Fehlerbehebung bei intermittierenden Routing-Protokoll-Flaps mit EEM und EPC

Inhalt

Einleitung

In diesem Dokument wird die Fehlerbehebung bei zeitweiligen Routing-Protokoll-Flaps und BFD-Flaps in Cisco IOS® XE mit EEM und EPC beschrieben.

Voraussetzungen

Anforderungen

Es wird empfohlen, sich mit den Einzelheiten des Embedded Event Manager (EEM) und der Embedded Packet Capture (EPC) für die an der Fehlerbehebung beteiligten Plattformen sowie mit Wireshark vertraut zu machen. Darüber hinaus wird empfohlen, sich mit den grundlegenden Hello- und Keepalive-Funktionen für Routing-Protokolle und BFD (Bidirectional Forwarding Detection)

vertraut zu machen.

Verwendete Komponenten

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

Problemübersicht

Intermittierende Routingprotokoll-Flaps sind ein häufiges Problem in Produktionsnetzwerken, können aber aufgrund ihrer Unvorhersehbarkeit in Echtzeit schwer zu beheben sein. EEM bietet die Möglichkeit, die Datenerfassung zu automatisieren, indem die Datenerfassung mit Syslog-Strings ausgelöst wird, wenn es zu Flaps kommt. Mit EEM und EPC können Paketerfassungsdaten von beiden Enden der Adjacency erfasst werden, um potenzielle Paketverluste vor dem Flapping zu isolieren.

Intermittierende Flaps auf Routing-Protokolle beruhen immer auf einem Hello- oder Keepalive-Timeout (sofern es sich nicht um ein eindeutiges physisches Problem handelt, z. B. Link-Flaps, die in den Protokollen angezeigt werden). Dies ist also die Logik, die in diesem Dokument behandelt wird.

Methodik der Fehlerbehebung

Das Wichtigste, um zu bestimmen, wann ein Flapping mit einem Routing-Protokoll auftritt, ist, ob die Hello- oder Keepalive-Pakete zum Zeitpunkt des Problems auf beiden Geräten gesendet und empfangen wurden. Bei dieser Fehlerbehebungsmethode wird ein kontinuierlicher EPC auf einem Ringpuffer verwendet, bis die Klappe auftritt. An diesem Punkt verwendet EEM die entsprechende Syslog-Zeichenfolge, um die Ausführung einer Reihe von Befehlen auszulösen, von denen einer den EPC stoppt. Die Option "Ringpuffer" ermöglicht es dem EPC, neue Pakete zu erfassen, während die ältesten Pakete im Puffer überschrieben werden. Dadurch wird sichergestellt, dass das Ereignis erfasst wird und der Puffer sich nicht füllt und nicht vorher angehalten wird. Die Paketerfassungsdaten können dann mit dem Zeitstempel der Klappe korreliert werden, um festzustellen, ob die erforderlichen Pakete vor dem Ereignis an beiden Enden gesendet und empfangen wurden.

Dieses Problem tritt in der Regel bei Geräten auf, die eine Adjacency über ein zwischengeschaltetes Netzwerk bilden, z. B. ein Internet Service Provider (ISP). Die gleiche Methode kann jedoch unabhängig von den spezifischen Topologiedetails für jedes Flapping-Szenario mit intermittierenden Routing-Protokollen verwendet werden. Dasselbe kann in Instanzen geschehen, in denen das Nachbargerät von einem Drittanbieter verwaltet wird und kein Zugriff darauf möglich ist. In solchen Fällen kann das in diesem Dokument beschriebene Fehlerbehebungsverfahren nur auf das eine zugängliche Gerät angewendet werden, um zu überprüfen, ob es die erforderlichen Pakete vor der Klappe gesendet und empfangen hat. Wenn dies bestätigt wird, können die Daten dem Partner angezeigt werden, der den Nachbarn verwaltet,

um bei Bedarf weitere Fehlerbehebungen am anderen Ende durchzuführen.

Konfigurationsübersicht

In diesem Abschnitt finden Sie eine Reihe von Konfigurationsvorlagen, mit denen Sie diese automatisierte Datenerfassung einrichten können. Ändern Sie nach Bedarf die IP-Adressen, Schnittstellennamen und Dateinamen.

Vorlage für ACL-Konfiguration

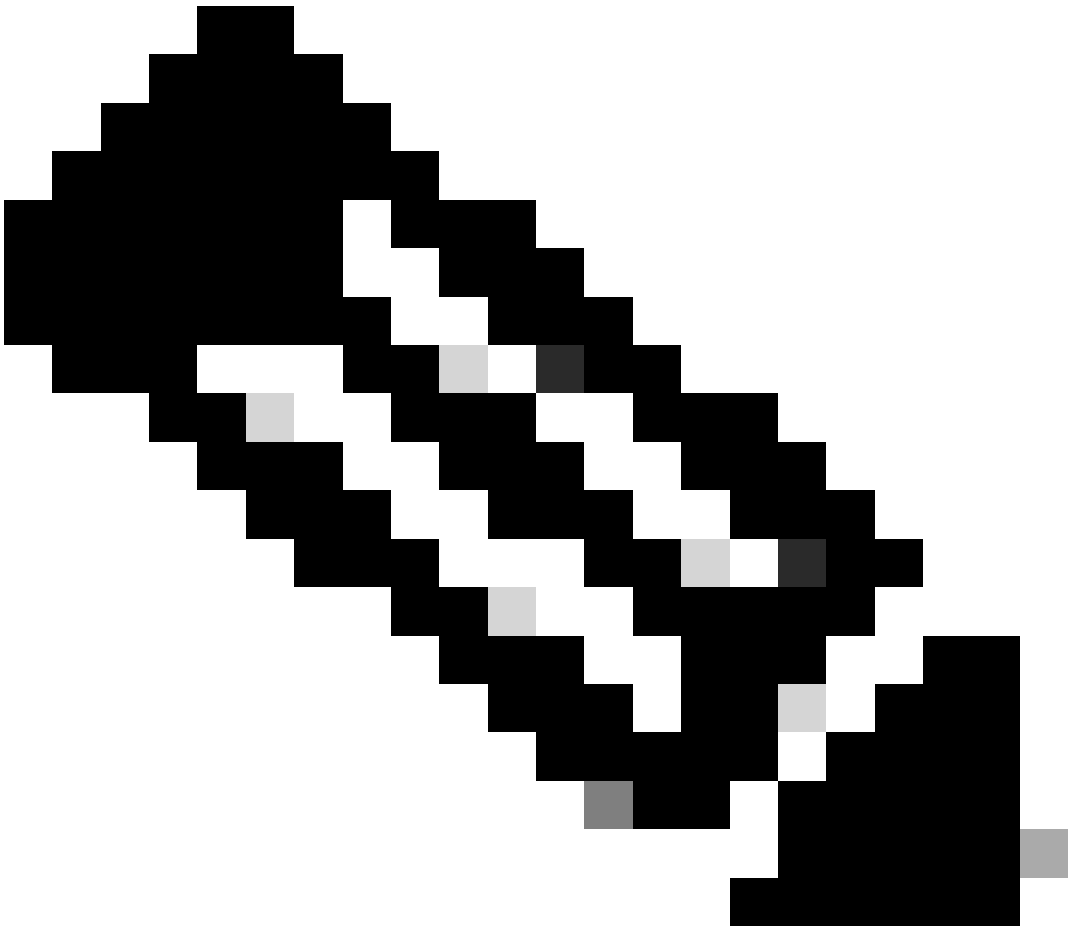
In den meisten Fällen stammt der einzige Datenverkehr, der von der IP-Adresse der Schnittstelle an beiden Enden einer Routing-Adjacency stammt, vom eigentlichen Routing-Kontrollverkehr. Eine ACL, die Datenverkehr von der IP-Adresse der lokalen Schnittstelle und der IP-Adresse des Nachbarn zu einem beliebigen Ziel zulässt, deckt somit die Anforderungen für alle Routing-Protokolle sowie für BFD ab. Wenn ein zusätzlicher Filter erforderlich ist, kann auch die auf dem Routing-Protokoll oder dem BFD-Modus basierende IP-Zieladresse angegeben werden. Definieren Sie die ACL-Parameter im Konfigurationsmodus:

```
config t

ip access-list extended




      permit ip host




       any permit ip host




        any end
```

Vorlage für EPC-Parameter

EPC-Parameter werden im privilegierten exec-Modus erstellt, nicht im Konfigurationsmodus. Überprüfen Sie die plattformspezifischen Konfigurationsleitfäden, um festzustellen, ob Einschränkungen in Bezug auf EPC bestehen. Erstellen Sie die Parameter für die gewünschte Schnittstelle, und ordnen Sie sie der ACL zu, um nach dem gewünschten Datenverkehr zu filtern:

- monitor capture <EPC-Name> interface <Schnittstelle> beide
- monitor capture <EPC-Name> access-list <ACL-Name>
- monitor capture <EPC-Name> buffer size 5, kreisförmig



> Anmerkung: Bei einigen Softwareversionen ist der lokal generierte Datenverkehr bei einem EPC auf Schnittstellenebene nicht sichtbar. In solchen Szenarien können die Erfassungsparameter so geändert werden, dass beide Richtungen des Datenverkehrs an der CPU erfasst werden:

- Kontrollebene von EPC erfassen und

- monitor capture <EPC-Name> access-list <ACL-Name>
- monitor capture <EPC-Name> buffer size 5, kreisförmig

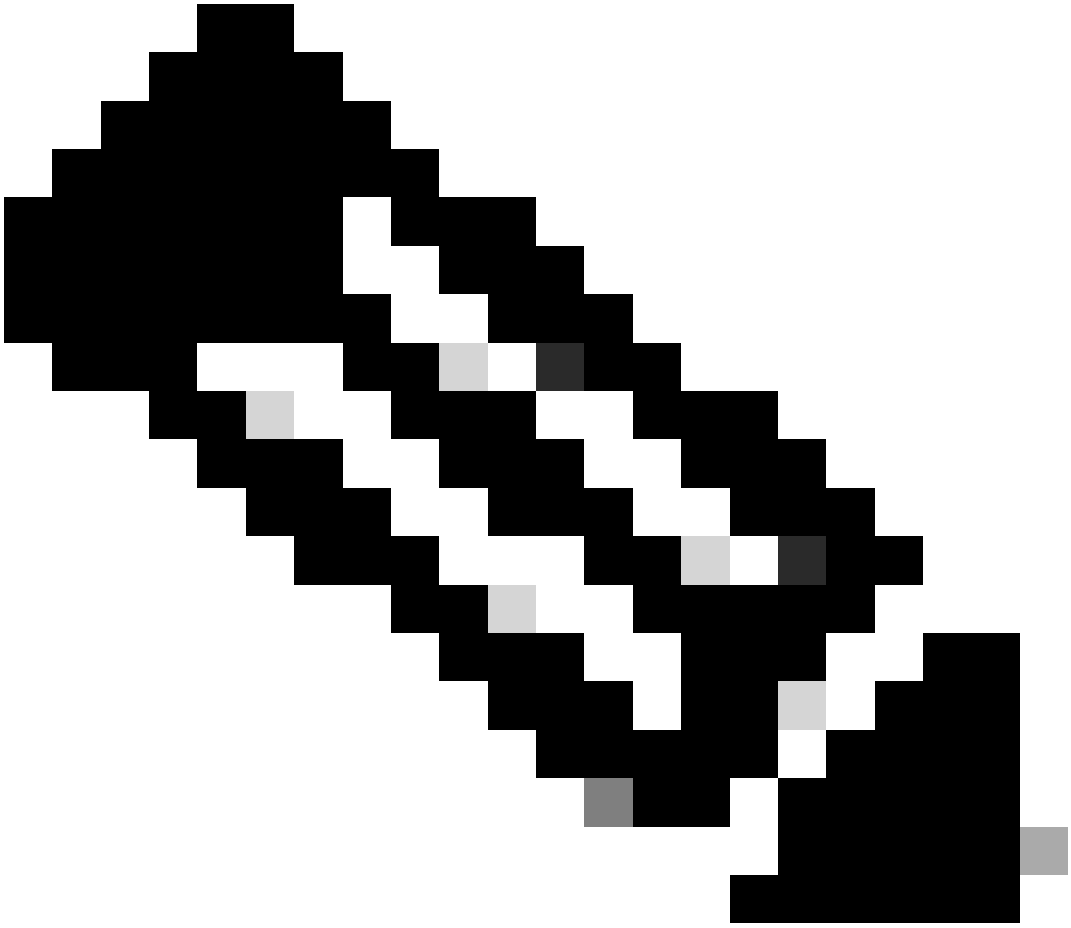
Starten Sie nach der Konfiguration den EPC:

- monitor capture <EPC-Name> start

Der EEM ist so eingestellt, dass die Erfassung gestoppt wird, wenn die Klappe auftritt.

Um sicherzustellen, dass Pakete in beide Richtungen erfasst werden, überprüfen Sie den Erfassungspuffer:

```
show monitor capture




      buffer brief
```



Anmerkung: Für die Catalyst Switching-Plattformen (wie Cat9k und Cat3k) muss die Erfassung beendet werden, bevor der Puffer angezeigt werden kann. Um sicherzustellen, dass die Erfassung funktioniert, beenden Sie die Erfassung mit dem Befehl monitor capture stop, zeigen Sie den Puffer an, und starten Sie ihn dann erneut, um Daten zu sammeln.

EEM-Konfigurationsvorlage

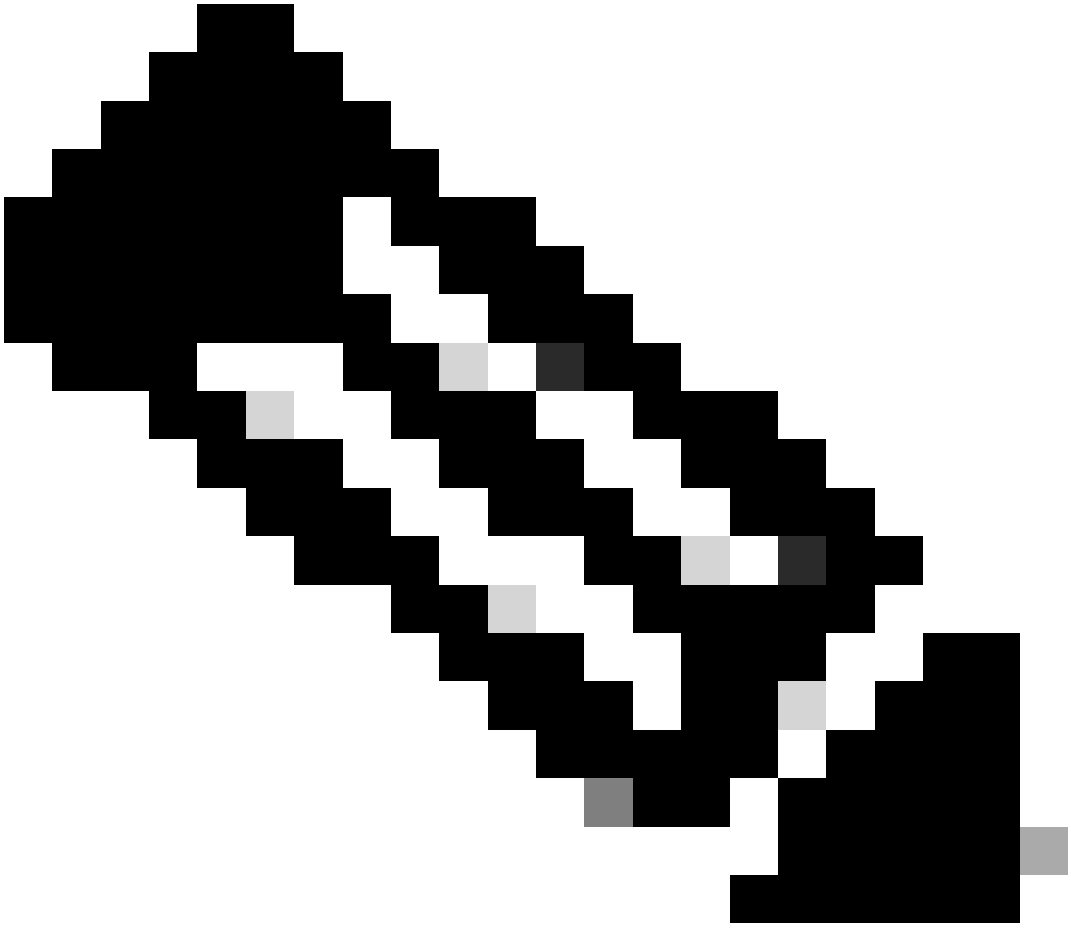
Der Hauptzweck des EEM besteht darin, die Paketerfassung zu stoppen und zusammen mit dem Syslog-Puffer zu speichern. Zusätzliche Befehle können hinzugefügt werden, um andere Faktoren zu überprüfen, z. B. die CPU, das Verwerfen von Schnittstellen oder die plattformspezifische Ressourcennutzung und die Verwerfungszähler. Erstellen Sie das EEM-Applet im Konfigurationsmodus:

```
config t
event manager applet
```

```
authorization bypass event syslog pattern "

 " maxrun 120 ratelimit 100000 action 000 cli command "enable" action 005 cli command "show clock

  .txt" action 010 cli command "show logging | append bootflash:

   .txt" action 015 cli command "show process cpu sorted | append bootflash:

    .txt" action 020 cli command "show process cpu history | append bootflash:

     .txt" action 025 cli command "show interfaces | append bootflash:

      .txt" action 030 cli command "monitor capture

       stop" action 035 cli command "monitor capture

        export bootflash:

         .pcap" action 040 syslog msg "Saved logs to bootflash:
```

.txt and saved packet capture to bootflash:

.pcap" action 045 cli command "end" end



Anmerkung: Auf Catalyst Switching-Plattformen (wie Cat9k und Cat3k) unterscheidet sich der Befehl zum Exportieren der Erfassung geringfügig. Ändern Sie für diese Plattformen den in Aktion 035 verwendeten CLI-Befehl:

```
action 035 cli command "monitor capture

      export location bootflash:

       .pcap"
```

Der ratelimit-Wert im EEM wird in Sekunden angegeben und gibt an, wie viel Zeit vergehen muss, bis der EEM wieder ausgeführt werden kann. In diesem Beispiel ist sie auf 100000 Sekunden (27,8 Stunden) festgelegt, damit der Netzwerkadministrator genügend Zeit hat, um zu erkennen, ob der Vorgang abgeschlossen wurde, und um die Dateien vom Gerät abzurufen, bevor es erneut ausgeführt wird. Wenn der EEM nach dieser Ratenlimitierungsphase wieder selbstständig ausgeführt wird, werden keine neuen Paketerfassungsdaten erfasst, da der EPC manuell gestartet werden muss. Neue Ausgabe des Befehls show wird jedoch an die Textdateien angehängt.

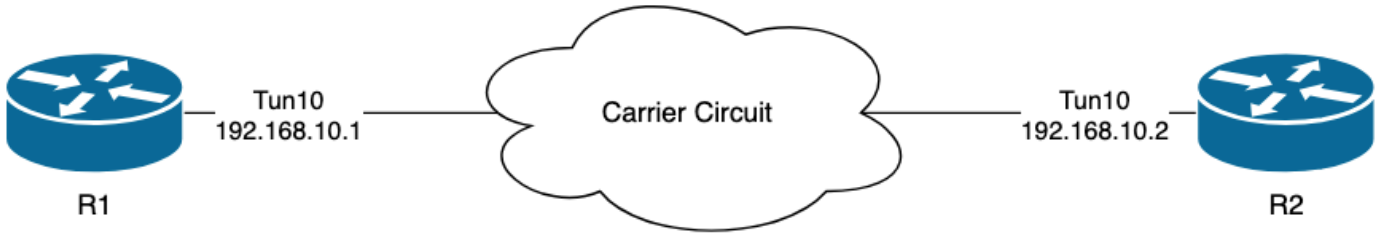
Der EEM kann nach Bedarf geändert werden, um plattformspezifische Paketverwerfungsinformationen zu erfassen und zusätzliche Funktionen für das Szenario zu erhalten.

Fehlerbehebung: Protokoll-Flaps mit unregelmäßigen Abständen

Beispiel: EIGRP

In diesem Beispiel sind alle Timer auf den Standardwert eingestellt (5-Sekunden-Hellos, 15-Sekunden-Haltezeit).

Topologie



Aus den Protokollen auf R1 geht hervor, dass es intermittierende EIGRP-Flaps gegeben hat, die mehrere Stunden voneinander entfernt aufgetreten sind:

```
R1#show logging | i EIGRP
*Jul 16 20:45:08.019: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.2 (Tunnel10) is down: Interf
*Jul 16 20:45:12.919: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.2 (Tunnel10) is up: new adja
*Jul 17 10:25:42.970: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.2 (Tunnel10) is down: holdin
*Jul 17 10:25:59.488: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.2 (Tunnel10) is up: new adja
*Jul 17 14:39:02.970: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.2 (Tunnel10) is down: holdin
*Jul 17 14:39:16.488: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.2 (Tunnel10) is up: new adja
```

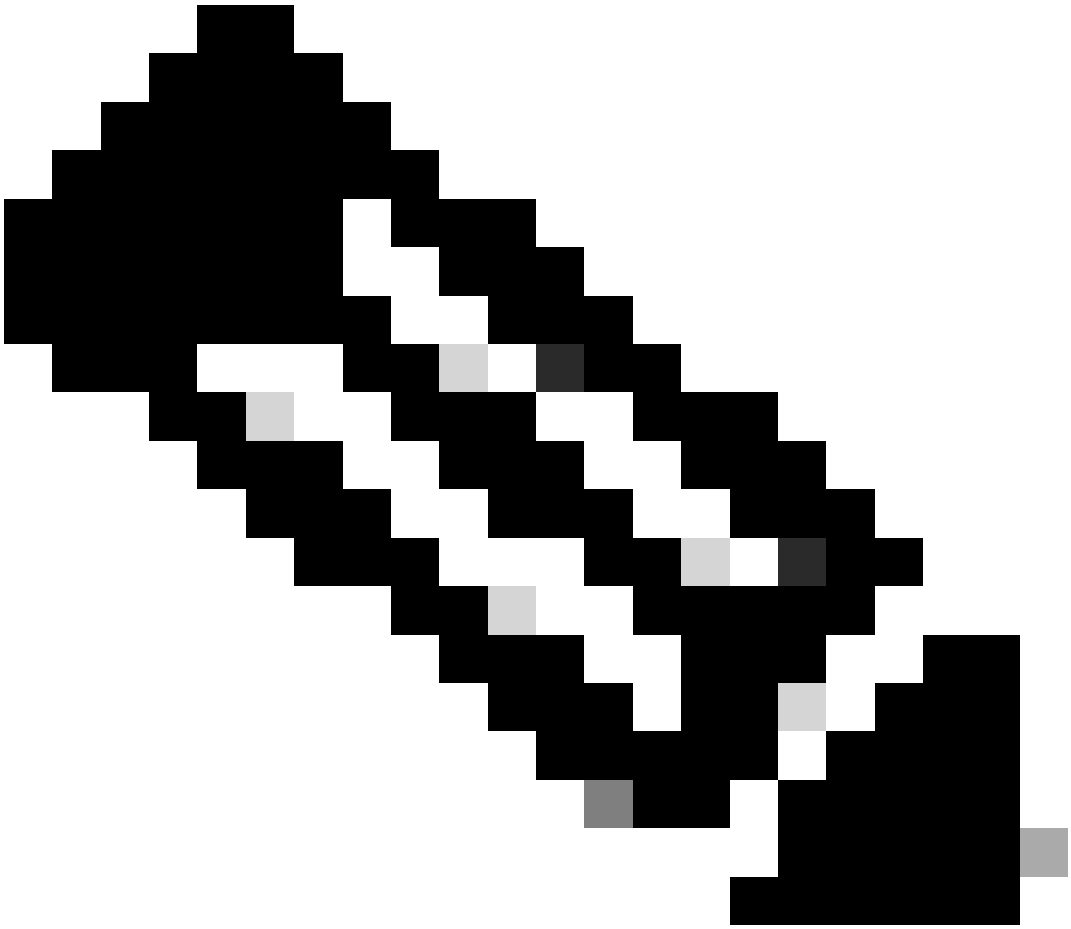
Der Paketverlust kann in beide Richtungen auftreten. Haltezeit abgelaufen zeigt an, dass dieses

Gerät innerhalb der Haltezeit keinen Hello-Anruf vom Peer empfangen oder verarbeitet hat, und dass Interface PEER-TERMINATION received anzeigt, dass der Peer die Adjacency beendet hat, weil er innerhalb der Haltezeit keinen Hello-Anruf empfangen oder verarbeitet hat.

Konfiguration

1. Konfigurieren Sie die ACL mit den IP-Adressen der Tunnelschnittstelle, da dies die Quell-IP-Adressen der Hellos sind:

```
R1#conf t
R1(config)#ip access-list extended FLAP_CAPTURE
R1(config-ext-nacl)#permit ip host 192.168.10.1 any
R1(config-ext-nacl)#permit ip host 192.168.10.2 any
R1(config-ext-nacl)#end
```



Anmerkung: Die gezeigten Konfigurationen stammen aus R1. Das Gleiche gilt für R2 für

---

die relevanten Schnittstellen und mit geänderten Dateinamen für den EEM. Wenn weitere Spezifitäten erforderlich sind, konfigurieren Sie die ACL mit der EIGRP-Multicast-Adresse 224.0.0.10 als Ziel-IP-Adresse, um die Hellos zu erfassen.

---

2. Erstellen Sie den EPC, und ordnen Sie ihn der Schnittstelle und der ACL zu:

```
R1#monitor capture CAP interface Tunnel10 both
R1#monitor capture CAP access-list FLAP_CAPTURE
R1#monitor capture CAP buffer size 5 circular
```

3. Starten Sie den EPC, und bestätigen Sie, dass die Pakete in beide Richtungen erfasst werden:

```
R1#monitor capture CAP start
R1#show monitor capture CAP buffer brief
 ----------------------------------------------------------------------------
 #   size   timestamp     source             destination      dscp    protocol
 ----------------------------------------------------------------------------
   0   74    0.000000   192.168.10.1     ->  224.0.0.10       48 CS6  EIGRP
   1   74    0.228000   192.168.10.2     ->  224.0.0.10       48 CS6  EIGRP
   2   74    4.480978   192.168.10.2     ->  224.0.0.10       48 CS6  EIGRP
   3   74    4.706024   192.168.10.1     ->  224.0.0.10       48 CS6  EIGRP
```

4. Konfigurieren Sie EEM:

```
R1#conf t
R1(config)#event manager applet R1_EIGRP_FLAP authorization bypass
R1(config-applet)#event syslog pattern "%DUAL-5-NBRCHANGE" maxrun 120 ratelimit 100000
R1(config-applet)#action 000 cli command "enable"
R1(config-applet)#action 005 cli command "show clock | append bootflash:R1_EIGRP_FLAP.txt"
R1(config-applet)#action 010 cli command "show logging | append bootflash:R1_EIGRP_FLAP.txt"
R1(config-applet)#action 015 cli command "show process cpu sorted | append bootflash:R1_EIGRP_FLAP.txt"
R1(config-applet)#action 020 cli command "show process cpu history | append bootflash:R1_EIGRP_FLAP.txt
R1(config-applet)#action 025 cli command "show interfaces | append bootflash:R1_EIGRP_FLAP.txt"
R1(config-applet)#action 030 cli command "monitor capture CAP stop"
R1(config-applet)#action 035 cli command "monitor capture CAP export bootflash:R1_EIGRP_CAP.pcap"
R1(config-applet)#action 040 syslog msg "Saved logs to bootflash:R1_EIGRP_FLAP.txt and saved packet cap
R1(config-applet)#action 045 cli command "end"
R1(config-applet)#end
```

5. Warten Sie auf die nächste Klappe auftreten, und kopieren Sie die Dateien von bootflash über Ihre bevorzugte Übertragungsmethode für die Analyse:

```
R1#show logging
```

```
*Jul 17 16:51:47.154: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.2 (Tunnel10) is down:
```

- Der Protokollpuffer auf dem Router gibt an, dass eine EIGRP-Klappe aufgetreten ist, und die Dateien wurden vom EEM gespeichert.

Analyse

Korrelieren Sie an diesem Punkt die Zeit der im Protokollpuffer gefundenen Klappe mit den erfassten Paketerfassungen, um zu bestimmen, ob die Hello-Pakete an beiden Enden gesendet und empfangen wurden, als die Klappe auftrat. Da die empfangene Schnittstelle PEER-TERMINATION auf R1 erkannt wurde, bedeutet dies, dass R2 verlorene Hellos erkannt haben muss und daher die Haltezeit abgelaufen ist, was in der Protokolldatei zu sehen ist:

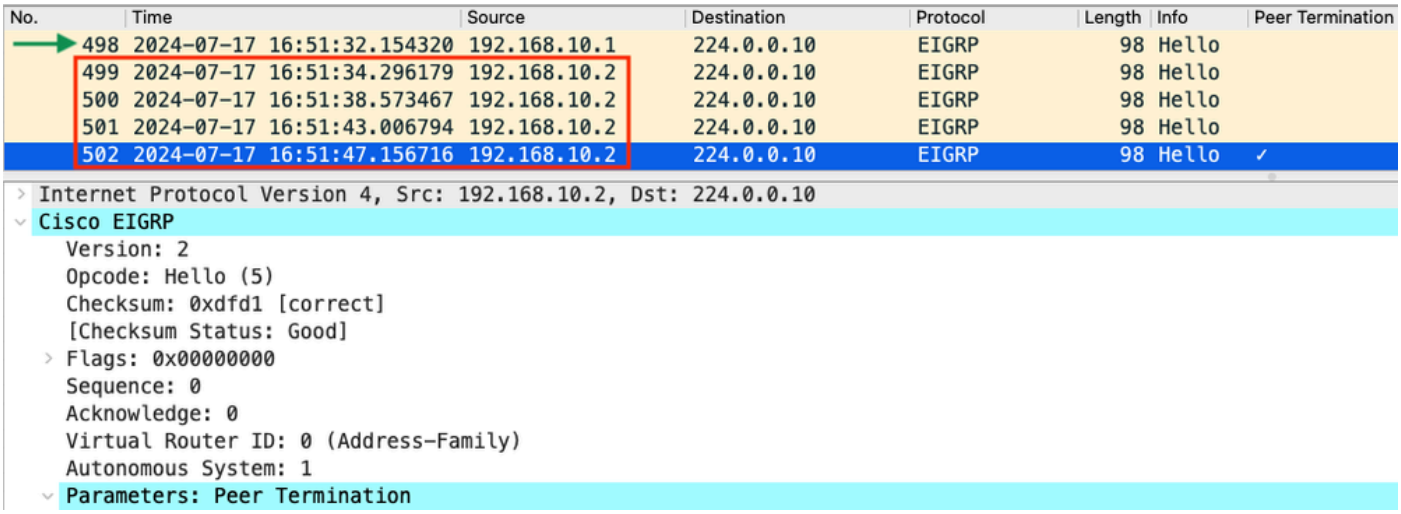
```
*Jul 17 16:51:47.156: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.1 (Tunnel10) is down: holdin
*Jul 17 16:51:51.870: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 192.168.10.1 (Tunnel10) is up: new adja
```

Da die von R2 erkannte Haltezeit abgelaufen ist, überprüfen Sie, ob in den 15 Sekunden vor der Klappe in der auf R1 erfassten Erfassung Hellos von R1 gesendet wurden:

No.	Time	Source	Destination	Protocol	Length	Info	Peer Termination
503	2024-07-17 16:51:32.150713	192.168.10.1	224.0.0.10	EIGRP	98	Hello	
504	2024-07-17 16:51:34.293604	192.168.10.2	224.0.0.10	EIGRP	98	Hello	
505	2024-07-17 16:51:36.802191	192.168.10.1	224.0.0.10	EIGRP	98	Hello	
507	2024-07-17 16:51:38.571024	192.168.10.2	224.0.0.10	EIGRP	98	Hello	
508	2024-07-17 16:51:41.456619	192.168.10.1	224.0.0.10	EIGRP	98	Hello	
510	2024-07-17 16:51:43.004216	192.168.10.2	224.0.0.10	EIGRP	98	Hello	
511	2024-07-17 16:51:46.457320	192.168.10.1	224.0.0.10	EIGRP	98	Hello	
513	2024-07-17 16:51:47.154111	192.168.10.2	224.0.0.10	EIGRP	98	Hello	✓

- Die Erfassung zeigt Hellos von 192.168.10.1 (R1) und 192.168.10.2 (R2) in den 15 Sekunden vor dem PEER-TERMINATION Hello-Paket, das R2 um 16:51:47 (Paket 513) sendet.
- Die Pakete 503, 505, 508 und 511 (durch die grünen Pfeile gekennzeichnet) waren alle Hellos, die in diesem Zeitraum von R1 gesendet wurden.

Der nächste Schritt besteht darin, zu überprüfen, ob alle von R1 gesendeten Hellos von R2 zu dem Zeitpunkt empfangen wurden. Daher muss die von R2 erfasste Erfassung überprüft werden:



- Die Erfassung zeigt, dass der letzte Hello, der von 192.168.10.1 (R1) empfangen wurde, um 16:51:32 (durch den grünen Pfeil gekennzeichnet) war. Danach werden in den nächsten 15 Sekunden nur Hellos angezeigt, die von R2 gesendet werden (durch das rote Kästchen gekennzeichnet). Die Pakete 505, 508 und 511 in der Erfassung von R1 werden in der Erfassung von R2 nicht angezeigt. Dies führt dazu, dass R2 den Haltezeitgeber als abgelaufen erkennt und das PEER-TERMINATION-Hello-Paket um 16:51:47 Uhr sendet (Paket 502).

Die Schlussfolgerung aus diesen Daten ist, dass der Paketverlust irgendwo im Trägernetz zwischen R1 und R2 liegt. In diesem Fall lag der Verlust in Richtung von R1 bis R2. Um weiter zu untersuchen, muss der Träger beteiligt sein, um den Pfad auf Tropfen zu überprüfen.

OSPF

Dieselbe Logik kann zur Fehlerbehebung bei zeitweiligen OSPF-Flaps verwendet werden. In diesem Abschnitt werden die wichtigsten Faktoren beschrieben, die das Protokoll in Bezug auf Timer, IP-Adressfilter und Protokollmeldungen von anderen Routing-Protokollen unterscheiden.

- Die Standard-Timer sind 10-Sekunden-Hellos und 40-Sekunden-Dead-Timer. Bestätigen Sie immer die Timer, die in Ihrem Netzwerk verwendet werden, wenn Sie Probleme mit abgelaufenen Flaps für den Dead-Timer beheben.
- Hello-Pakete stammen von den IP-Adressen der Schnittstelle. Wenn zusätzliche ACLs benötigt werden, lautet die Multicast-Zieladresse für OSPF Hellos 224.0.0.5.
- Die Protokollmeldungen auf den Geräten unterscheiden sich geringfügig. Im Gegensatz zum EIGRP gibt es bei OSPF kein Konzept für eine Peer-Terminierungsnachricht. Stattdessen protokolliert das Gerät, das den abgelaufenen Dead Timer erkennt, dies als Flapping-Grund, und dann enthalten die Hellos, die es sendet, nicht mehr die Router-ID des Peers, wodurch der Peer in den INIT-Zustand versetzt wird. Wenn die Hellos erneut erkannt werden, geht die Adjacency durch, bis sie den Status "VOLL" erreicht. Beispiele:

R1 erkennt den abgelaufenen Dead-Timer:

```
R1#show logging | i OSPF
*Jul 30 15:29:14.027: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.2 on Tunnel20 from FULL to DOWN, Neighb
```

```
*Jul 30 15:32:30.278: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.2 on Tunnel20 from LOADING to FULL, Loa
*Jul 30 16:33:19.841: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.2 on Tunnel20 from FULL to DOWN, Neighb
*Jul 30 16:48:10.504: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.2 on Tunnel20 from LOADING to FULL, Loa
```

R2 zeigt die Protokollmeldungen jedoch nur an, wenn OSPF zurück auf FULL wechselt. Wenn der Status zu INIT wechselt, wird keine Protokollmeldung angezeigt:

```
R2#show logging | i OSPF
*Jul 30 16:32:30.279: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on Tunnel20 from LOADING to FULL, Loa
*Jul 30 16:48:10.506: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on Tunnel20 from LOADING to FULL, Loa
```

Verwenden Sie "%OSPF-5-ADJCHG" als Syslog-Muster, um den EEM auf beiden Geräten auszulösen. So wird sichergestellt, dass der EEM auf beiden Geräten ausgelöst wird, solange er ausfällt und wieder aktiv ist. Der konfigurierte ratelimit-Wert stellt sicher, dass bei mehreren Protokollen mit dieser Zeichenfolge innerhalb kurzer Zeit keine doppelte Auslösung erfolgt. Der Schlüssel liegt darin, zu bestätigen, ob Hellos in den Paketerfassungen auf beiden Seiten gesendet und empfangen werden.

BGP

Dieselbe Logik kann auch für die Fehlerbehebung bei zeitweiligen BGP-Flaps verwendet werden. In diesem Abschnitt werden die wichtigsten Faktoren beschrieben, die das Protokoll in Bezug auf Timer, IP-Adressfilter und Protokollmeldungen von anderen Routing-Protokollen unterscheiden.

- Die Standard-Timer sind 60-Sekunden-Keepalives und eine Haltezeit von 180 Sekunden. Bestätigen Sie immer die Timer, die in Ihrem Netzwerk verwendet werden, wenn Sie Probleme mit abgelaufenen Flaps für die Haltezeit beheben.
- Keepalive-Pakete werden als Unicast-Pakete zwischen den benachbarten IP-Adressen an den TCP-Zielport 179 gesendet. Falls zusätzliche ACL-Spezifität erforderlich ist, lassen Sie den TCP-Datenverkehr von den Quell-IP-Adressen an den Ziel-TCP-Port 179 zu.
- Die Protokollmeldungen für BGP sehen auf beiden Geräten ähnlich aus, aber das Gerät, das eine abgelaufene Haltezeit erkennt, zeigt an, dass es die Benachrichtigung an den Nachbarn gesendet hat, während das andere Gerät anzeigt, dass es die Benachrichtigung erhalten hat. Beispiele:

R1 erkennt die abgelaufene Haltezeit und sendet eine Benachrichtigung an R2:

```
R1#show logging | i BGP
*Jul 30 17:49:23.730: %BGP-3-NOTIFICATION: sent to neighbor 192.168.30.2 4/0 (hold time expired) 0 byte
*Jul 30 17:49:23.731: %BGP-5-NBR_RESET: Neighbor 192.168.30.2 reset (BGP Notification sent)
*Jul 30 17:49:23.732: %BGP-5-ADJCHANGE: neighbor 192.168.30.2 Down BGP Notification sent
*Jul 30 17:49:23.732: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.30.2 IPv4 Unicast topology base remove
```

R2 erhält Benachrichtigung von R1, weil R1 erkannte, dass die Haltezeit abgelaufen ist:

```
R2#show logging | i BGP
*Jul 30 17:49:23.741: %BGP-3-NOTIFICATION: received from neighbor 192.168.30.1 4/0 (hold time expired)
*Jul 30 17:49:23.741: %BGP-5-NBR_RESET: Neighbor 192.168.30.1 reset (BGP Notification received)
*Jul 30 17:49:23.749: %BGP-5-ADJCHANGE: neighbor 192.168.30.1 Down BGP Notification received
*Jul 30 17:49:23.749: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.30.1 IPv4 Unicast topology base remove
```

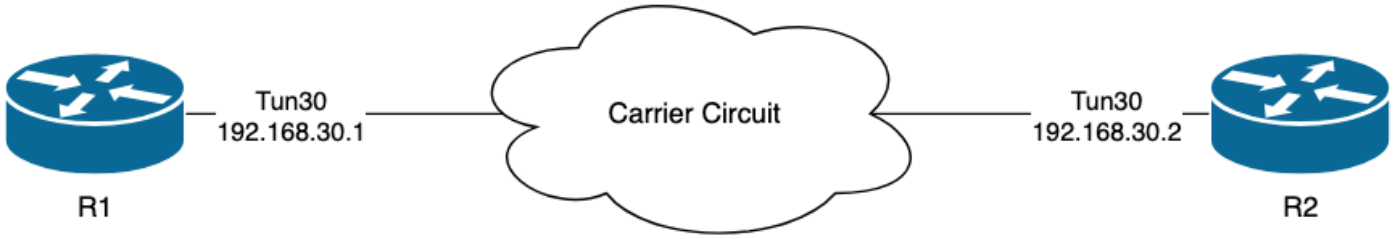
Verwenden Sie "%BGP_SESSION-5-ADJCHANGE" als Syslog-Muster, um den EEM für eine BGP-Klappe auszulösen. Alle anderen "%BGP"-Syslog-Meldungen, die auch nach der Klappe protokolliert werden, können ebenfalls zum Auslösen des EEM verwendet werden.

Fehlerbehebung: zeitweilige BFD-Flaps

Die gleiche Methode kann bei der Fehlerbehebung bei zeitweiligen BFD-Flaps angewendet werden, wobei für die Analyse einige geringfügige Unterschiede bestehen. In diesem Abschnitt werden einige grundlegende BFD-Funktionen behandelt. Außerdem wird ein Beispiel für die Verwendung von EEM und EPC zur Fehlerbehebung bereitgestellt. Weitere Informationen zur BFD-Fehlerbehebung finden Sie unter Fehlerbehebung bei bidirektionaler Weiterleitungserkennung in Cisco IOS XE.

In diesem Beispiel werden die BFD-Timer auf 300 ms mit einem Multiplikator von 3 festgelegt, d. h. alle 300 ms werden Echos gesendet. Ein Echofehler wird erkannt, wenn nicht drei aufeinander folgende Echopakete zurückgegeben werden (dies entspricht einer Haltezeit von 900 ms).

Topologie



Beispiel: BFD-Echo-Modus

Im BFD-Echo-Modus (dem Standardmodus) werden die BFD-Echo-Pakete mit der lokalen Schnittstellen-IP als Quelle und Ziel gesendet. Auf diese Weise kann der Nachbar das Paket auf Datenebene verarbeiten und an das Quellgerät zurücksenden. Jedes BFD-Echo wird mit einer Echo-ID im BFD-Echo-Nachrichten-Header gesendet. Diese können verwendet werden, um festzustellen, ob ein gesendetes BFD-Echo-Paket zurückempfangen wurde, da ein beliebiges BFD-Echo-Paket zweimal vorkommen muss, wenn es tatsächlich vom Nachbarn zurückgesendet wurde. Die BFD-Steuerungspakete, mit denen der Status der BFD-Sitzung gesteuert wird, werden per Unicast zwischen den IP-Adressen der Schnittstellen gesendet.

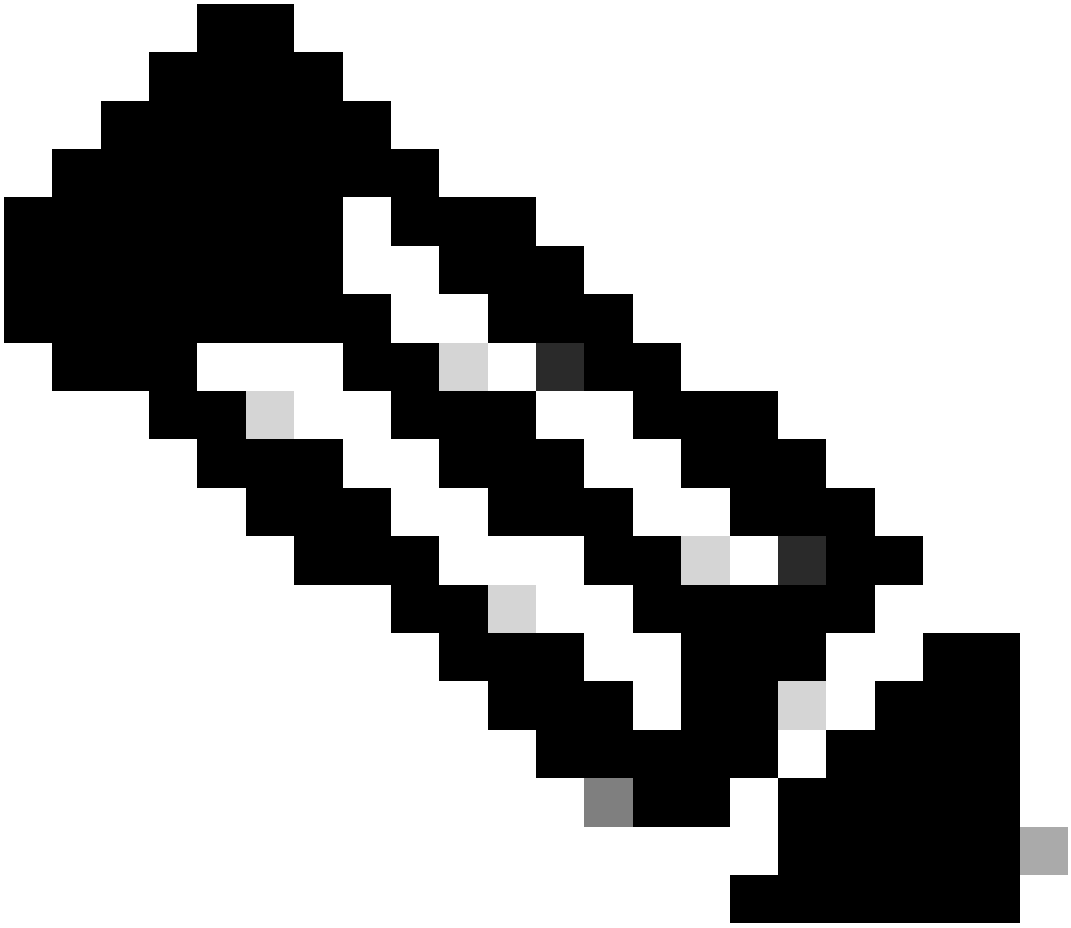
Aus den Protokollen von R1 geht hervor, dass die BFD-Adjacency aufgrund von ECHO FAILURE mehrfach abgenommen hat, d. h. dass R1 in diesen Intervallen 3 eigene Echo-Pakete nicht von R2 empfangen oder verarbeitet hat.

```
R1#show logging | i BFD
*Jul 18 13:41:09.007: %BFDFSM-6-BFD_SESS_DOWN: BFD-SYSLOG: BFD session ld:4097 handle:1,is going Down R
*Jul 18 13:41:09.009: %BGP-5-NBR_RESET: Neighbor 192.168.30.2 reset (BFD adjacency down)
*Jul 18 13:41:09.010: %BGP-5-ADJCHANGE: neighbor 192.168.30.2 Down BFD adjacency down
*Jul 18 13:41:09.010: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.30.2 IPv4 Unicast topology base remove
*Jul 18 13:41:09.010: %BFD-6-BFD_SESS_DESTROYED: BFD-SYSLOG: bfd_session_destroyed,  ld:4097 neigh proc
*Jul 18 13:41:13.335: %BFDFSM-6-BFD_SESS_UP: BFD-SYSLOG: BFD session ld:4097 handle:1 is going UP
*Jul 18 13:41:18.576: %BFD-6-BFD_SESS_CREATED: BFD-SYSLOG: bfd_session_created, neigh 192.168.30.2 proc
*Jul 18 13:41:19.351: %BFDFSM-6-BFD_SESS_UP: BFD-SYSLOG: BFD session ld:4097 handle:1 is going UP
*Jul 18 15:44:08.360: %BFDFSM-6-BFD_SESS_DOWN: BFD-SYSLOG: BFD session ld:4097 handle:1,is going Down R
*Jul 18 15:44:08.362: %BGP-5-NBR_RESET: Neighbor 192.168.30.2 reset (BFD adjacency down)
*Jul 18 15:44:08.363: %BGP-5-ADJCHANGE: neighbor 192.168.30.2 Down BFD adjacency down
*Jul 18 15:44:08.363: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.30.2 IPv4 Unicast topology base remove
*Jul 18 15:44:08.363: %BFD-6-BFD_SESS_DESTROYED: BFD-SYSLOG: bfd_session_destroyed,  ld:4097 neigh proc
*Jul 18 15:44:14.416: %BFDFSM-6-BFD_SESS_UP: BFD-SYSLOG: BFD session ld:4097 handle:1 is going UP
*Jul 18 15:44:14.418: %BFD-6-BFD_SESS_DESTROYED: BFD-SYSLOG: bfd_session_destroyed,  ld:4097 neigh proc
*Jul 18 15:44:18.315: %BFD-6-BFD_SESS_CREATED: BFD-SYSLOG: bfd_session_created, neigh 192.168.30.2 proc
```

Konfiguration

1. Konfigurieren Sie die ACL mit den IP-Adressen der Tunnelschnittstelle, da dies die Quell-IP-Adressen der BFD-Echo- und Steuerungspakete sind:

```
R1#conf t
R1(config)#ip access-list extended FLAP_CAPTURE
R1(config-ext-nacl)#permit ip host 192.168.30.1 any
R1(config-ext-nacl)#permit ip host 192.168.30.2 any
```



Anmerkung: Die gezeigten Konfigurationen stammen aus R1. Das Gleiche gilt für R2 für die relevanten Schnittstellen und mit geänderten Dateinamen für den EEM. Falls weitere Spezifität erforderlich ist, konfigurieren Sie die ACL für UDP mit den Zielports 3785 (Echo-Pakete) und 3784 (Kontrollpakete).

2. Erstellen Sie den EPC, und ordnen Sie ihn der Schnittstelle und der ACL zu:

```
R1#monitor capture CAP interface Tunnel30 both
R1#monitor capture CAP access-list FLAP_CAPTURE
R1#monitor capture CAP buffer size 5 circular
```

3. Starten Sie den EPC, und bestätigen Sie, dass die Pakete in beide Richtungen erfasst werden:

```
R1#monitor capture CAP start
```

```
R1#show monitor capture CAP buff brief
 ----------------------------------------------------------------------------
 #   size   timestamp     source             destination      dscp    protocol
 ----------------------------------------------------------------------------
   0   54    0.000000   192.168.30.2     ->  192.168.30.2     48 CS6  UDP
   1   54    0.000000   192.168.30.2     ->  192.168.30.2     48 CS6  UDP
   2   54    0.005005   192.168.30.1     ->  192.168.30.1     48 CS6  UDP
   3   54    0.005997   192.168.30.1     ->  192.168.30.1     48 CS6  UDP
```

4. Konfigurieren Sie EEM:

```
R1#conf t
R1(config)#event manager applet R1_BFD_FLAP authorization bypass
R1(config-applet)#event syslog pattern "%BFDFSM-6-BFD_SESS_DOWN" maxrun 120 ratelimit 100000
R1(config-applet)#action 000 cli command "enable"
R1(config-applet)#action 005 cli command "show clock | append bootflash:R1_BFD_FLAP.txt"
R1(config-applet)#action 010 cli command "show logging | append bootflash:R1_BFD_FLAP.txt"
R1(config-applet)#action 015 cli command "show process cpu sorted | append bootflash:R1_BFD_FLAP.txt"
R1(config-applet)#action 020 cli command "show process cpu history | append bootflash:R1_BFD_FLAP.txt"
R1(config-applet)#action 025 cli command "show interfaces | append bootflash:R1_BFD_FLAP.txt"
R1(config-applet)#action 030 cli command "monitor capture CAP stop"
R1(config-applet)#action 035 cli command "monitor capture CAP export bootflash:R1_BFD_CAP.pcap"
R1(config-applet)#action 040 syslog msg "Saved logs to bootflash:R1_BFD_FLAP.txt and saved packet captu
R1(config-applet)#action 045 cli command "end"
R1(config-applet)#end
```

5. Warten Sie auf die nächste Klappe auftreten, und kopieren Sie die Dateien von bootflash über Ihre bevorzugte Übertragungsmethode für die Analyse:

```
R1#show logging

      *Jul 18 19:09:47.482: %BFDFSM-6-BFD_SESS_DOWN: BFD-SYSLOG: BFD session ld:4097 handle:1,is going
```

- Der Protokollpuffer gibt an, dass um 19:09:47 Uhr eine BFD-Klappe aufgetreten ist, und die Dateien wurden vom EEM gespeichert.

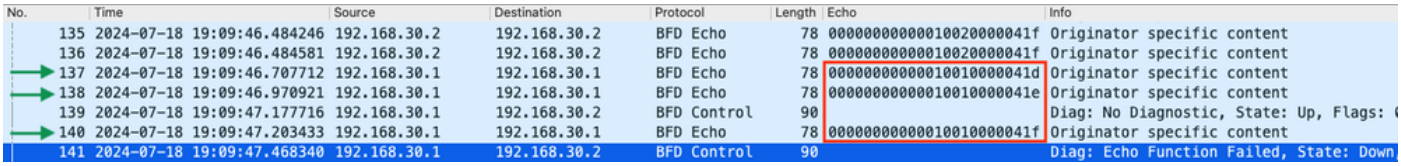
Analyse

Korrelieren Sie an diesem Punkt die Zeit der im Protokollpuffer gefundenen Klappe mit den

erfassten Paketerfassungen, um zu ermitteln, ob die BFD-Echos an beiden Enden gesendet und empfangen wurden, als das Problem auftrat. Da der Grund für die Klappe auf R1 "ECHO FAILURE" ist, bedeutet dies, dass ein Kontrollpaket an R2 gesendet wurde, um die BFD-Sitzung zu beenden. Dies spiegelt sich in der Protokolldatei wider, die von R2 gesammelt wurde, in der der Grund für den BFD-Ausfall "RX DOWN" zu sehen ist:

```
*Jul 18 19:09:47.468: %BFDFSM-6-BFD_SESS_DOWN: BFD-SYSLOG: BFD session ld:4098 handle:2,is going Down R
*Jul 18 19:09:47.470: %BGP-5-NBR_RESET: Neighbor 192.168.30.1 reset (BFD adjacency down)
*Jul 18 19:09:47.471: %BGP-5-ADJCHANGE: neighbor 192.168.30.1 Down BFD adjacency down
*Jul 18 19:09:47.471: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.30.1 IPv4 Unicast topology base remove
*Jul 18 19:09:47.471: %BFD-6-BFD_SESS_DESTROYED: BFD-SYSLOG: bfd_session_destroyed,  ld:4098 neigh proc
```

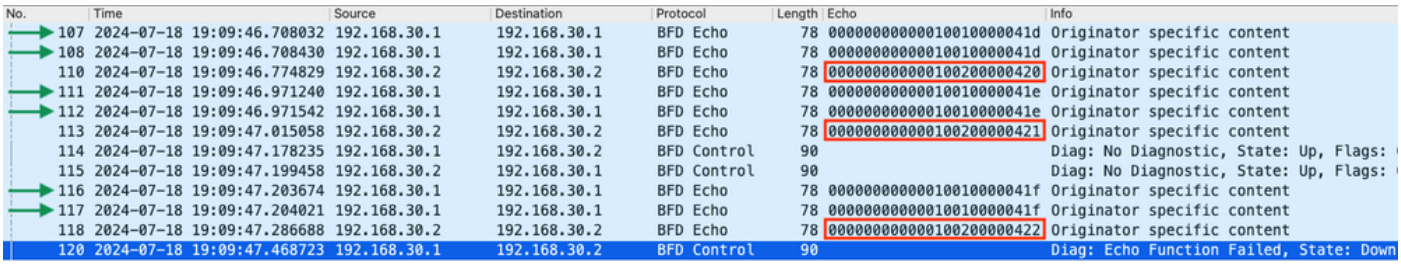
Da R1 einen ECHO-FEHLER erkannte, überprüfen Sie die auf R1 erfasste Paketerfassung, um festzustellen, ob in den 900 ms vor der Klappe BFD-Echos gesendet und empfangen wurden.



No.	Time	Source	Destination	Protocol	Length	Echo	Info
135	2024-07-18 19:09:46.484246	192.168.30.2	192.168.30.2	BFD Echo	78	0000000000001002000041f	Originator specific content
136	2024-07-18 19:09:46.484581	192.168.30.2	192.168.30.2	BFD Echo	78	0000000000001002000041f	Originator specific content
137	2024-07-18 19:09:46.707712	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041d	Originator specific content
138	2024-07-18 19:09:46.970921	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041e	Originator specific content
139	2024-07-18 19:09:47.177716	192.168.30.1	192.168.30.2	BFD Control	90		Diag: No Diagnostic, State: Up, Flags:
140	2024-07-18 19:09:47.203433	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041f	Originator specific content
141	2024-07-18 19:09:47.468340	192.168.30.1	192.168.30.2	BFD Control	90		Diag: Echo Function Failed, State: Down

- Die Erfassung zeigt, dass R1 aktiv BFD-Echopakete bis zum Zeitpunkt der Klappe gesendet hat, diese jedoch nicht von R2 zurückgegeben wurden. Daher sendet R1 ein Steuerungspaket, um die Sitzung um 19:09:47.468 Uhr zu beenden.
- Dies wird dadurch deutlich, dass die Pakete 137, 138 und 140 (durch die grünen Pfeile angedeutet) nur einmal in der Erfassung gesehen werden, was aus den BFD-Echo-IDs (im roten Kasten) ermittelt werden kann. Wenn die Echos zurückgegeben wurden, gäbe es eine zweite Kopie jedes dieser Pakete mit derselben BFD-Echo-ID. Das Feld IP-Identifizierung im IP-Header (hier nicht abgebildet) kann ebenfalls zur Verifizierung verwendet werden.
- Diese Erfassung zeigt auch, dass nach Paket 136 keine BFD-Echos von R2 empfangen wurden, was ein weiterer Hinweis auf einen Paketverlust in Richtung R2 bis R1 ist.

Im nächsten Schritt muss überprüft werden, ob alle von R1 gesendeten BFD-Echo-Pakete von R2 empfangen und zurückgegeben wurden. Die von R2 erfasste Erfassung muss daher überprüft werden:



No.	Time	Source	Destination	Protocol	Length	Echo	Info
107	2024-07-18 19:09:46.708032	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041d	Originator specific content
108	2024-07-18 19:09:46.708430	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041d	Originator specific content
110	2024-07-18 19:09:46.774829	192.168.30.2	192.168.30.2	BFD Echo	78	0000000000001002000000420	Originator specific content
111	2024-07-18 19:09:46.971240	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041e	Originator specific content
112	2024-07-18 19:09:46.971542	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041e	Originator specific content
113	2024-07-18 19:09:47.015058	192.168.30.2	192.168.30.2	BFD Echo	78	0000000000001002000000421	Originator specific content
114	2024-07-18 19:09:47.178235	192.168.30.1	192.168.30.2	BFD Control	90		Diag: No Diagnostic, State: Up, Flags:
115	2024-07-18 19:09:47.199458	192.168.30.2	192.168.30.1	BFD Control	90		Diag: No Diagnostic, State: Up, Flags:
116	2024-07-18 19:09:47.203674	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041f	Originator specific content
117	2024-07-18 19:09:47.204021	192.168.30.1	192.168.30.1	BFD Echo	78	0000000000001001000041f	Originator specific content
118	2024-07-18 19:09:47.286688	192.168.30.2	192.168.30.2	BFD Echo	78	0000000000001002000000422	Originator specific content
120	2024-07-18 19:09:47.468723	192.168.30.1	192.168.30.2	BFD Control	90		Diag: Echo Function Failed, State: Down

- Diese Erfassung zeigt, dass alle von R1 gesendeten BFD-Echos von R2 empfangen und zurückgegeben wurden (durch grüne Pfeile gekennzeichnet). Die Pakete 107 und 108 sind das gleiche BFD-Echo, die Pakete 111 und 112 das gleiche BFD-Echo und die Pakete 116 und 117 das gleiche BFD-Echo.

- Diese Erfassung zeigt auch, dass R2 aktiv Echopakete (mit roten Kästchen gekennzeichnet) gesendet hat, die bei der Erfassung auf R1 nicht zu sehen sind, was weiterhin einen Paketverlust zwischen den Geräten in Richtung von R2 nach R1 anzeigt.

Die Schlussfolgerung aus diesen Daten ist, dass der Paketverlust irgendwo im Trägernetz zwischen R1 und R2 liegt, und alle Hinweise an dieser Stelle deuten darauf hin, dass die Richtung des Verlustes von R2 nach R1 ist. Um weiter zu untersuchen, muss der Träger beteiligt sein, um den Pfad auf Tropfen zu überprüfen.

Asynchroner BFD-Modus

Dieselbe Methode kann angewendet werden, wenn der asynchrone BFD-Modus verwendet wird (Echofunktion deaktiviert). Die EEM- und EPC-Konfiguration können beibehalten werden. Der Unterschied im asynchronen Modus besteht darin, dass die Geräte Unicast-BFD-Steuerungspakete analog zu einer typischen Routing-Protokoll-Adjacency als Keepalive untereinander senden. Das bedeutet, dass nur Pakete vom UDP-Port 3784 gesendet werden. In diesem Szenario bleibt BFD im aktiven Zustand, solange ein BFD-Paket vom Nachbarn innerhalb des erforderlichen Intervalls empfangen wird. Wenn dies nicht geschieht, lautet der Fehlergrund DETECT TIMER EXPIRED (TIMER ERKANNT), und der Router sendet ein Steuerungspaket an den Peer, um die Sitzung zu beenden.

Um die Erfassungen auf dem Gerät zu analysieren, das den Fehler erkannt hat, suchen Sie nach den Unicast-BFD-Paketen, die der Peer in der Zeit vor dem Ausfall der Klappe empfangen hat. Wenn das TX-Intervall beispielsweise mit einem Multiplikator von 3 auf 300 ms festgelegt wird, gibt dies einen potenziellen Paketverlust an, wenn in den 900 ms vor der Klappe keine BFD-Pakete empfangen wurden. Aktivieren Sie in der Erfassung, die vom Nachbarn über den EEM erfasst wird, dasselbe Zeitfenster. Wenn die Pakete während dieser Zeit gesendet wurden, wird bestätigt, dass irgendwo zwischen den Geräten ein Verlust auftritt.

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.