

Fehlerbehebung bei unbeaufsichtigten Neuladevorgängen auf Catalyst 9300/3850/3650-Switches

Inhalt

[Einleitung](#)

[Fehlerbehebung/Befehle anzeigen](#)

[SifInfo](#)

[SifRacStatus](#)

[SifRacControl](#)

[SifExceptionInterruptA4](#)

[SifExceptionInterruptA8](#)

[Andere Stacking-Register](#)

[LesenRegister aus dem Linux-Kernel](#)

[ASIC in Dope.sh ändern](#)

[Probleme beim automatischen Neuladen](#)

[Schritt 1](#)

[Schritt 2](#)

[Schritt 3](#)

[Schritt 4](#)

[Timeouts/Neuladevorgänge bei Stack-Elementen - Anwenderbericht](#)

[Symptome](#)

[Abkürzungen](#)

Einleitung

In diesem Dokument wird die Fehlerbehebung bei Befehlen/Registern für Probleme beschrieben, die speziell mit Stacking-Port-/Kabelproblemen und automatischen Neuladevorgängen zusammenhängen.

Fehlerbehebung/Befehle anzeigen

Erfassung und Analyse von nützlichen Registern (für jeden ASIC und Core). Es gibt drei Hauptkategorien:

- SifInfo
- SifRacStatus
- SifRacControl

```
show platform hardware fed switch active fwd-asic register read register-name <name>
```

SifInfo

Das erste Bit sagt uns, ob Basic verfügbar ist oder nicht. Er ist auf 0x1 festgelegt. Wenn 0x0 eingestellt ist, treten Weiterleitungsprobleme auf. Fehlerindikatoren oder Fehlerboxen können Pakete nicht richtig wiederherstellen.

```
Switch#sh platform hardware fed switch active fwd-asic register read register-name SifInfo
```

```
For asic 0 core 0
```

```
Module 0 - SifInfo[0][0]
```

```
available           : 0x1 <---- should be 0x1 indicating balloting is completed
headerVersion       : 0x0
nodeAllLinksAvailable : 0x1
nodeId              : 0x4 <---- asic ID (unique across all asics in the stack)
numNodes            : 0x8 <---- how many asics are there in whole stack
serdesSpeed         : 0x2
sifAllLinksAvailable : 0x1
sifSupStall         : 0x0
wrappedAtRac0       : 0x0 <---- If a single stack port is down, 3 of 6 should wrap w/ value
wrappedAtRac1       : 0x0           of 0x1. Will appears in groups for 0, 2 and 4 or 1, 3 and 5.
wrappedAtRac2       : 0x0
wrappedAtRac3       : 0x0
wrappedAtRac4       : 0x0
wrappedAtRac5       : 0x0
```



Hinweis: Jedes Stack-Kabel hat sechs Rack-Ringe (Ringzugriffskontrolle), drei ausgehend/drei eingehend mit je 40Gig. `WrappedAtRac` 0 bis 5 gibt an, ob eine Stapelverbindung ausgefallen ist oder nicht. Wenn die Dinge gut sind, wird es als `0x0` angezeigt (sechs Verbindungen pro Basis, drei ausgehende, drei eingehende). Beispielsweise sind ungerade Nummern ausgehend und gerade Nummern eingehend oder umgekehrt).

SifRacStatus

Um jedes Racs detailliert zu überprüfen, werden kritische Aspekte angezeigt; `active/linkOk/syncOk` Bits, die uns sagen, ob ein bestimmter Rack eine Verbindung hergestellt hat oder nicht (wenn OK, wird es als `0x1` angezeigt).

```
Switch#sh plat hardware fed sw active fwd-asic register read register-name SifRacStatus
```

```
For asic 0 core 0
```

Module 0 - SifRacStatus[0][0]

```
active           : 0x1 <----
available        : 0x1
copyOk           : 0x1
disabled         : 0x0
insertOk        : 0x1
linkOk           : 0x1 <----
messageOk        : 0x1
noDataOnRing    : 0x0
pcsAlignmentOk   : 0x1
pcsCodewordSync : 0xf
reOrderOk       : 0x1
slapId          : 0x0
stripOk         : 0x1
syncOk          : 0x1 <----
toPbcOk         : 0x1
transmitOk      : 0x1
```

SifRacControl

Prüfen Sie, ob Rac ausgeschaltet ist oder nicht. Überprüfen Sie den Parameter greenPowerDisable. Dies zeigt 0x0 für alle Racs (mindestens für Nyquist-Plattform). Es gibt einige Ausnahmen, bei denen aufgrund der HW-Beschränkung des Stack-Kabels selbst erwartet wird, dass der Parameter "Racs power down" oder "greenPowerDisable" als 0x1 angezeigt wird, z. B. der Switch 3650, der die untere Endbox darstellt. Das Stack-Kabel unterstützt dann nur noch zwei Racs pro Basis. Die verbleibenden zwei Racs sind ausgeschaltet.

```
Switch#sh plat hardware fed sw active fwd-asic register read register-name SifRacControl
```

```
For asic 0 core 0
```

Module 0 - SifRacControl[0][0]

```
copyEn           : 0x1
deployToken      : 0x0
disablePmaChecks : 0x0
forceSync        : 0x0
greenPowerDisable : 0x0 <----
init             : 0x0
initRacInfoLinkedList : 0x0
insertEn         : 0x1
messageEn        : 0x1
reOrderEn       : 0x1
stripEn         : 0x1
toPbcEn         : 0x1
transmitEn      : 0x1
```

SifExceptionInterruptA4

Dies wird ausgelöst, weil im System ein Linkwechsel stattfindet (Up/Down-Situation). Der Interrupt wird auf Softwareebene abgewickelt. Es wird verarbeitet, um zu sehen, ob es irgendwelche Link-bezogenen Änderungen, und dann wird es veröffentlicht (log generiert).

```
Switch#sh plat hardware fed sw active fwd-asic register read register-name SifExceptionInterruptA4
```

```
For asic 0 core 0
```

```
Module 0 - SifExceptionInterruptA4[0][0]
```

```
sifRac0LinkOkChange      : 0x0
sifRac0LinkedListSpill   : 0x0
sifRac0SyncOkChange     : 0x1
sifRac0TransitFifoSpill  : 0x0
sifRac1LinkOkChange      : 0x0
sifRac1LinkedListSpill   : 0x0
sifRac1SyncOkChange     : 0x1
sifRac1TransitFifoSpill  : 0x0
sifRac2LinkOkChange      : 0x0
sifRac2LinkedListSpill   : 0x0
sifRac2SyncOkChange     : 0x1
sifRac2TransitFifoSpill  : 0x0
sifRac3LinkOkChange      : 0x0
sifRac3LinkedListSpill   : 0x0
sifRac3SyncOkChange     : 0x1
sifRac3TransitFifoSpill  : 0x0
sifRac4LinkOkChange      : 0x0
sifRac4LinkedListSpill   : 0x0
sifRac4SyncOkChange     : 0x1
sifRac4TransitFifoSpill  : 0x0
sifRac5LinkOkChange      : 0x0
sifRac5LinkedListSpill   : 0x0
sifRac5SyncOkChange     : 0x1
sifRac5TransitFifoSpill  : 0x0
```

SifExceptionInterruptA8

Dies ist der Hardware-Interrupt, der uns Details beim Abstimmen gibt (Abstimmen = Grundinitialisierung). Nachdem A8 abgeschlossen ist, prüft das System, ob ein grundlegendes verfügbares Bit richtig eingestellt ist. Ist dies nicht der Fall, wird die Stimmabgabe erneut durchgeführt.



Hinweis: Wenn die maximale Anzahl erreicht ist, wird der Switch mit einem Fehler neu geladen, der besagt, dass das verfügbare HW-Bit nicht gesetzt wurde oder Balloting nicht abgeschlossen wurde.

```
Switch#sh plat hardware fed sw active fwd-asic register read register-name SifExceptionInterruptA8
```

```
For ASIC 0 core 0
```

```
Module 0 - SifExceptionInterruptA8[0][0]
```

```
sifBallotDone          : 0x0
sifBallotOverallTimerExpires : 0x0
sifBallotPerStateTimerExpires : 0x0
sifBallotSpeedChangeNeeded : 0x0
sifBallotStart         : 0x1
sifDebugSent           : 0x0
sifEastNeighborChange  : 0x1
sifMessageReceiveBufferCreditsEmpty : 0x0
sifMessageReceived     : 0x1
sifMessageSent         : 0x1
```

```

sifNodeIdChanged          : 0x1
sif0ob3in2DropCntOverflow : 0x0
sif0obFlushDropCntOverflow : 0x0
sif0obStackSifCreditDropCntOverflow : 0x0
sif0obStackSifMtuDropCntOverflow : 0x0
sif0obSupSifMtuDropCntOverflow : 0x0
sifRacInfoLinkedListInitDone0 : 0x1
sifRacInfoLinkedListInitDone1 : 0x1
sifRacInfoLinkedListInitDone2 : 0x1
sifRacInfoLinkedListInitDone3 : 0x1
sifRacInfoLinkedListInitDone4 : 0x1
sifRacInfoLinkedListInitDone5 : 0x1
sifSegmentBuffer0LinkedListSpill : 0x0
sifSegmentBuffer1LinkedListSpill : 0x0
sifSegmentBufferLinkedListInitDone0 : 0x1
sifSegmentBufferLinkedListInitDone1 : 0x1
sifStackTopologyChange    : 0x1
sifUnmappedDestIndex      : 0x0
sifWestNeighborChange     : 0x1

```

Mit dem nächsten Befehl werden SIF-Zähler angezeigt, die SDP- und SIF-Verwaltungsmeldungen enthalten. Konzentrieren Sie sich ggf. auf die fehlgeschlagenen Nachrichten.

```

Switch#show platform software sif switch active r0 counters
Stack Interface (SIF) Counters

```

Stack Discovery Protocol (SDP) Messages

Message	Tx Success	Tx Fail	Rx Success	Rx Fail
Discovery	0	0	0	0
Neighbor	0	0	0	0
Forward	455966	0	1355818	107

SIF Management Messages

Message	Success	Fail
Link Status	16	0
Link Management	0	0
Chassis Num	1	0
Topo Change	3	0
Active Declare	1	0
Template set	2	0

Es gibt einen zusätzlichen Befehl, der ausgeführt werden könnte und nur dann Informationen anzeigt, wenn ein Interrupt den Schwellenwert überschreitet. Der Befehl lautet `show platform software sif switch active R0 exceptions` Hier ist die Ausgabe, wenn keine Probleme auf den Interrupts vorhanden sind:

```
Switch#
Switch#show platform software sif switch active R0 exceptions
Switch#
```

Wenn Interrupts vorhanden sind, ist die Ausgabe ähnlich wie das nächste Skript. In einigen Szenarien (Bootvorgang, Plug-and-Unplug usw.) werden Interrupts erwartet. Wenn also ein echtes Problem vorliegt und es zu kontinuierlichen Interrupts kommt, führen Sie den Befehl wiederholt für einen Zeitraum von Sekunden/Minuten aus.

```
Switch#show platform software sif switch active r0 exceptions
*****
Asicnum: 0
SIF INT : SIFEXCEPTIONINTERRUPTA1_SIFRAC5PMARECEIVEFIFOSPILL3_FIELD_IDX
Occurred count: 1
First Time: Fri May 18 08:03:23 2018
Last Time: Fri May 18 08:03:23 2018
-----
SIF INT : SIFEXCEPTIONINTERRUPTA1_SIFRAC5PMARECEIVEFIFOSPILL2_FIELD_IDX
Occurred count: 1
First Time: Fri May 18 08:03:23 2018
Last Time: Fri May 18 08:03:23 2018
-----
SIF INT : SIFEXCEPTIONINTERRUPTA1_SIFRAC5PMARECEIVEFIFOSPILL1_FIELD_IDX
Occurred count: 1
First Time: Fri May 18 08:03:23 2018
Last Time: Fri May 18 08:03:23 2018
-----
SIF INT : SIFEXCEPTIONINTERRUPTA1_SIFRAC5PMARECEIVEFIFOSPILL0_FIELD_IDX
Occurred count: 1
First Time: Fri May 18 08:03:23 2018
Last Time: Fri May 18 08:03:23 2018
```

Diese Tabelle enthält die häufigsten SIF-Ausnahmen von show platform software sif switch active R0 exceptions:

Ausnahme#	Feldname	Schweregrad	Nutzung	Beschreibung
0	sifRac{0:5}PmaTransmitFifoSpill{0:3}	Major (Schwerwiegend)	Statistik	Dies wird aus Push-Pull-FIF Systemuhr un verschüttet wi passieren. We es wahrschein darauf, dass d deaktiviert wu Programmieru fehlerhafte Se nicht auf ein Programmieru zurückzuführe Problem. Aber

				selbst. Und da kleinen Probleme Segment oder eine Neuinit. V Problem war, auftritt, dann n dieses CHIEF ausgelöst, und der Zustand n auftritt. Diese Übertragungsv Toast.
1	sifRac{0:5}PmaReceiveFifoSpill{0:3}	Major (Schwerwiegend)	Statistik	Dies wird ausg Push-Pull-FIFO Systemuhr un verschüttet wi passieren. We es wahrschein darauf, dass d deaktiviert wu Programmierung fehlerhafte Se nicht auf ein Programmierung zurückzuführen Problem. Aber selbst. Und da kleinen Probleme Segment oder eine Neuinit. V Problem war, auftritt, dann n dieses CHIEF ausgelöst, und der Zustand n auftritt. Diese Übertragungsv Toast.
2	sifRac{0:5}SerdesLossOfLock{0:3}	Major (Schwerwiegend)	Statistik	In Korrelation sifRac{0:5}Pm zu verwenden Zustand der e Uhren mit norm Betriebszustar Wenn sie nich entsprechen, h Timer den Unt

				ausgleichen. In dies ein Problem sicherzustellen dass die Empfänger funktionieren,
3	sifRac{0:5}ClockLossOfLock{0:3}	Major (Schwerwiegend)	Statistik	In Korrelation mit sifRac{0:5}Pm zu verwenden. Zustand der Empfänger Uhren mit normaler Betriebszustand. Wenn sie nicht entsprechen, kann Timer den Uhrzeit ausgleichen. In dies ein Problem sicherzustellen dass die Empfänger funktionieren,
4	sifRac{0:5}syncOkChange	geringfügig	Überwachung	Anzeige der L
	sifRac{0:5}linkOkChange	geringfügig	Überwachung	Anzeige der L
	sifRac{0:5}linkedListSpill	Major (Schwerwiegend)	Überwachung	RAC-verknüpfte Reorder-Algorithmen die maximal nicht überschritten. und bedeutet, jetzt Tail-Drop und OOB-Nachfrage RAC ist. Dies der Stack falsch in der verknüpfte Fehler aufgetreten Ausnahmen 9
	sifRac{0:5}transitFifoSpill	Major (Schwerwiegend)	Statistik	Der für das Verhalten durch das SIF verantwortlich wahrscheinlich Fehlkonfiguration IdleDensityTimer tatsächlichen Offset (parts per Switch im Verkehr verschüttet.
5	sifRac{0:5}missingToken	Major (Schwerwiegend)	Statistik	Die Stack-Contingent gegangen, bereitgestellt u

				wahrscheinlich dass ein Bit-H einen SifToken Das ist sehr un SIF kann so k dass es auf ve damit umgeht. Neustart, erne eines Tokens Bereitstellung
	sifRac{0:5}doubleToken	Major (Schwerwiegend)	Statistik	
	sifRac{0:5}tokenDeployed	Info	Statistik	
6	sifRac{0:5}RwCrcErrorCntOverflow	geringfügig	Statistik	Wahrscheinlich für eine Komp Stack-Kabels benachbarten vorhanden. Im Debuggen auf aufgeteilt. Im I syncOkChang alles was Sie v der Erfassung BER müssen und zählen, w korrekte Zähl umgestellt we oder pcsCode ist, wird die C nicht überprüf können Sie all BER summier
	sifRac{0:5}DataCrcErrorCntOverflow	geringfügig	Statistik	
	sifRac{0:5}InvalidRwErrorCntOverflow	geringfügig	Statistik	
	sifRac{0:5}PcsCodeWordErrorCntOverflow	geringfügig	Statistik	
7	sifRac{0:5}RdispErrorCntOverflow	geringfügig	Statistik	
	sifRac{0:5}PrbsUnLockErrorCntOverflow	Info	Statistik	Rufen Sie Sta bei der Suche Konfiguration verwenden kö Programmieru
	sifRac{0:5}PrbsBitErrorCntOverflow	Info	Statistik	
	sifRac{0:5}ErrorCaptureCntOverflow	Info	Labor	Statistiken zur fehlerhafter Ri um zu überprü Stack passiert

8	sifRacInfoLinkedListInitDone{0:5}	Info	Überwachung	Die HW-Initialisierung der verknüpften R... abgeschlossen
	sifDroppedSegmentCntOverflow	Info	Statistik	
	sifPbcInconsistentSopEopCntOverflow	Info	Statistik	Worst-Case-S... ob die Daten v... Protokollformu
	sifPbcErrorCntOverflow	Info	Statistik	
	sifSupInconsistentSopEopCntOverflow	Info	Statistik	Worst-Case-S... Sie, ob die Da... Protokollformu... ankommen.
	sifSupErrorCntOverflow	Info	Statistik	
	sifReorderInconsistentSopEopCntOverflow	Info	Statistik	Angabe, dass... Segmentindika... wurde.
	sifDebuggenGesendet	Info	Labor	Anzeige zum L... Debugsegment... aufrufen.
	SifNachrichtGesendet	Info	Labor	Aufgrund der... OOBM sind di... Laborsituation
	SifNachrichtErhalten	Info	Labor	
	SifNachrichtAbgebrochen	Info	Labor	
	sifMessageReceiveBufferCreditsEmpty	geringfügig	Überwachung	Aktualisieren S... wenn dies nicht... Kreditniveau v... damit dieses n
	sifNicht zugeordnetZielIndex	geringfügig	Statistik	Während des... destIndex nicht... und eine portC... portStrip auf '1... weist auf ein k... hin.
	sifSegmentBuffer{0:1}linkedListSpill	Major (Schwerwiegend)	Überwachung	Segmentverkn... der Neuordnun... maximal mögl... überschritten... darauf, dass d... Datensegment... Nachrichten p... Dies ist nur m... Stack falsch k... der verknüpft... aufgetreten ist

				und 10.
	sifSegmentBufferLinkedListInitDone{0:1}	Info	Überwachung	HW-Initialisierung Linkliste ist ab
	sifAbstimmungFertig	Info	Überwachung	Indikationsabs abgeschlossen
	sifBallotSpeedChangeErforderlich	Info	Überwachung	Seit dem letzte Wahlgang ist e Geschwindigk Verbindung er bedeutet, dass Stack eingetre Dynamik der S Geschwindigk langsamer als Geschwindigk nach unten an sie schneller s kann das Erge kürzeren Kabe
	sifOstNachbarnÄndern	Info	Überwachung	Überwachung Aktivierungs-, und Wrap-Sze
	sifWestNachbarnÄndern	Info	Überwachung	
	SifKnotenIDChanged	Info	Überwachung	Angabe, dass letzten Wahlg SifInfo.nodeld
	sifStackTopologieänderung	Info	Überwachung	Überwachung Aktivierungs-, und Wrap-Sze
9	sifRaInfoBuffer{0:5}EccCorrected	Major (Schwerwiegend)	Überwachung	sifRaInfoBuff einem weiche Dies ist nicht g Fall jedoch kö außerhalb der Reihenfolge o Paketverluste Ausgangsdate Zurücksetzen nicht erforderl
	sifRaInfoBuffer{0:5}EccDetected	Major (Schwerwiegend)	Überwachung	
	sifRaInfoLinkedListBuffer{0:5}EccCorrected	Major (Schwerwiegend)	Überwachung	sifRaInfoLink wurde mit eine getroffen. Je n HA-Richtlinie f müssen Sie de

				zurücksetzen. Leistungsprob führen.
	sifRacInfoLinkedListBuffer{0:5}EccDetected	Major (Schwerwiegend)	Überwachung	
	sifSegmentLinkedListBuffer{0:1}EccCorrected	Major (Schwerwiegend)	Überwachung	sifRacInfoLink wurde mit eine getroffen. Je n HA-Richtlinie f müssen Sie de zurücksetzen. Leistungsprob führen.
	sifSegmentLinkedListBuffer{0:1}EccDetected	Major (Schwerwiegend)	Überwachung	
10	ZielIndexTabelleParitätsfehler	Major (Schwerwiegend)	Überwachung	Der Speicher v Paritätsfehler Inhalte neu, un dass einige Pa fälschlicherwe entfernt wurde des Dopplers nicht erforderl
	GlobalZuLokalPortTabelle	Major (Schwerwiegend)	Überwachung	
	CPU-IndexTabelle	Major (Schwerwiegend)	Überwachung	
	HashtabelleA	Major (Schwerwiegend)	Überwachung	
	HashtabelleB	Major (Schwerwiegend)	Überwachung	
	MessageQueueInfo	Major (Schwerwiegend)	Überwachung	Die Nachrichte wurden mit ein getroffen. Dies Problem kann fehlgeleiteten OOB führen. D heilen und erfo Reset, da neu Einträge hier c überschreiben
	MessageQueueLinkBuffer	Major (Schwerwiegend)	Überwachung	

Andere Stacking-Register

- SifRacStatus
- SifStatistik
- SifRacEingefügt
- SifRacCopiedCnt
- SifRacPmaControl
- SifBallotWatchDogTimer
- SifPbcSifErrorCnt
- SifMessageStatus
- SifControl
- SupStackInterfaceControl
- SifSifPbcCnt0
- SifSifPbcCnt1
- SifSifPbcDroppedCnt
- SifSerdesHssMakroStatus
- SifSerdesHssChannelStatusRx
- SifSerdesHssChannelStatusTx

um die Details für jedes Register zu verstehen.

CLI zur Überwachung des Zustands von Stack-Ports:

```
show platform hardware fed switch <> fwd-asic register read registername SifSerdesHssMacroStatus
show platform hardware fed switch <> fwd-asic register read registername SifInfo
show platform hardware fed switch <> fwd-asic register read registername SifRacStatus
show platform hardware fed switch <> fwd-asic register read registername SifRacControl
show platform hardware fed switch <> fwd-asic register read registername SifExceptionInterruptA8
show platform hardware fed switch <> fwd-asic register read registername SifExceptionInterruptA4
show platform hardware fed switch <> fwd-asic register read registername SifStatistics
show platform hardware fed switch <> fwd-asic register read registername SifRacInsertedCnt
show platform hardware fed switch <> fwd-asic register read registername SifRacCopiedCnt
show platform hardware fed switch <> fwd-asic register read registername SifRacPmaControl
```

```
show platform hardware fed switch <> fwd-asic register read registername SifBallotWatchDogTimer
show platform hardware fed switch <> fwd-asic register read registername SifPbcSifErrorCnt
show platform hardware fed switch <> fwd-asic register read registername SifMessageStatus
show platform hardware fed switch <> fwd-asic register read registername SifControl
show platform hardware fed switch <> fwd-asic register read registername SupStackInterfaceControl
show platform hardware fed switch <> fwd-asic register read register-name SifSifPbcCnt0
show platform hardware fed switch <> fwd-asic register read register-name SifSifPbcCnt<>
show platform hardware fed switch <> fwd-asic register read registername SifPbcDroppedCnt
show platform hardware fed switch <> fwd-asic register read registername SifSerdesHssChannelStatusRx
show platform hardware fed switch <> fwd-asic register read registername SifSerdesHssChannelStatusTx
show platform hardware fed switch <> fwd-asic register read registername SifRacDataCrcErrorCnt
show platform hardware fed switch <> fwd-asic register read registername SifgRacRwCrcErrorCnt
show platform software sif switch <> R0 counters
show platform software sif switch <> R0 exception
```

Lesen von Registern aus dem Linux-Kernel

Wenn Sie sich in der Linux Shell befinden, fahren Sie mit dem nächsten Skript fort:

<#root>

```
[Switch_2_RP_0:~]$ dope.sh Num Asics: 0 Cat9300 platform dope vft ***** DOPpler Examine
```

ASIC in Dope.sh ändern

Das vorherige Skript liest den Schalter 1, im Grunde Null. Ändern Sie diese Einstellung, indem Sie dieses Skript ausführen:

```
dope[0,0]> asic 1 <--- changes to asic 1
dope[1,0]>
```




Hinweis: Dope.sh (Doppler-Shell) ist die niedrigste Ebene in der Hardware-Programmierung. So lesen Sie die Klingelwerte direkt von der Hardware. Verwenden Sie die **anderen Stapelregister** im vorherigen Skript nach dem Befehl,rdsp um die detailliertesten Daten abzurufen (falls erforderlich).

Probleme beim automatischen Neuladen

Bei jedem automatischen Neuladen (**kein Crashdump/system_report generiert**) gibt es Crash-Trace-Protokolle, die bestimmte Dateien anzeigen, um weitere Informationen zu erhalten, was das Ereignis verursachen könnte.

Schritt 1

Zuerst können wir **stack_mgr_R0** betrachten und aus seiner Perspektive den Grund für das Neuladen erkennen. Beispiele:

```
2018/04/26 19:26:01.363 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UUID: 0, ra: 0, TID: 0 (note): Entity RIPC channel terminated
2018/04/26 19:26:01.363 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UUID: 0, ra: 0, TID: 0 (note): Entity Mgr server connection dead
2018/04/26 19:26:01.363 [stack_mgr_R0-0]{1}: [mqipc] [14948]: UUID: 0, ra: 0, TID: 0 (ERR): record read: error [104] reading notification
2018/04/26 19:26:01.363 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UUID: 0, ra: 0, TID: 0 (ERR): stack MQIPC reader channel disconnected
2018/04/26 19:26:01.534 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UUID: 0, ra: 0, TID: 0 (note): reload req message swnum 255 REQ
2018/04/26 19:26:01.534 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UUID: 0, ra: 0, TID: 0 (note): STACK_WAIT_RELOAD_ACT_TIMER Timer not running
2018/04/26 19:26:01.534 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UUID: 0, ra: 0, TID: 0 (note): All switches acked. Reloading local chassis
2018/04/26 19:26:01.534 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UUID: 0, ra: 0, TID: 0 (note): Chassis 1 reloading, reason - Reload command
2018/04/26 19:26:01.534 [stack_mgr_R0-0]{1}: [errmsg] [14948]: UUID: 0, ra: 0, TID: 0 (): (1): %STACKMGR-1-RELOAD: Reloading due to reason Reload command
/tmp/stack_mgr_R0-0.14948_0.20180426172950.bin: DECODE(416:416:0:13)
```

Schritt 2

Wir können jetzt zu pvp-Protokollen wechseln. Verwenden Sie die Zeitstempel, die aus **stack_mgr_R0** extrahiert wurden (insbesondere, wenn ein Neuladen stattgefunden hat), und sehen Sie sich **pvp_F0** und **pvp_R0 an**, um zu ermitteln, wann die Terminierungssequenz des Prozesses gestartet wurde, bevor die gesamte Neuladeorchestrierungssequenz ausgeführt wird. Beispiele:

```
2018/04/25 18:17:39.842 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): IMOTIFY /tmp/rp/pvp/process/ DELETE linux_iosd_image*rp_0_0#10647
2018/04/25 18:17:39.843 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): PROCESS: dead or held-down, process linux_iosd_image fsb rp_0_0#0 pid 10647
2018/04/25 18:17:39.843 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): PROCESS: failure action expected 'critical', scope 'per_bay'
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): Checking exit code 70 file /tmp/rp/pvp/process_state/linux_iosd_image*rp_0_0#10647_exitcode
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): PROCESS: exit code for linux_iosd_image was 70
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): PROCESS: exit with code RELOAD_CHASSIS
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (info): (std): PROCESS: touch /tmp/rp/pvp/work/switchover_done_sent_inel
2018/04/25 18:17:39.862 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): quiet_death file NOT exists (/tmp/rp/chasfs/etc/quiet_death), its a crash, do sync issu crash file
*/flash/pvp.log" [Incomplete last line] 66 lines, 11270 characters
```



Hinweis: Es kann **pvp_F0** und **pvp_R0** anzeigen.

```
-rw-r--r-- 1 root root 4476 Apr 24 21:38 pvp_F0-0.13136_0.20180424012429.bin.gz
-rw-r--r-- 1 root root 4405 Apr 24 01:12 pvp_F0-0.14840_0.20180403072736.bin.gz
-rw-rw-rw- 1 root root 10094 Apr 25 22:36 pvp_R0-0.8079_0.20180425223247.bin.gz
-rw-rw-rw- 1 root root 2938 Apr 26 17:26 pvp_R0-0.8079_1.20180425223618.bin.gz
```



Hinweis: Stellen Sie sicher, dass Sie beide überprüfen, da Sie sehen konnten, dass der **linux_iodd_image**-Prozess in **pvp_R0** endet, aber ein anderer Prozess innerhalb von **pvp_F0** zuvor beendet wurde. Dies ist ein Schlüsselfaktor, weil der erste Prozess, der getötet wird. Dann kann es auf den Auslöser des Problems verweisen.

Schritt 3

Innerhalb von **pvp_F0** und **pvp_R0** gibt es auch einen Exit-Code, der nach dem Prozess tot/niedergehalten wird. Bei realen Prozessabstürzen werden die Exitcodes 129 und so verwendet. Auf diese Weise erkennt pvp, dass **crashdump/system_report** erstellt werden muss. Ohne **crashdump/system_report** ist der Exitcode normalerweise Null. Beispiele:

```
2018/04/25 18:17:39.843 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): PROCESS: failure action expected 'critical', scope 'per_bay
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): Checking exit code 70 file /tmp/tp/pvp/process_state/linux_
load_image=rp_0_0#10647_exitcode
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UUID: 0, ra: 0, TID: 0 (note): PROCESS: exit code for linux_load_image was 70
```

Schritt 4

Nachdem Sie den verantwortlichen Prozess identifiziert haben, gehen Sie zu den prozessbezogenen btrace-Protokollen, und überprüfen Sie, ob weitere Details vorhanden sind.

Timeouts/Neuladevorgänge bei Stack-Elementen - Anwenderbericht

Ein fehlerhaftes Kabel zwischen zwei Switches kann dazu führen, dass ein beliebiger Switch im Stack aufgrund verlorener Keepalives neu geladen wird.

Symptome

Stack-Traces oder Switches, bei denen das Problem aktiv auftritt, führen zu folgenden Fehlern:

- 9300-1# show platform software trace message stack-mgr switch active R0 | inc reagiert nicht
- 2018 <<tel:2018>>/05/10 13:57:30.397 [stack_mgr] [24459]: UUID: 0, ra: 0, TID: 0 (Hinweis): Peer 4 reagiert nicht, für 8000 <<tel:8000>> ms Bookkeep=3EFDD last_msg = 3EFD5
- 2018 <<tel:2018>>/05/10 13:57:29.396 [stack_mgr] [24459]: UUID: 0, ra: 0, TID: 0 (Hinweis): Peer 6 antwortet nicht, für 8000 <<tel:8000>> ms Bookkeep=3EFDC last_msg = 3EFD4

Die Broschüre prüft jede Sekunde, wann sie das letzte Mal von jedem Switch im Stack gehört wurde (aus der Perspektive des Switches, auf dem die Broschüre ausgeführt wird). Nach 8000 ms ohne Keepalives drucken wir Spuren, die man noch nicht gehört hat. Bei einer Geschwindigkeit von 16000 ms werden die fraglichen Switches bei Verlust der Keepalives neu geladen.

```
9300-1#sh switch stack-ports sum Load for five secs: 8%/4%; one minute: 9%; five minutes: 9% Time source is NTP, 11:53:11.196 EDT Thu May 17 2018
```

Diese Zeitüberschreitung trat auch auf, wenn die Stack-Verbindung zwischen den beiden Switches sehr instabil war, was dazu führte, dass einer der Switches den Stack-Port als aktiv einstufte und den Datenverkehr weiterleitete, der andere jedoch für ausgefallen hielt.

Der Stapelring arbeitet sowohl im Uhrzeigersinn als auch im Gegenuhrzeigersinn. Der Datenverkehr im Ring kann über beide Pfade laufen, unabhängig vom Ziel. Das heißt, wenn der Switch 2 einen Keepalive an den Switch 1 senden möchte, kann er über die Switches 3, 4, 5, 6, 7, 8 und dann 1 oder einfach von 2 direkt zu 1 gelangen. Der Datenrückverkehr von Switch 1 zu Switch 2, der per Hash zu Switch 8 geleitet wird, wurde verworfen, was zu den im vorherigen Skript festgestellten Zeitüberschreitungen geführt hätte.

Abkürzungen

- OOB: Out-of-Band
- SIF: Stack-Schnittstelle
- RAC: Ring Access Controller

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.