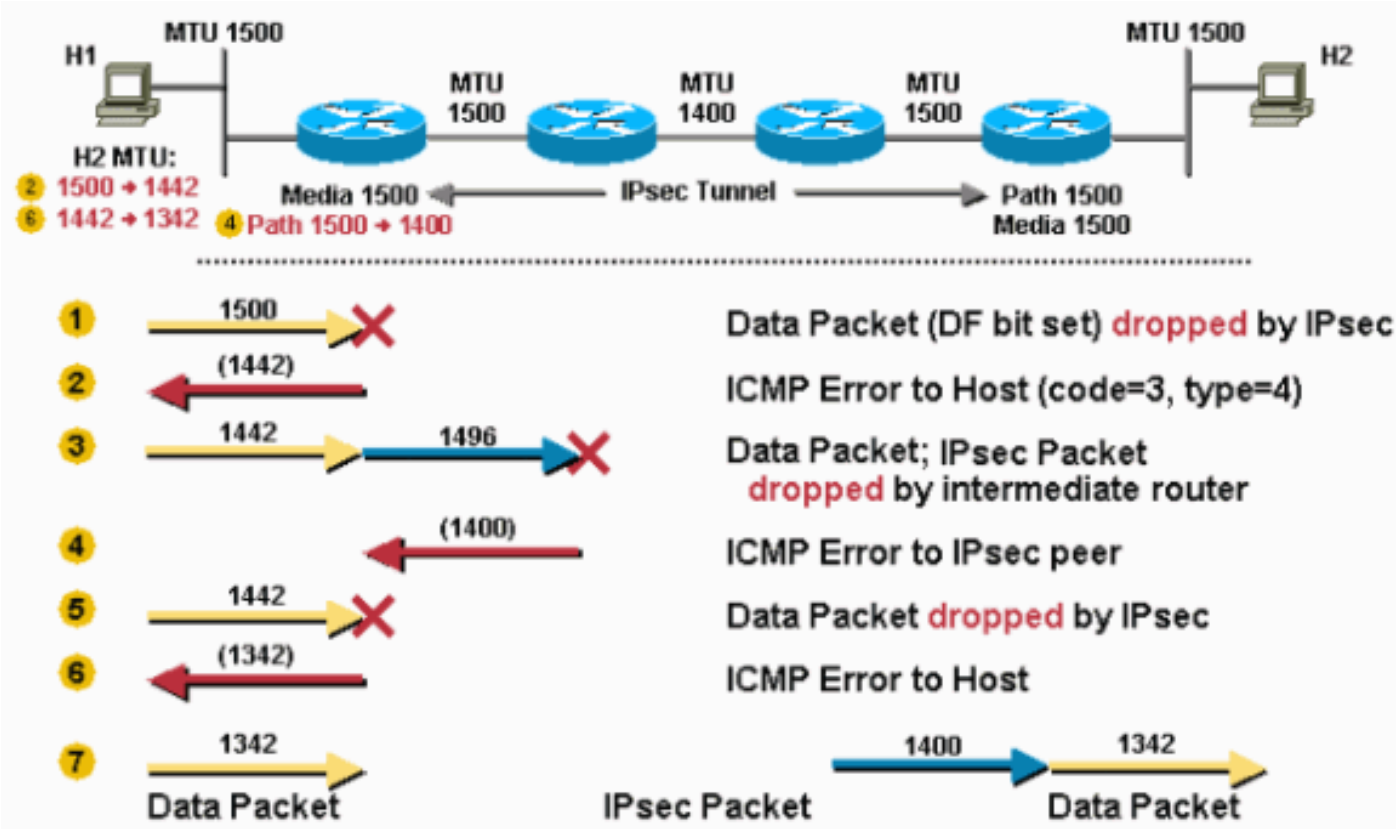
1. O roteador recebe um pacote de 1.500 bytes (cabeçalho IP de 20 bytes + virulência TCP de 1.480 bytes) destinado ao Host 2.
2. O pacote de 1500 bytes é criptografado por IPsec e 52 bytes de sobrecarga são adicionados (cabeçalho IPsec, trailer IPsec e cabeçalho IP adicional). Agora o IPsec precisa de enviar um pacote 1552-byte. Como o MTU de saída é 1500, este pacote terá que ser fragmentado.
3. Dois fragmentos são criados a partir do pacote de IPsec. Durante a fragmentação, um cabeçalho IP 20-byte adicional é adicionado para o segundo fragmento, tendo por resultado um fragmento 1500-byte e um fragmento IP 72-byte.
4. O roteador de peer do túnel de IPsec recebe os fragmentos, descasca o cabeçalho IP

adicional e coalesce os fragmentos IP de novo no pacote de IPsec original. Então o IPsec decifra este pacote.
5. O roteador encaminha o pacote de dados original de 1500 bytes para o Host 2.

Encenação 8

Esta encenação é similar à encenação 6 salvo que o bit DF é ajustado neste caso no pacote de dados originais e há um link no trajeto entre os pares do túnel de IPsec que tem um MTU inferior do que os outros links. Esta encenação demonstra como o roteador de IPSec peer executa ambos os papéis pmtud, como descrito o no roteador como um participante pmtud na seção do ponto final de um túnel.

Você verá nesta encenação como o IPsec PMTU muda a um valor mais baixo como consequência da necessidade para a fragmentação. Lembre-se de que o bit DF é copiado do cabeçalho IP interno para o cabeçalho IP externo quando o IPsec criptografa um pacote. Os media MTU e os valores PMTU são armazenados na associação de segurança IPSec (SA). A MTU de mídia tem com base a MTU da interface de roteador de saída e a PMTU tem como base a MTU mínima vista no caminho entre os correspondentes IPsec. Lembre-se de que IPsec encapsula/criptografa o pacote antes de tentar fragmentá-lo.



1. O roteador recebe um pacote 1500-byte e deixa-o cair porque a carga adicional de IPsec, quando adicionada, fará o pacote maior então o PMTU (1500).
2. O roteador envia uma mensagem ICMP ao Host 1 informando-o de que o MTU do próximo salto é 1442 (1500 - 58 = 1442). Esses 58 bytes são o valor máximo de overhead do IPsec quando o ESP e o ESPauth do IPsec estiverem sendo usados. A carga adicional real de IPsec pode ser tanto quanto os bytes 7 menos do que este valor. Hospede 1 grava esta informação, geralmente enquanto uma rota do host para o destino (host 2), em sua tabela de roteamento.
3. O host 1 abaixa seu PMTU para o host 2 1442, assim que o host 1 enviará (pacotes

menores do byte 1442) quando retransmite os dados para hospedar 2. O roteador recebe o pacote 1442-byte e o IPsec adiciona 52 bytes de carga adicionais de criptografia assim que o pacote de IPsec resultante é 1496 bytes. Porque este pacote tem o jogo do bit DF em seu encabeçamento obtém deixado cair pelo roteador intermediária com o link 1400-byte MTU.
4. O roteador intermediária que deixou cair o pacote envia-o a um mensagem ICMP ao remetente do pacote de IPsec (primeiro roteador) que diz que o salto seguinte MTU é 1400 bytes. Esse valor é registrado no PMTU do IPsec SA.
5. Da próxima vez que o Host 1 retransmitir o pacote de 1442 bytes (ele não recebeu uma confirmação para isso), o IPsec descartará o pacote. Outra vez o roteador deixará cair o pacote porque a carga adicional de IPsec, quando adicionada ao pacote, o fará maior do que o PMTU (1400).
6. O roteador envia um mensagem ICMP para hospedar 1 que diz lhe que o salto seguinte MTU é agora 1342. (1400 - 58 = 1342). O host 1 gravará outra vez esta informação.
7. Quando o host 1 retransmite outra vez os dados, usará o pacote menor do tamanho (1342). Esse pacote não exigirá fragmentação e fará isso por meio do túnel Ipsec para o Host2.

GRE e IPsec juntos

Mais interações complexas para fragmentação e PMTUD ocorrem quando o IPsec é usado a fim cifrar túneis GRE. O IPsec e o GRE são combinados desse modo porque o IPsec não apoia pacotes do Protocolo IP multicast, assim que significa que você não pode executar um protocolo de roteamento dinâmico sobre a rede do IPSec VPN. Os túneis GRE apoiam o Multicast, assim que um túnel GRE pode ser usado a primeiramente encapsula o pacote de transmissão múltipla do protocolo de roteamento dinâmico em um pacote do unicast IP GRE, que possa então ser cifrado pelo IPsec. Ao fazer isto, o IPsec geralmente está implementado no modo de transporte no ponto máximo de GRE, porque os peers de IPsec e os pontos finais do túnel GRE (os roteadores) são os mesmos e o modo de transporte salvará 20 bytes de overhead de IPsec.

Um caso interessante é quando um pacote de IP é dividido em dois fragmentos e encapsulado pelo GRE. Neste caso o IPsec verá dois pacotes independentes GRE +IP. Frequentemente em uma configuração padrão uma destes pacotes seja grande bastante que deverão ser fragmentados depois que foram cifrados. O ipsec peer terá que remontar este pacote antes da descriptografia. Esta “dupla fragmentação” (uma vez antes do GRE e outra vez depois que IPsec) no roteador de emissão aumenta a latência e abaixa a taxa de transferência. Também, a remontagem é comutado por processo, tão lá será uns acertos da CPU no roteador de recepção sempre que esta acontece.

Esta situação pode ser evitada ajustando “o MTU IP” na interface do túnel GRE baixo bastante para levar em consideração as despesas gerais do GRE e do IPsec (a interface do túnel GRE “MTU IP” é ajustada à revelia à interface real que parte MTU - bytes da carga adicional de GRE).

Esta tabela alista os valores sugeridos MTU para cada túnel/combinação de modo que supõe que a relação de física de saída tem um MTU de 1500.

Combinação de Túneis	MTU específico necessário	MTU recomendado
GRE + IPsec (modo de transporte)	1440 bytes	1400 bytes
GRE + IPsec (modo de túnel)	1420 bytes	1400 bytes

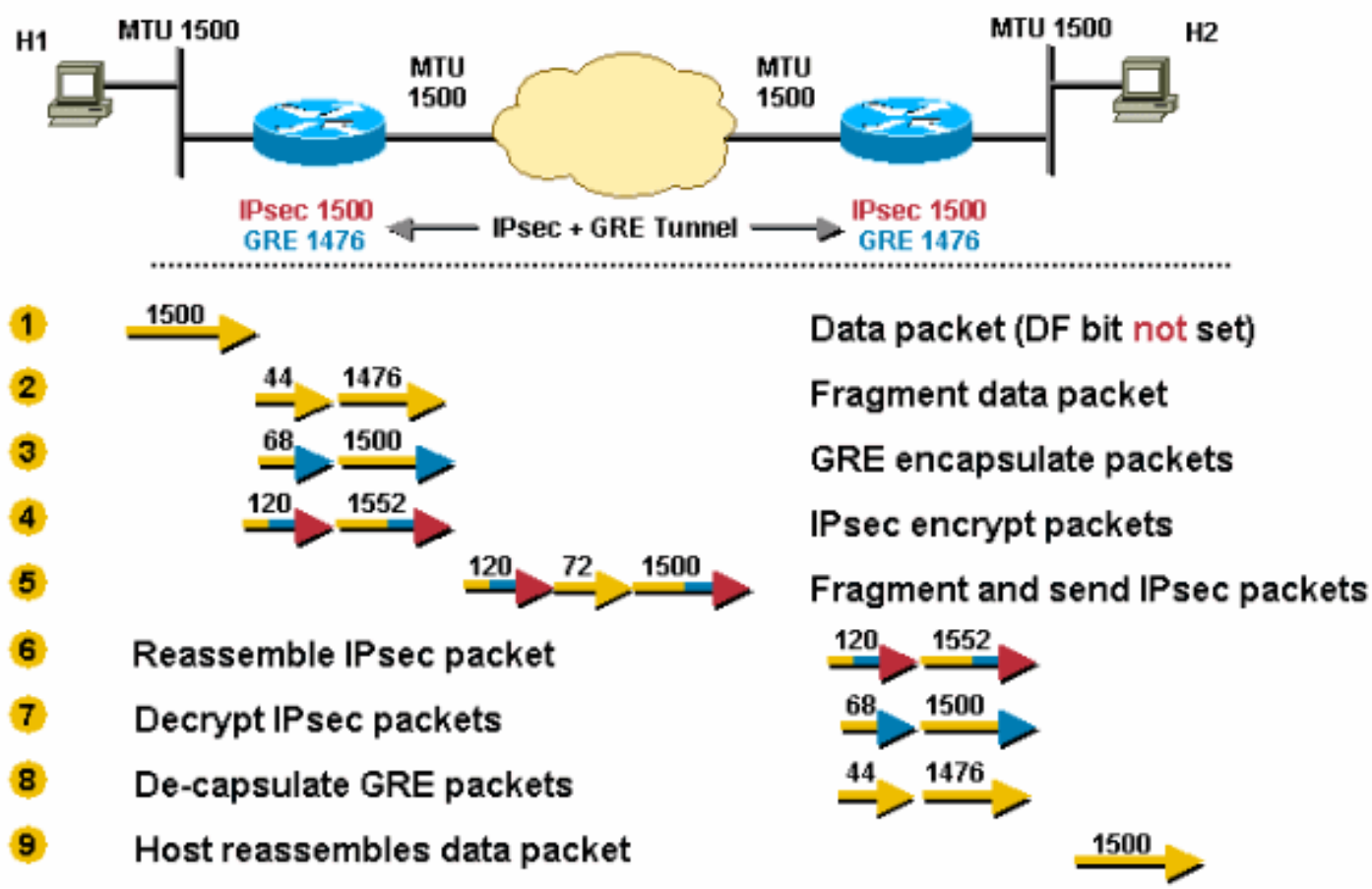
Nota: O valor MTU de 1400 é recomendado porque cobre o mais comum combinações de modo GRE + de IPsec. Também, não há nenhuma redução discernível a permitir uns 20 ou

40 bytes extra em cima. É mais fácil recordar quase e ajustar todos os cenários das tampas de um valor e deste valor.

Cenário 9

O IPsec é distribuído sobre o GRE. O MTU físico de saída é 1500, o PMTU de IPSec é 1500 e o MTU de IP do GRE é 1476 (1500 - 24 = 1476). Devido a isto, os pacotes TCP/IP serão fragmentados duas vezes, uma vez antes do GRE e uma vez após o IPsec. O pacote será fragmentado antes do encapsulamento GRE e um desses pacotes GRE será fragmentado novamente após a criptografia Ipsec.

Configurar “MTU 1440" IP (modo transporte de IPpsec) ou “MTU 1420" IP (modo do túnel de IPsec) no túnel GRE removeria a possibilidade de dupla fragmentação nesta encenação.



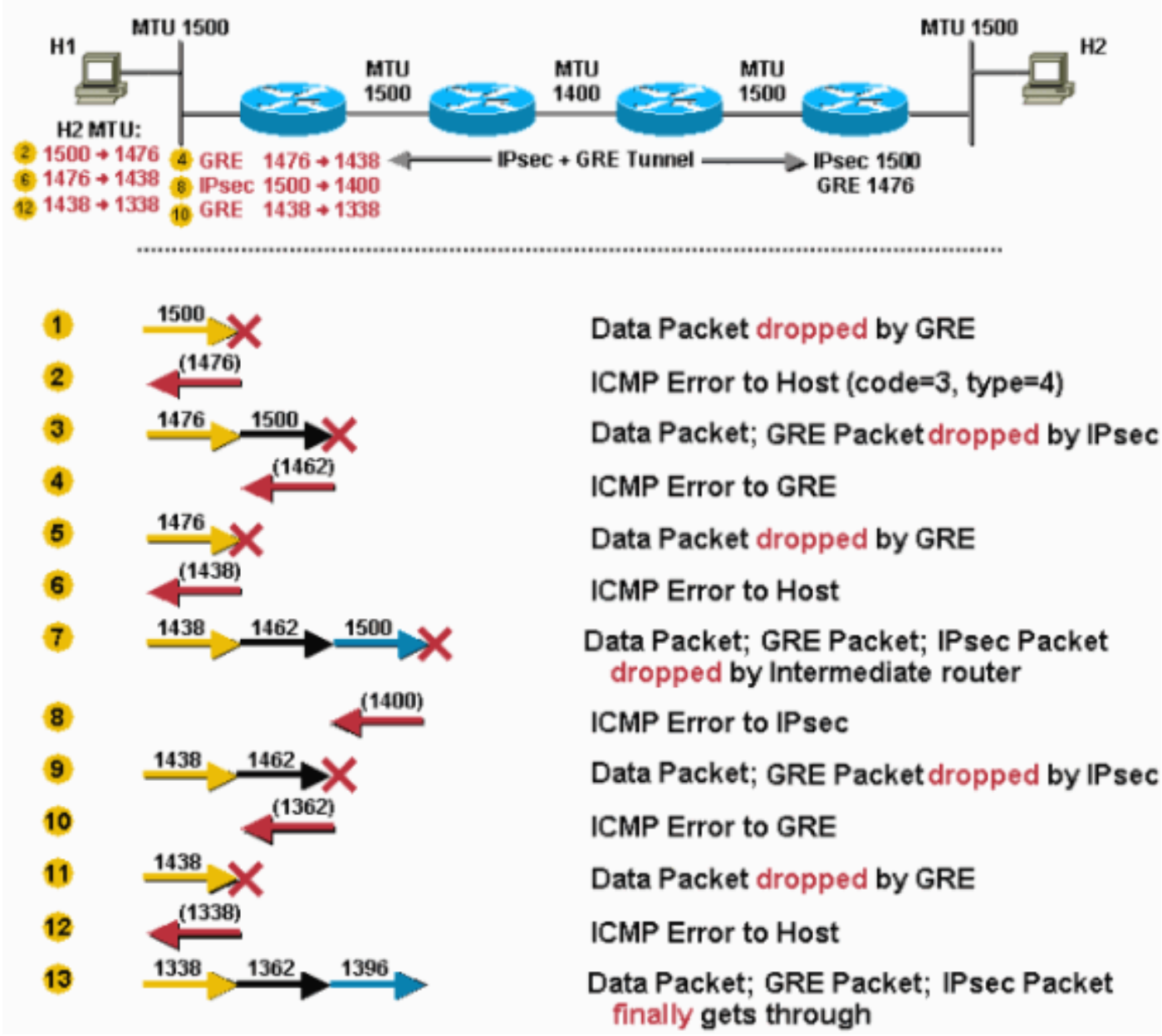
1. O roteador recebe um datagrama de 1.500 bytes.
2. Antes do encapsulamento, o GRE fragmenta o pacote 1500-byte em dois 1476 (1500 - 24 = 1476) e 44 (24 dados + cabeçalho IP 20) bytes das partes.
3. O GRE encapsula os fragmentos IP, que adiciona 24 bytes a cada pacote. Isso resulta em dois pacotes GRE + IPsec de 1500 (1476 + 24 = 1500) e 68 (44 + 24) bytes cada.
4. O IPsec cifra os dois pacotes, adicionando 52 byes (modo de túnel do IPsec) da carga adicional de encapsulamento a cada um, a fim dar um 1552-byte e um pacote do 120-byte.
5. O pacote de IPsec 1552-byte é fragmentado pelo roteador porque é maior do que o MTU de partida (1500). O pacote de 1552 bytes é dividido em pedaços, um pacote de 1500 bytes e outro pacote de 72 bytes (52 bytes de "payload" mais um cabeçalho de IP adicional de 20 bytes para o segundo fragmento). Os três pacotes de 1500 bytes, 72 bytes e 120 bytes são encaminhados para o peer IPsec + GRE.

6. O roteador de recepção remonta os dois fragmentos do IPsec (1500 bytes e 72 bytes) a fim obter o pacote original do IPsec+GRE 1552-byte. Nada precisa de ser feito ao pacote do IPsec+GRE do 120-byte.
7. O IPsec decifra 1552-byte e pacotes do IPsec+GRE do 120-byte a fim obter os pacotes GRE 1500-byte e 68-byte.
8. GRE sem capsulamento os pacotes GRE 1500-byte e 68-byte a fim obter fragmentos do pacote IP 1476-byte e 44-byte. Esses fragmentos de pacotes IP são encaminhados ao host de destino.
9. O host 2 remonta estes fragmentos IP a fim obter o IP datagram 1500-byte original.

O Cenário 10 é semelhante ao Cenário 8, exceto pela existência de um link MTU mais baixo no caminho de túnel. Esse é o "pior" cenário para o primeiro pacote enviado do Host 1 para o Host 2. Após a última etapa nesta encenação, o host 1 ajusta o PMTU correto para o host 2 e tudo é bem para as conexões de TCP entre o host 1 e os fluxos de TCP do host 2. entre o host 1 e os outros anfitriões (alcançáveis através do túnel do IPsec+GRE) terão que somente atravessar as últimas três etapas da encenação 10.

Nesta encenação, o **comando tunnel path-mtu-discovery** é configurado no túnel GRE e o bit DF é ajustado nos pacotes TCP/IP que originam do host 1.

Encenação 10



1. O roteador recebe um pacote 1500-byte. Este pacote é deixado cair pelo GRE porque o GRE não pode fragmentar ou enviar o pacote porque o bit DF é ajustado, e o tamanho do pacote excede a interface externa “MTU IP” após ter adicionado a carga adicional de GRE (24 bytes).
2. O roteador envia um mensagem ICMP para hospedar 1 a fim deixá-lo saber que o salto seguinte MTU é 1476 (1500 - 24 = 1476).
3. Hospede 1 muda seu PMTU para o host 2 1476 e envia o tamanho menor quando retransmite o pacote. O GRE encapsular-lo e entrega-o o pacote 1500-byte ao IPsec. O IPsec deixa cair o pacote porque o GRE copiou o DF mordido (ajuste) do cabeçalho IP interno, e com a carga adicional de IPsec (máximo 38 bytes), o pacote é demasiado grande enviar para fora a interface física.
4. O IPsec envia um mensagem ICMP ao GRE que indica que o salto seguinte MTU é 1462 bytes (desde que um máximo 38 bytes será adicionado para a criptografia e o IP em cima). O GRE grava o valor 1438 (1462 - 24) como “o MTU IP” na interface de túnel.Nota: Esta mudança no valor é armazenada internamente e não pode ser considerada na saída do **comando interface tunnel<-> da mostra IP**. Você verá essa alteração se acionar o comando use debug tunnel.
5. A próxima vez o host 1 retransmite o pacote 1476-byte, GRE deixa-o cair.

6. O roteador envia um mensagem ICMP para hospedar 1 que indica que 1438 são o salto seguinte MTU.
7. O host 1 abaixa o PMTU para o host 2 e retransmite um pacote 1438-byte. Esta vez, o GRE aceita o pacote, encapsular-lo, e entrega-o fora ao IPsec para a criptografia. O pacote IPSec é encaminhado para o roteador intermediário e descartado porque tem uma MTU de interface de saída de 1.400.
8. O roteador intermediário envia um mensagem ICMP ao IPsec que lhe diz que o salto seguinte MTU é 1400. Este valor é gravado pelo IPsec no valor PMTU IPsec associado SA.
9. Quando o Host 1 retransmite o pacote de 1.438 bytes, o GRE o encapsula e entrega-o ao IPSec. O IPsec deixa cair o pacote porque mudou seu próprio PMTU a 1400.
10. O IPsec envia um erro ICMP ao GRE que indica que o salto seguinte MTU é 1362, e o GRE grava o valor 1338 internamente.
11. Quando o host 1 retransmitir o pacote original (porque não recebeu um reconhecimento), o GRE deixa-o cair.
12. O roteador envia um mensagem ICMP para hospedar 1 que indica que o salto seguinte MTU é 1338 (1362 - 24 bytes). O host 1 abaixa seu PMTU para o host 2 1338.
13. O host 1 retransmite um pacote 1338-byte e esta vez onde possa finalmente conseguir por completo hospedar 2.

Outras recomendações

Configurar o **comando tunnel path-mtu-discovery em uma** interface de túnel pode ajudar o GRE e a interação de IPsec quando são configurados no mesmo roteador. Lembre-se que sem o comando tunnel path-mtu-discovery configurado, o bit DF sempre seria apagado no cabeçalho IP GRE. Isto permite o pacote IP GRE seja fragmentado mesmo que o cabeçalho IP dos dados encapsulados tenha o jogo do bit DF, que normalmente não permitiria que o pacote fosse fragmentado.

Se o **comando tunnel path-mtu-discovery** é configurado na interface do túnel GRE, esta acontecerá.

1. O GRE copiará o DF mordido do cabeçalho IP dos dados ao cabeçalho IP GRE.
2. Se o bit DF é ajustado no cabeçalho IP GRE e o pacote será “demasiado grande” após a criptografia IPSec para o IP MTU na interface enviada física, a seguir o IPsec deixará cair o pacote e notificará o túnel GRE para reduzir seu tamanho do MTU IP.
3. O IPsec faz o PMTUD para seus próprios pacotes e se o IPsec PMTU muda (se está reduzido), a seguir IPsec não notifica imediatamente o GRE, mas quando um outro “demasiado grande” pacote vem completo, a seguir o processo em etapa 2 ocorre.
4. O IP MTU do GRE é agora menor, assim que deixará cair todos os pacotes IP dos dados com o jogo do bit DF que forem agora demasiado grandes e enviará um mensagem ICMP ao host de emissão.

O comando tunnel path-mtu-discovery ajuda a interface GRE a definir seu MTU IP dinamicamente, em vez de estatisticamente com o comando ip mtu. Recomenda-se realmente que os comandos both estão usados. **O comando ip mtu** é usado fornecer a sala para o GRE e a carga adicional de IPsec relativo ao IP MTU da interface enviada do local físico. O comando tunnel path-mtu-discovery permite que o MTU IP do túnel GRE seja reduzido mais ainda se houver um enlace de MTU do IP no caminho entre os peers de IPsec.

Estão abaixo algumas das coisas que você pode fazer se você está tendo problemas com

PMTUD em uma rede onde haja GRE + túneis de IPsec configurados.

Esta lista começa com a maioria de solução desejável.

- Fixe o problema com o PMTUD que não trabalha, que é causado geralmente por um roteador ou por um Firewall que obstrua o ICMP.
- Use o comando ip tcp adjust-mss nas interfaces de túnel para que o roteador reduza o valor de TCP MSS no pacote TCP SYN. Isso ajudará os dois hosts finais (o TCP emissor e receptor) a usar pacotes suficientemente pequenos, de modo que o PMTUD não seja necessário.
- Use o roteamento de política na interface de ingresso do roteador e configurar um mapa de rota para cancelar o bit DF no cabeçalho IP dos dados antes que obtenha à interface do túnel GRE. Isso permite que o pacote IP de dados seja fragmentado antes do encapsulamento de GRE.
- Aumente “o MTU IP” na interface do túnel GRE para ser igual à interface externa MTU. Isto permitirá que o pacote IP dos dados seja GRE encapsulado sem fragmentá-lo primeiramente. O pacote GRE será então IPsec cifrado e fragmentado então para sair a interface externa física. Neste caso você não configuraria o **comando tunnel path-mtu-discovery na** interface do túnel GRE. Isto pode reduzir drasticamente o throughput porque a remontagem do pacote IP no peer do IPSec é feita no modo de switching de processo.

Informações Relacionadas

- Página de Suporte do IP Routing
- Página de suporte do IPSec (protocolo de segurança IP)
- Calculadora da carga adicional de IPsec (calcule o tamanho do pacote com protocolos do encapsulamento de IPSec)
- Descoberta de MTU de caminho RFC 1191
- Opções de descoberta de MTU RFC 1063 IP
- Protocolo de Internet RFC 791
- Protocolo de controle de transmissão RFC 793
- RFC 879 - O tamanho máximo do segmento de TCP e tópicos relacionados
- RFC 1701 Generic Routing Encapsulation (GRE)
- Esquema 1241 A de RFC para um protocolo de encapsulamento de Internet
- RFC 2003 IP Encapsulation within IP
- Suporte Técnico - Cisco Systems