"Warn" 또는 "Over" 상태에서 Essmgr/Aaamgr 문제 해결

목차

소개

이 문서에서는 "경고" 또는 "오버" 상태의 sessmgr 또는 aaamgr 문제를 해결하는 방법에 대해 설명합니다.

개요

세션 관리자(Sessmgr) - 여러 세션 유형을 지원하는 가입자 처리 시스템이며 가입자 트랜잭션을 처리합니다. Sessmgr은 일반적으로 AAAManagers와 쌍을 이룹니다.

Aaamgr(Authorization, Authentication, and Accounting Manager) - 시스템 내의 가입자 및 관리 사용자에 대해 모든 AAA 프로토콜 작업 및 기능을 수행합니다.

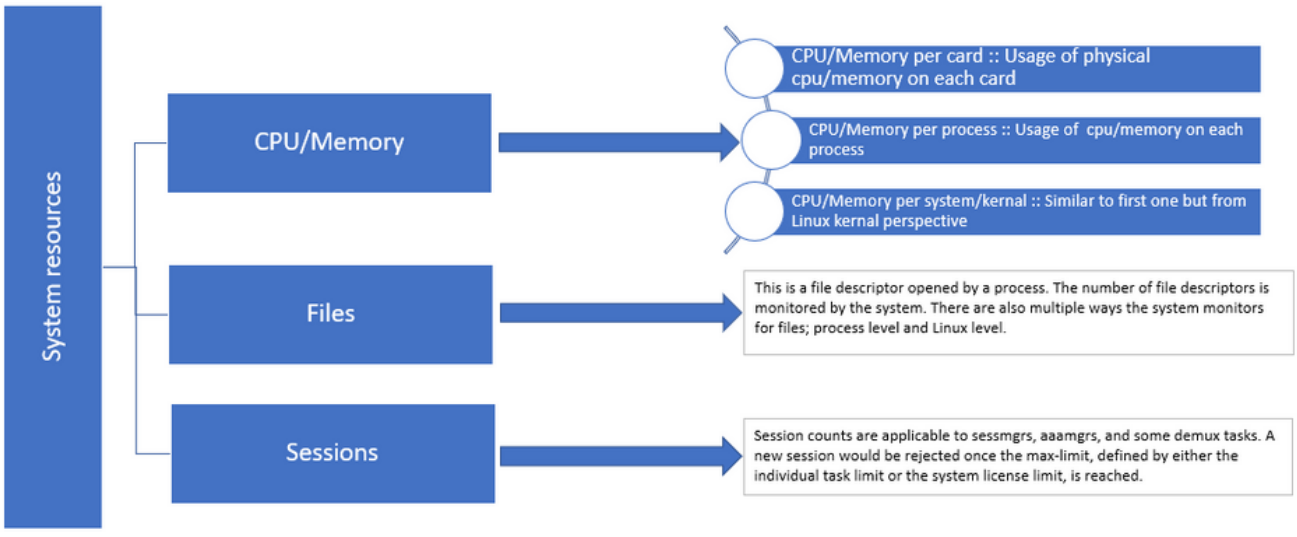


그림 1 :: Staros 리소스 분포

로그/기본 검사

기본 검사

문제에 대한 자세한 정보를 수집하려면 사용자에게 이 정보를 확인해야 합니다.

1. Sessmgr/aaamgr이 "경고" 또는 "오버" 상태로 유지된 기간은 얼마나 됩니까?
2. 이 문제로 인해 영향을 받는 세션/aaamgrs는 몇 개입니까?
3. 메모리 또는 CPU로 인해 essmgr/aaamgr이 "경고" 또는 "오버" 상태인지 확인해야 합니다.
4. 또한 트래픽이 갑자기 증가했는지 확인해야 합니다. 이는 세션당 세션 수를 검토하여 평가할 수 있습니다.

이 정보를 확보하면 당면한 문제를 더 잘 이해하고 해결할 수 있습니다.

로그

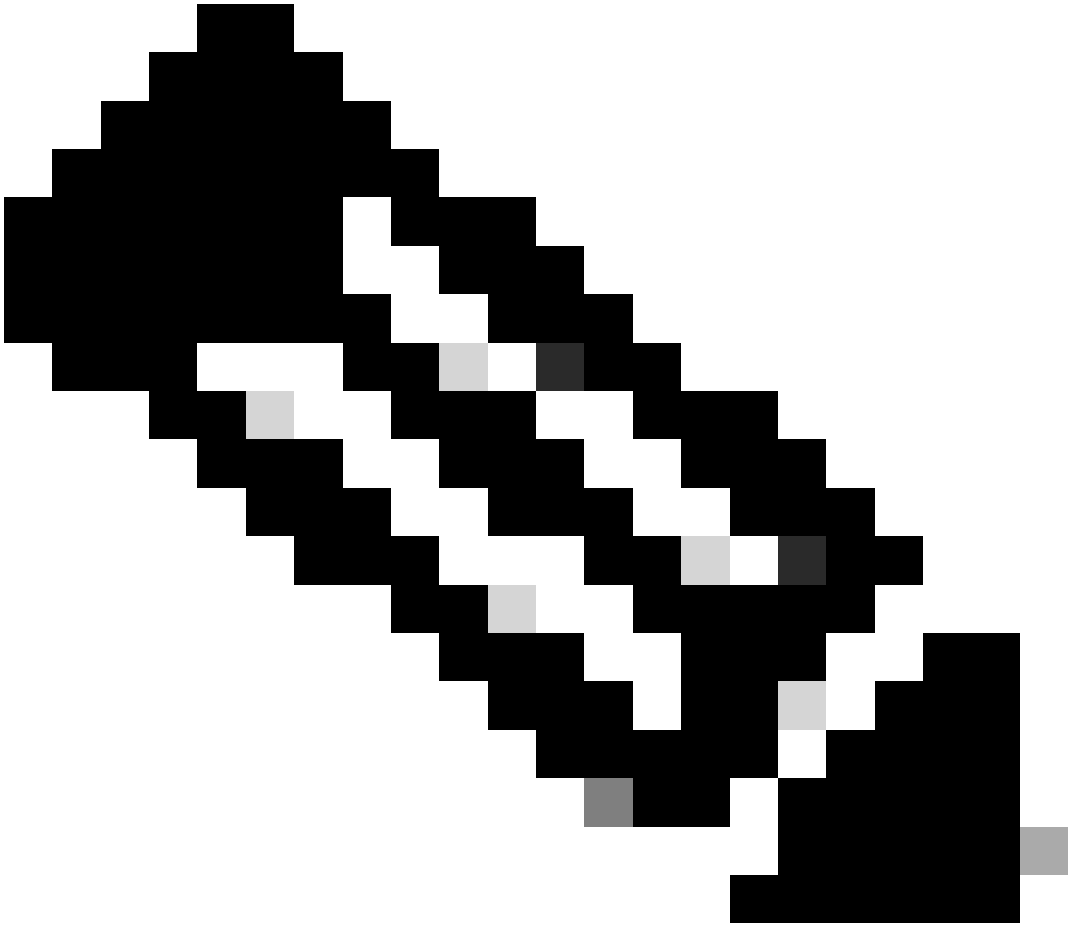
1. 문제가 되는 타임스탬프를 캡처하는 SSD(Show Support Details) 및 syslogs를 가져옵니다. 트리거 포인트를 식별하려면 문제가 발생하기 최소 2시간 전에 이러한 로그를 수집하는 것이 좋습니다.

2. 문제가 있는 세션관리자/아아암거 및 문제가 없는 세션관리자/아암거 모두의 코어 파일을 캡처합니다. 이에 대한 자세한 내용은 분석 섹션에서 확인할 수 있습니다.

분석

1단계. 영향을 받는 sessmgr/aaamgr의 상태를 명령으로 확인합니다.

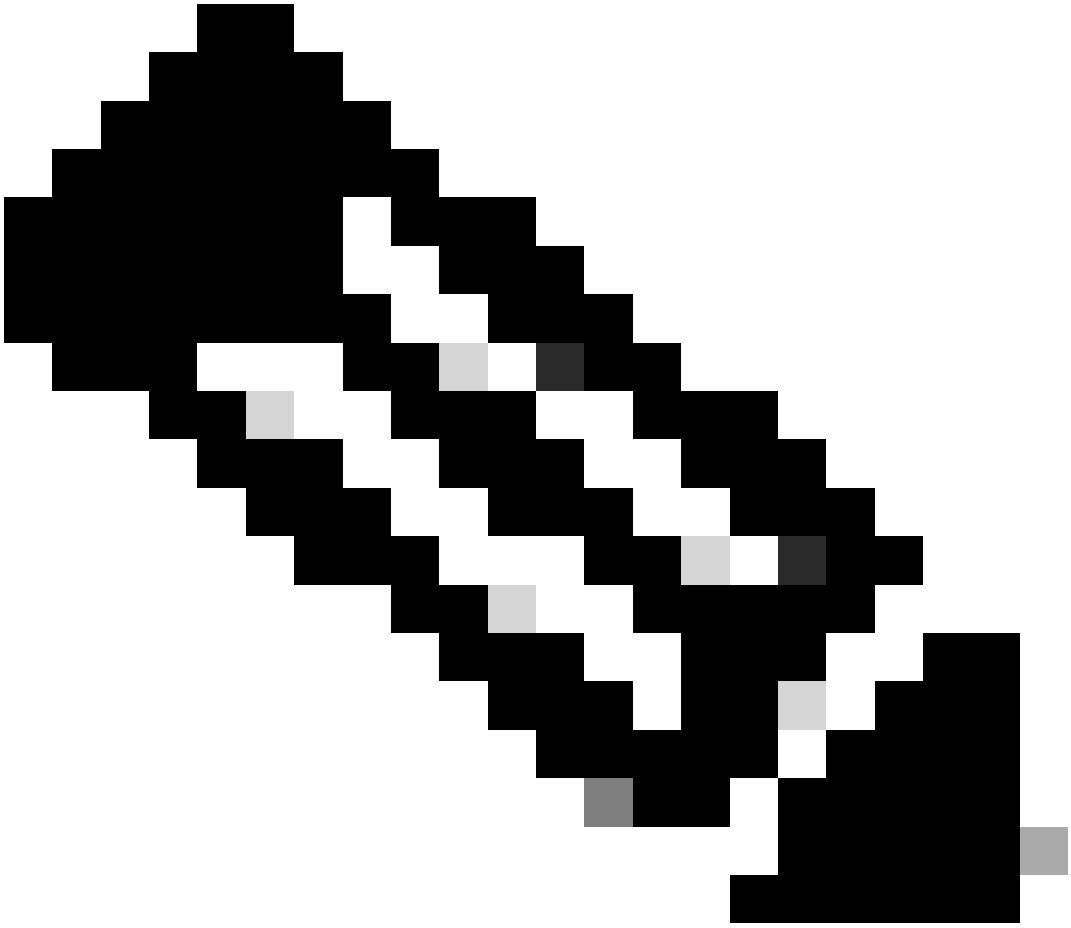
```
show task resources         -
--------- to check detail of sessmgr/aamgr into warn/over state and from the same you also get to know

Output ::

******** show task resources *******
Monday May 29 08:30:54 IST 2023
     task          cputime      memory         files      sessions
cpu facility inst used  alloc used alloc    used allc    used allc   S   status
---------------------- ----------- ------------ --------- ------------ ------
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
```



참고: sessmgr당 세션 수는 명령 출력에 표시된 대로 이 명령으로 확인할 수 있습니다.

다음 두 명령 모두 노드가 다시 로드된 이후의 최대 메모리 사용량을 확인하는 데 도움이 됩니다.

```
show task resources max
show task memory max

 ******** show task memory max *******
Monday May 29 08:30:53 IST 2023
             task   heap     physical                    virtual
cpu facility inst   max          max     alloc           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
```



참고: memory max 명령은 노드가 다시 로드된 이후 사용된 최대 메모리를 제공합니다. 이 명령은 문제가 최근 다시 로드 이후에 시작되었는지 또는 최대 메모리 값을 확인할 수 있는 최근 다시 로드가 있었는지 등 문제와 관련된 모든 패턴을 식별하는 데 도움이 됩니다. 반면 "show task resources" 및 "show task resources max"는 유사한 출력을 제공하며, max 명령은 다시 로드 이후 특정 sessmgr/aaamgr에서 사용하는 메모리, CPU 및 세션의 최대값을 표시한다는 점이 다릅니다.

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

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

실행 계획

시나리오 1. 메모리 사용률이 높기 때문에

1. sessmgr 인스턴스를 다시 시작/종료하기 전에 SSD를 수집합니다.
2. 영향받는 Sessmgr에 대한 코어 덤프를 수집합니다.

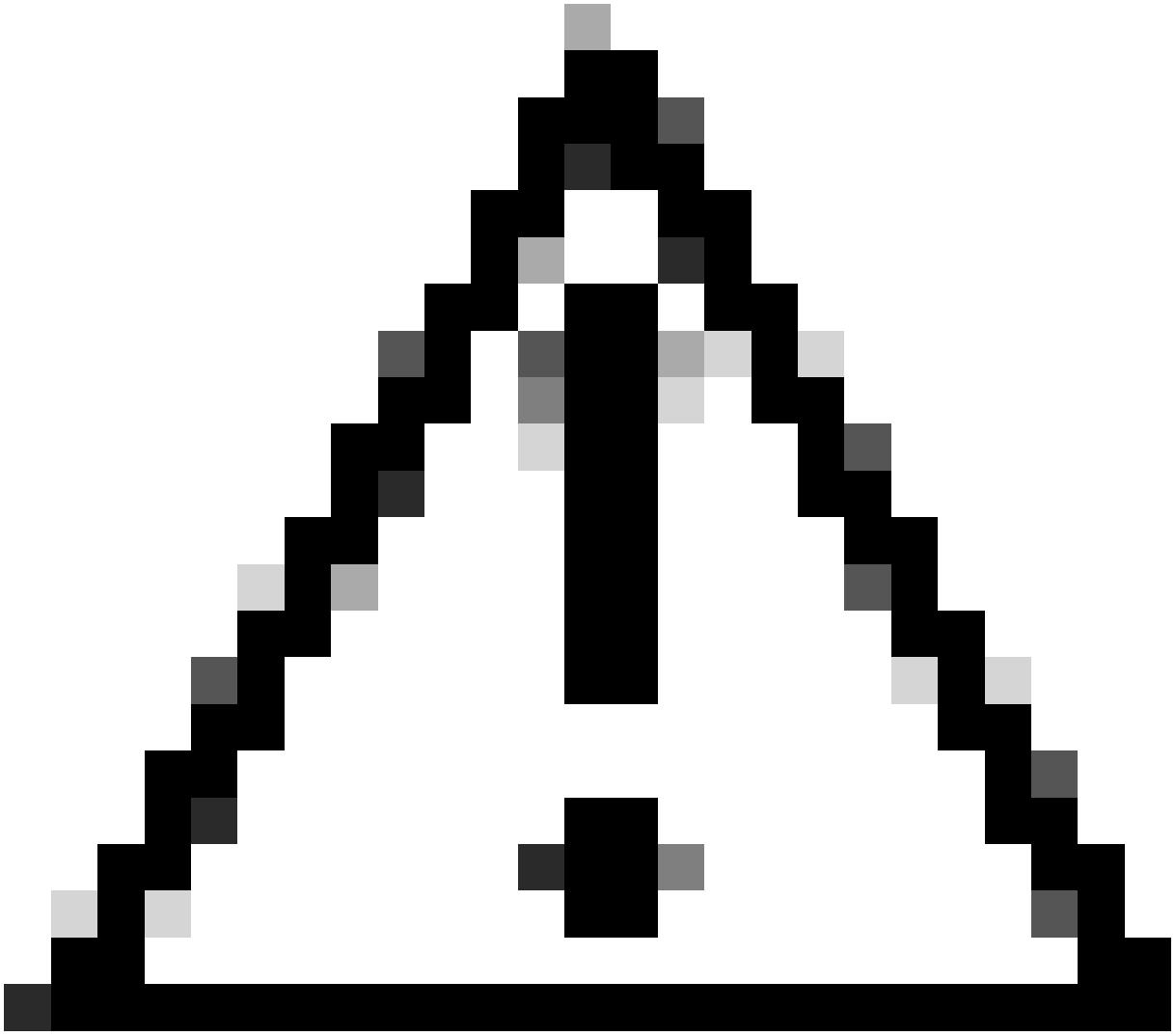
```
task core facility sessmgr instance <instance-value>
```

3. 영향을 받는 동일한 sessmgr 및 aaamgr에 대해 숨김 모드에서 이러한 명령을 사용하여 힙 출력을 수집합니다.

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

Heap outputs:

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

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

4. 다음 명령을 사용하여 sessmgr 작업을 다시 시작합니다.

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



주의: "경고" 또는 "오버" 상태의 세션이 여러 개 있는 경우 2~5분 간격으로 세션을 다시 시작하는 것이 좋습니다. 처음에는 2~3개의 세션만 다시 시작한 다음 최대 10~15분 동안 기다렸다가 해당 세션이 정상 상태로 돌아갈지 확인합니다. 이 단계는 재시작의 영향을 평가하고 복구 진행 상황을 모니터링하는 데 도움이 됩니다.

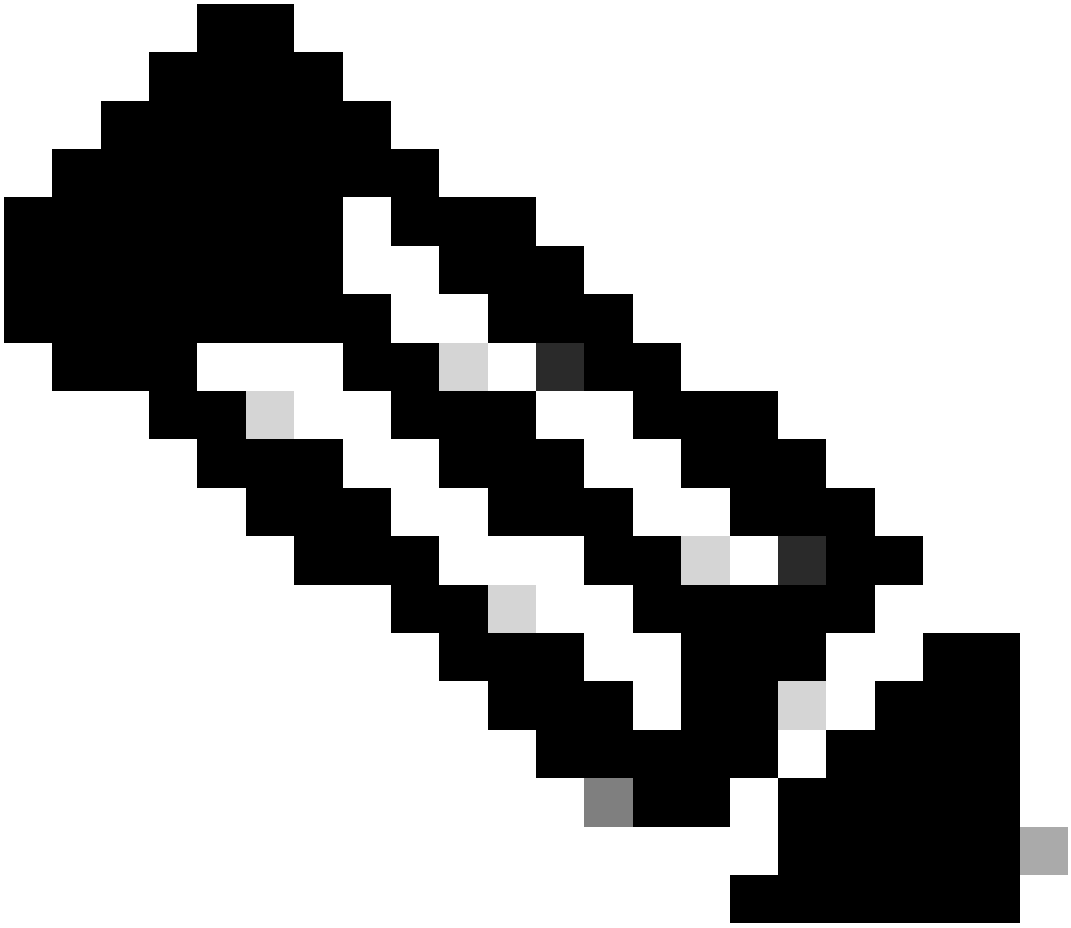
5. Sessmgr 상태를 확인합니다.

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

6. 다른 SSD를 수집합니다.

7. 3단계에서 언급한 모든 CLI 명령의 출력을 수집합니다.

8. 2단계에서 언급한 명령을 사용하여 정상적인 sesmgr 인스턴스에 대한 코어 덤프를 수집합니다.

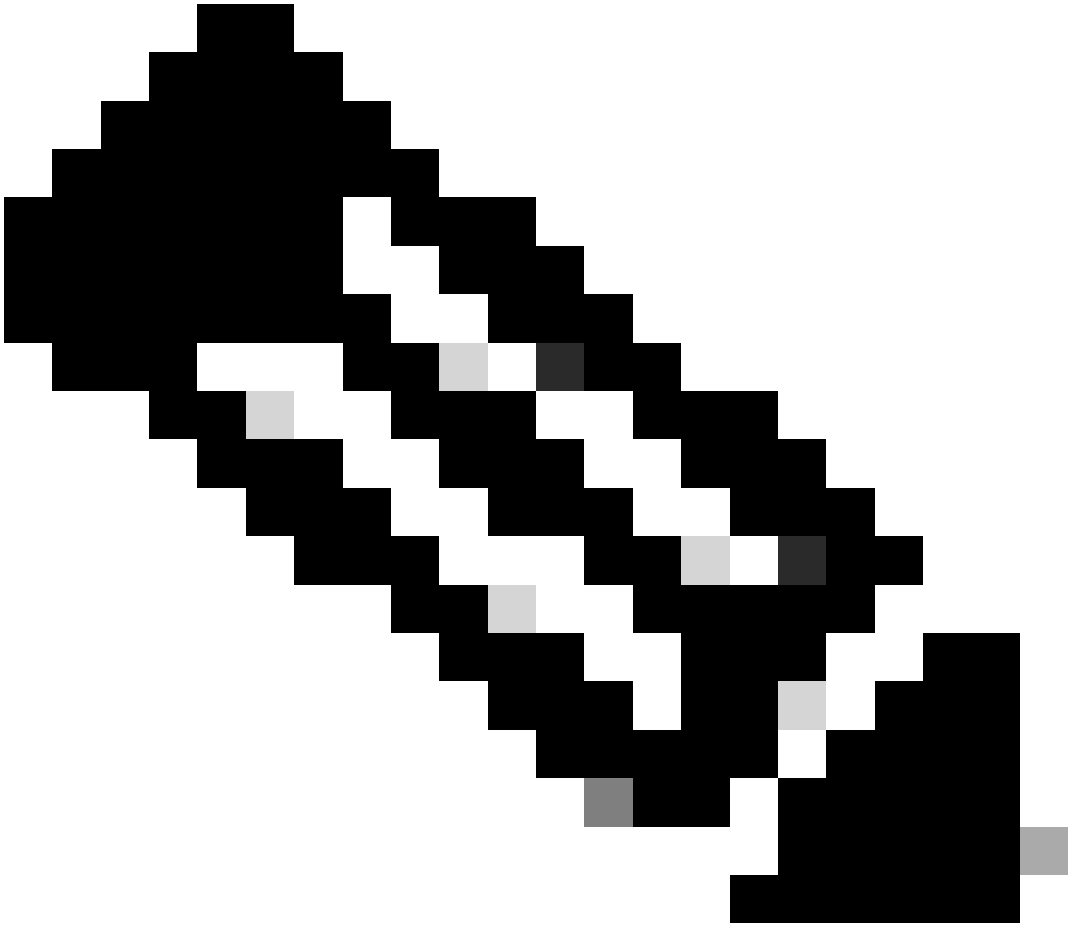


참고: 문제가 있는 설비와 문제가 없는 설비에 대한 코어 파일을 가져오려면 두 가지 옵션을 사용할 수 있습니다. 첫째, 재시작 후 정상으로 돌아간 후 동일한 essmgr의 코어 파일을 수집할 수 있습니다. 또는 다른 정상적인 sessmgr에서 코어 파일을 캡처할 수 있습니다. 이 두 가지 접근 방식은 분석 및 문제 해결에 유용한 정보를 제공합니다.

힙 출력을 수집했으면 Cisco TAC에 문의하여 정확한 힙 소비 테이블을 확인하십시오.

이러한 힙 출력에서 더 많은 메모리를 사용하는 함수를 확인해야 합니다. 이를 바탕으로 TAC은 기능 사용의 목적을 조사하고, 그 사용이 트래픽/트랜잭션 증가 규모와 일치하는지 또는 기타 문제 원인을 확인합니다.

힙 출력은 Memory-CPU-data-sorting-tool로 제공된 링크에서 액세스하는 툴을 사용하여 정렬할 수 있습니다.



참고: 이 툴에서는 여러 가지 다양한 기능을 사용할 수 있습니다. 그러나 힙 출력을 업로드하고 도구를 실행하여 출력을 정렬된 형식으로 가져오는 "힙 소비 테이블"을 선택해야 합니다.

시나리오 2. CPU 사용률이 높기 때문에

1. sessmgr 인스턴스를 다시 시작하거나 종료하기 전에 SSD를 수집합니다.
2. 영향받는 Sessmgr에 대한 코어 덤프를 수집합니다.

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

3. 영향을 받는 동일한 sessmgr/aamgr에 대해 이러한 명령의 힙 출력을 숨김 모드로 수집합니다.

<#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. 다음 명령을 사용하여 sessmgr 작업을 다시 시작합니다.

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

5. Sessmgr 상태를 확인합니다.

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

6. 다른 SSD를 수집합니다.
7. 3단계에서 언급한 모든 CLI 명령의 출력을 수집합니다.

8. 2단계에서 언급한 명령을 사용하여 정상적인 sesmgr 인스턴스에 대한 코어 덤프를 수집합니다.

높은 메모리와 CPU 시나리오를 모두 분석하려면 대량 통계를 검토하여 트래픽 트렌드가 합법적으로 증가했는지 확인하십시오.

또한 Card/CPU level statistics(카드/CPU 레벨 통계)에 대한 bulkstats를 확인합니다.

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.