

"Warn" ; または "Over" ; 状態の Sessmgr/Aaamgrのトラブルシューティング

内容

[はじめに](#)

[概要](#)

[ログ/基本的なチェック](#)

[基本チェック](#)

[ログ](#)

[分析](#)

[アクションプラン](#)

[シナリオ 1. 高いメモリ使用率が原因](#)

[シナリオ 2. 高いCPU使用率が原因](#)

はじめに

このドキュメントでは、「warn」または「over」状態のsessmgrまたはaaamgr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。

概要

セッションマネージャ(Sessmgr) : 複数のセッションタイプをサポートし、加入者トランザクションを処理する加入者処理システムです。Sessmgrは通常、AAAManagerと組み合わせられています。

Authorization, Authentication, and Accounting (AAA ; 認可、認証、アカウントリング) マネージャ : システム内の加入者と管理ユーザに対するすべてのAAAプロトコルの動作と機能を実行します。

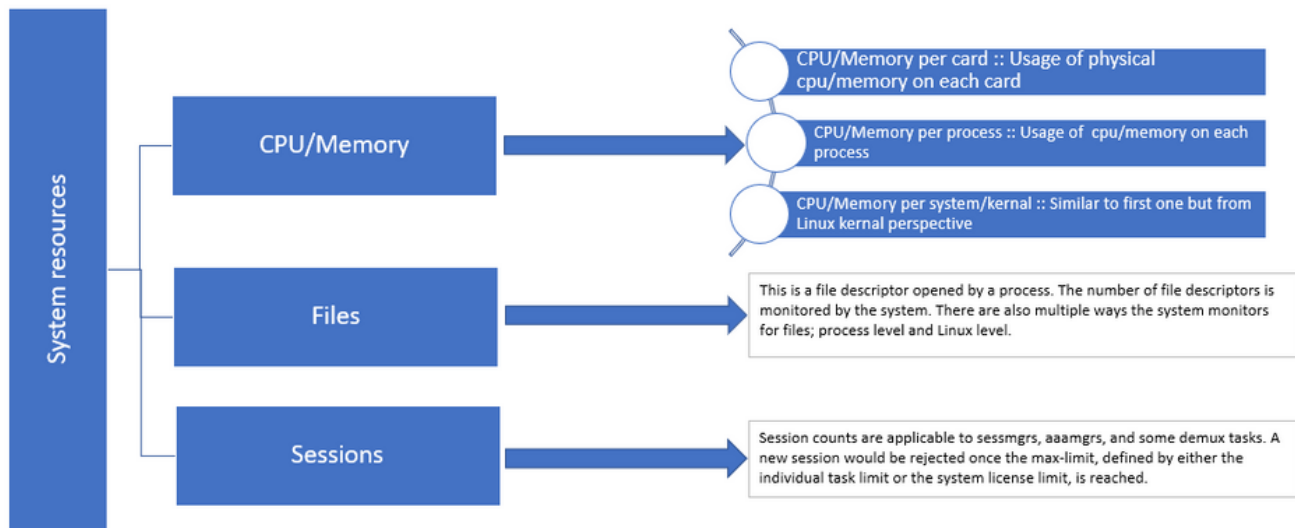


図1:Starosリソースのディストリビューション

ログ/基本的なチェック

基本チェック

この問題に関する詳細を収集するには、次の情報をユーザに確認する必要があります。

1. sessmgr/aaamgrが「warn」または「over」状態になっている期間
2. この問題の影響を受けるセッション/チームはいくつですか。
3. sessmgr/aaamgrがメモリまたはCPUによって「warn」または「over」状態になっているかどうかを確認する必要があります。
4. また、トラフィックが突然増加したかどうかを確認する必要があります。このトラフィックは、sessmgrごとのセッション数を調べることによって評価できます。

この情報を入手することで、現在の問題をよりよく理解して対処できます。

ログ

1. Show Support Details(SSD)と、問題のあるタイムスタンプをキャプチャしたsyslogを取得します。トリガーポイントを特定するために、問題が発生する少なくとも2時間前にこれらのログを収集することを推奨します。
2. 問題のあるsessmgr/aaamgrと問題のないsessmgr/aaamgrの両方のコアファイルをキャプチャします。詳細については、「分析」セクションを参照してください。

分析

ステップ 1：影響を受けるsessmgr/aaamgrのステータスをコマンドで確認します。

show task resources

-

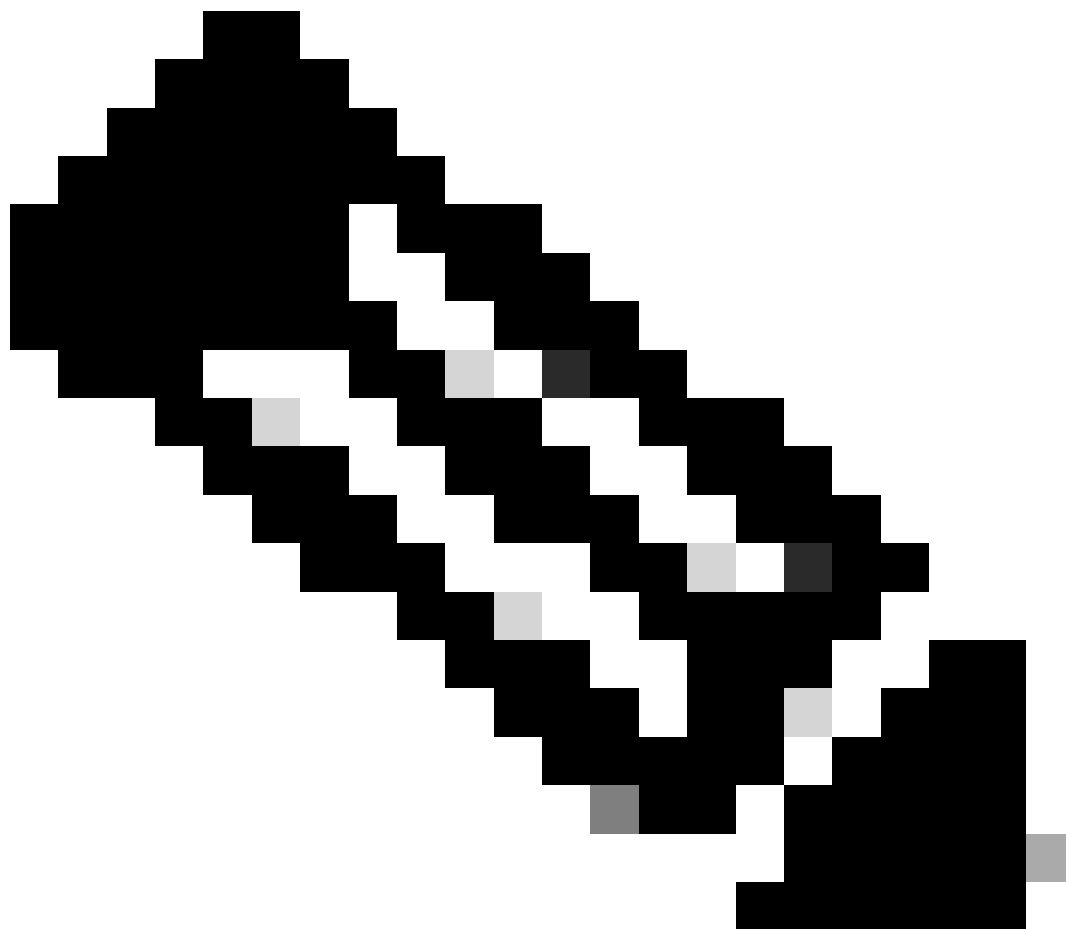
----- to check detail of sessmgr/aamgr into warn/over state and from the same you also get to know

Output ::

***** show task resources *****

Monday May 29 08:30:54 IST 2023

task			cputime		memory		files		sessions		S	status
cpu	facility	inst	used	alloc	used	alloc	used	allc	used	allc		
2/0	sessmgr	297	6.48%	100%	604.8M	900.0M	210	500	1651	12000	I	good
2/0	sessmgr	300	5.66%	100%	603.0M	900.0M	224	500	1652	12000	I	good
2/1	aaamgr	155	0.90%	95%	96.39M	260.0M	21	500	--	--	-	good
2/1	aaamgr	170	0.89%	95%	96.46M	260.0M	21	500	--	--	-	good



注: sessmgrあたりのセッション数は、コマンド出力に示されているように、このコマンドで確認できます。

これらのコマンドは、ノードがリロードされた後の最大メモリ使用量のチェックに役立ちます。

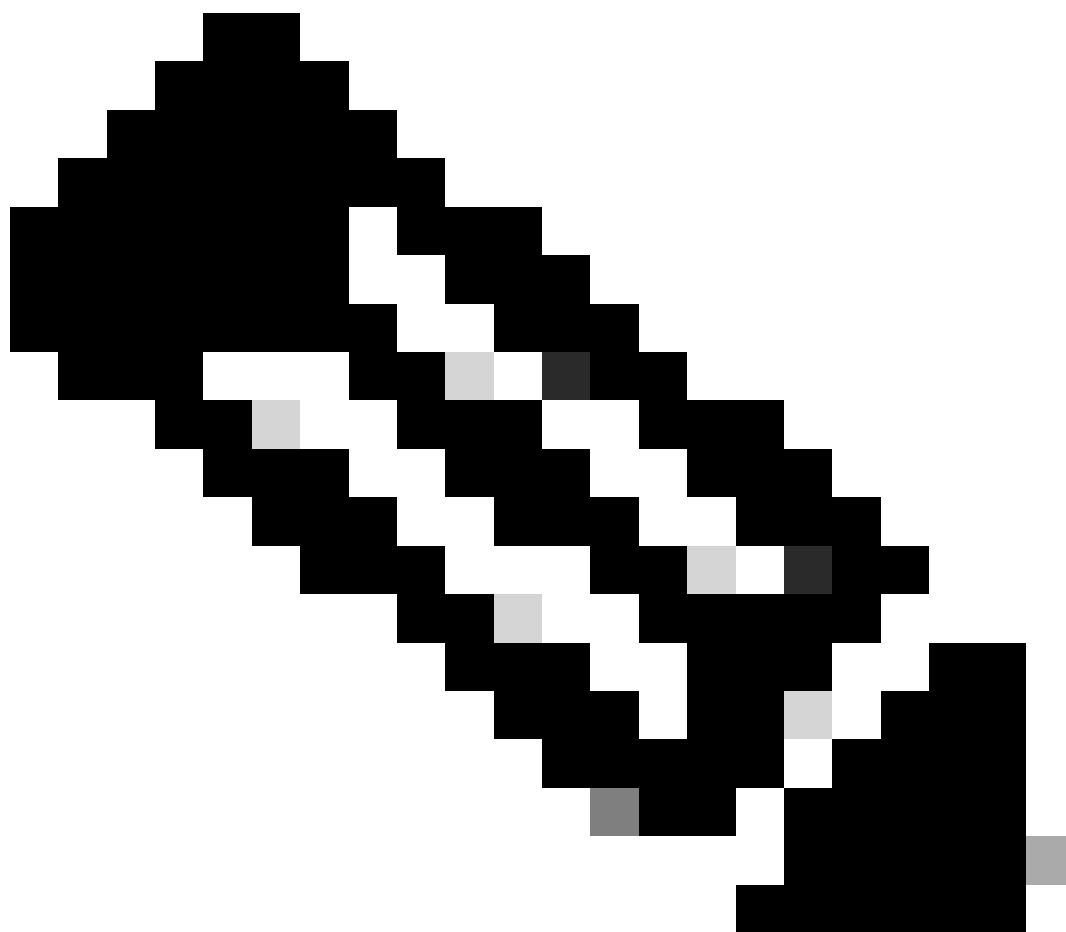
```
show task resources max
```

```
show task memory max
```

```
***** show task memory max *****
```

```
Monday May 29 08:30:53 IST 2023
```

cpu	facility	task inst	heap max	physical max	alloc		virtual max	alloc	status
2/0	sessmgr	902	548.6M	66% 602.6M	900.0M	29%	1.19G	4.00G	good
2/0	aaamgr	913	68.06M	38% 99.11M	260.0M	17%	713.0M	4.00G	good



注:memory maxコマンドは、ノードがリロードされてから使用されている最大メモリを示します。このコマンドは、問題が最近のリロード後に発生したのが、最新のリロードがあり最大メモリ値をチェックできたのかなど、問題に関連するパターンを特定するのに役立ちます。一方、「show task resources」と「show task resources max」でも同様の出力が得られます。maxコマンドでは、リロード後に特定のsessmgr/aamgrが使用したメモリ、CPU、およびセッションの最大値が表示されるという違いがあります。

```
show subscriber summary apn <apn name> smgr-instance <instance ID> | grep Total
```

----- to check no of subscribers for that particular APN in sessmg

アクションプラン

シナリオ 1.高いメモリ使用率が原因

1. sessmgrインスタンスを再起動または強制終了する前にSSDを収集します。
2. 影響を受けるsessmgrのコアダンプを収集します。

```
task core facility sessmgr instance <instance-value>
```

3. 影響を受ける同じsessmgrおよびaaamgrについて、隠しモードでこれらのコマンドを使用してヒープ出力を収集します。

```
show session subsystem facility sessmgr instance <instance-value> debug-info verbose  
show task resources facility sessmgr instance <instance-value>
```

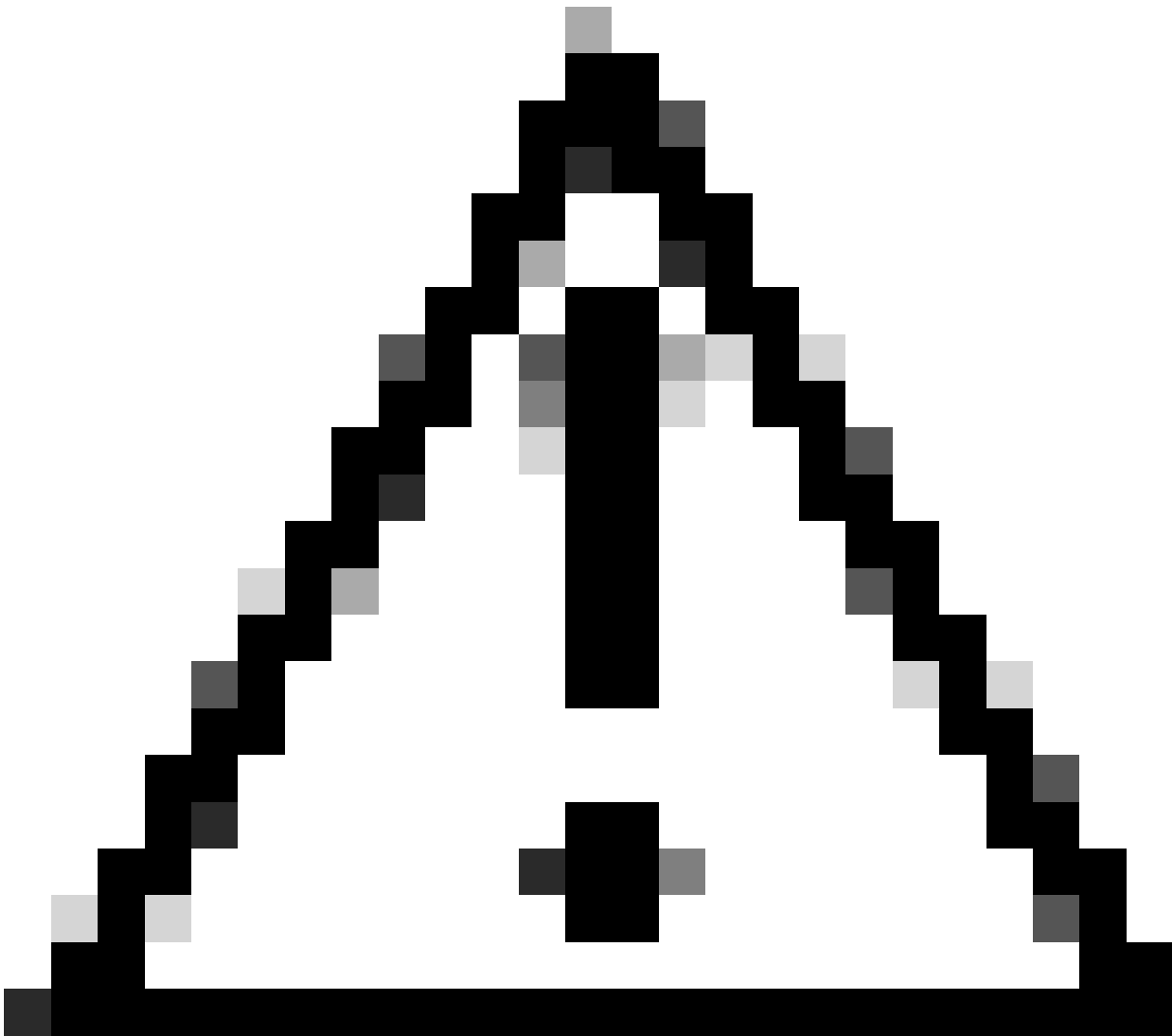
Heap outputs:

```
show messenger procllet facility sessmgr instance <instance-value> heap depth 9  
show messenger procllet facility sessmgr instance <instance-value> system heap depth 9  
show messenger procllet facility sessmgr instance <instance-value> heap  
show messenger procllet facility sessmgr instance <instance-value> system
```

```
show snx sessmgr instance <instance-value> memory ldbuf  
show snx sessmgr instance <instance-value> memory mblk
```

4. 次のコマンドを使用して、sessmgrタスクを再起動します :

```
task kill facility sessmgr instance <instance-value>
```



注意: 「warn」または「over」状態のsessmgrが複数ある場合は、sessmgrを2～5分の間隔で再起動することをお勧めします。最初は2～3個のsessmgrだけを再起動し、次に最大で10～15分待機して、これらのsessmgrが通常状態に戻るかどうかを確認します。この手順は、再起動の影響を評価し、リカバリの進行状況を監視するのに役立ちます。

5. sessmgrのステータスを確認します。

```
show task resources facility sessmgr instance <instance-value> ----- to check if sessmgr is back in
```

6. 別のSSDを収集します。

7. ステップ3で説明したすべてのCLIコマンドの出力を収集します。

8. ステップ2で説明したコマンドを使用して、正常なsessmgrインスタンスのコアダンプを収集します。



注：問題のあるファシリティと問題のないファシリティの両方のコアファイルを取得するには、2つのオプションがあります。1つは、再起動後に正常に戻った後に、同じ sessmgrのコアファイルを収集する方法です。別の方法として、別の正常な sessmgr からコアファイルをキャプチャすることもできます。これらの両方のアプローチは、分析とトラブルシューティングに役立つ情報を提供します。

ヒープ出力を収集したら、Cisco TACに連絡して、正確なヒープ消費表を見つけてください。

これらのヒープ出力から、より多くのメモリを利用している関数を確認する必要があります。これに基づいて、TACは機能使用率の目的を調査し、その使用率がトラフィック/トランザクション量の増加や他の問題のある理由と一致するかどうかを判断します。

ヒープ出力は、[Memory-CPU-data-sorting-tool](#)というリンクからアクセスされるツールを使用してソートできます。



注：このツールには、さまざまな機能に対する複数のオプションがあります。ただし、ヒープ出力をアップロードする「ヒープ消費表」を選択し、ツールを実行して出力を並べ替えた形式で取得する必要があります。

シナリオ 2. 高いCPU使用率が原因

1. sessmgrインスタンスを再起動または強制終了する前にSSDを収集します。
2. 影響を受けるsessmgrのコアダンプを収集します。

```
task core facility sessmgr instance <instance-value>
```

3. 影響を受ける同じsessmgr/aamgrの非表示モードでこれらのコマンドのヒープ出力を収集します。

<#root>

```
show session subsystem facility sessmgr instance <instance-value> debug-info verbose
show task resources facility sessmgr instance <instance-value>
show cpu table
show cpu utilization

show cpu info ----- Display detailed info of CPU.
show cpu info verbose ----- More detailed version of the above
```

Profiler output for CPU

This is the background cpu profiler. This command allows checking which functions consume the most CPU time. This command requires CLI test command password.

```
show profile facility <facility instance> instance <instance ID> depth 4
show profile facility <facility instance> active facility <facility instance> depth 8
```

4. 次のコマンドを使用して、sessmgrタスクを再起動します：

```
task kill facility sessmgr instance <instance-value>
```

5. sessmgrのステータスを確認します。

```
show task resources facility sessmgr instance <instance-value> ----- to check if sessmgr is back in
```

6. 別のSSDを収集します。

7. ステップ3で説明したすべてのCLIコマンドの出力を収集します。

8. ステップ2で説明したコマンドを使用して、正常なsessmgrインスタンスのコアダンプを収集します。

高メモリとCPUの両方のシナリオを分析するには、トラフィックの傾向に正当な増加があるかどうかを判断するためにバルク統計を調べてください。

さらに、カード/CPUレベルの統計情報のbulkstatsを確認します。

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。