

CallsiteによるIOS XEのメモリ問題のトラブルシューティング

内容

[はじめに](#)

[前提条件](#)

[使用するコンポーネント](#)

[背景説明](#)

[コールサイトとは何か](#)

[DiffコールとDiffバイト](#)

[メモリ消費またはリーク問題の症状](#)

[トラブルシューティングのシナリオ](#)

[メモリ不足のためデバイスがクラッシュした](#)

[デバイスはまだクラッシュしていませんが、メモリ使用量の警告があります](#)

[コールサイトの特定](#)

[モジュール、データベース、およびメッセージングでのプロセスメモリの消費](#)

[コールサイトのデバッグ](#)

[関連記事およびドキュメント](#)

はじめに

このドキュメントでは、コールサイトのリークが発生した場合に、ルータやスイッチなどのCisco IOS® XEベースのデバイスでメモリの問題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。

前提条件

Cisco IOS XEソフトウェアベースデバイスのメモリ管理に関する知識。

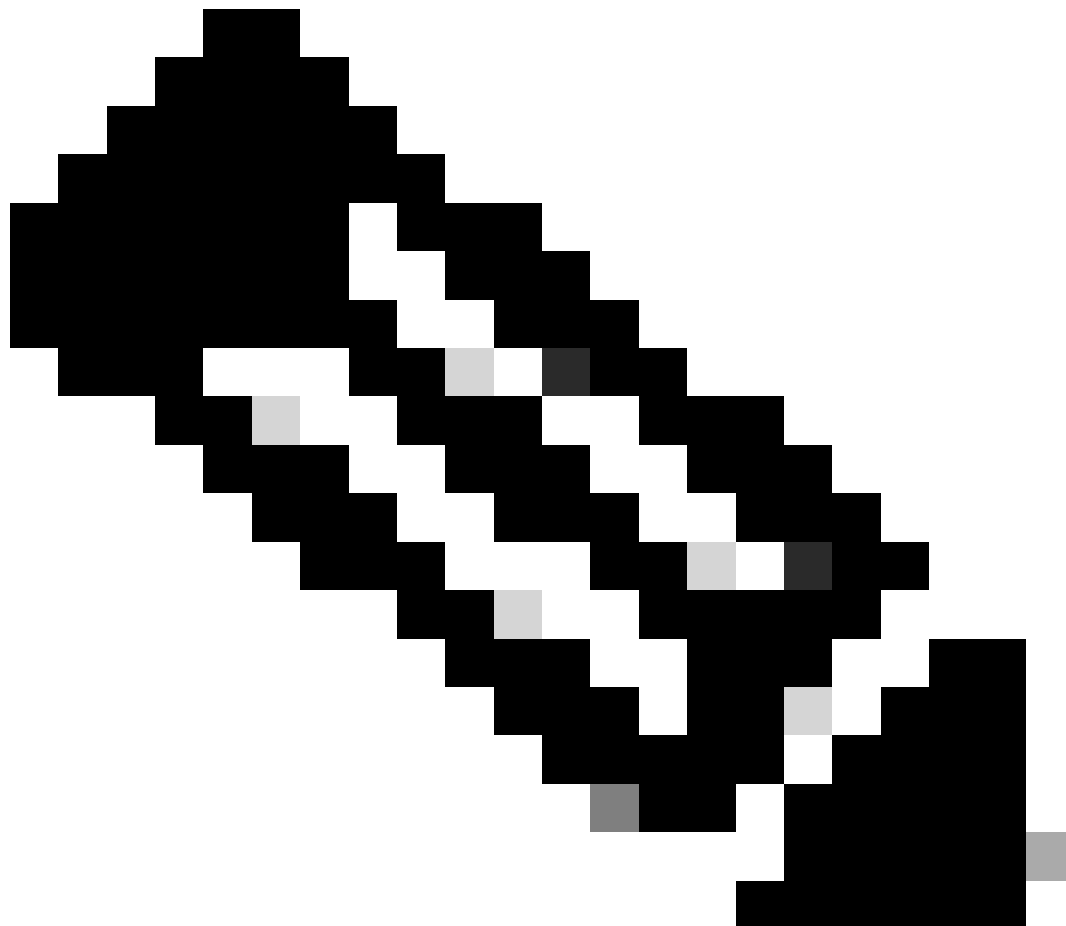
使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。Cisco IOS XEソフトウェアベースのプラットフォームのルーティングおよびスイッチングに適用されます。

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、クリアな（デフォルト）設定で作業を開始しています。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

デルタ増分に関するデバイスの本番メモリの使用状況を監視し、予測どおりの時間がかかるかどうかを確認します。このドキュメントでは、コールサイトとは何か、およびコールサイトがメモリ問題の迅速なトラブルシューティングにどのように役立つかについて説明します。



注：このドキュメントでは、ルーティングプロセッサ(RP)のダイナミックランダムアクセスメモリ(DRAM)のメモリ使用量のトラブルシューティングを中心に説明しています。

コールサイトとは何か

callsiteは、Cisco Technical Assistance Center(TAC)が使用するタグで、Cisco IOS-XE関連のプロセスによってメモリが割り当てられる間に呼び出されるソースコードの機能を確認および追跡するために使用されます。

お客様は、TACケースをオープンする前にこのタグを提供して、より迅速に解決できます。また

、この記事で後述するコマンドを使用して、このタグのデバッグを支援することもできます。

DiffコールとDiffバイト

Diffコールは、メモリ割り当てのコール数と割り当て解除のコール数の差を監視します。通常、大量のdiffコールはメモリ関連の問題を示している可能性があります。これは、差分が過剰に発生している場合に発生します。これは、システムがメモリを解放しておらず、割り当てが累積されていることを示します。

show processes memory platform accountingコマンドは、diffコールとdiffバイトの両方を表示できます。

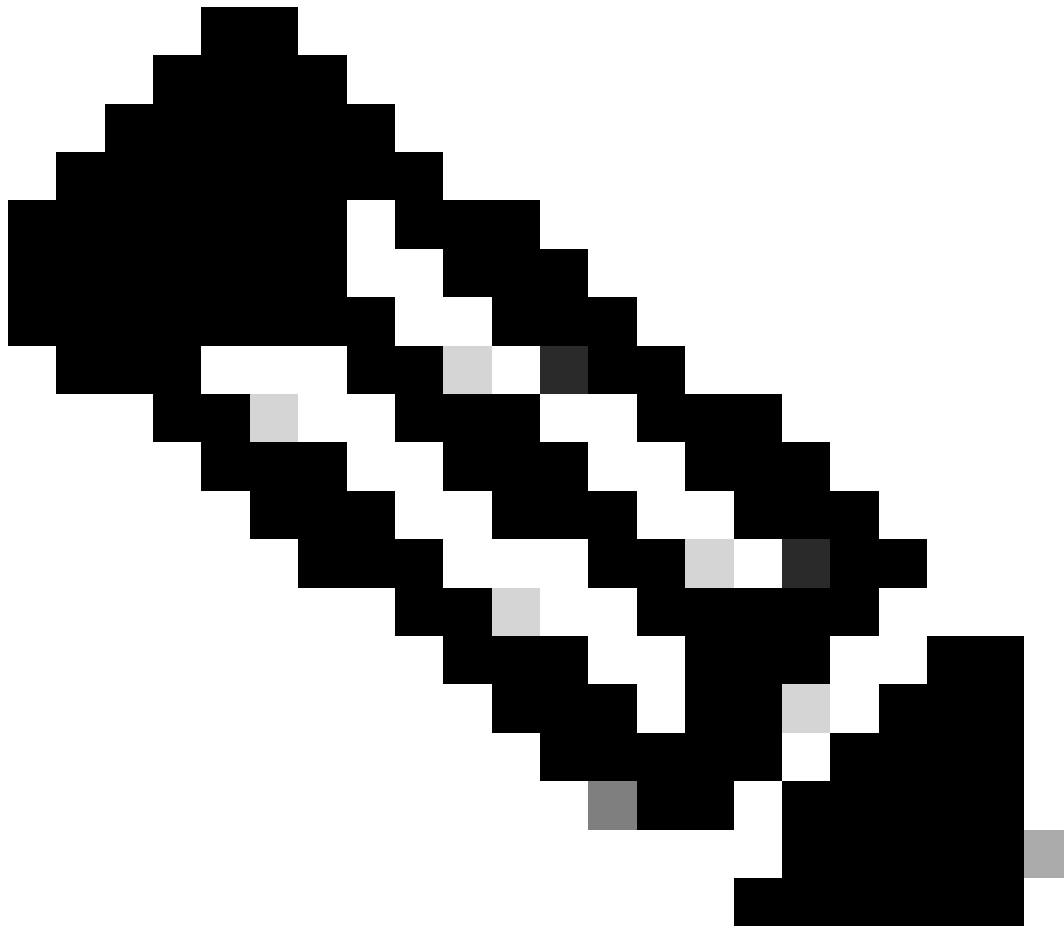
```
test1#show processes memory platform accounting
Hourly Stats
```

process	callsite_ID(bytes)	max_diff_bytes	callsite_ID(calls)	max_diff
sessmgrd_rp_0	F8E78C86E08C8003	1579683	E6A19D3ED0064000	
cli_agent_rp_0	A5E99693AA3B8004	1268440	5D11C89CA87A8003	
smand_rp_0	3DFF8F3C424F400A	918144	C34A609190E3C001	

システムには、メモリ警告と重大レベルのsyslogをトリガーする内部メモリ使用量のしきい値があります。これらのしきい値に基づくメモリ使用率は、show platform resourcesコマンドを使用して表示できます。

```
test1#show platform resources
**State Acronym: H - Healthy, W - Warning, C - Critical
Resource          Usage          Max          Warning      Critical      State
-----
RP0 (ok, active)
Control Processor  1.17%          100%         80%          90%          H
  DRAM             2639MB (34%)   7753MB       88%          93%          H
  bootflash        856MB (13%)    6338MB       88%          93%          H
  harddisk         0MB (0%)       0MB          88%          93%          H
ESP0(ok, active)
QFP
  TCAM             10cells (0%)   131072cells  65%          85%          H
  DRAM             89761KB (2%)   3670016KB    85%          95%          H
  IRAM             13525KB (10%)  131072KB     85%          95%          H
  CPU Utilization  1.00%          100%         90%          95%          H
  Crypto Utilization 3.00%          100%         90%          95%          H
  Pkt Buf Mem (0)   67KB (0%)      524288KB     85%          95%          H

test1#
```



注:TACケースを提出して、diffコールまたはdiffバイトが特定のプロセスに関連しているかどうかを判断してください。一般に、show processes memory platform sortedコマンドで確認できるように空きシステムメモリが少ない場合は、さらに詳しく調べる価値があります。

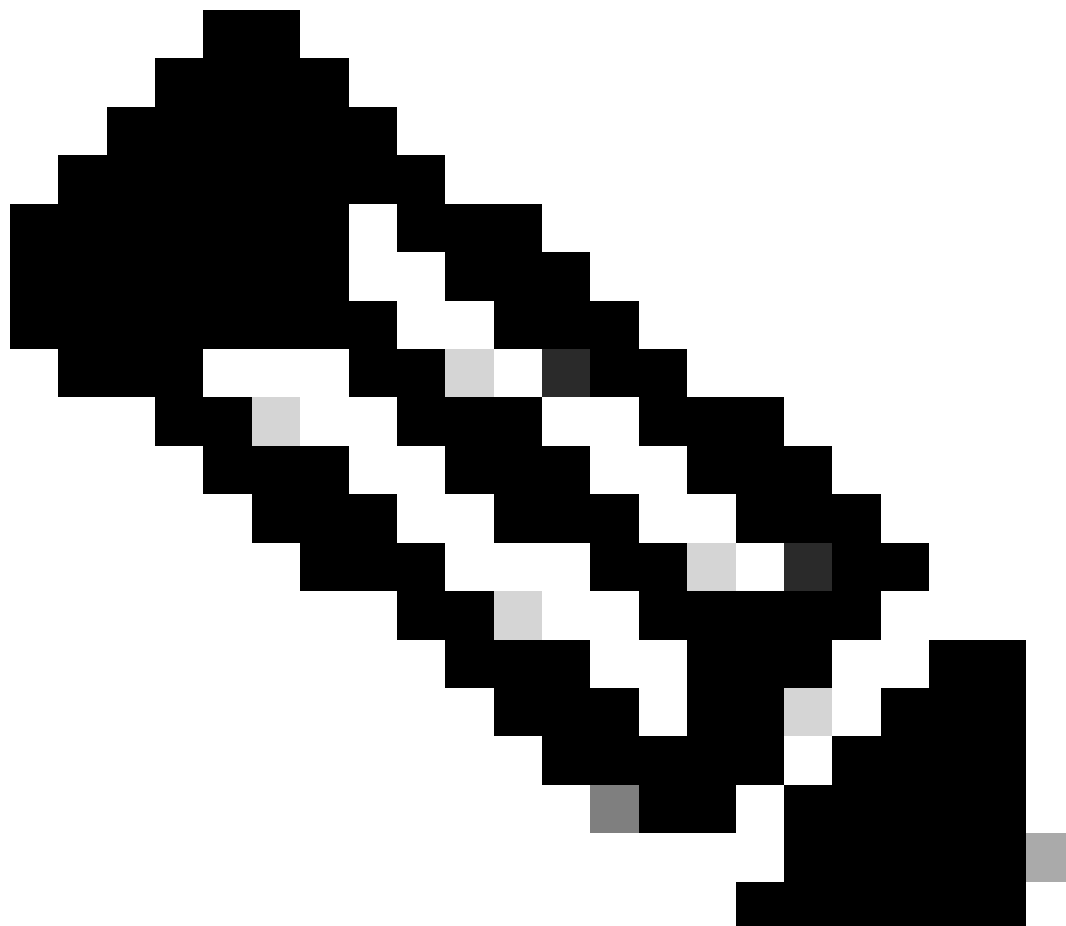
メモリ消費またはリーク問題の症状

Cisco IOS XE側でメモリ消費またはリークの問題が発生した場合、通常は次のような警告またはクリティカルアラームが生成されます。

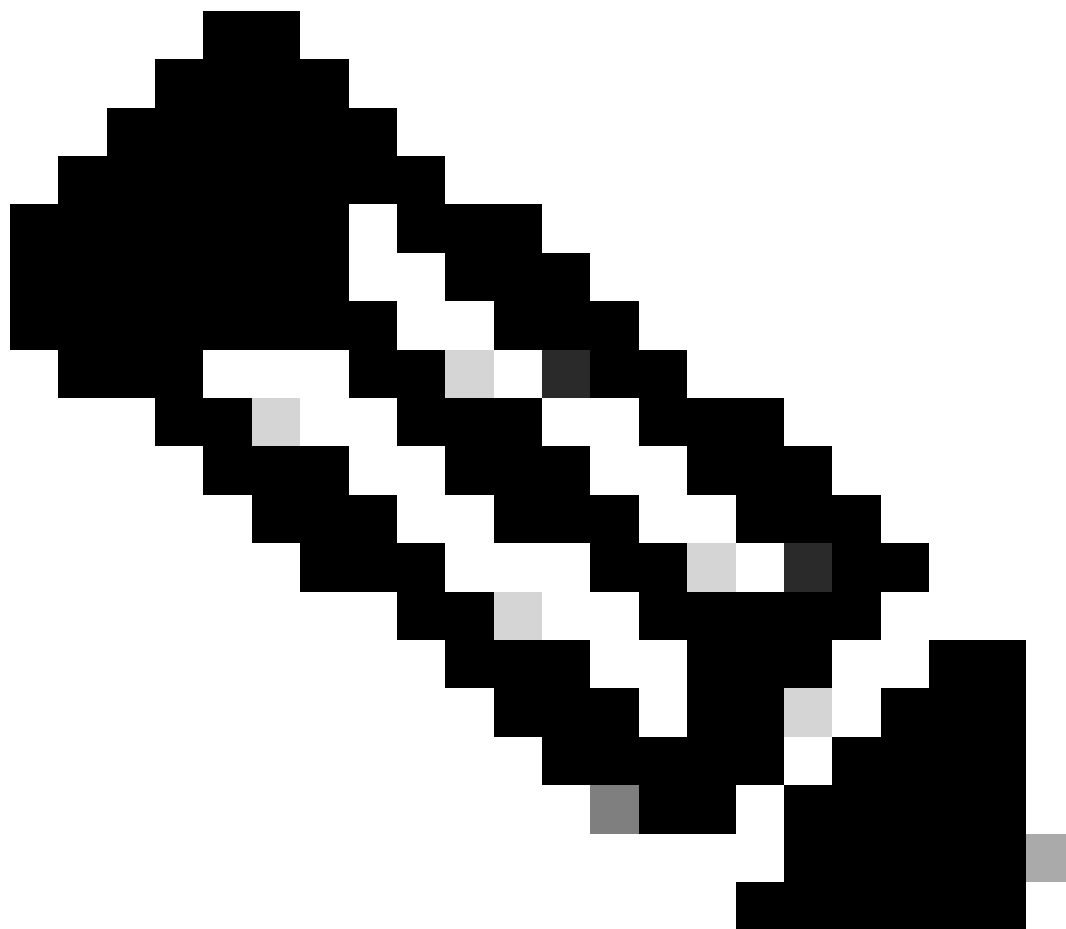
```
Nov 22 11:37:16.770: %PLATFORM-4-ELEMENT_WARNING: R0/0: smand: RP/0: Used Memory value 89% exceeds warn
```

このタイプのアラームでは、トラブルシューティングの開始点として重要な情報が強調表示されます。

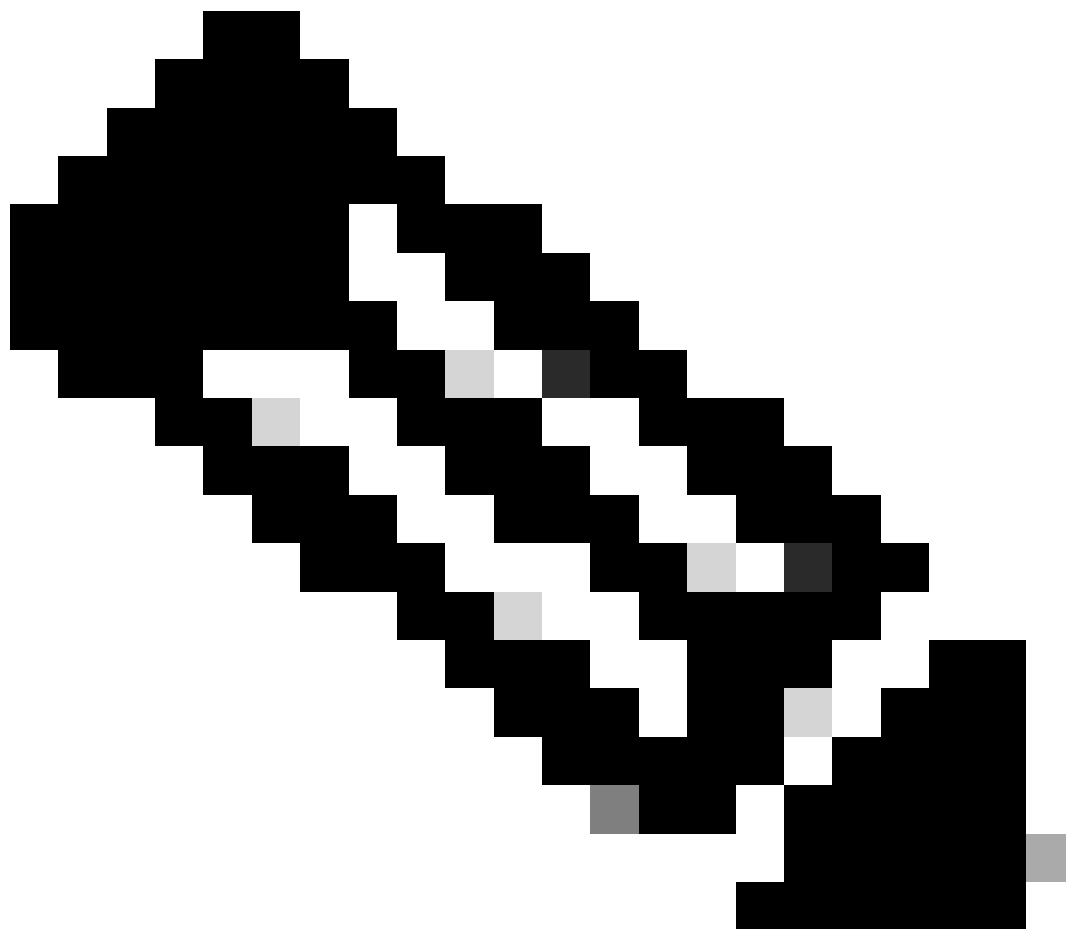
- アラームを受信した日時
 - 使用率
 - 影響を受けるコンポーネント
 - 上位のCisco IOS XEは、差分コールに基づいて上位のコンシューマとしてシステムが検出したプロセスを処理します。
-



注:%PLATFORM-4-ELEMENT_WARNINGアラームは、必ずしもメモリ消費問題の根本原因分析(RCA)を取得するための決定的なデータポイントではありません。



注:Temporal File System(TMPFS)、Quantum Flow Processor(QFP)、Cisco IOS daemon(IOSd)などの異なるコンポーネントに関連付けられている症状やメモリ使用量のアラームは他のタイプのものもありますが、これらのタイプについてはこのドキュメントでは説明していません。

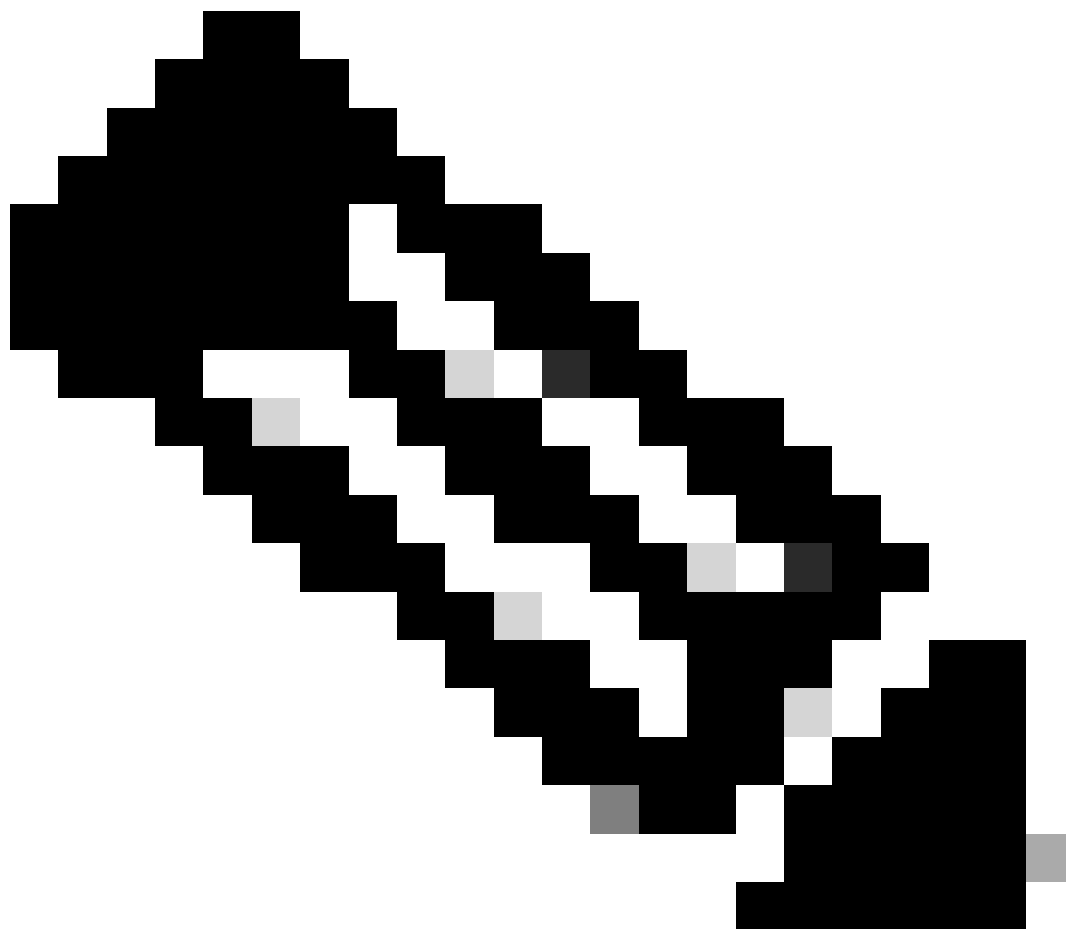


注：このドキュメントでは、Cisco IOSデーモン(IOSd)でメモリの問題を示すSYS-2-MALLOCFAIL syslogメッセージのトラブルシューティングについては説明していません。

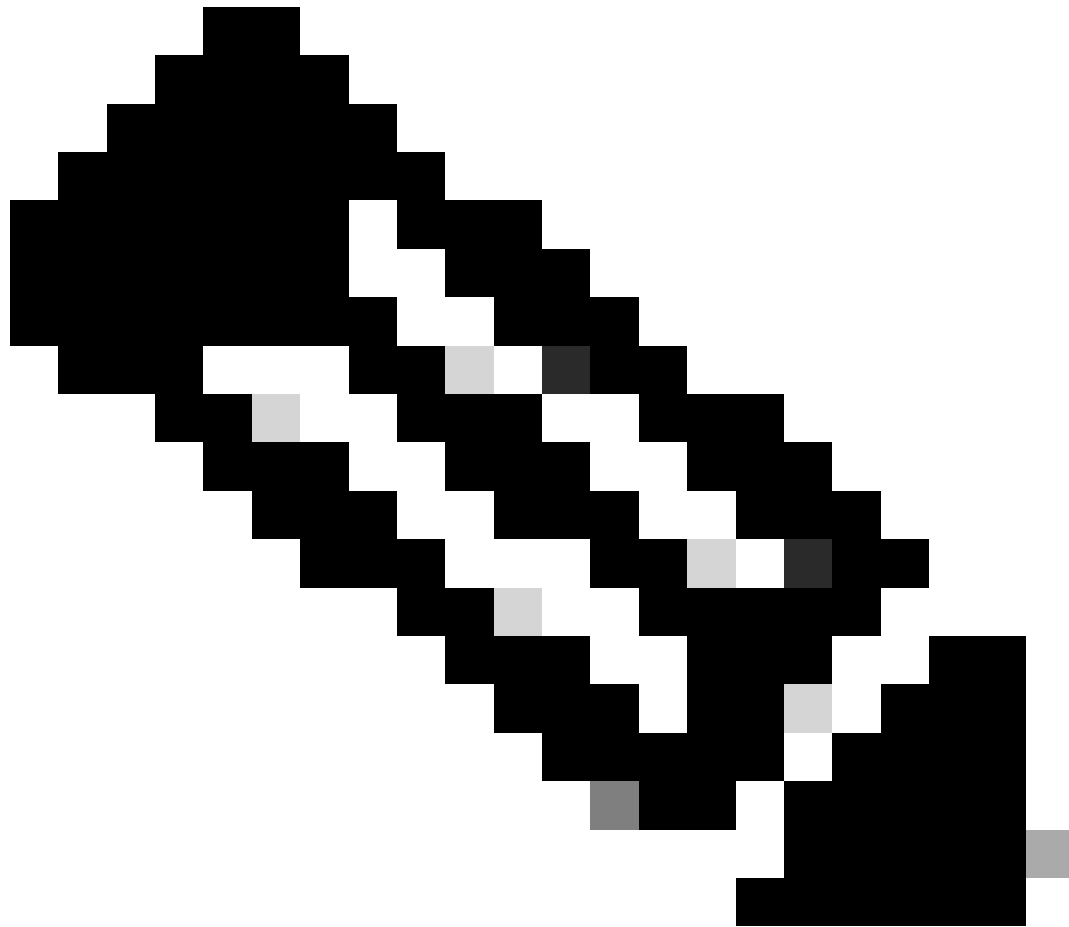
トラブルシューティングのシナリオ

メモリ不足のためデバイスがクラッシュした

メモリリソースの不足が原因でデバイスがクラッシュした場合、syslogメッセージ「%PLATFORM-4-ELEMENT_WARNING: R0/0: smand: RP/0: Used Memory value X% exceeds warning level Y% is present.



注：メモリ不足が原因でクラッシュが発生すると、ローカルDRAMバッファからsyslogが
消去されるので注意してください。そのため、クラッシュイベントが必要になる前に
syslogサーバからアーカイブログを確認します。syslogサーバがまだ設定されていない場
合は、『[Cisco IOSでのロギングの設定方法](#)』を参照してください。

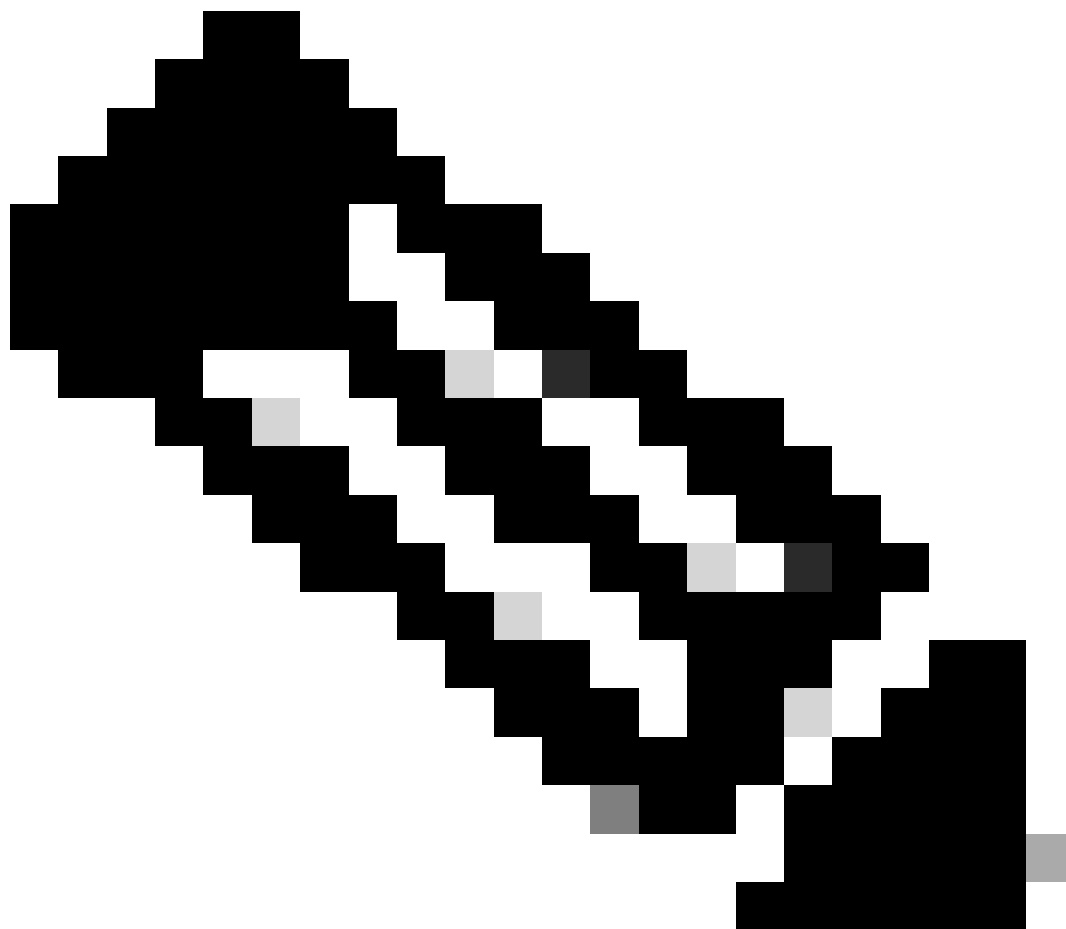


注：クラッシュイベント後の%PLATFORM-4-ELEMENT_WARNING: R0/0: smand: RP/0: Used Memory value X% exceeds warning level Y%アラームは、デコードされたCisco IOSトレースログでも確認できます。詳細は、『[統合ロギング拡張機能によるトレースログの収集と管理](#)』を参照してください。

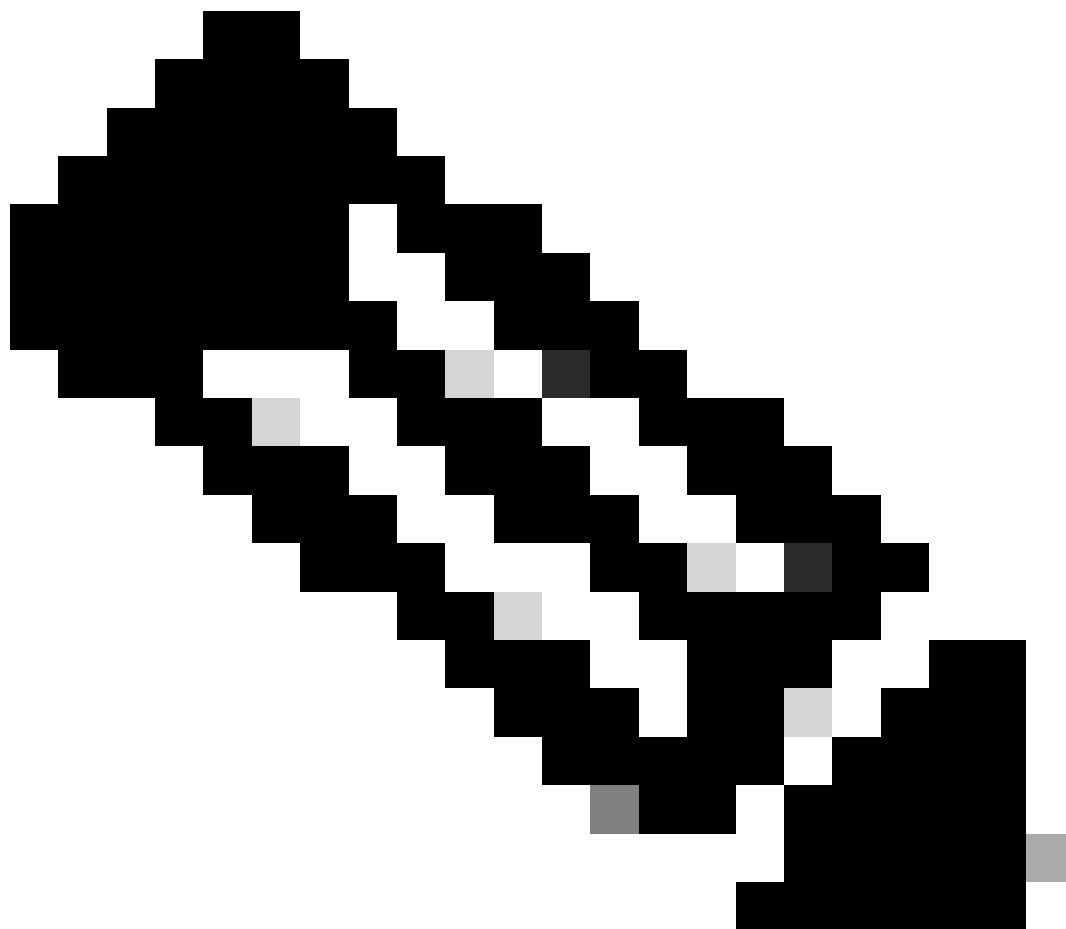
メモリ不足のため、システムはクラッシュしました。これにより、システムレポートが生成されます。このレポートは、メモリの問題の調査に利用できる関連データを含む.tar.gzファイルです。詳細については、『[システムレポートを使用したトラブルシューティング](#)』を参照してください。

解凍すると、システムレポートのtmpディレクトリ内にmaroon statsというディレクトリが作成されます。Maroon statsは、さまざまなCisco IOS XEプロセスのdiffコールおよびバイト内のメモリ割り当ておよび割り当て解除を追跡するコードに実装されたサービスアビリティ機能です。

Maroon統計スナップショットはシステムレポート内に含まれ、メモリ消費またはリーク問題のRCAを特定したり、詳細にデバッグして理解するために潜在的な原因となるコールサイトを特定するのに役立ちます。



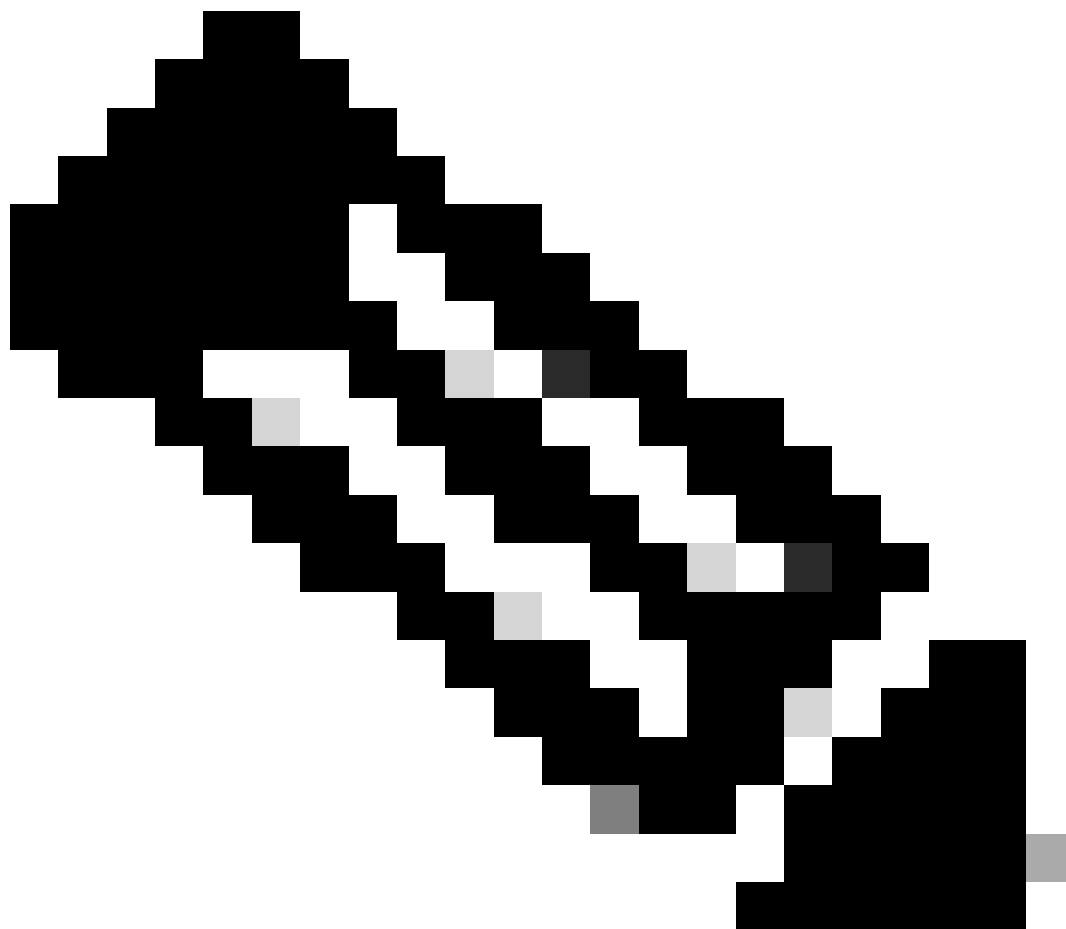
注：システムレポートからmaroon statsディレクトリをデコードできるのは、TACだけです。このディレクトリには、コードの内部関数および機密関数が含まれており、TACエンジニアがコードのどの関数にメモリが割り当てられているかを理解するために役立ちます。TACケースを提出し、システムレポートを提供してください。



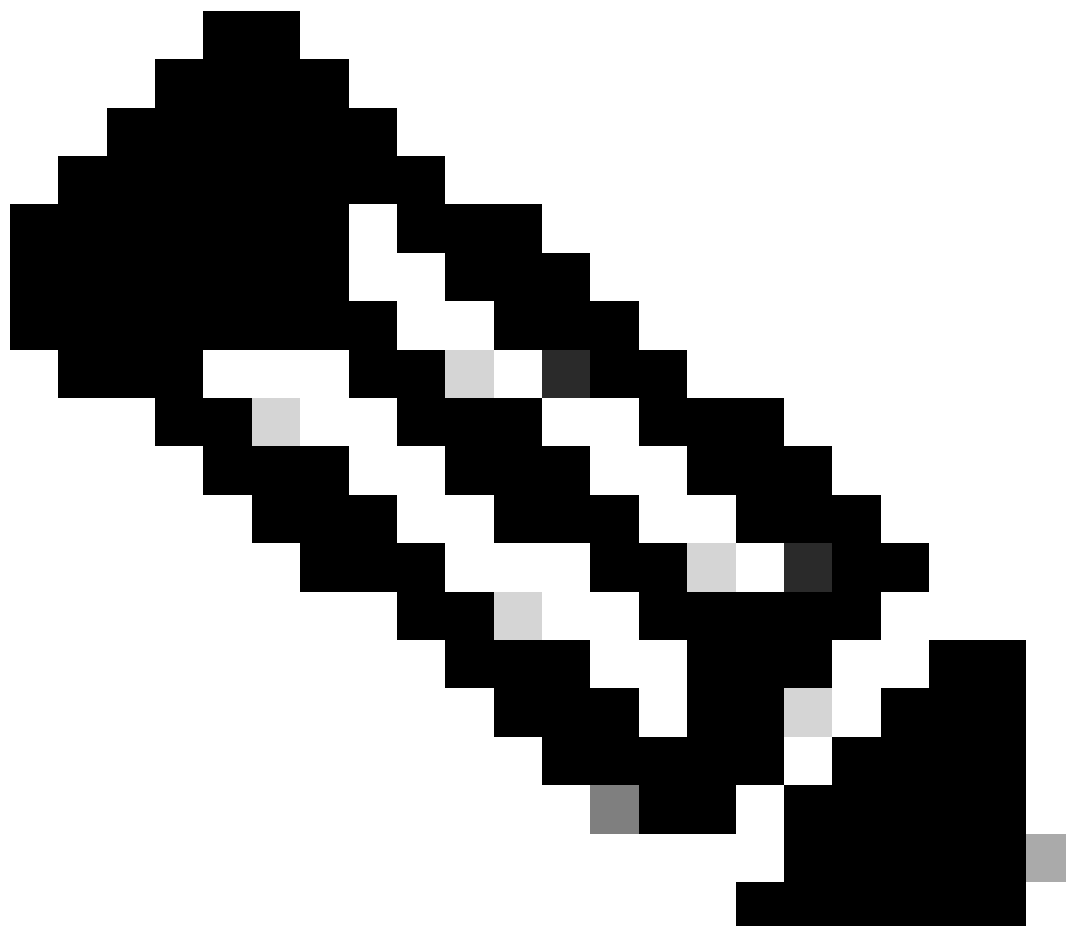
注：システムレポートには、メモリ不足が原因で発生するメモリクラッシュを把握するのに十分な量のデータが提供されますが、場合によっては、さらにメモリの追跡、モニタリング、デバッグ、およびトラブルシューティングが必要になります。

デバイスはまだクラッシュしていませんが、メモリ使用量の警告があります

show platform resources コマンドは、警告と重大なメモリ使用量のしきい値を表示します。



注：メモリ消費またはリークがどれだけ速く発生するかによって、デバイスはメモリリソースの不足が原因でクラッシュする可能性があるため、デバッグのためにメモリ関連の出力コマンドをさらに収集することがベストプラクティスです。



注：メモリ使用量の警告が表示された場合は、TACにケースを提出して、show tech-supportコマンドと

show tech-support memoryを発行します。このメモリは、TACエンジニアが問題を最初にトリアージし、潜在的に迅速にRCAを見つけるのに役立ちます。

デバイスがまだクラッシュしておらず、ローカルsyslogバッファでメモリアラームを生成しているか、監視ツールを介してsyslogサーバから受信されている場合は、show processes memory platform sortedの出力を収集して、問題のプロセスで消費されているバイトを特定します。

```
Router#show processes memory platform sorted
```

```
System memory: 4027884K total, 2580612K used, 1447272K free,
```

```
Lowest: 1447272K
```

Pid	Text	Data	Stack	Dynamic	RSS	Total	Name
21240	263436	858000	136	308	858000	3632460	linux_iosd-imag
27232	12877	195460	136	23592	195460	2231316	fman_fp_image
26797	90	157260	136	22308	157260	1741996	cpp_cp_svr

19194	7325	102756	136	2376	102756	1318608	fman_rp
27179	18745	242708	136	448	242708	1160248	qfp-ucode-utah

この出力では、Resident Set Size (RSS)列を確認します。これは、各Cisco IOS XEプロセスが消費しているキロバイト数を示します。

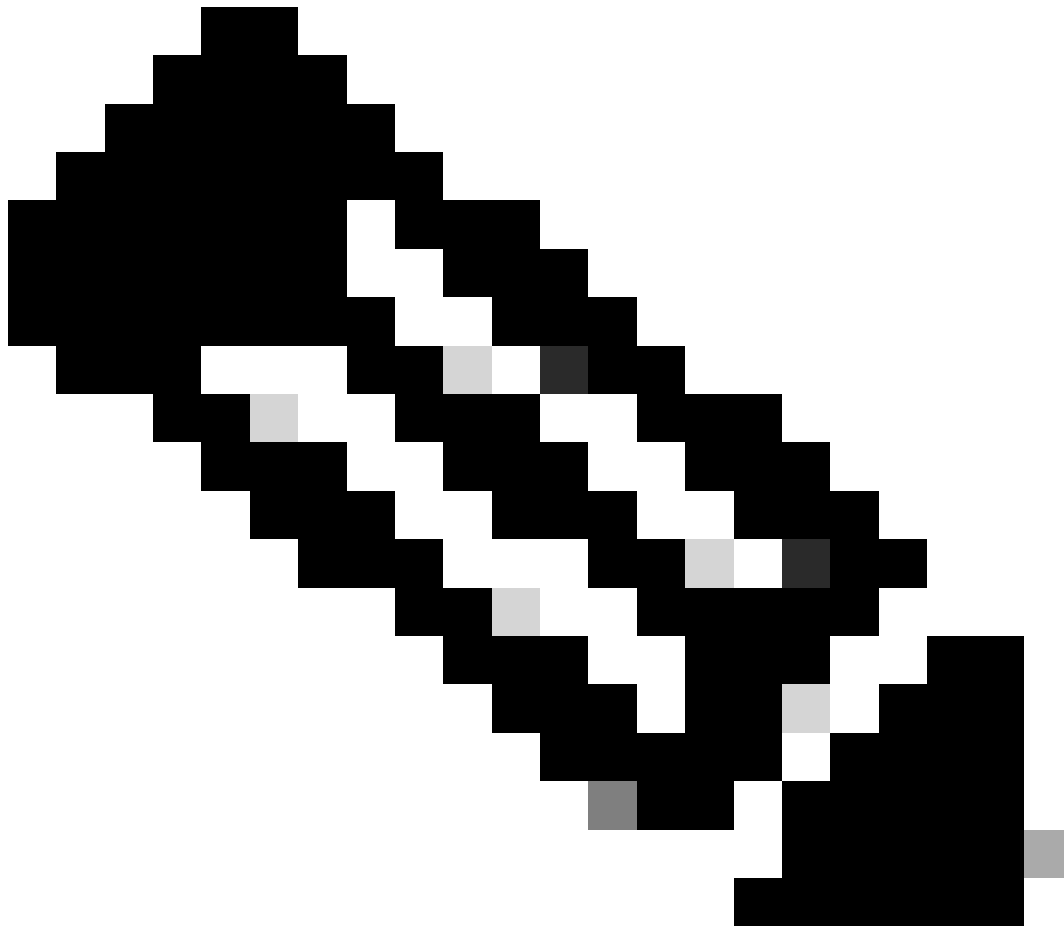
コールサイトの特定

次に、異なるプロセスのdiffコールとバイトの値を示すshow processes memory platform accountingの出力を収集します。通常は、より大きな値に焦点を当てます。

diff call bytesは潜在的なメモリリークがあるかどうかを判断する良い指標です。これは、あるプロセスによって、システムにリリースされずにシステムが保持しているメモリのバイト数を示しています。

このデータに基づいて、どちらが不正なプロセスのcallsiteタグで、より大きなdiff calls値とbytes値を持っているのかを特定できます。

show process memory platform accountingでは、これらのdiffコールとバイトが時間の経過とともに追跡されます。場合によっては、このコマンド出力の最後にバックトレースが含まれています。バックトレースは内部ツールを使用してデコードされるため、これはTACエンジニアにとって重要であり、潜在的なメモリリークを引き起こす可能性のあるコードの機能を判別するのに役立ちます。



注：メモリリークの問題をトラブルシューティングするのに十分な情報がshow process memory platform accounting コマンドで提供されない場合、多くの場合、プロセスのデバッグをさらに行う必要があります。

コールサイトを識別する第2の方法については、このドキュメントの「コールサイトのデバッグ」も参照してください。

モジュール、データベース、およびメッセージングでのプロセスメモリの消費

Cisco IOS XEプロセスのメモリリークをさらにデバッグするには、特定のCisco IOS XEプロセスで次のコマンドを収集する必要があります。

```
# Allocations and frees per module  
show platform software memory
```

```
show platform software memory
```

```
bri
```

```
# Database diff and entries statistics  
show platform software memory database
```

```
| ex diff:0  
show platform software memory database
```

```
bri | ex _0_
```

```
# Messaging diff and entries statistics  
show platform software memory messaging
```

```
| ex diff:0  
show platform software memory messaging
```

```
brief | ex _0_
```


これらのコマンド出力は、プロセスによって発生したメモリリークの調査を補完するものであり、最初の基本的なトリアージコマンドで十分な情報が得られない場合に必要となります。

コールサイトのデバッグ

コールサイトを識別する2番目の方法は、サイトをデバッグすることです。次のコマンドが必要です。

```
debug platform software memory
```

```
alloc callsite start  
show platform software memory
```

```
alloc callsite brief  
debug platform software memory
```

```
alloc backtrace start
```

```
depth 10  
show platform software memory
```

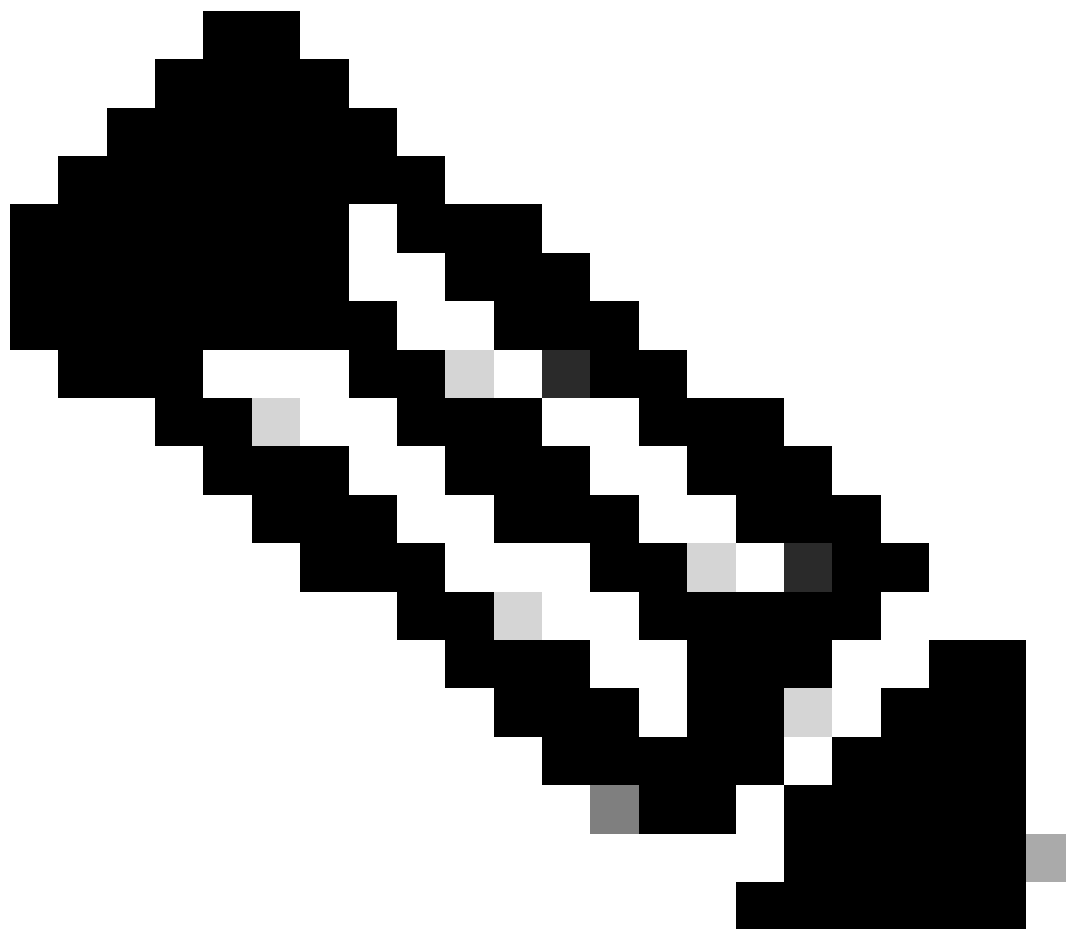
```
alloc backtrace
```

最初のコマンドは、プロセスの呼び出しサイトの割り当てのデバッグを有効にします。新しいバージョンでは、このコマンドはデフォルトで有効になっており、サービスに影響を与えることはありません。

show platform software memory <process> <location> alloc callsite brief コマンドにより、そのプロセスのコールサイト、および各コールサイトの差分コールおよびバイトを示すテーブルが提供されます。たとえば、次の例ではCisco IOSプロセスの出力を示していますが、他の任意のプロセスについて収集できます。

```
test1# show platform software memory ios r0 alloc callsite brief
The current tracekey is : 1#b1ba773f123f8d990fd84c82c1d0e1d3
```

callsite	thread	diff_byte	diff_call
3DFF8F3C424F4004	4115	57384	1
ABB2D8F932038000	4115	57360	1
3869885745FC8000	4115	16960	1
DF884D58A8EF0004	4115	8208	1
DF884D58A8EF0008	4115	8208	1
FAE69298A17B8000	4115	4243	165
FAE69298A17B8001	4115	2640	165
FAE69298A17B8002	4115	1958	12



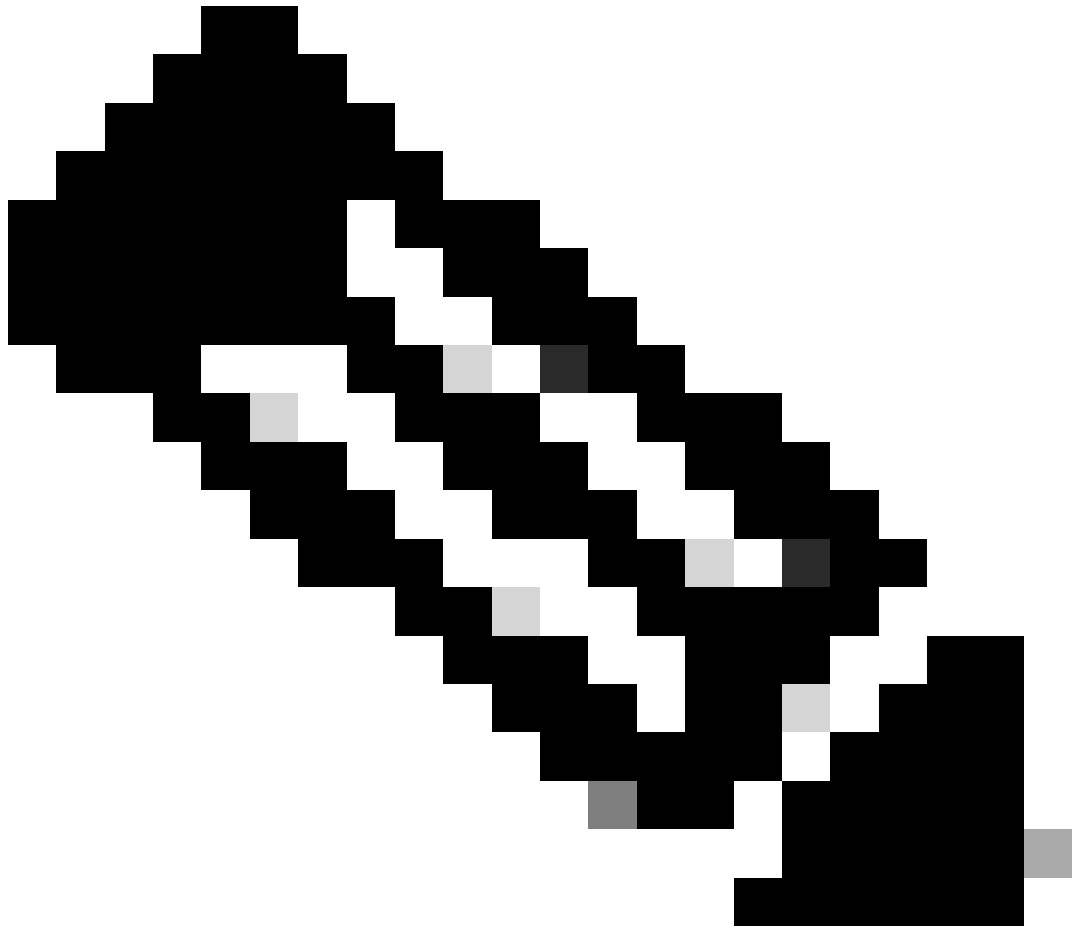
注：コマンドshow plat soft memory <process> <location> alloc callsite briは、diff callまたはbytesカラムの増加を見つけるまで複数回実行する必要があります。これは、システムがこのようなメモリをリリースせずに保持していることを示しています。

コールサイトのリークが特定されたら、そのコールサイトに対してコマンドdebug platform software memory <process> <location> alloc backtrace start <callsite> depth 10を実行する必要があります。このコマンドはそのまま残してもかまいません。サービスに影響を与えることはありません。

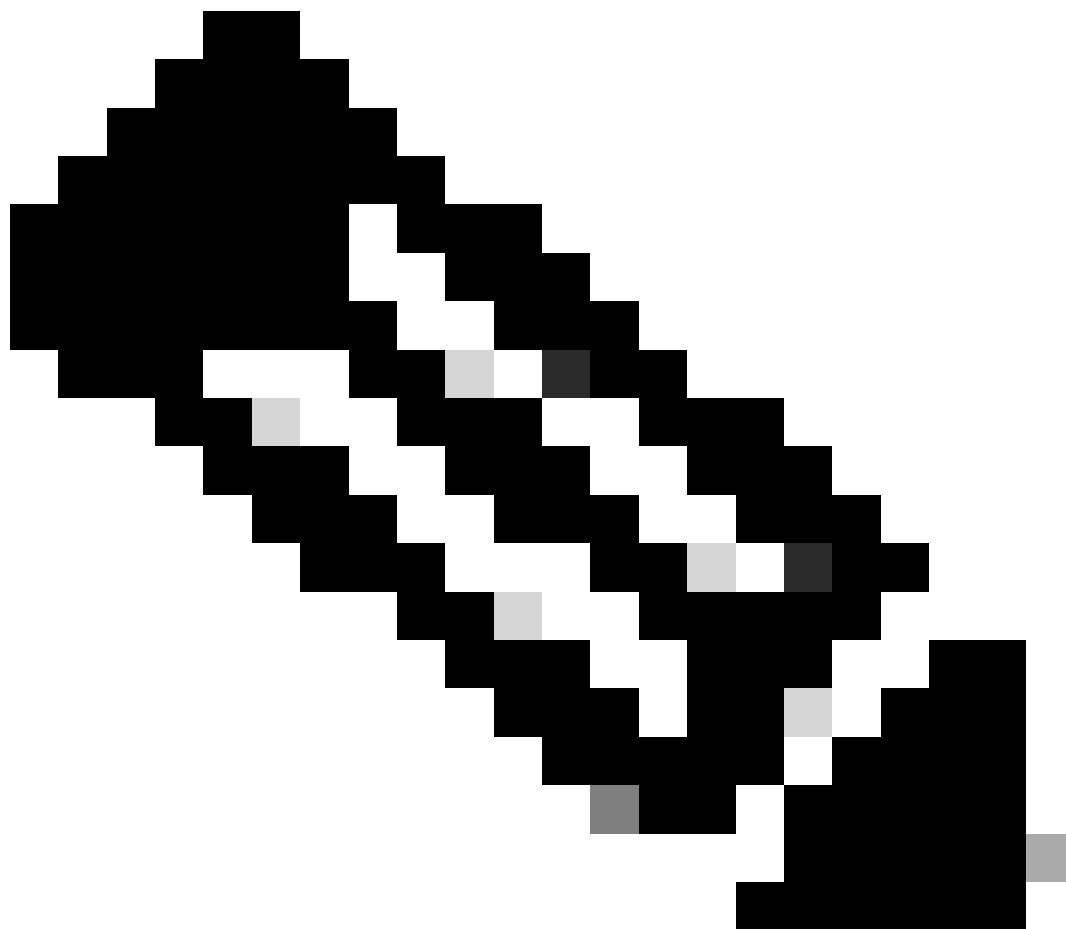
show plat soft memory <process> <location> alloc callsite briコマンドを、diff calls/bytesの増加が見られるまで再度実行する必要があるのは、識別されたcallsiteのデバッグを有効にした後で、これはそのcallsiteにメモリを割り当てるコードの機能を追跡するためです。その後、show platform software memory <process> <location> alloc backtraceを使用して、バックトレースを収集できます。次に例を示します。

```
show platform software memory install-manager switch active R0 alloc back
```

backtrace: 1#83e58872a4792de086bf7191551098d7 maroon:7FCBACB87000+4642 maroon:7FCBACB87000+579C repm
callsite: 7BD5593C00E30000, thread_id: 15556
allocs: 70, frees: 0, call_diff: 70



注：この出力をTACに提供してバックトレースをデコードすると、TACエンジニアがコード内の動作を確認し、既存の不具合があるかどうかを判断したり、動作をよりよく理解したりできます。必要に応じて、TACから開発者チームに連絡できます。



注：ソフトウェアを必ず最新の状態にしてください。新しいソフトウェア不具合が見つかった場合、TACはDeveloper Teamと協力して詳細なデバッグを行い、状況を調査します。

関連記事およびドキュメント

- [ASR 1000 シリーズ ルータ メモリのトラブルシューティング ガイド](#)
- [IOS-XEのメモリ使用量の調査](#)
- [EEMによるメモリ使用率の問題の報告](#)
- [Cisco 4000シリーズISRのメモリトラブルシューティングガイド](#)

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。