Funktionsweise von GRE-Keepalives

Inhalt

Einführung

Dieses Dokument bietet eine Übersicht über die Funktionsweise von GRE-Keepalives (Generic Routing Encapsulation).

Voraussetzungen

Anforderungen

Die Leser dieses Dokuments sollten folgende Themen kennen:

- GRE-Tunnel-Keepalive
- Befehle des Keepalive-Konfigurationsmodus

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf den folgenden Software- und Hardwareversionen:

- Cisco Router der Serie 7505
- Cisco IOS® Software, die GRE über IPSec unterstützt

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie

die potenziellen Auswirkungen eines Befehls verstehen.

Konventionen

Weitere Informationen zu Dokumentkonventionen finden Sie in den Cisco Technical Tips Conventions.

Hintergrundinformationen

Die GRE-Keepalive-Funktion aktiviert den **Keepalive-**Schnittstellenbefehl für Tunnel und ermöglicht die Konfiguration von Keepalives für Point-to-Point-GRE-Tunnel. Sie können Keepalives mit dem **keepalive-**Befehl und optional mit der neuen Erweiterung konfigurieren.

GRE-Tunnel bieten eine Methode, um beliebige Pakete innerhalb eines Transportprotokolls zu kapseln. Sie bieten außerdem eine Architektur, die die Services bereitstellt, die für die Implementierung eines standardmäßigen Point-to-Point-Kapselungsschemas erforderlich sind. Hier einige der Vorteile von GRE-Tunneln:

- GRE-Tunnel stellen lokale Multi-Protocol-Netzwerke über einen einzigen Protokoll-Backbone bereit.
- GRE-Tunnel stellen Workarounds für Netzwerke bereit, die Protokolle mit begrenzten Hop-Zählungen enthalten.
- GRE-Tunnel verbinden separate Subnetze.
- GRE-Tunnel ermöglichen VPNs über WANs hinweg.

Bei der aktuellen Implementierung von GRE-Tunneln kann ein konfigurierter Tunnel jedoch das Leitungsprotokoll eines Tunnelendpunkts nicht deaktivieren, wenn das Gegenstück nicht erreichbar ist. Der aus dem Tunnel gesendete Verkehr ist also schwarz gehalten, und er kann keine alternativen Pfade folgen, da der Tunnel immer in Betrieb ist.

Dies gilt für Tunnel, die auf statischen Routen oder Routing-Protokollen basieren, die Routen aggregieren, um eine Route zum Tunnelziel zu finden. Dies gilt auch für Situationen, in denen die Daten auf der Kontrollebene einem anderen Pfad folgen als die Daten auf der Datenebene.

Der TunnelKeepalive-Mechanismus

Dieser Abschnitt enthält eine Funktionsbeschreibung des Mechanismus für die TunnelKeepalive-Funktion mithilfe eines Beispiels. In diesem Abschnitt werden auch die Softwareelemente aufgeführt, die durch diese Funktion geändert werden, sowie die Auswirkungen auf Arbeitsspeicher und Leistung.

Funktionsbeschreibung

Der Tunnel-Keepalive-Mechanismus aktiviert, erweitert und implementiert einen schnittstellenspezifischen Befehl für Tunnelschnittstellen und ermöglicht das Herunterfahren des Leitungsprotokolls eines Tunnels. Weitere Informationen finden Sie im Abschnitt Befehle und Konfiguration.

Der Tunnel-Keepalive-Mechanismus erfüllt darüber hinaus folgende zusätzliche Anforderungen:

- Der Tunnelkeepalive-Mechanismus funktioniert auch dann, wenn der Tunnelendpunkt Keepalives nicht unterstützt.
- Der Tunnelkeepalive-Mechanismus erzeugt Keepalives.
- Der Tunnelkeepalive-Mechanismus verarbeitet Keepalives.
- Der Tunnel-Keepalive-Mechanismus antwortet auf Keepalive-Pakete des Gegenstandes, selbst wenn das Leitungsprotokoll des Tunnels ausgefallen ist.

Im Folgenden sehen Sie ein Beispiel für die Funktionsweise des Tunnelkeepalive-Mechanismus (siehe Abbildung 1):

Abbildung 1: Beispiel für einen Keepalive-Mechanismus

Ausgabe

```
interface tunnel 0                          interface tunnel 0
ip address 1.1.1.1 255.255.255.240          ip address 1.1.1.2 255.255.255.240
tunnel source 128.8.8.8                     tunnel source 129.9.9.9
tunnel destination 129.9.9.9                tunnel destination 128.8.8.8
keepalive 5 4                               keepalive 5 4
interface loopback 0                        interface loopback 0
ip address 128.8.8.8 255.255.255.255       ip address 129.9.9.9 255.255.255.255
```

Ein Keepalive-Paket, das von A bis B stammt

```
---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
```

Wenn Sie Keepalives auf dem Tunnel-Endpunkt von Router A aktivieren, erstellt der Router in jedem Intervall den inneren IP-Header. Am Ende des Headers fügt der Router auch einen GRE-Header mit dem Protokoll-Typ (PT) 0 und keine andere Payload an. Der Router sendet das Paket dann durch den Tunnel. Dies führt zur Kapselung mit dem äußeren IP-Header und einem GRE-Header mit dem PT der IP. Der Zähler für den Tunnel-Keepalive erhöht sich um eins. Wenn es eine Möglichkeit gibt, den Tunnelendpunkt am anderen Ende zu erreichen, und das Tunnelleitungsprotokoll aus anderen Gründen nicht heruntergefahren ist, erreicht das Paket den Router B. Er wird dann mit Tunnel 0 abgeglichen, entkapselt und an die Ziel-IP weitergeleitet, die Tunnelquelle, Router A. Bei der Ankunft auf Router A wird das Paket erneut entkapselt und die PT überprüft. Wenn das Ergebnis der PT-Prüfung 0 ist, bedeutet dies, dass es sich um ein Keepalive-Paket handelt. In diesem Fall wird der Zähler für die Keepalive-Keepalive-Verbindung auf 0 zurückgesetzt, und das Paket wird verworfen.

Falls Router B nicht erreichbar ist, wird Router A weiterhin die Keepalive-Pakete zusammen mit dem normalen Datenverkehr erstellen und senden. Wenn das Leitungsprotokoll ausgefallen ist, werden die Keepalives nicht zu Router A zurückgeleitet. Der Keepalive-Zähler nimmt daher weiter

zu. Das Tunnelleitungsprotokoll bleibt nur so lange verfügbar, wie der Tunnel-Keepalive-Zähler 0 bleibt oder weniger als ein konfigurierter Wert. Wenn diese Bedingung nicht zutrifft, wird das Leitungsprotokoll beim nächsten Versuch, einen Keepalive an Router B zu senden, deaktiviert, sobald der Keepalive-Zähler den konfigurierten Keepalive-Wert erreicht. Im Ein-/Ausschaltzustand leitet oder verarbeitet der Tunnel außer den Keepalive-Paketen keinen Datenverkehr weiter. Damit dies nur bei Keepalive-Paketen funktioniert, muss der Tunnel eine Vor-und-Empfangen-freundliche Funktion aufweisen. Der Tunnelsuchalgorithmus muss daher in allen Fällen erfolgreich sein und nur die Datenpakete verwerfen, wenn das Verbindungsprotokoll ausgefallen ist. Wenn ein Keepalive-Paket empfangen wird, impliziert dies, dass der Tunnel-Endpunkt wieder erreichbar ist. Der Tunnel-Keepalive-Zähler wird dann auf 0 zurückgesetzt, und das Line-Protokoll wird wieder aktiviert.

Speicher- und Performance-Auswirkungen

Durch diese Funktion wird der Systemspeicher des Routers fast nicht zusätzlich beansprucht, und die Leistung wird durch die Hinzufügung voraussichtlich nicht beeinträchtigt. Keepalive-Pakete werden als normale Pakete behandelt, sodass sie unter hohen Verkehrsbedingungen verworfen werden können. Im Moment können Sie die Anzahl der Wiederholungsversuche für dieses Problem ändern. Sollte sich dies als unzulänglich erweisen, können Sie lokal generierte Keepalive-Pakete in eine Warteschlange mit hoher Priorität zur Übertragung stellen. Sie können dann den TOS-Wert in den IP-Headern auf einen geeigneteren Wert festlegen, der nicht dem Standard- oder dem konfigurierten Wert entspricht.

Verpackungsüberlegungen

Diese Funktion ist im grundlegenden IP-Tunnel-Code und im GRE-Subsystem enthalten. Daher muss es mit einem grundlegenden IP-Paket verfügbar sein, das den Tunnel und die GRE-Subsysteme hat.

Befehle und Konfiguration

In diesem Abschnitt wird der **Keepalive-**Befehl behandelt, der durch diese Funktion aktiviert und erweitert wurde, nur unter der Cisco Bug-ID CSCuk26449. Weitere Befehle sind in den *entsprechenden Cisco IOS-Konfigurationsanleitungen und Befehlsreferenzen* dokumentiert. Der **[no] keepalive <*period*> <*retries*>**-Befehl wird mit einem zweiten Parameter aktiviert und erweitert und ist in Cisco IOS Software Version 12.2(8)T und höher verfügbar. Es wurde auch unter der Cisco Bug-ID CSCuk29980 und CSCuk29983 auf die Cisco IOS Software Releases 12.1E und 12.2S portiert.

Da **keepalive** ein Schnittstellenkonfigurationsbefehl ist, der Keepalives auf der Tunnelschnittstelle aktiviert, werden derzeit nur Keepalives für den GRE/IP-Modus unterstützt. Der zweite Parameter des Befehls (**erneute *Versuche***) ist sichtbar und nur für Tunnelschnittstellen verfügbar. Die Werte des ersten Parameters können zwischen 1 und 32767 liegen. Wenn der Wert 0 ist, entspricht er "no keepalive". Dieser Parameter hat den Standardwert 10. Die Werte für den zweiten Parameter können zwischen 1 und 255 liegen. Sie geben die Anzahl der Keepalives an, die gesendet, aber nicht zurückgegeben werden, nach denen die Tunnelschnittstelle das Leitungsprotokoll abruft. Keepalives auf Tunnelschnittstellen sind standardmäßig deaktiviert.

Beispielformate für Ausgabe und Bildschirm

Dieser Abschnitt enthält Beispielausgaben.

```
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
```

Zugehörige Informationen

- Generic Routing Encapsulation (GRE) Tunnel Keepalive
- GRE-Beispielkonfigurationen
- Technischer Support und Dokumentation