

StarOS S4-SGSNでのグレースフルアサート処理

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[問題](#)

[解決方法](#)

[設定](#)

[確認](#)

[トラブルシューティング](#)

概要

このドキュメントでは、StarOSのリリース21.5.0で導入された「グレースフルアサート処理」機能について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- StarOs
- Serving GPRS Support Node(SGSN)

使用するコンポーネント

このドキュメントの情報は、StarOS R21.5以降に基づくものです。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

この機能は、次の公式ドキュメントで確認できます。 [SGSN-Admin](#)ガイド

グレースフルアサート処理フレームワークは、コール実行時にASSERT条件がヒットしたサブス

クライバセッションのグレースフルハンドリングを可能にします。これは、同じエンドポイント上の他のサブクライバセッションに影響を与えずに実現されます。

通常、ASSERT条件が満たされると、Session Manager(SessMgr)のプロンプトが再起動し、AAA Manager(AAAMgr)からすべてのサブクライバセッションが回復されます。リカバリされたサブクライバセッションはIDLE状態に移行します。

グレースフルアサート処理が有効な場合、SessMgrプロキシは再起動されません。その代わりに、SessMgrプロキシは、影響を受けたサブクライバのセッションのみをAAAMgrから回復し、SessMgr上の既存のサブクライバのセッションをクリアします。リカバリされたサブクライバセッションはIDLE状態に移行します。回復手順の時点で、サブクライバに向けられたすべてのメッセージがドロップされます。回復後、サブクライバは自身に向けられたメッセージを引き続き処理します。この手順では、SessMgr上に残っているサブクライバセッションは影響を受けません。

問題

根本原因の修正が複雑であるか、根本原因が不明であるコーナーまたは衝突ケースがあります。このような場合、セッションマネージャの完全な再起動を回避するために、グレースフルアサートアプローチが採用されます。

解決方法

グレースフルアサートでは、グレースフルアサート条件に一致する1セッションをクリーンアップして復元できます。

同じsessmgr上の他のセッションへの影響はありません。

グレースフルリスタートのためのSNMPトラップやsyslogはありません。

グレースフルアサートの場合、KPIの損失はありません。タスク自体は再起動されません。

ただし、グレースフルアサートは他のクラッシュと同様に処理されるため、show crash listにエントリが表示されます。

SSDからのグレースフルアサートの識別方法：

- コアを使用したシステム開始状態ダンプ。 - スタックの前に「show crash」の出力に表示されます。
- クラッシュしたproclletはユーザが開始したものか非ボクサーか、スタックの後の「debug console cpu」出力に表示されます。
- pid 7939 facility sessmgr failover 5132->94 - 「debug console CPU」の下で、グレースフルアサートの場合にログが記録されず、表示されない

設定

グレースフルアサート処理は、次のように設定できます。

```
configure
  debug controlled-assert s4sgsn
    [ disable | enable ] core-generation
    limit-per-assert assert_value
    [ no ] test file-name file_name line-number line_num [ sequence-number seq_num ]
  end
```

注：

- controlled-assert:制御アサートフレームワークを設定します。
- s4sgsn:S4-SGSN制御アサートを設定します。
- コア世代：制御アサートのコア生成を設定します。デフォルト：有効
- limit-per-assert:制御されたアサートのアサートごとの制限を設定します。デフォルト：5.
- test file-name file_name line-number line_num [sequence-number seq_num]:制御アサートテスト処理を設定します。file-name file_name:アサート制御が必要なファイル名を構成します。file_nameは1から254文字の英数字文字列である必要があります。line-number line_num:assert制御が必要な回線番号を設定します。line_numは1 ~ 4294967295の整数である必要があります。sequence-number seq_num:assert制御が必要なシーケンス番号を設定します。seq_numは1 ~ 100の整数である必要があります。デフォルト：1.
- disable：制御されたアサートフレームワークの指定されたアクションを無効にします。
- enable:制御されたアサートフレームワークに対して指定されたアクションを有効にします。
- いいえ：制御されたアサートフレームワークに関連する指定されたテスト構成を削除します。

例

```
***** CRASH #93 *****
```

```
SW Version : 21.5.19
```

```
Similar Crash Count : 8
```

```
Time of First Crash : 2019-May-21+06:57:14
```

```
Fatal Signal 6: Aborted
```

```
PC: [ffffe430/X] __kernel_vsyscall()
```

```
Note: System-initiated state dump w/core. <<< This note indicates a graceful assert.
```

```
Process: card=10 cpu=0 arch=X pid=11573 cpu=~16% argv0=sessmgr
```

```
Crash time: 2019-May-23+06:00:13 UTC
```

```
Recent errno: 11 Resource temporarily unavailable
```

```
Build_number: 71813
```

確認

ここでは、設定が正常に機能しているかどうかを確認します。

すべてのアクティブなsessmgrの制御されたアサート統計を取得する例：

```
# zcat ssd_s4sgn.log.gz | sed -n -e '/\*\{7\} show session subsystem facility sessmgr all debug-  
info /,/\*\{7\}/p' | sed -e '/^SessMgr: /,/^Controlled Assert Stats/{/^SessMgr: /!{/^Controlled  
Assert Stats/!d}}' | grep -E "SessMgr: Instance [0-9]{1,3}$" -A 10
```

出力例 :

```
SessMgr: Instance 135  
Controlled Assert Stats  
  Module Name :SGW_DRV  
  Assert Count:0  
  CountFile:Line  
  Last Assert hit time(in  
sec)  
Module Name :S4_SGSN Assert Count:1 Count File:Line Last Assert hit time(in sec) 1  
sess/sgsn/sgsn-app/s4_sm/s4_smn_egtpc.c:3164 2019/01/30 09:28:11 UTC
```

この情報 (カウントと回線番号) は、sessmgrが他のクラッシュのために再起動するとリセット
されます。最大回数 (デフォルトは5) に達すると、コアは生成されません。

トラブルシューティング

現在、この設定に関する特定のトラブルシューティング情報はありません。

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。