

Verzamel gegevens wanneer NSO hoge CPU verbruikt

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Te verzamelen gegevens](#)

[Aanvullende informatie](#)

[Gerelateerde informatie](#)

Inleiding

In dit document wordt de gegevensverzameling van Network Services Orchestrator (NSO) beschreven die nodig is wanneer het CPU-verbruik tot 100-150% toeneemt.

Voorwaarden

Vereisten

Er zijn geen specifieke vereisten van toepassing op dit document.

Gebruikte componenten

Dit document is niet beperkt tot specifieke software- en hardware-versies.

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

Achtergrondinformatie

Wanneer meerdere transacties van NB worden verwerkt, stijgt het NSO CPU-verbruik tot ongeveer 100-150% van het normale verbruik. Wanneer dit gebeurt, moet u de oorzaak vinden die de CPU-prestaties verlaagt. Tegelijkertijd reageert de NSO niet correct op RESTCONF-vragen (indien gebruikt).

In dit artikel worden alle belangrijke gegevens beschreven die tijdens het probleem moeten worden verzameld, zodat het probleem correct kan worden opgelost. Verder worden enkele stappen voorgesteld om het probleem op te lossen.

Te verzamelen gegevens

Vanuit Linux-perspectief:

- vlek
- bovenkant
- vrij -h
- vmstat
- cat /proc/meminfo
- Pastree -c
- ps-auxw | soort

Opmerking: U kunt deze gegevens (behalve 'lscpu') op regelmatige tijdstippen vastleggen om te begrijpen hoe het systeem zich gedraagt als de verzoeken van NB komen.

Vanuit het perspectief van de nationale veiligheidsinstantie:

- NCS —status | griepsloot
- Schakel het voortgangsspoor in: admin@ncs(config)# voer droog uit cli{ lokaal knooppunt { voortgang van gegevens { + spoor alle { + bestemming { + file progress-all.txt; + logboek formaat; +} +} } }}admin@ncs(config)# commit
- Leg de volgende informatie elke 'n' seconden vast (dit kan als een script worden uitgevoerd):
seq=0
terwijl ncs —status >& /dev/null; do
NCS —debug-dump ncs.dd.\$((seq++));
ncs —status > ncs.stat.\$(seq++);
slaap 30; #Configured according aan gebruiker
gereed

Vervolgens zijn er een aantal corrigerende maatregelen die ook kunnen worden uitgevoerd om de kwestie te verhelpen:

1. Beperk het aantal sessies als volgt (op dit moment heeft u deze set niet):

```
<sessielimieten>
```

```
<sessielimiet>
```

```
<context>rest</context>
```

```
<max-sessies>100</max-sessies>
```

```
</sessielimiet>
```

```
</sessielimieten>
```

- b. Laat controleregel toe om te verifiëren of het proces NSO door iets is gedood en als in het geval dat het was, registreer het in audit.log:

```
sudo auditctl -a exit,altijd -F arch=b64 -S kill -k audit_kill
```

Om problemen op te lossen en te analyseren, hebt u de vorige details nodig samen met de audit.log, devel.log (bij voorkeur ingesteld op level=trace), ncs-java-vm.log en NB logs.

Aanvullende informatie

V. Hoe behandelt NSO RESTCONF verzoeken van een NB applicatie eigenlijk?

A. Wanneer een noordwaartse toepassing een RESTCONF-verzoek verstuurt, wordt deze behandeld als een unieke transactie op basis van NSO. Dit betekent dat NSO de gehele CDB kan vergrendelen en geen andere transacties kan toestaan totdat de huidige transactie is voltooid. Als dit gebeurt, blijft de transactionele aard van NSO behouden en zorgt het ervoor dat een terugdraaiing kan worden gedaan in geval van problemen.

De NSO commit-wachtrij kan elk volgend transactieverzoek verwerken als het voltooit, en u kunt het transactieslot volgen in devel.log als ze starten/voltoeien. In gebruikgevallen waarin een grote hoeveelheid zoekopdrachten wordt gedaan, introduceert dit een grote hoeveelheid overheadkosten bij NSO; en de transacties zijn in de commit-wachtrij langer dan verwacht. Indien de RESTCONF-verzoeken zouden worden gegroepeerd, zou de doorvoersnelheid toenemen, aangezien de transactieoverheadkosten zouden worden verminderd. Ook zou NSO in staat zijn om binnen één transactie zoveel mogelijk tegelijk te doen. Als een transactie bijvoorbeeld 2 wijzigingen in de apparaatconfiguratie bevat, kan NSO de CDB vergrendelen, beide apparaten tegelijkertijd uitreiken en bewerken, dan de transactie voltoeien. Dit is in tegenstelling tot 2 transacties die elk 1 apparaat bevatten en beide worden gewijzigd; aangezien NSO de CDB voor de eerste transactie kan vergrendelen, het eerste apparaat bewerken, de transactie voltoeien, dan dezelfde stappen voor het tweede apparaat.

Gerelateerde informatie

- [Cisco technische ondersteuning en downloads](#)

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document ([link](#)) te raadplegen.