

Como as manutenções de atividades do GRE funcionam

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Conventions](#)

[Informações de Apoio](#)

[O mecanismo de manutenção de atividade do túnel](#)

[Descrição funcional](#)

[Impacto na memória e no desempenho](#)

[Considerações sobre embalagem](#)

[Comandos e configuração](#)

[Exemplo de formatos de tela e saída](#)

[Informações Relacionadas](#)

[Introduction](#)

Este documento fornece uma visão geral de como os keepalives do Generic Routing Encapsulation (GRE) funcionam.

[Prerequisites](#)

[Requirements](#)

Os leitores deste documento devem estar cientes destes tópicos:

- [Manutenção de atividade do túnel GRE](#)
- [Comandos do modo de configuração de manutenção de atividade](#)

[Componentes Utilizados](#)

As informações neste documento são baseadas nestas versões de software e hardware:

- Cisco 7505 Router
- Software Cisco IOS® que suporta GRE sobre IPSec

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is

live, make sure that you understand the potential impact of any command.

Conventions

For more information on document conventions, refer to the [Cisco Technical Tips Conventions](#).

Informações de Apoio

O recurso de keepalive GRE habilita o comando de interface **keepalive** para túneis e permite configurar keepalives para túneis GRE ponto a ponto. Você pode configurar keepalives com o comando **keepalive** e, opcionalmente, com seu novo ramal.

Os túneis GRE fornecem um método para encapsular pacotes arbitrários dentro de um protocolo de transporte. Eles também oferecem uma arquitetura projetada para fornecer os serviços necessários para implementar qualquer esquema padrão de encapsulamento ponto a ponto. Aqui estão algumas das vantagens dos túneis GRE:

- Os túneis GRE fornecem redes locais multiprotocolo sobre um backbone de protocolo único.
- Os túneis GRE fornecem soluções alternativas para redes que contêm protocolos com contagens limitadas de saltos.
- Os túneis GRE conectam sub-redes descontínuas.
- Os túneis GRE permitem VPNs em WANs.

No entanto, na implementação atual de túneis GRE, um túnel configurado não tem a capacidade de desativar o protocolo de linha de um dos pontos finais do túnel, se a extremidade distante não puder ser alcançada. Assim, o tráfego enviado do túnel é o black-holed e não pode seguir caminhos alternativos porque o túnel sempre permanece ativo.

Essa situação é verdadeira para túneis que dependem de rotas estáticas ou de protocolos de roteamento que agregam rotas para encontrar uma rota para o destino do túnel. Também é verdade em situações em que os dados no plano de controle seguem um caminho diferente dos dados no plano de dados.

O mecanismo de manutenção de atividade do túnel

Esta seção fornece uma descrição funcional para o mecanismo de manutenção de atividade do túnel com a ajuda de um exemplo. Esta seção também lista os elementos de software que este recurso modifica e discute o impacto na memória e no desempenho.

Descrição funcional

O mecanismo de manutenção de atividade do túnel ativa, estende e implementa um comando específico de interface para interfaces de túnel e fornece a capacidade de desativar o protocolo de linha de um túnel. Para obter mais informações, consulte a seção [Comandos e configuração](#).

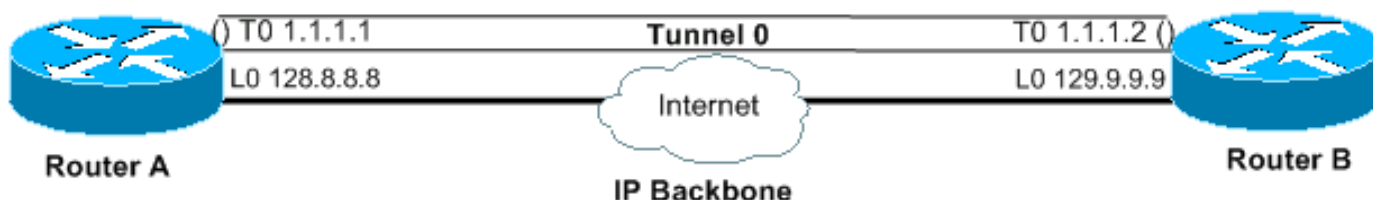
O mecanismo de manutenção de atividade do túnel também aborda estes requisitos adicionais:

- O mecanismo de manutenção de atividade do túnel funciona mesmo se o ponto de extremidade do túnel distante não suportar keepalives.
- O mecanismo de manutenção de atividade de túnel origina keepalives.

- O mecanismo de manutenção de atividade do túnel processa keepalives.
- O mecanismo de manutenção de atividade do túnel responde aos pacotes de manutenção de atividade da extremidade distante, mesmo quando o protocolo de linha do túnel está inativo.

Aqui está um exemplo de como o mecanismo de manutenção de atividade do túnel funciona (consulte a [Figura 1](#)):

Figura 1: Exemplo do mecanismo de manutenção de atividade do túnel



Saída

```
interface tunnel 0
ip address 1.1.1.1 255.255.255.240
tunnel source 128.8.8.8
tunnel destination 129.9.9.9
keepalive 5 4
interface loopback 0
ip address 128.8.8.8 255.255.255.255

interface tunnel 0
ip address 1.1.1.2 255.255.255.240
tunnel source 129.9.9.9
tunnel destination 128.8.8.8
keepalive 5 4
interface loopback 0
ip address 129.9.9.9 255.255.255.255
```

Um pacote keepalive que se origina de A a B

```
---outer IP header---'      ---inner IP header---'
=====
|IP | IP src | IP dst | GRE | IP | IP src | IP dst | GRE |
|  |128.8.8.8|129.9.9.9|PT=IP|   |129.9.9.9|128.8.8.8| PT=0|
=====
          -----'          ----'
                   GRE header                   GRE header
```

Quando você ativa keepalives no ponto final do túnel do Roteador A, o roteador em cada intervalo constrói o cabeçalho IP interno. No final do cabeçalho, o roteador também acrescenta um cabeçalho GRE com um tipo de protocolo (PT) igual a 0, e nenhum outro payload. Em seguida, o roteador envia esse pacote pelo túnel, o que resulta em seu encapsulamento com o cabeçalho IP externo, e um cabeçalho GRE com o PT de IP. O contador de keepalive do túnel é incrementado em um. Se houver uma maneira de alcançar o ponto de extremidade distante do túnel e o protocolo de linha do túnel não estiver inativo devido a outros motivos, o pacote chega ao Roteador B. Em seguida, ele é comparado ao túnel 0, é desencapsulado e encaminhado ao IP de destino, que é a origem do túnel, o roteador A. Na chegada ao Roteador A, o pacote é novamente desencapsulado e o PT é verificado. Se o resultado da verificação PT for 0, significa que este é um pacote keepalive. Nesse caso, o contador de manutenção de atividade do túnel é redefinido para 0 e o pacote é descartado.

Caso o Roteador B não possa ser alcançado, o Roteador A continua a construir e enviar os pacotes keepalive junto com o tráfego normal. Se o protocolo de linha estiver inativo, os keepalives não voltarão para o Roteador A. Portanto, o contador keepalive continua a aumentar. O protocolo de linha de túnel permanece ativo somente enquanto o contador de manutenção de atividade de túnel permanecer zero, ou menor que um valor configurado. Se essa condição não for verdadeira, na próxima vez que você tentar enviar uma keepalive para o Roteador B, o protocolo de linha será desativado, assim que o contador keepalive alcançar o valor de keepalive

configurado. No estado up/down, o túnel não encaminha nem processa nenhum tráfego além dos pacotes keepalive. Para que isso funcione somente para pacotes keepalive, o túnel deve ser amigável de encaminhamento e recebimento. Portanto, o algoritmo de pesquisa de túnel deve ser bem-sucedido em todos os casos e deve descartar apenas os pacotes de dados se o protocolo de linha estiver inativo. Quando um pacote keepalive é recebido, isso implica que o ponto final do túnel é novamente alcançável. O contador de manutenção de atividade do túnel é então redefinido como 0, e o protocolo de linha volta a funcionar.

Impacto na memória e no desempenho

O recurso quase não impõe nenhuma demanda adicional na memória do sistema do roteador e espera-se que o desempenho não seja afetado pela sua adição. Os pacotes keepalive são tratados como pacotes comuns e, portanto, é possível que eles possam ser descartados em condições de tráfego intenso. Por enquanto, você pode alterar o número de novas tentativas para lidar com esse problema. Se isso se revelar inadequado no final, você poderá colocar pacotes keepalive gerados localmente em uma fila de alta prioridade para transmissão. Você pode então definir o valor TOS nos cabeçalhos IP para um valor mais adequado, diferente do valor padrão ou configurado.

Considerações sobre embalagem

O recurso está incluído no código de túnel IP básico e no subsistema GRE. Portanto, ele deve estar disponível com um pacote IP básico que tenha o túnel e os subsistemas GRE.

Comandos e configuração

Esta seção aborda o comando **keepalive** ativado e estendido por este recurso somente na ID de bug Cisco CSCuk26449. Outros comandos estão documentados nos *respectivos Guias de Configuração do Cisco IOS e Referências de Comando*. O comando **[no] keepalive <period> <retries>** está ativado e estendido com um segundo parâmetro e está disponível no Cisco IOS Software Release 12.2(8)T e posterior. Ele também foi portado sob o bug da Cisco ID CSCuk29980 e CSCuk29983 para o Cisco IOS Software Releases 12.1E e 12.2S.

Como o **keepalive** é um comando de configuração de interface que permite keepalives na interface de túnel, apenas keepalives para o modo GRE/IP são suportados atualmente. O segundo parâmetro do comando (*novas tentativas*) está visível e disponível somente para interfaces de túnel. Os valores do primeiro parâmetro podem variar de 1 a 32767. Quando o valor é 0, é equivalente a "no keepalive". Esse parâmetro tem um valor padrão de 10. Os valores para o segundo parâmetro podem variar de 1 a 255, e indica o número de keepalives que são enviados mas não retornados, após o que a interface de túnel retira o protocolo de linha. As manutenções de atividade em interfaces de túnel são desativadas por padrão.

Exemplo de formatos de tela e saída

Esta seção fornece saídas de exemplo.

```
cisco-7505#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
cisco-7505(config)#interface tunnel 1
cisco-7505(config-if)#?
    access-expression    Build a bridge boolean access expression
```

```

.....
keepalive          Enable keepalive<=====
.....
timeout           Define timeout values for this interface

cisco-7505(config-if)#keepalive ?<=====
<0-32767>  Keepalive period (default 10 seconds)

cisco-7505(config-if)#keepalive 5 ?<=====
<1-255>    Keepalive retries (default 3 times)
cisco-7505(config-if)#keepalive 5 4<=====
cisco-7505(config-if)#end

cisco-7505#show interfaces tunnel 1

Tunnel1 is up, line protocol is up
Hardware is Tunnel
Internet address is 10.1.1.1/24
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec, rely 255/255, load 1/255
Encapsulation TUNNEL, loopback not set
Keepalive set (5 sec), retries 4<=====
Tunnel source 9.2.2.1, destination 6.6.6.2
Tunnel protocol/transport GRE/IP, key disabled, sequencing disabled
Tunnel TOS 0xF, Tunnel TTL 128
Checksumming of packets disabled, fast tunneling enabled
Last input never, output 00:57:05, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/0, 1 drops; input queue 0/75, 0 drops
30 second input rate 0 bits/sec, 0 packets/sec
30 second output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  3 packets output, 1860 bytes, 0 underruns
  0 output errors, 0 collisions, 0 interface resets
  0 output buffer failures, 0 output buffers swapped out

```

[Informações Relacionadas](#)

- [Manutenção de atividade do túnel GRE \(Generic Routing Encapsulation\)](#)
- [Configurações de exemplo de GRE](#)
- [Suporte técnico e documentação](#)