Dit is om te testen artikel publiceren met Licensing hoe te

Inleiding

Dit document beschrijft de algemene methodologie voor het oplossen van problemen met een trage APIC GUI-ervaring.

Snel starten

Het is vaak gebleken dat langzame APIC GUI problemen het gevolg zijn van een hoog percentage API verzoeken afkomstig van een script, integratie of toepassing. Het access.log van een APIC registreert elk verwerkt API-verzoek. Het access.log van een APIC kan snel worden geanalyseerd met het Access Log Analyzer script binnen het Github Datacenter group aci-tac-scripts project.

Achtergrondinformatie

APIC als webserver - NGINX

NGINX is de DME die verantwoordelijk is voor de API-eindpunten die beschikbaar zijn op elke APIC. Als NGINX niet beschikbaar is, kunnen API-aanvragen niet worden afgehandeld. Als NGINX overbelast is, is de API overbelast. Elk APIC heeft zijn eigen NGINX-proces, dus het is mogelijk dat slechts één APIC NGINX-problemen kan hebben als alleen die APIC wordt gericht door agressieve queriers.

De APIC UI voert meerdere API-verzoeken uit om elke pagina te vullen. Op dezelfde manier zijn alle APIC-show-opdrachten (NXOS Style CLI) wrappers voor python-scripts die meerdere API-verzoeken uitvoeren, de respons afhandelen en vervolgens aan de gebruiker dienen.

Relevante logs

bestandsnaam van logboek	Location (Locatie)	In welke technische ondersteuning zit het	Opmerkingen
access.log	/var/log/dme/log	APIC 3of3	ACI-agnost, geeft 1 regel per API-verzoek
error.log	/var/log/dme/log	APIC 3of3	ACI Agnostic, toont nginx fouten (throttling inbegrepen)

nginx.bin.log	/var/log/dme/log	APIC 3of3	ACI-specifieke logs, DME-transacties
nginx.bin.warnplus.log	/var/log/dme/log	APIC 3of3	ACI Specific bevat logs die de ernst van de waarschuwing+ weergeven

methodologie

initiële trigger isoleren

Wat wordt beïnvloed?

- Welke APIC's worden beïnvloed; één, vele of alle APIC's?
- Waar wordt traagheid gezien; via UI, CLI-opdrachten of beide?
- Welke specifieke UI-pagina's of -opdrachten zijn traag?

Hoe wordt de traagheid ervaren?

- Is dit te zien in meerdere browsers voor een enkele gebruiker?
- Meerdere gebruikers melden traagheid of slechts een enkele / subset van gebruikers?
- Delen de getroffen gebruikers een vergelijkbare geografische locatie of netwerkpad van browser naar APIC?

Wanneer werd de traagheid voor het eerst opgemerkt?

- Is er onlangs een ACI-integratie of -script toegevoegd?
- Is er onlangs een browserextensie ingeschakeld?
- Is er onlangs een wijziging opgetreden in de ACI-configuratie?

Gebruik en gezondheid van NGINX controleren

Access.log-invoerindeling

access.log is een kenmerk van NGINX en is daarom APIC-agnostisch. Elke regel vertegenwoordigt 1 HTTP-verzoek dat de APIC heeft ontvangen. Verwijs naar dit logboek om het NGINX-gebruik van een APIC te begrijpen.

De standaardindeling access.log op ACI versie 5.2+:

```
log_format proxy_ip '$remote_addr ($http_x_real_ip) - $remote_user [$time_local]'
                    '"$request" $status $body_bytes_sent '
                    '"$http_referer" "$http_user_agent"';
```

Deze regel vertegenwoordigt een item access.log wanneer een moquery -c fvTenant wordt

uitgevoerd:

```
127.0.0.1 (-) - - [07/Apr/2022:20:10:59 +0000]"GET /api/class/fvTenant.xml HTTP/1.1" 200 15863 "-" "Pyt
```

Kaart van voorbeeld access.log entry naar log_format:

log_format-veld	Inhoud uit voorbeeld	Opmerkingen
$remote_addr	127.0.0.1	IP van de host die dit verzoek heeft verzonden
$http_x_real_ip	-	IP van laatste aanvrager als proxies in gebruik zijn
$remote_user	-	Niet algemeen gebruikt. Controleer nginx.bin.log om bij te houden welke gebruiker is aangemeld om aanvragen uit te voeren
$time_local	07/apr/2022:20:10:59 +0000	Wanneer het verzoek is verwerkt
$-aanvraag	GA NAAR /api/class/fvTenant.xml HTTP/1.1	Http-methode (GET, POST, DELETE) en URI
$status	200	HTTP-statuscode voor respons
$body_bytes_sent	1586	grootte van de responslast
$http_referer	-	-
$http_user_agent	Python-urllib	Welk type klant heeft het verzoek verzonden

Access.log-gedrag

Aanvraag met hoog tarief barst gedurende een lange periode:

- Voortdurende uitbarstingen van meer dan 40 verzoeken per seconde kunnen UI-traagheid veroorzaken
- Bepaal welke host(s) verantwoordelijk zijn voor de vragen
- Verminder of schakel de bron van vragen uit om te zien of dit de APIC-responstijd verbetert.

Consistente 4xx- of 5xx-antwoorden:

- Als gevonden, identificeer de foutmelding van nginx.bin.log

Het access.log van een APIC kan snel worden geanalyseerd met het Access Log Analyzer script binnen het Github Datacenter group aci-tac-scripts project.

Gebruik NGINX-bronnen controleren

NGINX CPU en geheugengebruik kunnen worden gecontroleerd met de bovenste opdracht van de APIC:

<#root>

```
top - 13:19:47 up 29 days, 2:08, 11 users, load average: 12.24, 11.79, 12.72
Tasks: 785 total, 1 running, 383 sleeping, 0 stopped, 0 zombie
%Cpu(s): 3.5 us, 2.0 sy, 0.0 ni, 94.2 id, 0.1 wa, 0.0 hi, 0.1 si, 0.0 st
KiB Mem : 13141363+total, 50360320 free, 31109680 used, 49943636 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 98279904 avail Mem

PID   USER PR NI VIRT    RES   SHR   S %CPU %MEM TIME+     COMMAND
21495 root 20 0  4393916 3.5g 217624 S
```

2.6

2.8 759:05.78

nginx.bin

Een hoog gebruik van NGINX-bronnen kan direct correleren met een hoog percentage verwerkte verzoeken.

Controleren op cores

Een NGINX-crash is niet typisch voor Slow APIC GUI-problemen. Als er echter NGINX-kernen worden gevonden, bevestig deze dan aan een TAC SR voor analyse. Raadpleeg de ACI Techsupport-gids voor stappen om te controleren op kernen.

Client-naar-server latentie controleren

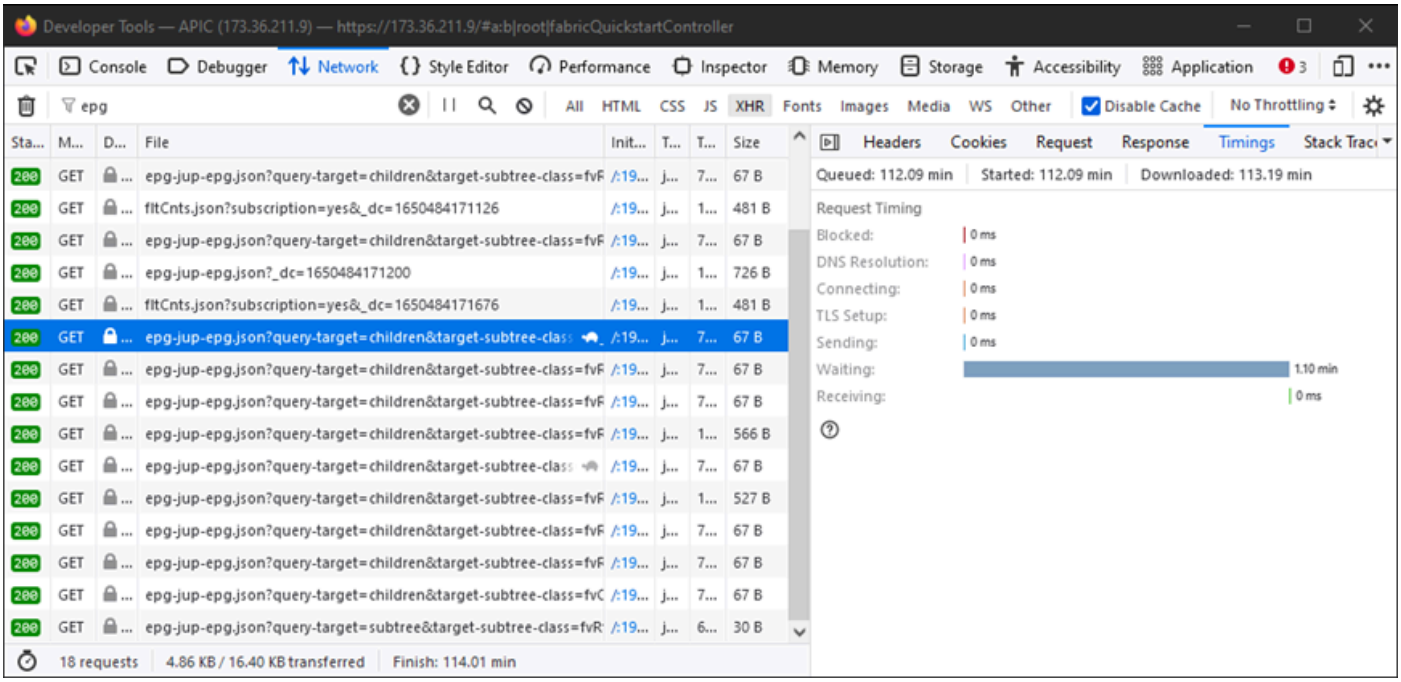
Als snelle verzoeken niet worden gevonden, maar een gebruiker blijft UI traagheid vertonen, kan het probleem client (browser) naar server (APIC) latentie.

In deze scenario's valideert u het gegevenspad van de browser naar de APIC (geografische afstand, VPN, enz.). Implementeer en test indien mogelijk de toegang vanaf een jumpserver in dezelfde geografische regio of hetzelfde datacenter als de APIC's om te isoleren. Valideren als andere gebruikers een vergelijkbare latentie vertonen.

Tabblad Browserontwikkelingshulpmiddelen Netwerk

Alle browsers hebben de mogelijkheid om HTTP-verzoeken en -antwoorden te valideren via de toolkit voor browserontwikkeling, meestal op een tabblad Netwerk.

Deze tool kan worden gebruikt om de hoeveelheid tijd te valideren die het kost voor elke fase van browserverzoeken zoals weergegeven in de afbeelding.



Voorbeeld van de browser die 1,1 minuten wacht totdat de APIC reageert

Verbeteringen voor specifieke UI-pagina's

Pagina Beleidsgroep:

Cisco bug ID CSCvx14621 - APIC GUI laadt langzaam op IPG-beleid op het tabblad Fabric.

Interface onder Voorraadpagina:

Cisco bug ID CSCvx90048 - De eerste lading van het operationele tabblad "Layer 1 Physical Interface Configuration" is lang/induceert 'freeze'.

Algemene aanbevelingen voor client > Serverlatentie

Bepaalde browsers, zoals Firefox, maken standaard meer webverbindingen per host mogelijk.

- Controleer of deze instelling kan worden geconfigureerd in de gebruikte browserversie
- Dit is belangrijker voor pagina's met meerdere query's, zoals de pagina Beleidsgroep

VPN en afstand tot APIC verhogen de algehele traagheid van de gebruikersinterface, gezien verzoeken van de clientbrowser en de reistijd van de APIC-respons. Een springplank die geografisch lokaal is voor de APIC's, reduceert de browser aanzienlijk tot APIC-reistijden.

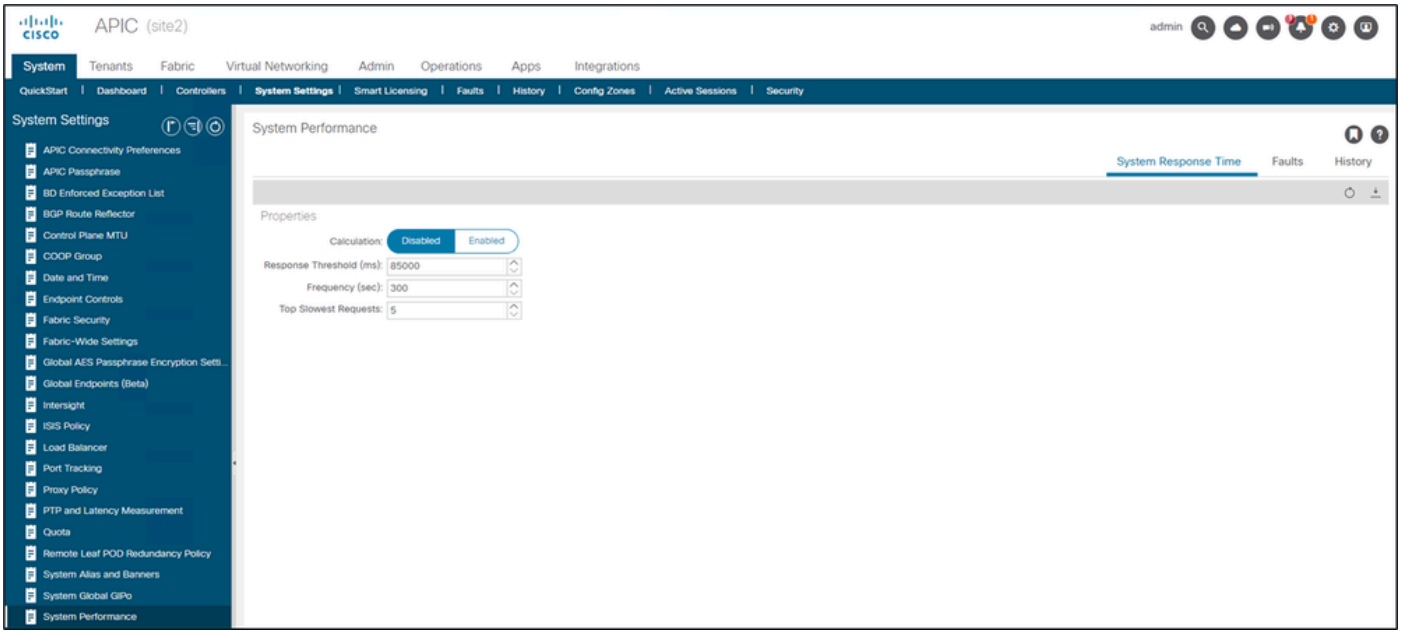
Controleren op lange webverzoeken

Als een webserver (NGINX op APIC) een groot aantal Long-Web-verzoeken verwerkt, kan dit van invloed zijn op de prestaties van andere verzoeken die parallel worden ontvangen.

Dit geldt met name voor systemen met gedistribueerde databases, zoals APIC's. Een enkele API-aanvraag kan extra verzoeken en lookups vereisen die naar andere knooppunten in de fabric worden verzonden, wat kan resulteren in verwachte langere responstijden. Een uitbarsting van deze Long-Web-verzoeken binnen een klein tijdsbestek kan de hoeveelheid benodigde middelen vergroten en leiden tot onverwacht langere responstijden. Bovendien kunnen ontvangen verzoeken dan time-out (90 seconden), wat resulteert in onverwacht systeemgedrag vanuit het perspectief van de gebruiker.

Responstijd van het systeem - Berekening van de responstijd van de server inschakelen

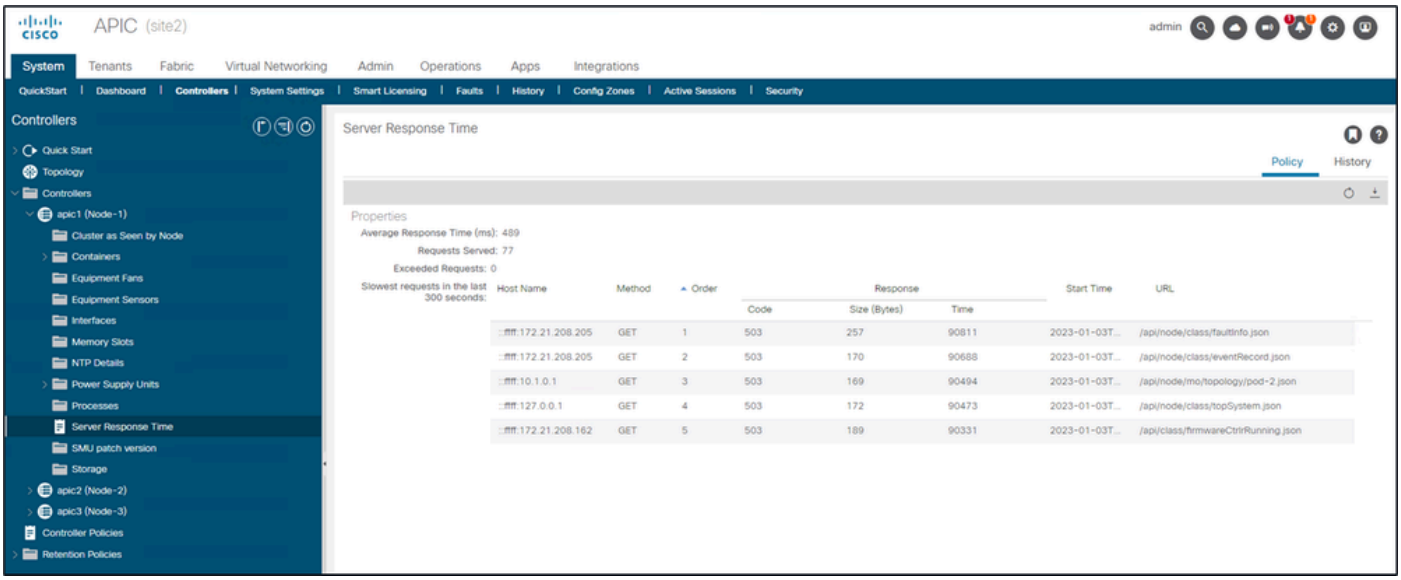
In 4.2(1)+ kan een gebruiker "System Performance Calculation" inschakelen die API-verzoeken bijhoudt en markeert die lang hebben geduurd om te verwerken.



Berekening kan worden ingeschakeld vanuit Systeem - Systeeminstellingen - Systeemprestaties

Zodra "Berekening" is ingeschakeld, kan een gebruiker navigeren naar specifieke APIC's onder

Controllers om de langzaamste API-verzoeken binnen de laatste 300 seconden te bekijken.



Systeem - Controllers - Map Controllers - APIC x - Responstijd van server

Overwegingen voor APIC API-gebruik

Algemene aanwijzingen om ervoor te zorgen dat een script Nginx niet schaadt

- Elke APIC heeft zijn eigen NGINX DME.
  - Alleen APIC 1's NGINX verwerkt verzoeken tot APIC 1. De NGINX van APIC 2 en 3 verwerkt deze verzoeken niet.
- Over het algemeen verzwakken meer dan 40 API-verzoeken per seconde over een lange periode NGINX.
  - Indien gevonden, verminderen de agressiviteit van de verzoeken.
  - Als de host-verzoeken niet kunnen worden gewijzigd, overweeg dan NGINX-tarieflimieten op de APIC.

Inefficiëntie van adresscript

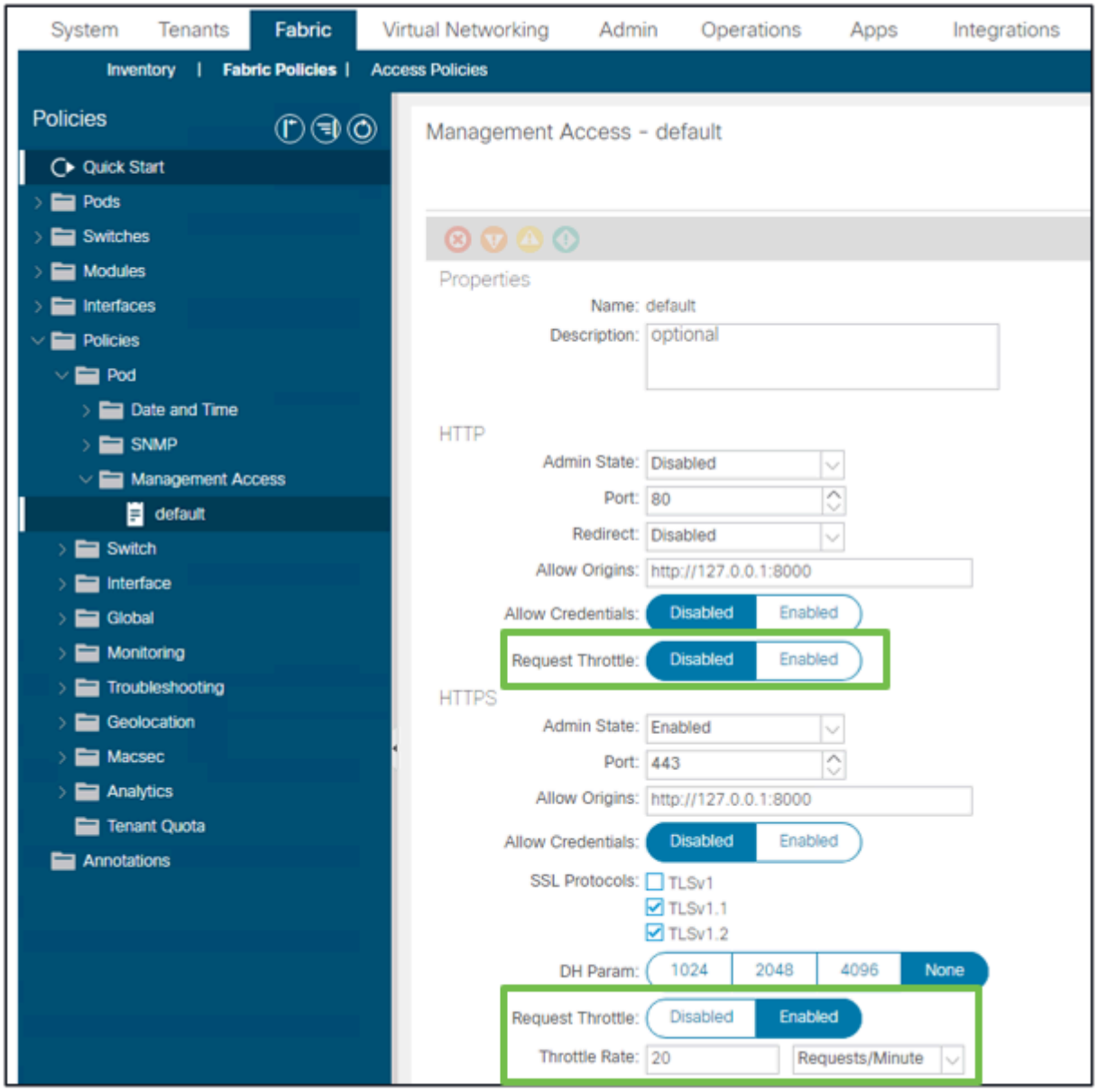
- Log niet in/log uit voor elke API-aanvraag.
  - De standaard time-out voor één inlogsessie is 10 minuten. Dezelfde sessie kan worden gebruikt voor meerdere verzoeken en kan worden vernieuwd om de geldigheidsperiode te verlengen.
  - Zie Cisco APIC REST API Configuration Guide - Access the REST API - Authenticating and Maintaining an API Session (Toegang tot de REST API - Authenticatie en onderhoud van een API-sessie).
- Als uw script veel DN's opvraagt die een bovenliggende query delen, vouwt u de query's samen in één logische bovenliggende query met queryfilters.
  - Zie de configuratiehandleiding voor de API van Cisco APIC REST - REST API-query's samenstellen - Query-scopefilters toepassen.

- Als u updates van een object of klasse van object nodig hebt, overweeg dan websocket-abonnementen in plaats van snelle API-verzoeken.

NGINX Aanvraaggaspedaal

Verkrijgbaar in 4.2(1)+, kan een gebruiker request throttle tegen HTTP en HTTPS onafhankelijk inschakelen.

Opmerking: vanaf ACI versie 6.1(2) werd het ondersteunde maximale tarief voor deze functie verlaagd naar 40 verzoeken per seconde (r/s) of 2400 verzoeken per minuut (r/m) van 10.000 r/m.



Fabric - Verbindingsbeleid - Map met beleidsregels - Map met beheertoegang - Standaard

Wanneer ingeschakeld:

- NGINX wordt opnieuw gestart om wijzigingen in het configuratiebestand toe te passen

    - Een nieuwe zone, httpsClientTagZone, wordt naar nginx config geschreven
- De gaspedaal kan worden ingesteld in Verzoeken per minuut (r/m) of Verzoeken per seconde (r/s).
- Request Throttle is gebaseerd op de snelheidslimietimplementatie die is opgenomen in NGINX
    - API-verzoeken tegen de API/api/URI gebruiken de door de gebruiker gedefinieerde Throttle Rate + burst= (Throttle Rate x 2) + nodelay
        - Er is een niet-configureerbaar gaspedaal (zone aaaApiHttps) voor api/aaaLogin en api/aaaRefresh die snelheidslimieten bij 2r/s + burst=4 + nodelay
    - Request Throttle wordt bijgehouden per client-ip-adres
    - API-verzoeken afkomstig van de APIC self-ip (UI + CLI) omzeilen het gaspedaal
    - Elk IP-adres van de client dat de door de gebruiker gedefinieerde throttle rate + burst-drempel overschrijdt, ontvangt een 503-respons van de APIC
    - Deze 503's kunnen worden gecorreleerd binnen de toegangslogboeken
    - error.log bevat vermeldingen die aangeven wanneer throttling is geactiveerd (zone httpsClientTagZone) en tegen welke clienthosts

<#root>

```
apic#
```

`less /var/log/dme/log/error.log`

```
...
2023/04/17 20:19:14 [error] ...
```

`limiting requests`

```
, excess: 40.292 by zone "
```

`httpsClientTagZone`

```
", client: h.o.s.t, ... request: "GET /api/class/...", host: "a.p.i.c"
2023/04/17 20:19:14 [error] ...
```

`limiting requests`

```
, excess: 40.292 by zone "
```

`httpsClientTagZone`

```
", client: h.o.s.t, ... request: "GET /api/node/...", host: "a.p.i.c"
```

Als algemene regel geldt dat Request Throttle alleen dient om de server (APIC) te beschermen tegen DDOS-achtige symptomen die worden veroorzaakt door query-agressieve clients. Begrijpen en isoleren van de vraag-agressieve client voor definitieve oplossingen in de app / script logica.

Aanbevelingen

Deze aanbevelingen zijn bedoeld om de belasting en operationele stress op de APIC te helpen verminderen, met name in scenario's waarin geen enkele bron verantwoordelijk is voor een groot

aantal API-oproepen. Door deze best practices te implementeren, kunt u onnodige verwerking, logging en het genereren van gebeurtenissen in uw fabric minimaliseren, wat resulteert in verbeterde systeemstabiliteit en -prestaties. Deze suggesties zijn vooral relevant in omgevingen waar gezamenlijk gedrag in plaats van geïsoleerde incidenten bijdragen aan APIC-spanning.

ACL-logboekregistratie uitschakelen

Zorg ervoor dat de ACL-logboekregistratie is uitgeschakeld tijdens normale bewerkingen. Schakel het alleen in tijdens geplande onderhoudsvensters voor probleemoplossing of foutopsporing. Continue logging kan leiden tot overmatige informatieberichten, vooral bij grote verkeersdalingen over meerdere switches, waardoor de APIC-werklast toeneemt.

Raadpleeg voor meer informatie de Cisco APIC Security Configuration Guide (link naar 5.2.x-handleiding):
https://www.cisco.com/c/en/us/td/docs/dcn/aci/apic/5x/security-configuration/cisco-apic-security-configuration-guide-release-52x/security-policies-52x.html

Syslog-conversie beperken tot kritieke gebeurtenissen

Configureer het systeem zodat alleen syslog-berichten met ALERT-ernst worden geconverteerd naar eventRecords. Vermijd het converteren van INFORMATION-niveau (waaronder ACL.logging) om te voorkomen dat lawaaierige gebeurtenissen de APIC overweldigen:

1. Navigeer naar Fabric → Fabric Policies → Policies → Monitoring → Common Policy → Syslog Message Policies → Standaard.
2. Pas het faciliteitsfilter aan om de syslog-ernst in te stellen op alert.

Niet-essentiële gebeurteniscodes opheffen

Onderdruk (squelch)-gebeurteniscodes die niet relevant zijn voor uw bewakingsbehoeften om ruis te verminderen.
Als u gebeurteniscode E4204939 wilt opheffen, gebruikt u deze opdracht op een APIC-CLI:

```
bash
icurl -k -sX POST -d '<fabricInst><monCommonPol><eventSevAsnP code="E4204939" sev="squelched"/></monCom
```

Zo verifieert u:

```
bash
icurl -k -sX GET 'https://localhost/api/node/class/eventSevAsnP.xml' | xmllint --format -
```

U kunt ook controleren via de gebruikersinterface:
Fabric > Verbindingsbeleid > Beleid > Controle > Gemeenschappelijk beleid > Toewijzingsbeleid voor ernst van gebeurtenissen

Abonnement op ND optimaliseren Vernieuwt

Voor verbindingen die worden beheerd door ND-versies die ouder zijn dan 3,2,2 m of 4,1,1 g, moet u upgraden naar een van deze versies of later om de vernieuwingsintervallen van het abonnement te optimaliseren. Eerdere versies worden elke 45 seconden per MO vernieuwd, wat op schaal kan resulteren in meer dan 300.000 APIC-verzoeken per dag. Bijgewerkte versies verhogen de time-out van het abonnement tot 3600 seconden (1 uur), waardoor het aantal vernieuwingen wordt teruggebracht tot ongeveer 5000 per dag.

Intervisiegerelateerde zoekopdrachten bewaken

Dankzij intersight kunnen verbindingen periodieke topsysteemquery's genereren vanuit de DC-connector (elke 15 seconden), wat bijdraagt aan de APIC-belasting.
In versie 6.1.2 en later is deze query geoptimaliseerd voor minder overhead.

Retentiebeleid voor records afstemmen

Stel het retentiebeleid voor eventRecord, faultRecord en healthRecord in op 1.000 om overmatige accumulatie van records te voorkomen. Dit is vooral handig wanneer u deze records regelmatig extraheert voor een specifieke operationele activiteit. Beoordeel altijd de impact van het verminderen van de granulariteit van de monitoring op basis van uw operationele en probleemoplossende vereisten.

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.