

通过系统报告对堆栈管理器重新加载的3650/3850进行故障排除

目录

[简介](#)

[背景信息:](#)

[系统报告与交换机报告](#)

[收集系统/交换机报告的位置](#)

[系统报告中的相关部分](#)

[故障类型](#)

[重置原因：\[stack-manager\]请求重置/重新加载。\[ISSU不兼容\]](#)

[重置原因：服务\[iosd\] pid:\[xyz\]异常终止\[11\]。](#)

[hap sup reset:重置原因：\[stack-manager\]请求重置/重新加载。\[堆栈合并\]](#)

[hap sup reset:原因代码：\[4\]重置原因：\[堆栈管理器\]请求重置/重新加载。\[堆栈合并，因不兼容\]](#)

[重置原因：\[stack-manager\]请求重置/重新加载。\[ASIC投票后遇到错误的邻居\]](#)

[hap sup reset:原因代码：\[4\]重置原因：\[堆栈管理器\]请求重置/重新加载。\[同时丢失主用和备用\]](#)

[hap sup reset:原因代码：\[1\]重置原因：\[堆栈管理器\]请求重置/重新加载。\[Keepalive Timeout\]](#)

[hap sup reset:重置原因：\[stack-manager\]请求重置/重新加载。\[Reload Command\]](#)

[相关错误](#)

[诊断潜在的堆栈布线/端口问题](#)

[其他提示](#)

[1. 存档Crashinfo目录](#)

[2. 恢复不稳定的堆栈](#)

[3. 手动生成系统报告](#)

简介

在NGWC交换平台上，通常使用stackwise技术通过系统报告排除堆栈在没有崩溃时重新加载的故障。当前文档仅限于系统报告的使用，本指南旨在说明如何利用这些报告诊断通常与堆叠问题有关的问题。本指南特别适用于运行支持堆叠功能的IOS-XE的Catalyst 3650/3850交换架构。

堆叠wise技术的大多数问题源自堆叠内成员之间的通信问题。成员之间信息的任何不一致或连接的丢失都可能导致贯穿整个堆栈的问题，最终导致使用堆栈管理器重置。本文档将重点介绍堆栈管理器重新加载时出现的一些常见故障类型、系统报告的使用以及可用于诊断和不同类型问题的相关CLI。

背景信息:

系统报告与交换机报告

系统报告是成员对堆栈状态的感知的综合报告。这不是`crashinfo`（它将转储内存以供进一步调试），而是包含在IOS-XE下运行的各种服务的日志和调试信息的报告，这些信息对于开发以跟踪该服务的状态非常有用。当堆栈管理器重新加载交换机、发生进程崩溃或用户在实时操作期间手动生成

系统报告时，可以生成系统报告。

在许多情况下，堆栈中的一台交换机可能发生故障，但其余成员可能保持不变。为了收集指定时间堆栈状态的信息，引入了switch_reports，以便其余成员在检测到成员已关闭时生成一个。switch_report将是该成员如何感知堆栈当前状态的本地报告。

注意：这些报告会被编写和压缩，因此无法使用“more”将其打印到终端。它们需要从交换机转出并解压才能查看日志。

收集系统/交换机报告的位置

系统报告通常写在crashinfo中：堆栈中成员的目录。例如，如果有x成员交换机堆叠，则每台交换机将拥有自己的crashinfo目录，可使用“dir crashinfo-x”访问该目录，其中“x”对应于堆叠中的该成员。

```
3850#dir crashinfo-1:
```

```
crashinfo/
```

```
11 -rwx 355 201581407:48:17 -04:00 last_systemreport_log
```

```
12 -rwx 20141072401507:14:32 -04:00 system-report_1_20141015-111342-UTC.gz
```

```
3850#dir crashinfo-2:
```

```
crashinfo-2/
```

```
11 -rwx 357 201581407:50:49 -04:00 last_systemreport_log
```

```
12 -rwx 20141075134006:41:12 -04:00 system-report_1_20141015-104022-UTC.gz
```

注意：请务必为该堆栈中的每台交换机收集“dir crashinfo-x：”的输出，因为“show tech”不会列出可用的文件系统或crashinfo文件。您必须拥有堆栈中每个成员的完整图片。更新：从IOS-XE版本>3.6E的更新版本开始，show tech将反映“dir crashinfo：”+“show file systems”输出。参见[CSCun50428](#)。

系统报告中的相关部分

从TAC的角度来看，这些是系统报告中一些更常查看的条目，可帮助诊断特定问题的事件。此处包含开发人员可能希望查看的其他服务的其他日志。

日志文件:/mnt/pss/sup_sysmgr_reset.log

这是一个简短的输出，可以非常笼统地理解为何出现重置。请参阅以下故障类型部分，了解这些原因的含义和背景。

日志文件:IOS

这是从IOSd中维护的日志缓冲区。本节将介绍由用户或IOSd中生成的系统日志发出的任何命令。最

近的日志接近此输出的结尾。

跟踪缓冲区：stack-mgr-events

跟踪从堆栈管理器看到的事件，其中包括当其他成员加入/从堆栈中删除时，或消息通过的堆叠端口。

跟踪缓冲区：redundancy-timer-ha_mgr

显示堆栈中交换机之间的保持活动事件。时间戳有助于确定通信中断的开始时间。

故障类型

本部分将重点介绍堆栈管理器进程通常调用的系统报告中常见的一些重置。堆叠管理器是一个Linux进程，用于管理堆叠中的成员，并将监督堆叠中交换机之间角色的任何更改。如果堆栈管理器在初始化或角色选举期间检测到问题，它将向各个交换机发送重新加载信号，以便堆栈重置。下面还将列出与相应故障类型关联的已知错误。

注意：并非所有堆栈管理器重新加载都归因于软件问题。事实上，更常见的情况是，这些问题表现为基础硬件问题的次要/受害者问题。

重置原因：[stack-manager]请求重置/重新加载。[ISSU不兼容]

当尝试在堆栈中的所有成员之间同步活动上的配置时出现批量同步故障时，您可能会看到这种重置类型。正在检查日志文件：IOS或活动交换机的日志可能会突出显示促成此重置的配置。

重置原因：服务[iosd] pid:[xyz]异常终止[11]。

当交换机在IOSd进程中崩溃时，就会出现这种情况。查看任何crashinfo文件+核心转储的crashinfo目录可用于进一步调试此故障。

hap_sup_reset:重置原因：[stack-manager]请求重置/重新加载。[堆栈合并]

当有两台或多台交换机认为它们是堆栈的活动交换机时，会看到堆栈合并。当堆栈的环中出现断裂（即两根电缆从堆栈断开），导致主用和备用设备都失去与其他成员的通信时，就会发现这种情况。将已通电的交换机添加到现有堆叠可能导致堆叠合并，因为堆叠中将有两个活动交换机。

[CSCuh58098](#) — 当存在堆栈布线问题时，3850堆栈可能会重新加载

[CSCui56058](#) — 为堆栈电缆启用反跳逻辑

[CSCup53338](#) - 3850 IOSD崩溃 | Signal=SIGSEGV(11)@ pm_port_data_from_swidb

hap_sup_reset:原因代码：[4]重置原因：[堆栈管理器]请求重置/重新加载。[堆栈合并，因不兼容]

在堆栈中存在主用和备用交换机时，就会出现这种情况。如果主用交换机与备用交换机失去通信

, 则备用交换机将尝试接管为主用交换机, 即使主用交换机仍处于工作状态。

[CSCuo49555](#) /CSCup58016 - 3850/3650由于管理端口上的单播泛洪而崩溃

[CSCur07909](#) — 由于主用和备用连接丢失, 堆栈合并

重置原因: [stack-manager]请求重置/重新加载。[ASIC投票后遇到错误的邻居]

交换机在启动过程中参与ASIC投票, 以确定其堆叠中的相邻交换机。当计时器到期, 邻居可以声明自己, 或者在广播会话期间出现逻辑错误时, 可以看到此重置。在堆叠电缆故障(通过更换已解决)的情况下, 已发现这种情况。

[CSCun60777](#) — 由于ASIC投票后遇到邻居错误, 交换机重新加载

[CSCud93761](#) — 由于ASIC投票后遇到邻居错误, 交换机重新加载

hap_sup_reset:原因代码: [4]重置原因: [堆栈管理器]请求重置/重新加载。[同时丢失主用和备用]

这通常从堆栈上不处于活动或备用角色的成员中看到。当主用交换机发生故障时, 如果没有备用交换机来承担堆栈的主用角色, 则整个堆栈将重置。如果堆栈处于不稳定状态或冗余策略尚未同步, 则可以看到此情况。这可能是主用/备用交换机关闭原因的牺牲品, 也可能是冗余未正确同步的迹象。当在半环设置中配置堆栈时, 也可以看到这一点。

[CSCup53882](#) — 成员交换机在3850堆叠重新启动中 — [丢失主用和备用]

hap_sup_reset:原因代码: [1]重置原因: [堆栈管理器]请求重置/重新加载。[Keepalive_Timeout]

当未从堆栈中的交换机收到保持连接消息时显示。查看“跟踪缓冲区: redundancy-timer-ha_mgr”应显示保持连接消息的交换, 并提供通信中断开始时的时间视角。从堆栈的其余部分收集交换机报告并查看时间段内的日志可能有助于此处。

hap_sup_reset:重置原因: [stack-manager]请求重置/重新加载。[Reload Command]

这是非常直观的重置原因 — 当堆栈管理器收到可通过CLI或通过管理设备(SNMP)从外部调用的重新加载请求时, 会出现此情况。在[CSCuj17317](#)中, 这也将显示为发出“reload命令”。从日志文件: IOS:

```
CMD: 'reload'
%SYS-5-RELOAD: Reload requested by console. Reload Reason: Reload command.
%STACKMGR-1-RELOAD_REQUEST: 1 stack-mgr: Received reload request for all switches, reason
Reload command
%STACKMGR-1-RELOAD: 1 stack-mgr: Reloading due to reason Reload command
```

相关错误

[CSCur76872](#) — 当系统用完SDP缓冲区时, 堆栈管理器会关闭。

[CSCup49704](#) - 3850 FED崩溃 — 正在等待SPI通道FED_SPI_FLCD、FED_SPI_FAST ...

诊断潜在的堆栈布线/端口问题

症状1)堆栈布线问题的任何迹象在重置之前堆栈端口出现抖动时都会明显显示。查看“logfile:IOS报告在重置之前通常是一个很好的开始位置。以下示例显示了当前SW2和备用SW1上注册的堆叠端口抖动的情况。在重置的每个实例中，此相同的堆叠端口都抖动，并且通过更换堆叠电缆得到解决：

```
===== log file: IOS =====
.
.
Aug  8 21:40:14.532 UTC: %STACKMGR-1-STACK_LINK_CHANGE: STANDBY:1 stack-mgr: Stack port 1 on
switch 1 is down (SW1-1)
Aug  8 21:40:17.242 UTC: %STACKMGR-1-STACK_LINK_CHANGE: STANDBY:1 stack-mgr: Stack port 1 on
switch 1 is up (SW1-1)
Aug  8 21:46:11.194 UTC: %STACKMGR-1-STACK_LINK_CHANGE: 2 stack-mgr: Stack port 2 on switch 2
is down
Aug  8 21:46:12.937 UTC: %STACKMGR-1-STACK_LINK_CHANGE: 2 stack-mgr: Stack port 2 on switch 2
is up
Aug  8 21:48:23.063 UTC: %STACKMGR-1-STACK_LINK_CHANGE: 2 stack-mgr: Stack port 2 on switch 2
is down
Aug  8 21:48:24.558 UTC: %STACKMGR-1-STACK_LINK_CHANGE: 2 stack-mgr: Stack port 2 on switch 2
is up
Aug  8 21:50:40.666 UTC: %STACKMGR-6-SWITCH_REMOVED: 2 stack-mgr: Switch 1 has been removed
from the stack.
Aug  8 21:50:40.671 UTC: Starting SWITCH-DELETE sequence, switch 1
```

症状2)根据使用的stackwise设置(180、480和+)，每个端口ASIC的传输环数将不同。这些命令将轮询全局寄存器，以保持每个传输环的读取错误总数。“Port-asic 0”对应于堆栈端口1，“port-asic 1”对应于堆栈端口2。应为每个交换机发出此命令，并且任何计数增加的迹象都可以隔离端口或堆栈电缆是否出现问题。

在几分钟内可以多次收集这些数据，以比较计数中的增量：

```
show platform port-asic <0-1> read register SifRacDataCrcErrorCnt switch <switch#>
```

- 带有数据CRC错误的数段

```
show platform port-asic <0-1> read register sifRacRwCrcErrorCnt switch <switch#>
```

- 在任何失败的CRC检查时递增

```
show platform port-asic <0-1> read register SifRacPcsCodeWordErrorCnt switch <switch#>
```

- 在无效PCS代码、未知PCS码字时递增，检测到运行视差错误

```
show platform port-asic <0-1> read register SifRacInvalidRingWordCnt switch <switch#>
```

- 堆栈上的位错误导致环字CRC错误

对于Polaris (16.X代码)，命令如下：

```
show plat hardware fed sw <#/active/standby> fwd-asic register read register-name
SifRacDataCrcErrorCnt asic <0-1>
```

```
show plat hardware fed sw <#/active/standby> fwd-asic register read register-name
SifRacRwCrcErrorCnt asic <0-1>
```

```
show plat hardware fed sw <#/active/standby> fwd-asic register read register-name
SifRacPcsCodeWordErrorCnt asic <0-1>
```

show plat hardware fed sw <#/active/standby> fwd-asic register read register-name SifRacInvalidRingWordCnt asic <0-1>

以下是一个示例，在该示例中，您看到堆栈合并事件看到2个成员堆栈的两个成员都没有出现任何抖动堆栈端口的迹象。您会看到交换机1的堆叠端口1上的环[0]随着CRC的增加而增加，最终更换了堆叠电缆以解决此问题。

```
3850#$show platform port-asic 0 read register SifRacRwCrcErrorCnt switch 1
Load for five secs: 11%/4%; one minute: 11%; five minutes: 12%
Time source is NTP, 14:02:49.119 EDT Thu Aug 20 2015
```

For asic 0

```
SifRacRwCrcErrorCnt on Asic 0
[0]
      count 0x00000031 <<
[1]
      count 0x00000001
[2]
      count 0x00000000
[3]
      count 0x00000001
[4]
      count 0x00000000
[5]
      count 0x00000001
```

```
3850#$show platform port-asic 0 read register SifRacRwCrcErrorCnt switch 1
Load for five secs: 9%/4%; one minute: 11%; five minutes: 12%
Time source is NTP, 14:02:53.550 EDT Thu Aug 20 2015
```

For asic 0

```
SifRacRwCrcErrorCnt on Asic 0
[0]
      count 0x000000c9 <<
[1]
      count 0x00000001
[2]
      count 0x00000000
[3]
      count 0x00000001
[4]
      count 0x00000000
[5]
      count 0x00000001
```

注意：根据正在查看的寄存器，掩码在每种情况下可能不同。在上例中，掩码将包绕最后14位。因此，当计数器达到0x00003FFF时，它将回包为0x00000000。

其他提示

1. 存档Crashinfo目录

堆叠中的交换机越多，意味着收集的报告文件就越多。生成的报告数量很容易被淹没。组织对于隔离故障至关重要。尽可能使用每个交换机为给定事件编写报告文件的时间戳来查找一致性。从那里，要求从给定的交换机获取这些非常具体的报告，以便客户不上传多个文件。crashinfo目录也可以存档，以便客户可以发送包含感兴趣报告的单个存档。以下内容将在闪存目录中创建名为

“crashinfo-archive.tar”的存档：

```
F340.03.10-3800-1#archive tar /create ?  
WORD Tar filename
```

```
F340.03.10-3800-1#archive tar /create crashinfo-archive.tar ?  
WORD Dir to archive files from
```

```
F340.03.10-3800-1#archive tar /create crashinfo-archive.tar crashinfo ?  
WORD File or Dir  
<cr>
```

```
F340.03.10-3800-1#archive tar /create crashinfo-archive.tar crashinfo:
```

2.恢复不稳定的堆栈

在堆栈选举过程发生后，在启动期间，您可能会看到堆栈重新加载中有几个成员。如果重新加载的交换机认为自己处于活动状态，则这通常会导致堆栈合并事件并进入引导环路状态。在这种情况下，建议客户：

- 关闭整个堆栈的电源，并牢固地重新拔插所有堆栈电缆。
- 逐个打开堆栈中每个成员交换机的电源，直到所有成员都收敛到其预期状态。
- 如果成员未能加入堆栈，请从堆栈中删除此项，然后尝试将此个人作为独立设备启动以进一步排除故障。

3.手动生成系统报告

手动创建系统报告需要启用“service internal”。这会将系统报告作为文本文件写入，可以按交换机执行。

```
3800-1#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
3800-1(config)#service internal  
3800-1(config)#exit
```

```
3800-1#resource create_system_report ?  
WORD system report filename
```

```
3800-1#resource create_system_report sysreport.txt ?  
switch Switch number  
<cr>
```

```
3800-1#resource create_system_report sysreport.txt switch ?  
<1-1> Switch number
```

```
3800-1#resource create_system_report sysreport.txt switch 1 ?  
<cr>
```