

Fehlerbehebung bei StarOS-Abstürzen

Inhalt

[Einleitung](#)

[Überblick](#)

[Szenario der Abstürze](#)

[Ursache des Absturzes](#)

[Verschiedene Arten von Abstürzen](#)

[Anfängliche Protokollanforderungen](#)

[Analyseschritte](#)

[Sitzungswiederherstellung](#)

Einleitung

In diesem Dokument wird beschrieben, wie StarOs Facility-Abstürze erkannt und behoben werden.

Überblick

Manchmal kann die Systemlogik fehlschlagen, was dazu führt, dass ein Software-Task neu gestartet wird, um die ordnungsgemäße Funktion wiederherzustellen. Dies kann zu einem Prozessabsturz führen. Abstürze von Aufgabeneinrichtungen werden in StarOS häufig gemeldet, und die erforderlichen Maßnahmen können basierend auf der Ursache des Absturzes ergriffen werden. Um Abstürze auf dem Knoten zu identifizieren, können Sie den folgenden CLI-Befehl verwenden:

```
***** show crash list *****
Saturday April 15 05:05:56 SAST 2023
=== =====
#           Time           Process  Card/CPU/      SW           HW_SER_NUM
           Time           Process  PID            VERSION      CF / Crash Card
=== =====
1  2022-Dec-02+14:08:46  confdmgr 02/0/19342  21.26.13     NA
2  2022-Dec-02+14:48:08  confdmgr 02/0/31546  21.26.13     NA
3  2022-Dec-04+19:10:50  sessmgr  03/0/12321  21.26.13     NA
4  2022-Dec-21+03:34:13  sessmgr  04/0/12586  21.26.13     NA
```

Ähnliche Abstürze werden in einem Datensatz zusammengefasst. Der Datensatz zeigt an, wie oft dieser Absturztyp aufgetreten ist.

```
***** CRASH #02 *****
SW Version      : 21.26.13
Similar Crash Count : 33 >>>>
Time of First Crash : 2022-Dec-02+14:10:05
```

```
Assertion failure at confdmgr/src/confdmgr_fsm.c:870
Note: State machine failure, state = 3
Function: confdmgr_fsm_state_wait_p0_handler()
```

```
Expression: 0
Code: CRASH
Proclet: confdmgr (f=1900,i=0)
Process: card=2 cpu=0 arch=X pid=31546 argv0=confdmgr
```

In `show snmp trap history verbose` zeigt, dass ein Prozess abgestürzt ist:

```
Fri Dec 26 08:32:20 2014 Internal trap notification 73 (ManagerFailure) facility sessmgr
instance 188 card 7 cpu 0
Fri Dec 26 08:32:20 2014 Internal trap notification 150 (TaskFailed) facility sessmgr instance
188 on card 7 cpu 0
Fri Dec 26 08:32:23 2014 Internal trap notification 1099 (ManagerRestart) facility sessmgr
instance 139 card 4 cpu 1
Fri Dec 26 08:32:23 2014 Internal trap notification 151 (TaskRestart) facility sessmgr
instance 139 on card 4 cpu 1
```

Szenario der Abstürze

Abstürze können aus verschiedenen Gründen auftreten:

1. Unterschiedliche Anruflusszenarien
2. Speicherprobleme
3. Konfigurationsproblem
4. Hardwarefehler

Ursache des Absturzes

In StarOS gibt es mehrere Aufgabenbereiche, deren individuelle Funktionalität auf den Funktionen basiert. Wenn eine Einrichtung auf solche Eingaben trifft, bei denen sie in einen problematischen Zustand gerät, stürzt sie die Einrichtung ab, um sich von diesem Fehlerzustand zu erholen.

Verschiedene Arten von Abstürzen

1. Erklärungsfehler:

```
***** CRASH #22 *****
SW Version      : 21.26.13
Similar Crash Count : 33
Time of First Crash : 2023-Apr-12+22:40:01
```

```
Assertion failure at sess/smgr/sessmgr_snx.c:9568 >>>>
Function: sessmgr_snx_send_drop_call()
Expression: result == SN_STATUS_SUCCESS
Proclet: sessmgr (f=87000,i=261)
Process: card=5 cpu=0 arch=X pid=12724 cpu=~0% argv0=sessmgr
```

2. Segmentierungsfehler:

```
***** CRASH #69 *****
SW Version : 21.13.3
Similar Crash Count : 2
Time of First Crash : 2019-Nov-25+07:53:54
Fatal Signal 11: Segmentation fault >>>>
Faulty address: 0x7ff6b4801036
Signal from: kernel
Signal detail: address not mapped to object
Process: card=8 cpu=1 arch=X pid=7316 argv0=vpp
Crash time: 2020-Feb-11+04:04:23 UTC
Build_number:
```

3. Schwerwiegendes Signal:

```
***** CRASH #01 *****
SW Version : 21.23.12
Similar Crash Count : 2
Time of First Crash : 2023-Jan-27+05:22:46

Fatal Signal 11: 11 >>>>>
PC: [04be6859/X] sessmgr_pgw_create_bearers()
Faulty address: 0x297116e4
Signal from: kernel
Signal detail: address not mapped to object
Process: card=9 cpu=1 arch=X pid=10383 cpu=~8% argv0=sessmgr
```

Anfängliche Protokollanforderungen

Das Absturzprotokoll dient als wertvolle Quelle für Informationen über das Absturzereignis. Bei einem Software-Absturz erfasst und speichert StarOS relevante Daten, anhand derer die Ursache des Absturzes ermittelt werden kann. Diese Informationen können im Systemspeicher abgelegt oder auf einem Netzwerkserver übertragen und gespeichert werden.

Core-Datei oder Mini-Core-Datei: Beachten Sie, dass die Core-Dateien den PID(s) entsprechen, in denen der Absturz aufgetreten ist. Core-Dateien werden im Format "Absturz-<Kartennr>-<CPU>-<PID>-<Unixtime>-Core" benannt. Diese Informationen finden Sie in der Befehlsausgabe "show crash list".

Minicore-Datei: Diese Datei enthält Informationen über den ausgefallenen Task, einschließlich der aktuellen Stapelüberwachung, vergangenen Profilerproben, vergangenen Speicheraktivitätsproben und anderen gebündelten Daten in einem proprietären Dateiformat.

Core Dump (oder Full Core): Ein Core Dump stellt ein vollständiges Speicherdump des Prozesses unmittelbar nach dem Absturz bereit. Dieses Speicherabbild ist häufig für die Identifizierung der Ursache des Softwareabsturzes unerlässlich.

Crash Signatures (Absturzsignaturen): Die Absturzsignaturen können von der freigegebenen Show Support Details (SSD) oder anderen relevanten Quellen überprüft werden.

```

***** show crash list *****
Saturday April 15 05:05:56 SAST 2023
=== =====
#           Time           Process  Card/CPU/      SW           HW_SER_NUM
           Time           Process  PID           VERSION      CF / Crash Card
=== =====

1  2022-Dec-02+14:08:46  confdmgr  02/0/19342  21.26.13      NA
2  2022-Dec-02+14:48:08  confdmgr  02/0/31546  21.26.13      NA
3  2022-Dec-04+19:10:50  sessmgr   03/0/12321  21.26.13      NA

```

Wenn Sie jetzt die Signatur für Absturz 1 kennen möchten, suchen Sie in SSD mit **CRASH #01** oder in CLI mit **show Absturz Nummer 1**.

From SSD

```

***** CRASH #01 *****
SW Version           : 21.26.13
Similar Crash Count  : 1
Time of First Crash  : 2022-Dec-02+14:08:46

```

```

Assertion failure at confdmgr/src/confdmgr_fsm.c:758
Note: State machine failure, state = 5
Function: confdmgr_fsm_state_wait_p1_handler()
Expression: 0
Code: CRASHâ€œ

```

Using CLI

```

[local]abc# show crash number 1
Friday June 09 06:41:53 CDT 2023
***** CRASH #01 *****
SW Version           : 21.12.20.77760
Similar Crash Count  : 1
Time of First Crash  : 2021-Mar-31+15:58:06

```

```

Fatal Signal 6: Aborted
PC: [ffffe430/X] __kernel_vsyscall()
Note: User-initiated state dump w/core.
Signal from: sitmain pid=6999 uid=0
Process: card=9 cpu=0 arch=X pid=9495 cpu=~0% ar

```

Überprüfen Sie die Show Support Details (SSD) und die Syslogs während des spezifischen Zeitstempels, wenn das Problem aufgetreten ist.

Analyseschritte

1. Muss den Crash-Stack/die Crash-Signatur überprüfen und überprüfen, ob Fehler für diese spezielle Crash-Signatur vorliegen.
2. Muss die Core-Datei/Minicore analysieren, um den Backtrace zu analysieren und einen Hinweis darauf zu erhalten, welche Funktion die Einrichtung abgestürzt ist.
3. Sobald das Debuggen von Kerndateien abgeschlossen ist, müssen Sie die Symptome des Softwarefehlers

überprüfen und überprüfen, ob ein Softwarefehler für eine ähnliche Absturzsignatur und einen ähnlichen Rücklauf vorhanden ist.

Sitzungswiederherstellung

Die StarOs-Software wurde für die Verarbeitung vorhersehbarer und unvorhergesehener Bedingungen und Ereignisse entwickelt. Cisco strebt zwar eine perfekte Software an, es sind jedoch unweigerlich Fehler und Abstürze möglich. Aus diesem Grund ist die Funktion zur Sitzungswiederherstellung so wichtig.

Die Funktion zur Sitzungswiederherstellung ermöglicht ein nahtloses Failover und die Wiederherstellung von Teilnehmersitzungsinformationen, wenn ein Hardware- oder Softwarefehler im System die Trennung einer vollständig verbundenen Benutzersitzung verhindert. Die Sitzungswiederherstellung erfolgt durch Spiegelung wichtiger Softwareprozesse (z. B. Sitzungsmanager und AAA-Manager) innerhalb des Systems. Diese gespiegelten Prozesse verbleiben in einem Inaktivitätszustand (Standby-Modus), in dem sie keine Verarbeitung durchführen, bis sie bei einem Softwarefehler (z. B. Abbruch einer Session-Manager-Aufgabe) benötigt werden.

Aufgaben wie Demux-Aufgaben, AAA-Manager und VPN-Manager verfügen über integrierte automatische Wiederherstellungsmechanismen in unseren Systemen, insbesondere für den Umgang mit Teilnehmerinformationen. Die Sitzungswiederherstellung bezieht sich in erster Linie auf Szenarien, bei denen ein Fehler in der Sessmgr-Aufgabe oder ein Fehler auf Kartenebene vorliegt und Sitzungen ohne Anrufverlust wiederhergestellt werden müssen.

- Im System ist auf jeder Verarbeitungskarte ein Standby-Sitzungsmanager aktiv, der bei einem Ausfall schnell als primärer Sessmgr übernehmen kann. Anschließend wird ein neuer Standby-Sessmgr auf der Verarbeitungskarte erstellt.
- Wenn ein Sessmgr-Prozess unerwartet fehlschlägt, ruft der Standby-Sessmgr gesicherte Informationen vom aaamgr (AAA-Manager) ab und erstellt seine Sitzungen neu.
- Wenn aaamgr ausfällt, fragt der Standby-aaamgr den sessmgr ab, um die relevanten Teilnehmerinformationen zu synchronisieren.
- Bei einem Ausfall von demux-mgr werden alle Sessmgrs abgefragt und die Datenbank mit den Anrufverteilungsinformationen rekonstruiert.
- Um Redundanz auf Kartenebene sicherzustellen, dient eine Prozessorkarte als Standby-Karte, die bei einem Hardware- oder Softwareausfall sofort einsatzbereit ist. Die Standby-Karte stellt dann die Sitzungen von einem Peer-Aaamgr wieder her, der auf einer anderen Karte ausgeführt wird.

```
***** show session recovery status verbose *****
Saturday April 15 05:11:17 SAST 2023
Session Recovery Status:
  Overall Status      : Ready For Recovery  >>>>
  Last Status Update  : 5 seconds ago

-----sessmgr----- ----aaamgr----- demux
cpu state  active standby active standby active status
-----
3/0 Active  40    1    40    1    0    Good
4/0 Active  40    1    40    1    0    Good
5/0 Active  40    1    40    1    0    Good
6/0 Active  40    1    40    1    0    Good
7/0 Active  0     0     0     0    10   Good (Demux)
8/0 Active  40    1    40    1    0    Good
9/0 Active  40    1    40    1    0    Good
```

10/0	Active	40	1	40	1	0	Good
11/0	Standby	0	40	0	40	0	Good

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.