ASR5x00 Session Manager-Aufgaben - Beschreibung der Funktions-, Absturz-, Wiederherstellungs- und Absturzprotokolle

Inhalt

Einführung

In diesem Dokument werden die Softwarezuverlässigkeit, Serviceverfügbarkeit und Failover-Funktionen für die Cisco Aggregation Services Router (ASR) der Serie 5x00 beschrieben und erläutert. Es stellt die Definition für einen Softwareabsturz auf ASR5x00 und die Auswirkungen des Softwareabsturzes dar. Der Artikel stellt fest, dass die ASR5x00 selbst bei unerwarteten Softwareabstürzen das Ziel einer "Carrier-Class"-Verfügbarkeit dank der inhärenten Ausfallsicherheit und Verfügbarkeitsfunktionen der Software erreichen kann. Der mobile Teilnehmer sollte nie über die Verfügbarkeit des Dienstes nachdenken müssen. Cisco verfolgt das Ziel, keine Sitzungsverluste aufgrund von Hardware- oder Softwareausfällen zu verursachen, die den Verlust eines kompletten Systems, also der Zuverlässigkeit der Sprachqualität, beinhalten. Die Funktionen für die Softwarezuverlässigkeit des ASR5x00 sind so konzipiert, dass sie die Ziele für die Dienstverfügbarkeit der Carrier-Klasse auch dann erreichen können, wenn im Netzwerk eines Betreibers unvorhergesehene Ausfälle auftreten können.

Softwarearchitektur: Ausgelegt auf Ausfallsicherheit

Der ASR5x00 umfasst eine Reihe von Softwareaufgaben, die auf die Packet Services Card (PSC)- oder Data Processing Card (DPC)- und System Management Card (SMC)- oder Management and I/O (MIO)-Karten verteilt sind und die für die Ausführung einer Vielzahl spezifischer Funktionen konzipiert sind.

Beispielsweise ist die Sitzungsverwaltungsaufgabe für die Behandlung der Sitzungen für eine Reihe von Teilnehmern und für die Ausführung von Inline-Diensten wie Peer-to-Peer (P2P), Deep Packet Inspection (DPI) usw. für Benutzerdatenverkehr zuständig. Die Verwaltungsaufgabe

"Authentication, Authorization, and Accounting (AAA)" ist für die Erstellung von Abrechnungsereignissen zuständig, um die Nutzung des Teilnehmerdatenverkehrs usw. aufzuzeichnen. Die Sitzungsmanager- und AAA-Verwaltungsaufgaben werden auf der PSC/DPC-Karte ausgeführt.

Die SMC/MIO-Karte ist für Betriebs- und Wartungsaufgaben (O&M) und plattformbezogene Aufgaben reserviert. Das ASR5x00-System ist praktisch in verschiedene Software-Subsysteme unterteilt, z. B. das Session-Subsystem zur Verarbeitung von Teilnehmersitzungen und das VPN-Subsystem, das für die Zuweisung von IP-Adressen, das Routing usw. verantwortlich ist. Jedes Teilsystem hat eine Controller-Aufgabe, die den Zustand des von ihm gesteuerten Teilsystems überwacht. Die Controller-Aufgaben werden auf der SMC/MIO-Karte ausgeführt. Die Sitzungsverwaltung und die Aufgaben des AAA-Managers werden zusammengefasst, um die Sitzungen eines Teilnehmers zu Steuerungs-, Datenverkehrs- und Abrechnungszwecken zu verwalten. Wenn die Sitzungswiederherstellung im System aktiviert ist, sichert jede Sitzungs-Manager-Aufgabe den Zustand ihrer Teilnehmerstaaten mit einer Peer-AAA-Manager-Aufgabe, die bei einem Absturz des Sitzungsmanagers wiederhergestellt wird.

Was ist ein Crash?

Eine Aufgabe im ASR5x00 kann möglicherweise abstürzen, wenn während des normalen Betriebs ein Fehler auftritt. Ein Absturz oder ein Softwarefehler im ASR5x00 ist definiert als *unerwartetes* Beenden oder Beenden einer Aufgabe im System. Ein Absturz kann auftreten, wenn der Softwarecode versucht, auf verbotene Speicherbereiche zuzugreifen (z. B. beschädigte Datenstrukturen), im nicht erwarteten Code eine Bedingung auftritt (z. B. eine ungültige Zustandsänderung) usw. Ein Absturz kann auch ausgelöst werden, wenn die Aufgabe für die Systemüberwachungsaufgabe nicht mehr reagiert und der Monitor versucht, die Aufgabe zu beenden und neu zu starten. Ein Absturzereignis kann im System auch explizit ausgelöst werden (im Gegensatz zu unerwarteten Ereignissen), wenn eine Aufgabe gezwungen ist, ihren aktuellen Zustand durch einen CLI-Befehl oder den Systemmonitor zu deaktivieren, um den Aufgabenstatus zu analysieren. Ein erwartetes Absturzereignis kann auch auftreten, wenn der Systemcontroller selbst neu startet, um eine Situation mit einer Manager-Aufgabe, die wiederholt fehlschlägt, möglicherweise zu korrigieren.

Auswirkungen eines Session Manager-Absturzes

Im normalen Betrieb verarbeitet ein Sitzungsverwalter eine Reihe von Teilnehmersitzungen und den zugehörigen Datenverkehr für die Sitzungen zusammen mit einer Peering-AAA-Managementaufgabe, die die Abrechnung für diese Teilnehmersitzungen übernimmt. Wenn ein Sitzungsmanager abstürzt, existiert er nicht mehr im System. Wenn die Sitzungswiederherstellung im System aktiviert ist, wird eine Standby-Sitzungsverwaltungsaufgabe ausgeführt, um auf derselben PSC/DPC-Karte aktiv zu werden. Diese neue Sitzungsverwaltungsaufgabe setzt die Teilnehmersitzungen bei der Kommunikation mit der Peer-AAA-Managementaufgabe wieder ein. Der Wiederherstellungsvorgang liegt zwischen 50 ms und einigen Sekunden, abhängig von der Anzahl der Sitzungen, die zum Zeitpunkt des Absturzes im Sitzungsmanager aktiv waren, und der CPU-Gesamtauslastung auf der Karte usw. Es besteht kein Verlust an Teilnehmersitzungen, die bereits im ursprünglichen Sitzungsmanager bei diesem Vorgang eingerichtet wurden. Alle Teilnehmersitzungen, die zum Zeitpunkt des Absturzes gerade eingerichtet wurden, werden wahrscheinlich auch aufgrund von Protokollübermittlungen usw. wiederhergestellt. Alle Datenpakete, die sich zum Zeitpunkt des Absturzes im System befanden, werden von den

kommunizierenden Einheiten der Netzwerkverbindung als Netzwerkverlust eingestuft und erneut übertragen. Die Verbindung wird dann vom neuen Sitzungsverwalter übernommen. Die Rechnungsinformationen für die vom Sitzungsmanager durchgeführten Sitzungen werden im Peer-AAA-Manager beibehalten.

Wann sollte sich der Betreiber Sorgen machen?

Wenn ein Sitzungsmanager-Absturz auftritt, erfolgt die Wiederherstellung wie oben beschrieben, und der Rest des Systems bleibt von diesem Ereignis unberührt. Ein Absturz in einem Sitzungsmanager hat keine Auswirkungen auf die anderen Sitzungsmanager. Wenn mehrere Sitzungsverwaltungsaufgaben *auf der gleichen PSC/DPC-Karte* gleichzeitig oder innerhalb von 10 Minuten voneinander abstürzen, kann es als Anleitung für den Operator zu Sitzungsverlusten kommen, da das System möglicherweise nicht in der Lage ist, neue Sitzungsmanager schnell genug zu starten, um die abgestürzten Aufgaben zu übernehmen. Dies entspricht einem Szenario mit zwei Fehlern, bei dem ein Sitzungsverlust auftreten kann. Wenn eine Wiederherstellung nicht möglich ist, wird der Sitzungsmanager einfach neu gestartet und kann neue Sitzungen akzeptieren.

Wenn ein bestimmter Sitzungsverwalter wiederholt abstürzt (z. B. immer wieder die gleiche Fehlerbedingung auftritt), nimmt die Sitzungs-Controller-Aufgabe Notiz und startet sich neu, um das Subsystem wiederherzustellen. Wenn die Sitzungs-Controller-Aufgabe das Sitzungs-Subsystem nicht stabilisieren kann und sich bei diesem Vorgang ständig neu startet, besteht der nächste Schritt der Eskalation darin, dass das System auf eine Standby-SMC/MIO-Karte umschaltet. Falls es unwahrscheinlich ist, dass keine Standby-SMC/MIO-Karte vorhanden ist oder beim Switchover ein Fehler auftritt, startet das System selbst neu.

Sitzungsmanager führen außerdem Statistiken für jeden Access Point-Namen (APN), alle Services, Funktionen usw. durch, die bei einem Absturz dauerhaft verloren gehen. Daher wird eine externe Einheit, die Bulkstats regelmäßig sammelt, einen Einbruch in die Statistik beobachten, wenn ein oder mehrere Abstürze auftreten. Dies kann als Dip in einer grafischen Darstellung der über eine Zeitachse gezeichneten Statistiken auftreten.

> **Hinweis**: Ein typisches Chassis mit 7-14 PSC- oder 4-10 DPC-Karten verfügt je nach Anzahl der PSC/DPC-Karten über etwa 120-160 Session Manager. Bei einem einzigen Absturz gehen etwa 1/40 oder 80· der Statistiken verloren. Wenn ein Standby-Sitzungsmanager die Kontrolle übernimmt, beginnt er erneut, die Statistiken von Null zu akkumulieren.

Woher weiß ich, ob ein Crash aufgetreten ist?

Ein Absturz löst ein SNMP-Trap-Ereignis für eine Netzwerküberwachungsstation aus, z. B. den Event Monitoring Service (EMS) und Syslog-Ereignisse. Die Abstürze im System können auch mit dem Befehl **show crash list** beobachtet werden. Beachten Sie, dass dieser Befehl sowohl unerwartete als auch erwartete Absturzereignisse wie oben beschrieben auflistet. Diese beiden Arten von Crash-Ereignissen können mithilfe eines Headers unterschieden werden, der jeden Crash beschreibt.

Ein Aufgabenabsturz gefolgt von einer erfolgreichen Sitzungswiederherstellung wird durch diese Protokollmeldung angezeigt:

```
"Death notification of task <name>/<instance id> on <card#>/<cpu#> sent to parent
task <parent name>/<instance id> with failover of <task name>/<instance id>
on <card#>/<cpu#>"
```

Ein Aufgabenabsturz, der nicht wiederhergestellt werden konnte, wird durch diese Protokollmeldung angezeigt:

```
"Death notification of task <name>/<instance id> on <card#>/<cpu#> sent to parent
task <parent name>/<instance id>"
```

Zusammenfassend lässt sich sagen, dass bei aktivierter Sitzungswiederherstellung die Abstürze in den meisten Fällen nicht bemerkt werden, da sie keine Auswirkungen auf den Teilnehmer haben. Man muss den CLI-Befehl eingeben oder die Protokolle oder die SNMP-Benachrichtigung überprüfen, um Abstürze zu erkennen.

Beispiel:

```
 ******** show crash list *******
Tuesday May 26 05:54:14 BDT 2015
=== ==================== ======== ========== =============== ======================
# Time Process Card/CPU/ SW HW_SER_NUM
PID VERSION MIO / Crash Card
=== ==================== ======== ========== =============== ======================

1 2015-May-07+11:49:25 sessmgr 04/0/09564 17.2.1 SAD171600WS/SAD172200MH
2 2015-May-13+17:40:16 sessmgr 09/1/05832 17.2.1 SAD171600WS/SAD173300G1
3 2015-May-23+09:06:48 sessmgr 03/1/31883 17.2.1 SAD171600WS/SAD1709009P
4 2015-May-25+15:58:59 sessmgr 09/1/16963 17.2.1 SAD171600WS/SAD173300G1
5 2015-May-26+01:15:15 sessmgr 04/0/09296 17.2.1 SAD171600WS/SAD172200MH

 ******** show snmp trap history verbose *******
Fri May 22 19:43:10 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:30 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 9 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
audit 1754 calls recovered 1754 all call lines 1754 time elapsed ms 1108.
Fri May 22 19:43:32 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:44:49 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:51 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 7 Cpu Number 0 fetched from aaa mgr 1741 prior to audit 1741 passed audit
1737 calls recovered 1737 all call lines 1737 time elapsed ms 1047.
Fri May 22 19:44:53 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:50:04 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 221 card 2 cpu 1
: Fri May 22 19:50:04 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:04 2015 Internal trap notification 151 (TaskRestart) facility
```

```
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:05 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 2 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
audit 1749 calls recovered 1750 all call lines 1750 time elapsed ms 1036.

 ******** show snmp trap history verbose *******
Fri May 22 19:43:10 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:30 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 9 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
audit 1754 calls recovered 1754 all call lines 1754 time elapsed ms 1108.
Fri May 22 19:43:32 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:44:49 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:51 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 7 Cpu Number 0 fetched from aaa mgr 1741 prior to audit 1741 passed
audit 1737 calls recovered 1737 all call lines 1737 time elapsed ms 1047.
Fri May 22 19:44:53 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:50:04 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 221 card 2 cpu 1
: Fri May 22 19:50:04 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:04 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:05 2015 Internal trap notification 183 (SessMgrRecoveryComplete
) Slot Number 2 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
audit 1749 calls recovered 1750 all call lines 1750 time elapsed ms 1036.

 ******** show logs  *******
2015-May-25+23:15:53.123 [sitmain 4022 info] [3/1/4850 <sitmain:31> sittask.c:4762]
[software internal system critical-info syslog] Readdress requested for facility
sessmgr instance 5635 to instance 114
2015-May-25+23:15:53.122 [sitmain 4027 critical] [3/1/4850 <sitmain:31>
crash_mini.c:908] [software internal system callhome-crash] Process Crash Info:
time 2015-May-25+17:15:52(hex time 556358c8) card 03 cpu 01 pid 27118 procname
sessmgr crash_details
Assertion failure at acs/acsmgr/analyzer/ip/acs_ip_reasm.c:2970
Function: acsmgr_deallocate_ipv4_frag_chain_entry()
Expression: status == SN_STATUS_SUCCESS
Proclet: sessmgr (f=87000,i=114)
Process: card=3 cpu=1 arch=X pid=27118 cpu=~17% argv0=sessmgr
Crash time: 2015-May-25+17:15:52 UTC
Recent errno: 11 Resource temporarily unavailable
Stack (11032@0xffffb000):
[ffffe430/X] __kernel_vsyscall() sp=0xffffbd28
[0af1de1f/X] sn_assert() sp=0xffffbd68
[0891e137/X] acsmgr_deallocate_ipv4_frag_chain_entry() sp=0xffffbde8
[08952314/X] acsmgr_ip_frag_chain_destroy() sp=0xffffbee8
[089d87d1/X] acsmgr_process_tcp_packet() sp=0xffffc568
[089da270/X] acs_process_tcp_packet_normal_path() sp=0xffffc5b8
[089da3fd/X] acs_tcp_analyzer() sp=0xffffc638
```

```
[0892fb39/X] do_acsmgr_process_packet() sp=0xffffc668
[08940045/X] acs_ip_lean_path() sp=0xffffc6b8
[0887e309/X] acsmgr_data_receive_merge_mode() sp=0xffffc9d8
[0887f323/X] acs_handle_datapath_events_from_sm_interface() sp=0xffffca08
[037c2e1b/X] sessmgr_sef_initiate_data_packet_ind() sp=0xffffca88
[037c2f50/X] sessmgr_pcc_intf_send_data_packet_ind() sp=0xffffcaf8
[061de74a/X] sessmgr_pcc_fwd_packet() sp=0xffffcb58
[0627c6a4/X] sessmgr_ipv4_process_inet_pkt_part2_slow() sp=0xffffcf68
[06318343/X] sessmgr_ipv4_process_inet_pkt_pgw_ggsn() sp=0xffffd378
[0632196c/X] sessmgr_med_ipv4_data_received() sp=0xffffd418
[0633da9a/X] sessmgr_med_data_receive() sp=0xffffd598
[0afb977c/X] sn_epoll_run_events() sp=0xffffd5e8
[0afbdeb8/X] sn_loop_run() sp=0xffffda98
[0ad2b82d/X] main() sp=0xffffdb08

2015-May-25+23:15:53.067 [rct 13038 info] [5/0/7174 <rct:0> rct_task.c:305]
[software internal system critical-info syslog] Death notification of task
sessmgr/114 on 3/1 sent to parent task sessctrl/0 with failover of sessmgr/5635 on 3/1
2015-May-25+23:15:53.065 [evlog 2136 info] [5/0/7170 <evlogd:0> odule_persist.c:3102]
[software internal system critical-info syslog] Evlogd crashlog: Request received to
check the state of persistent crashlog.
2015-May-25+23:15:53.064 [sitmain 4099 info] [3/1/4850 <sitmain:31> crash_mini.c:765]
[software internal system critical-info syslog] have mini core, get evlogd status for
logging crash file 'crashdump-27118'
2015-May-25+23:15:53.064 [sitmain 4017 critical] [3/1/4850 <sitmain:31> sitproc.c:1544]
[software internal system syslog] Process sessmgr pid 27118 died on card 3 cpu 1
signal=6 wstatus=0x86
2015-May-25+23:15:53.048 [sitmain 4074 trace] [5/0/7168 <sitparent:50> crashd.c:1130]
[software internal system critical-info syslog] Crash handler file transfer starting
(type=2 size=0 child_ct=1 core_ct=1 pid=23021)
2015-May-25+23:15:53.047 [system 1001 error] [6/0/9727 <evlogd:1> evlgd_syslogd.c:221]
[software internal system syslog] CPU[3/1]: xmitcore[21648]: Core file transmitted to
card 5 size=663207936 elapsed=0sec:908ms
2015-May-25+23:15:53.047 [system 1001 error] [5/0/7170 <evlogd:0> evlgd_syslogd.c:221]
[software internal system syslog] CPU[3/1]: xmitcore[21648]: Core file transmitted to
card 5 size=663207936 elapsed=0sec:908ms
2015-May-25+23:15:53.047 [sitmain 4080 info] [5/0/7168 <sitparent:50> crashd.c:1091]
[software internal system critical-info syslog] Core file transfer to SPC complete,
received 8363207936/0 bytes

 ******** show session recovery status verbose *******
Tuesday May 26 05:55:26 BDT 2015
Session Recovery Status:
Overall Status : Ready For Recovery
Last Status Update : 8 seconds ago

----sessmgr--- ----aaamgr---- demux
cpu state active standby active standby active status
---- ------- ------ ------- ------ ------- ------ -------------------------
1/0 Active 24 1 24 1 0 Good
1/1 Active 24 1 24 1 0 Good
2/0 Active 24 1 24 1 0 Good
2/1 Active 24 1 24 1 0 Good
3/0 Active 24 1 24 1 0 Good
3/1 Active 24 1 24 1 0 Good
4/0 Active 24 1 24 1 0 Good
4/1 Active 24 1 24 1 0 Good
5/0 Active 0 0 0 0 14 Good (Demux)
7/0 Active 24 1 24 1 0 Good
7/1 Active 24 1 24 1 0 Good
8/0 Active 24 1 24 1 0 Good
8/1 Active 24 1 24 1 0 Good
9/0 Active 24 1 24 1 0 Good
```

```
9/1 Active 24 1 24 1 0 Good
10/0 Standby 0 24 0 24 0 Good
10/1 Standby 0 24 0 24 0 Good
```

Architektur für die Absturzprotokollierung

Absturzprotokolle zeichnen alle möglichen Informationen auf, die zu einem Softwareabsturz (Voll-Core-Dump) führen. Aufgrund ihrer Größe können sie nicht im Systemspeicher gespeichert werden. Daher werden diese Protokolle nur generiert, wenn das System mit einer URL konfiguriert ist, die auf ein lokales Gerät oder einen Netzwerkserver verweist, auf dem das Protokoll gespeichert werden kann.

Das Absturzprotokoll ist ein permanentes Repository mit Informationen zu Crash-Ereignissen. Jedes Ereignis ist nummeriert und enthält Text, der mit einer CPU (Minicore), einer Netzwerkverarbeitungseinheit (NPU) oder einem Kernel-Absturz verknüpft ist. Die protokollierten Ereignisse werden in Datensätze mit fester Länge aufgezeichnet und in /flash/crashlog2 gespeichert.

Bei jedem Absturz werden diese Crash-Informationen gespeichert:

1. Der Ereignisdatensatz wird in der Datei /flash/crashlog2 (das Absturzprotokoll) gespeichert.
2. Die zugehörige Minicore-, NPU- oder Kernel-Dump-Datei wird im Verzeichnis /flash/crsh2 gespeichert.
3. Ein Volldump wird in einem benutzerdefinierten Verzeichnis gespeichert.

Synchronisierung von Absturzereignissen und Minicores zwischen Verwaltungskarten

Das Crashlog ist für jede Management-Karte eindeutig. Wenn also ein Absturz auftritt, wenn die Karte "8" aktiv ist, wird sie auf der Karte "8" angemeldet. Bei einem nachfolgenden Switchover wird der Absturz nicht mehr im Protokoll angezeigt. Um diesen Absturz zu beheben, muss ein Wechsel zurück zur Karte "8" durchgeführt werden. Das Crash Event Log und Dumps sind nur für aktive und Standby-Management-Karten verfügbar. Wenn also ein Crash auf einer aktiven Karte auftritt, werden das Crash Event Log und die zugehörigen Dumps nur auf einer aktiven Karte gespeichert. Diese Crash-Informationen sind auf der Standby-Karte nicht verfügbar. Immer wenn die Karten aufgrund eines Crashs in der aktiven Karte umschalten und auf der Karte, die die Karte übernimmt, keine Crash-Informationen mehr angezeigt werden, können die Crash-Informationen nur von der aktuellen aktiven Karte abgerufen werden. Um die Crash-Liste der anderen Karte abzurufen, ist ein erneuter Switchover erforderlich. Um diesen Switchover zu vermeiden und die Crash-Informationen von der Standby-Karte abzurufen, müssen zwei Management-Karten synchronisiert und die neuesten Crash-Informationen verwaltet werden.

Das ankommende Crash-Ereignis wird an die Standby-SMC/MIO gesendet und in ähnlicher Weise in der Crashlog-Datei des Standby-Geräts gespeichert. Minicore-, NPU- oder Kernel-Dumps im Flash der aktiven SMC/MIO müssen mit dem **rsync**-Befehl mit Standby-SMC/MMIO synchronisiert werden. Wenn ein Crashlog-Eintrag oder die gesamte Liste über den CLI-Befehl gelöscht wird, sollte er sowohl auf aktiven als auch auf Standby-SMCs/MIOs gelöscht werden. Es treten keine Auswirkungen auf den Speicher auf. Alle Synchronisierungsaktivitäten im Zusammenhang mit dem Absturz werden durch die Aktualisierung der Standby-SMC/MIO-Karte ausgeführt, da das Standby-Ereignis weniger geladen ist und die Standby-Karte genügend Platz für Synchronisierungsaktivitäten bietet. Daher wird die Leistung des Systems nicht beeinträchtigt.

Befehle

Diese Befehle können zur Fehlerbehebung verwendet werden:

```
#show support details

#show crash list

#show logs

#show snmp trap history verbose

#show session recovery status verbose

#show task resources facility sessmgr instance <>

#show task resources facility sessmgr all
```

Corefiles werden nach einem Absturz generiert. In der Regel speichern Operatoren diese auf einem externen Server. Der Corefile-Name sieht in der Regel aus wie Crash-<Cardnum>-<CPU Num>-<Hex timestamp>-coree.gcrash-09-00-5593a1b8-core.

Bei jedem Absturz werden diese Crash-Informationen gespeichert:

- Der Ereignisdatensatz wird in der Datei /flash/crashlog2 (das Absturzprotokoll) gespeichert.
- Die zugehörige Minicore-, NPU- oder Kernel-Dump-Datei wird im Verzeichnis /flash/crsh2 gespeichert.

Zusammenfassung

Die gesamte ASR5x00-Software wurde entwickelt, um vorhersehbare Bedingungen/Ereignisse und unvorhergesehene Ereignisse zu behandeln. Cisco bemüht sich um eine perfekte Software, aber es werden unweigerlich Fehler und Abstürze auftreten. Aus diesem Grund ist die Sitzungswiederherstellungsfunktion so wichtig. Das Streben von Cisco nach Perfektion minimiert das Auftreten von Abstürzen, und die Sitzungswiederherstellung ermöglicht die Fortsetzung der Sitzungen nach einem Absturz. Dennoch ist es wichtig, dass Cisco auch weiterhin bestrebt ist, die perfekte Software zu erhalten. Weniger Abstürze verringern die Wahrscheinlichkeit, dass mehrere Abstürze gleichzeitig auftreten. Während die Sitzungswiederherstellung einen einzelnen Absturz nahtlos heilt, ist die Wiederherstellung nach mehreren gleichzeitigen Abstürzen etwas anders ausgelegt. Bei den Betreibern sollten selten (oder nie) mehrere gleichzeitige Abstürze auftreten. Sollte dies jedoch der Fall sein, ist der ASR5x00 so konzipiert, dass die Systemintegrität als höchste Priorität wiederhergestellt wird, möglicherweise auf Kosten einiger Teilnehmersitzungen.