

# Catalyst 9300/3850/3650でのサイレントリロードのトラブルシューティング

## 内容

---

### [はじめに](#)

[トラブルシューティング/表示コマンド](#)

[SifInfo](#)

[SifRacStatus](#)

[SifRacControl](#)

[SifExceptionInterruptA4](#)

[SifExceptionInterruptA8](#)

[その他のスタッキングレジスタ](#)

[Linuxカーネルからのレジスタの読み込み](#)

[Dope.shのASICの変更](#)

### [サイレントリロードの問題](#)

[手順 1](#)

[手順 2](#)

[手順 3](#)

[手順 4](#)

### [スタックメンバーのタイムアウト/リロード - ケーススタディ](#)

[症状](#)

### [短縮形](#)

---

## はじめに

このドキュメントでは、特にスタック構成のポートやケーブルの問題とサイレントリロードに関連する問題のコマンドやレジスタをトラブルシューティングする方法について説明します。

### トラブルシューティング/表示コマンド

有用なレジスタを収集して分析します（各ASICおよびコアに対して）。3つの主なものがあります。

- SifInfo
- SifRacStatus
- SifRacControl

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

## SifInfo

最初のビットは、ASICが使用可能かどうかを示します。0x1に設定されています。0x0に設定されている場合は、転送の問題があります。エラーカウンタまたはボックスがパケットを正しく回復できません。

```
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
nodeAllLinksAvailble : 0x1
nodeId             : 0x4 <---- asic ID (unique across all asics in the stack)
numNodes           : 0x8 <---- how many asics are there in whole stack
serdesSpeed        : 0x2
sifAllLinksAvailble : 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
```



注：各スタックケーブルには6つのラックリング（リングアクセスコントロール）があり、それぞれ40ギガで3つの発信/着信があります。WrappedAtRac 0 ~ 5は、スタックリンクがダウンしているかどうかにかかわらず対応します。良好な場合は0x0（asicあたり6つのリンク、発信3つ、着信3つ）と表示されます。たとえば、奇数は送信され、偶数は受信されます（またはその逆）。

---

## SifRacStatus

各Racを詳細にチェックするには、確認すべき重要な項目が表示されます。アクティブ /linkOk/syncOkビットは、特定のRacがリンクしているかどうかを示します（OKの場合は0x1と表示されます）。

```
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

Racの電源が切れているかどうかを確認します。greenPowerDisableパラメータを確認します。これは、すべてのRacs ( 少なくともNyquistプラットフォーム ) の0x0を示します。ローエンドボックスである3650スイッチなど、スタックケーブル自体のハードウェア制限により、Racsの電源がオフになるか、greenPowerDisableパラメータが0x1と表示されることが予想される例外がいくつかあります。スタックケーブルは、ASICごとに2つのRacだけをサポートします。残りの2台のRacの電源がオフになります。

```
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

これは、システムでリンクが変更されたためにトリガーされます ( アップ/ダウンの状況 )。割り込みはソフトウェアレベルで処理されます。リンク関連の変更があるかどうかを確認するために処理され、発行 ( ログ生成 ) されます。

```
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

これはハードウェア割り込みであり、バロテイングが行われた時に詳細を提供します ( バロテイング=ASIC初期化プロセス )。A8が完了すると、システムはASICの使用可能なビットが正しく設定されているかどうかを確認します。そうでない場合は、再びバロテイングが実行されます。



注：最大数に到達すると、スイッチがリロードされ、「HW available bit was not set」または「Balloting did not complete」というエラーが表示されます。

---

```
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
```

```

sifObFlushDropCntOverflow : 0x0
sifObStackSifCreditDropCntOverflow : 0x0
sifObStackSifMtuDropCntOverflow : 0x0
sifObSupSifMtuDropCntOverflow : 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

```

次のコマンドでは、SDPメッセージとSIF管理メッセージに関連するSIFカウンタが表示されます。失敗したメッセージがある場合は、そのメッセージに注目します。

```

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

割り込みがしきい値を超えた場合にのみ実行して情報を表示する追加のコマンドがあります。コマンドは、show platform software sif switch active R0 exceptionsです。割り込みに問題がない場合の出力を次に示します。

```

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

```

Switch#

割り込みが存在する場合、出力は次のスクリプトのようになります。割り込みが予想されるシナリオ ( ブートアップ、プラグ/プラグ抜きなど ) もあるので、実際に問題が発生して割り込みが連続する場合は、コマンドを数秒から数分間繰り返し実行します。

```
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
```

次の表は、show platform software sif switch active R0 exceptionsからの最も一般的なSIF例外の詳細を示しています。

例外番号	フィールド名	重大度	用途	説明
0	sifRac{0:5}PmaTransmitFifoSpill{0:3}	major	統計情報	これは、システムクロックとSERDESクロック間のプッシュプルFIFOがオーバーフローした場合に起動されます。これは発生しません。発生している場合は、Serdesクロックが ( プログラミングまたは障害のあるSerdesによって ) 無効にされていることを示している可能性があります。これがプログラミングの問題によるものでない場合は、大きな問題です。しかし、SIFは自己修復します。そして、小さな問題の最終的な結果は、セグメントの損失か、極端な場合は再初期化です。これが小さな問題ではなく、まだ



				発生している場合は、このCHIEFを処理した後に再起動し、この時点で状況がまだ発生していることを伝えます。この送信リンクはtoastです。
1	sifRac{0:5}PmaReceiveFifoSpill{0:3}	major	統計情報	これは、システムクロックとSERDESクロック間のプッシュプルFIFOがオーバーフローした場合に起動されます。これは発生しません。発生している場合は、Serdesクロックが（プログラミングまたは障害のあるSerdesによって）無効にされていることを示している可能性があります。これがプログラミングの問題によるものでない場合は、大きな問題です。しかし、SIFは自己修復します。そして、小さな問題の最終的な結果は、セグメントの損失か、極端な場合は再初期化です。これが小さな問題ではなく、まだ発生している場合は、このCHIEFを処理した後に再起動し、この時点で状況がまだ発生していることを伝えます。この送信リンクはtoastです。
2	sifRac{0:5}SerdesLossOfLock{0:3}	major	統計情報	sifRac{0:5}PmaReceiveFifoSpill{0:3}と関連付けて使用し、受信したSerdesクロックの状態を通知します。ただし、この状態は正常な動作状態ではありません。仕様外の場合、IdleDensity Timerは差異を補正できません。一般に、これは受信側のSERDESが正しく動作しているという仮定が実際に正しいことを確認するための問題チェッカーです。
3	sifRac{0:5}ClockLossOfLock{0:3}	major	統計情報	sifRac{0:5}PmaReceiveFifoSpill{0:3}と関連付けて使用し、受信したSerdesクロックの状態を通知します。ただし、この状態は正常な動作状態ではありません。仕様外の場合、IdleDensity Timerは差異を補正できません。一般に、これは受信側のSERDESが正しく動作しているという仮定が実際に正しいことを確認するための問題チェッカーです。
4	sifRac{0:5}syncOkChange	マイナー	モニタ	リンクフラップの表示
	sifRac{0:5}linkOkChange	マイナー	モニタ	リンクフラップの表示

			タ	
	sifRac{0:5}linkedListSpill	major	モ ニ タ	リオーダー・ アルゴリズムの一部であるRACリンク・ リストが、可能な最大エントリ数を超えています。これは本当に問題があり、このRAC上のデータセグメントとOOBメッセージがテールドロップしていることを意味します。これは、スタックが正しく設定されていない場合、またはリンクされたリストにソフトウェアが発生していない限り発生しません。例外9および10を参照してください。
	sifRac{0:5}トランジットFifoSpill	major	統 計 情 報	SIF経由で他のノードにデータを移動するために応答されるtransitFifoがオーバーフローした原因は、このスイッチとネイバーの対比での、IdleDensityTimer w.r.t.の実際のSerdes clock ppm(parts per million)オフセットへの誤設定にある可能性があります。
5	sifRac{0:5}missingToken	major	統 計 情 報	Stack conchシェルが失われ、破損し、再分解されました。これは、スタック上のビットヒットがSifTokenDescをヒットしたことを示している可能性があります。あり得ないことだ。SIFは、さまざまな方法でこれを処理するように設定できます。再投票して最初からやり直るか、トークンを再導入するか、SIFの再導入を許可します。
	sifRac{0:5}duplicateToken	major	統 計 情 報	
	sifRac{0:5}tokenDeployed	info	統 計 情 報	
6	sifRac{0:5}RwCrcErrorCntOverflow	マイ ナー	統 計 情 報	おそらく、スタックケーブルまたはネイバーボックスのすべてのインジケータが比較されています。主にデバッグのために、この詳細に分かれています。通常の実行中は、syncOkChangeとlinkOkChangeだけが知っておく必要があります。LONG-TERM-BERを収集する場合、ビットエラーを適切にカウントするには、カウンタがロールオーバーするときにこれらのカウンタを監視

				してカウントする必要があります。 invalidRwまたはpcsCodeWordErrorが存在する場合は、CRCがチェックされない可能性があります。このようにして、BERのすべてのレジスタを合計できます。
	sifRac{0:5}DataCrcErrorCntOverflow	マイナー	統計情報	
	sifRac{0:5}InvalidRwErrorCntOverflow	マイナー	統計情報	
	sifRac{0:5}PcsCodeWordErrorCntOverflow	マイナー	統計情報	
7	sifRac{0:5}RdispErrorCntOverflow	マイナー	統計情報	
	sifRac{0:5}PrbsUnLockErrorCntOverflow	info	統計情報	最適なプログラムを見つけるためにIBM HSSマクロの最適な構成を見つけるのに役立つ統計を表示します。
	sifRac{0:5}PrbsBitErrorCntOverflow	info	統計情報	
	sifRac{0:5}ErrorCaptureCntOverflow	info	ラボ	エラーが発生したリングワードの形式をキャプチャする統計情報を表示して、スタックで何が起きているのかを調べます。
8	sifRacInfoLinkedListInitDone{0:5}	info	モニタ	RACリンクリストのハードウェア初期化が完了しました。
	sifDroppedSegmentCntOverflow	info	統計情報	
	sifPbcInconsistentSopEopCntOverflow	info	統計情報	最悪のシナリオ。PBCからのプロトコルフォームに従ってデータの到着を確認します。
	sifPbcErrorCntOverflow (オーバーフロー)	info	統	

			計 情 報	
sifSupInconsistentSopEopCntOverflow	info	統 計 情 報	最悪のシナリオ。SUP(OOBM)からのプロトコルフォームに従ってデータの到着を確認します。	
sifSupErrorCntOverflow	info	統 計 情 報		
sifReorderInconsistentSopEopCntOverflow	info	統 計 情 報	欠落セグメントインジケータがロールオーバーしたことを示します。	
sifDebugSent	info	ラ ボ	スタックへのデバッグセグメントの挿入の通知を起動します。	
sifメッセージ送信	info	ラ ボ	OOBMは自動化された性質を持つため、実際に役に立つのはラボ環境のみです。	
sifMessage受信	info	ラ ボ		
sifMessageドロップ	info	ラ ボ		
sifMessageReceiveBufferCreditsEmpty	マイ ナー	モ ニ タ	これが発生した場合は、クレジットを更新してください。クレジット・レベルはアクティブに監視され、トリップは発生しません。	
sifUnmappedDestIndex	マイ ナー	統 計 情 報	Copy/Stripの実行中にdestIndexをマップできず、portCopyが'0'に設定され、portStripが'1'に設定されました。これは、設定の問題を示しています。	
sifSegmentBuffer{0:1}linkedListSpill	major	モ ニ タ	リオーダーの一部であるセグメントリンクリストが、可能な最大入力数を超過しています。これは、リオーダーがデータセグメントとOOBメッセージをテールドロップしていることを示します。これは、スタックが正しく設定されていない場合、またはリンクされたリストにソフトウェアエラーが発生していない限り、発生しません。例外9および10を参照してください。	
sifSegmentBufferLinkedListInitDone{0:1}	info	モ ニ タ	セグメントリンクリストのハードウェア初期化が完了しました。	
sifBallotDone	info	モ	指示バロテイングが完了しました。	

			ニ タ	
	sifBallotSpeedChangeNeeded	info	モ ニ タ	最後の成功した投票から、スタックリンクに新しい速度が必要です。これは、ノードがスタックに入り、スタック速度の動的な値を変更したことを意味します。現在の速度より遅くどちらかで、スタックを調整する必要があります。以前よりも高速になることもできますこれは、新しい短いケーブルが原因である可能性があります。
	sifEastNeighborChange	info	モ ニ タ	スタックの起動、マージ、ラップシナリオを監視します。
	sifWestNeighborChange	info	モ ニ タ	
	sifNodeIdChanged	info	モ ニ タ	前回の投票の結果として SifInfo.nodeIdが変更されたことを示します。
	sifStackTopologyChange	info	モ ニ タ	スタックの起動、マージ、ラップシナリオを監視します。
9	sifRacInfoBuffer{0:5}EccCorrected	major	モ ニ タ	sifRacInfoBuffer{0:5}がソフトウェアでヒットしました。これは悪い問題ですが、最悪の結果は、不正なパケットが発生したり、出力データパスでパケットが後でドロップしたりすることです。ここではドップラーのリセットは不要です。
	sifRacInfoBuffer{0:5}Ecc検出	major	モ ニ タ	
	sifRacInfoLinkedListBuffer{0:5}EccCorrected	major	モ ニ タ	sifRacInfoLinkedListBuffer{0:5}がソフトウェアでヒットしました。このソフトウェアロードのオーバーアーチHAガイドラインに応じて、ドップラーをリセットします。これにより、SifReorderのパフォーマンスの問題が発生する可能性があります。
	sifRacInfoLinkedListBuffer{0:5}Ecc検出	major	モ ニ タ	
	sifSegmentLinkedListBuffer{0:1}Ecc訂正済み	major	モ ニ タ	sifRacInfoLinkedListBuffer{0:5}がソフトウェアでヒットしました。このソフトウェアロードのオーバーアーチHAガイ

			ドラインに応じて、ドップラーをリセットします。これにより、SifReorderのパフォーマンスの問題が発生する可能性があります。
	sifSegmentLinkedListBuffer{0:1}Ecc検出	major	モニタ
10	DestinationIndexTableParityError	major	モニタ メモリにパリティエラーが発生しました。コンテンツをリロードし、結果として一部の packets が誤ってコピーまたは削除される可能性があることを認識します。ドップラーのリセットはおそらく必要ありません。
	GlobalToLocalPortTable	major	モニタ
	CpuIndexTable	major	モニタ
	ハッシュテーブルA	major	モニタ
	ハッシュテーブルB	major	モニタ
	メッセージキューFifo	major	モニタ メッセージ制御メモリにソフトエラーが発生しました。これは一時的な問題であり、OOBの転送ミスや順序の誤りにつながる可能性があります。これは自己修復が可能で、ここでエントリの新しいユーザが古いユーザを上書きできるため、ドップラーをリセットする必要はありません。
	メッセージキューリンクバッファ	major	モニタ

これは「EDCS-757121:NG3K SIF Driver Software Functional Specification」に記載されています。

その他のスタッキングレジスタ

- SifRacStatus
- Sif統計情報
- SifRacInsertedCnt

- SifRacCopiedCnt
- SifRacPmaControl
- SifBallotWatchDogTimer
- SifPbcSifErrorCnt
- SifMessageStatus
- SifControl
- SupStackインターフェイス制御
- SifSifPbcCnt0
- SifSifPbcCnt1
- SifSifPbcDroppedCnt
- SifSerdesHssMacroStatus です
- SifSerdesHssChannelStatusRx ( 受信 )
- SifSerdesHssChannelStatusTx

各レジスタの詳細を理解する。

スタックポートの状態を監視するCLI:

show platform hardware fed switch <> fwd-asic register read register-name SifSerdesHssMacroStatus ( プラットフォームのハードウェア供給スイッチを表示 )

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

show platform hardware fed switch <> fwd-asic register read register-name SifRacStatus ( プラットフォームのハードウェア供給スイッチ  
<> fwd-asicレジスタ読み取りレジスタ名SifRacStatus )

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

show platform hardware fed switch <> fwd-asic register read register-name SifExceptionInterruptA8

show platform hardware fed switch <> fwd-asic register read register-name SifExceptionInterruptA4 ( プラットフォームのハードウェア供給スイッチ<> fwd-asicレジスタ読み取りレジスタ名 )

show platform hardware fed switch <> fwd-asic register read register-name SifStatistics ( プラットフォームのハードウェア供給スイッチ  
<> fwd-asicレジスタ読み取りレジスタ名SifStatistics )

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

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

show platform hardware fed switch <> fwd-asic register read register-name SifRacPmaControl ( プラットフォームのハードウェア供給スイッチ<> fwd-asicレジスタ読み取りレジスタ名SifRacPmaControl )

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

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

```
show platform hardware fed switch <> fwd-asic register read register-name SifMessageStatus ( プラットフォームのハードウェア供給スイッチ<> fwd-asicレジスタの読み取りレジスタ名SifMessageStatus )
show platform hardware fed switch <> fwd-asic register read register-name SifControl
show platform hardware fed switch <> fwd-asic register read register-name 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 register-name SifSifPbcDroppedCnt
show platform hardware fed switch <> fwd-asic register read register-name SifSerdesHssChannelStatusRx ( プラットフォームのハードウェア供給スイッチ<> fwd-asicレジスタ読み取りレジスタ名SifSerdesHssChannelStatusRx )
show platform hardware fed switch <> fwd-asic register read register-name SifSerdesHssChannelStatusTx ( ハードウェアフィード型スイッチの表示<> fwd-asicレジスタ読み取りレジスタ名SifSerdesHssChannelStatusTx )
show platform hardware fed switch <> fwd-asic register read register-name SifRacDataCrcErrorCnt
show platform hardware fed switch <> fwd-asic register read register-name SifgRacRwCrcErrorCnt
show platform software sif switch <> R0カウンタ
show platform software sif switch <> R0例外
```

Linuxカーネルからのレジスタの読み込み

Linuxシェルに入ったら、次のスクリプトに進みます。

<#root>

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

Dope.shのASICの変更

前のスクリプトは、スイッチ1、ASIC 0を読み取っています。次のスクリプトを実行して、これを変更します。

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





注: Dope.sh (Doppler shell)は、ハードウェアプログラミングの最下位レベルです。このようにして、リング値をハードウェアから直接読み取ります。rdspコマンドの後に前述のスキ립トでその他のスタッキングレジスタを使用すると、最も詳細なデータを取得できます ( 必要な場合 )。

---

#### サイレントリロードの問題

サイレントリロード(`crashdump/system_report`が生成されない)が発生するたびに、一部の特定ファイルが表示されるクラッシュトレースログが発生し、イベントの原因に関連する詳細情報を取得します。

手順 1

最初にstack\_mgr\_R0を調べ、リロードの理由をその観点から確認します。たとえば、次のとおりです。

```
2018/04/26 19:26:01.363 [stack_mgr_R0-0]{1}: [stack_mgr] [14948]: UID: 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]: UID: 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]: UID: 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]: UID: 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]: UID: 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]: UID: 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]: UID: 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]: UID: 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]: UID: 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)
```

## 手順 2

これでpvpログに移動できます。stack\_mgr\_R0から抽出されたタイムスタンプ（特にリロードが発生した場合）を使用し、pvp\_F0およびpvp\_R0を確認して、すべてのリロードオーケストレーションシーケンスを実行する前にプロセス終了シーケンスがいつ開始したかを特定します。たとえば、次のとおりです。

```
2018/04/25 18:17:39.842 [pvp_R0-0]{1}: [pvp] [8311]: UID: 0, ra: 0, TID: 0 (note): IMOTIFY /tmp/rp/pvp/process/ DELETE linux_iods_image=rp_0_0%#10647
2018/04/25 18:17:39.843 [pvp_R0-0]{1}: [pvp] [8311]: UID: 0, ra: 0, TID: 0 (note): PROCESS: dead or held-down, process linux_iods_image= fsb rp_0_0%# pid 10647
2018/04/25 18:17:39.843 [pvp_R0-0]{1}: [pvp] [8311]: UID: 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]: UID: 0, ra: 0, TID: 0 (note): Checking exit code 70 file /tmp/rp/pvp/process_state/linux_iods_image=rp_0_0%#10647_exitcode
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UID: 0, ra: 0, TID: 0 (note): PROCESS: exit code for linux_iods_image was 70
2018/04/25 18:17:39.858 [pvp_R0-0]{1}: [pvp] [8311]: UID: 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]: UID: 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]: UID: 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
```



注：これはpvp\_F0とpvp\_R0を表示できます。

---

```
-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
```



注：両方をチェックしてください。linux\_iosd\_imageプロセスはpvp\_R0で終了していますが、以前にpvp\_F0内で別のプロセスが終了していることが確認できます。これは重要な要素です。なぜなら、最初のプロセスが終了してしまうからです。これにより、問題のトリガーを示すことができます。

---

### 手順 3

pvp\_F0およびpvp\_R0内には、プロセスが停止または停止した後に提供される終了コードもあります。実際のプロセスのクラッシュの場合は、終了コード129などが使用されます。このようにして、pvpはcrashdump/system\_reportを作成する必要があることを認識します。crashdump/system\_reportがない場合、終了コードは通常ゼロです。たとえば、次のとおりです。

```
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
```

#### 手順 4

問題を引き起こしているプロセスを特定した後、btraceログに関連するプロセスに移動し、詳細を確認します。

#### スタックメンバーのタイムアウト/リロード - ケーススタディ

2台のスイッチ間の1本の不良ケーブルが原因で、スタック内のどのスイッチでも、キープアライブの喪失によるリロードが発生する可能性があります。

#### 症状

問題が発生しているスタックトレース (スイッチ) では、次のエラーが発生します。

- 9300-1# show platform software trace message stack-mgr switch active R0 | Incが応答していません
- 2018 <tel:2018>/05/10 13:57:30.397 [stack\_mgr] [24459]: UUID: 0, ra: 0, TID: 0 (注): Peer 4 not responding, for 8000 <tel:8000>ミリ秒Bookkeep=3EFDD last\_msg = 3EFD5
- 2018 <tel:2018>/05/10 13:57:29.396 [stack\_mgr] [24459]: UUID: 0, ra: 0, TID: 0 (注): Peer 6 not responding, for 8000 <tel:8000>ミリ秒ブックキープ=3EFDC last\_msg = 3EFD4

ブックキープは、スタック内の各スイッチから最後に受信した情報を1秒ごとにチェックします (ブックキープを実行しているスイッチから見て)。8000ミリ秒のキープアライブがない状態から、ピアが受信されていないトレースの出力を開始します。16000ミリ秒で、問題のスイッチはキープアライブが失われたことを確認してリロードします。

```
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
```

このタイムアウトは、スイッチ間のスタックリングが大量に不安定な場合にも見られました。あるスイッチではスタックポートがアップ状態でトラフィックを渡すことができると認識され、他のスイッチではダウンしたと認識されます。

スタックリングは、時計回りと反時計回りの両方の方向に動作します。リング上のトラフィックは、宛先に関係なく、どちらのパスでも通過できます。つまり、スイッチ2がスイッチ1にキープアライブを送信する場合、スイッチ3、4、5、6、7、8を経由して1に到達するか、直接2から1に到達します。スイッチ1からスイッチ2へのリターントラフィックで、スイッチ8へのハッシュが発生すると廃棄され、前のスクリプトでタイムアウトが発生します。

## 短縮形

- OOB : アウトオブバンド
- SIF:Stack Interface ( スタックインターフェイス )
- RAC:Ring Access Controller ( リングアクセスコントローラ )

## 翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。