Configuratie van Inx omgekeerde proxy voor VPN-minder toegang tot Cisco Finesse 12.6 ES 02

Inhoud

Inleiding

Dit document beschrijft hoe u een omgekeerde proxy kunt gebruiken om toegang te krijgen tot het Cisco Finse bureaublad zonder verbinding te maken met een VPN op basis van 12.6 ES02-versies van Cisco Finesse, Cisco Unified Intelligence Center (CUIC) en Cisco Identity Service (IDS).

> **Opmerking:** De installatie en configuratie van NGINX worden niet ondersteund door Cisco. Vragen over dit onderwerp kunnen worden besproken op de Cisco gemeenschapsforums.

> **Opmerking:** Voor ES02-implementaties van VPN-Minder, zie de releaseopmerkingen van de individuele componenten om de upgrades te plannen en de compatibiliteitsbeperkingen te controleren.

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- Cisco Unified Contact Center Enterprise (UCCE) release
- Cisco Finesse
- Linux-administratie
- Netwerkbeheer en Linux-netwerkbeheer

Gebruikte componenten

De informatie in dit document is gebaseerd op de volgende software- en hardware-versies:

- Finesse - 12.6 ES02
- CUIC - 12.6 ES02
- IDs - 12.6 ES02
- UCCE/Hosted Collaboration Solutions (HCS) voor contactcenters (CC) - 11.6 of hoger
- Packaged Contact Center Enterprise (PCCE) - 12.0 of hoger

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 levend is, zorg er dan voor dat u de mogelijke impact van om het even welke opdracht begrijpt.

Opmerking: De configuratie die in dit document is meegeleverd, is geconfigureerd, geharde en belastingsgetest met NEGINX reverse proxy (OpenResty) die is ingezet op CentOS 8.0, tegen een proefversie van 2000 UCCE-inzet van gebruikers. De referentie-informatie over het prestatieprofiel is in dit document beschikbaar.

Achtergrondinformatie

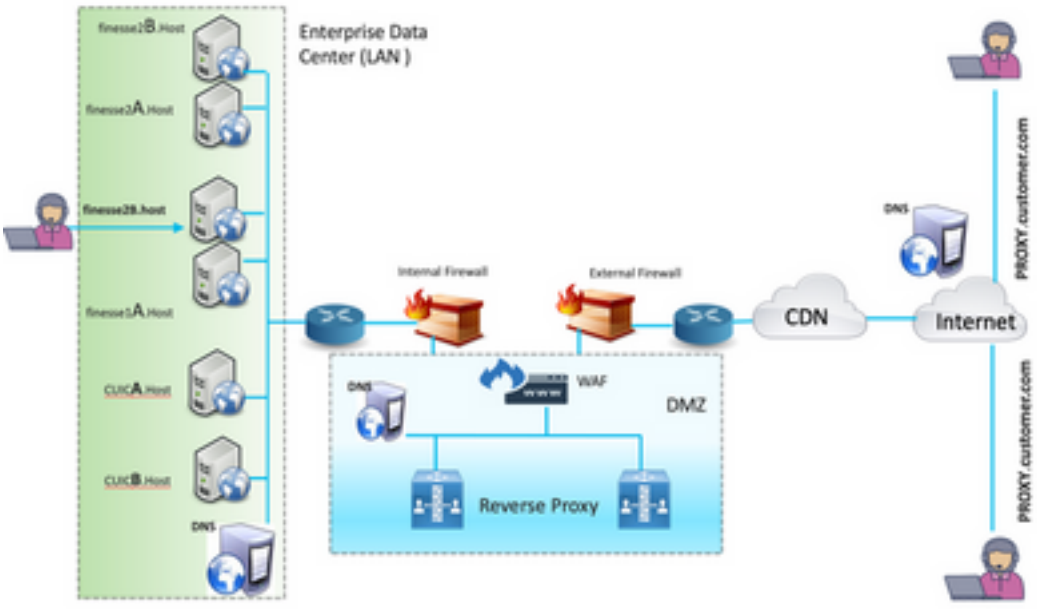
Dit inzetmodel wordt ondersteund voor UCCE/PCCE en HCS voor UCCE-oplossingen.

Plaatsing van een omgekeerde proxy wordt ondersteund (beschikbaar via 12.6 ES01) als een optie om toegang te krijgen tot het Cisco Finse bureaublad zonder verbinding te maken met een VPN. Deze functie biedt de flexibiliteit voor agenten om het bureaublad van Finse overal via het internet te bereiken.

Om deze optie in te schakelen, moet een omgekeerd proxy-paar worden ingezet in de gedemilitariseerde zone (DMZ).

De toegang tot de media blijft onveranderd in omgekeerde proxy-implementaties. Om aan de media te verbinden kunnen agenten Cisco Jabber via Mobile en Remote Access Solutions (MRA) of de Mobile Agent-mogelijkheid van UCCE gebruiken met een PSTN-netwerk (Public Switched Telephone Network) of mobiel eindpunt. Dit diagram toont hoe de netwerkplaatsing eruit zal zien wanneer u tot twee clusters van Finse en twee CUIC knopen door één enkel hoge beschikbaarheid (HA) paar van omgekeerde volmachtknopen toegang hebt.

Gelijktijdige toegang van agents op het internet en agents die via het LAN verbonden zijn, wordt ondersteund zoals in deze afbeelding.



Opmerking: Zie de functiehandleiding voor de selectiecriteria voor volmacht van derden in plaats van Nginx om deze plaatsing te ondersteunen.

- UCCE 12.6 functiegids - Hier vindt u een overzicht van de functies, ontwerp en configuratiegegevens voor de VPN-Minder functie.
- UCCE 12.6 Security Guide - biedt veiligheidsconfiguratievoorschriften voor de reverse proxy-toepassing.

Het wordt aanbevolen het VPN-Minder gedeelte van de functiehandleiding en de beveiligingsgids te bekijken voordat u dit document leest.

Wijzigingen in ES02

- Nieuwe functies

  Finse supervisor mogelijkheden worden nu ondersteund via omgekeerde proxy.CUIC RealTime en Historische rapporten worden nu ondersteund via Finesse gadgets in een vooraf gedefinieerde omgeving.

  Verificatie voor alle verzoeken/communicatie - hiervoor is Lua-ondersteuning nodig
  Alle verzoeken om Fins / CUIC / IM & Presence (IM&P) zijn bij de proxy geauthentificeerd voordat u het datacenter kunt invoeren.Websocket- en Live-gegevenssocketIO-verbindingen zijn ook beperkt en alleen toegestaan van klanten die met succes een beveiligd verzoek aan Finesse hebben ingediend.Brute force aanval met gevoel en loggen bij de proxy, die kan worden gebruikt met Fail2Ban om kwaadaardige IP-adressen te blokkeren.

- Verbeteringen in beveiliging tegen omgekeerde proxy-configuratie - vereist Lua-ondersteuning Wederzijdse beveiliging van transportlaag (TLS) tussen omgekeerde proxy en upstream componenten (Finesse/IDS/CUIC/Livedata).SeLinux-instellingen.Schakel Secure Socket Layer (SSL) vertrouwensverificatie in voor proxy- en componentserververzoeken.

- Verbeterde beveiliging van de proxy-configuratie om aanvallen van Denial-of-Service (DoS)/Distributed Denial-of-Service (DDoS) te voorkomen
  Uitgebreide NGINX-aanvraagtarieven voor verschillende delen van het systeem.Snelheidsbeperkingen voor IP-tabellen.Verificatie van statische bronverzoeken alvorens de upstream component server aan te vragen.Lichtbare en niet-echt bevonden pagina's die niet op de upstream component server terechtkomen.

- Diverse andere functies - vereist Lua-ondersteuning Auto-sensing Cross-Origin Resource Sharing (CORS)-reacties van de proxy om automatische configuratie te ondersteunen en de prestaties te verbeteren.

Upgradeopmerkingen voor ES01-gebaseerde VPN-minder configuraties

- Voor de configuratie van ES02 moet NGINX met Lua-ondersteuning worden geïnstalleerd.
- certificaatvereisten Cisco Finesse, CUIC en IDS vereisen dat het NGINX / OpenResty host-certificaat wordt toegevoegd aan de Tomcat-trustwinkel en dat deze opnieuw wordt gestart voordat de configuratie van NGINX ES02 met succes kan worden aangesloten op de upstream server.De certificaten van de servers van Cisco Finesse, CUIC, en IDS upstream moeten in de NGINX server worden geconfigureerd om de op ES02 gebaseerde configuratie te gebruiken.

  Opmerking: Het wordt aanbevolen de bestaande op ES01 gebaseerde NGEX-configuratie te verwijderen voordat u de ES02 NGINX-configuraties installeert.

Opmerking: ES02 configuratiescripts moeten ook de corresponderende ES02 COP-installatie in Cisco Finesse, CUIC en IDS hebben.

Verificatie

Finesse 12.6 ES02 introduceert authenticatie bij de volmacht. Verificatie wordt ondersteund voor Single aanmelding (SSO) en niet-SSO-implementaties.

Verificatie wordt afgedwongen voor alle verzoeken en protocollen die bij de proxy zijn geaccepteerd voordat ze worden doorgestuurd naar de upstream component servers, waar de verificatie die door de component servers wordt afgedwongen ook lokaal plaatsvindt. Alle authenticatie gebruikt de gemeenschappelijke Finse inlogreferenties om de verzoeken voor te echt te maken.

Persistente verbindingen, zoals websoires die afhankelijk zijn van toepassingsprotocollen zoals Extensible Messaging and Presence Protocol (XMPP) voor authenticatie en postverbinding, worden bij de proxy geauthenticeerd door het IP-adres te valideren waarna een succesvolle applicatie-verificatie is uitgevoerd voordat de socket verbinding tot stand is gebracht.

Niet-SSO-verificatie

Niet-SSO-verificatie vereist geen extra configuraties en zal werken met de NGINX-configuratiescripts nadat de gewenste script vervangen is. Verificatie is afhankelijk van de naam van de gebruiker en het wachtwoord dat wordt gebruikt om in te loggen op Voltooien. Toegang tot alle eindpunten zal worden gevalideerd door de Eenheid van Finse.

De lijst van geldige gebruikers wordt plaatselijk bij de volmacht gecached (werkt het cache elke 15 minuten bij), die wordt gebruikt om de gebruiker in een verzoek te valideren. Gebruikerscredentials valideren door het verzoek naar de geconfigureerde Finesse URI te sturen en vervolgens wordt de geloofsbrieven lokaal gecached (15 minuten gecached) om nieuwe verzoeken lokaal te authentiseren. Als de gebruikersnaam of het wachtwoord wordt gewijzigd, wordt deze pas na 15 minuten van kracht.

SSO-verificatie

Voor verificatie van SSO moet de beheerder de coderingssleutel van het IDS-token configureren op de server Nginx in het configuratiebestand. De sleutel voor de codering van het IDS-teken kan worden verkregen op de server IDs met de **show ids secret** CLI-opdracht. Ze moeten worden ingesteld als onderdeel van een van de vervangingen van de #must-change die de beheerder in de scripts moet uitvoeren voordat de SSO-autoriteit kan werken.

Raadpleeg de gebruikershandleiding van SSO voor de IDS SAML-configuraties die moeten worden uitgevoerd voor de proxy-resolutie om voor IDS te werken.

Zodra de SSO-verificatie is ingesteld kan een geldig paar penningen worden gebruikt om toegang te krijgen tot een van de eindpunten in het systeem. De proxy-configuratie bevestigt de aanmeldingsgegevens door de token-aanvragen van IDs te onderscheppen of door geldige penningen te decrypteren en deze vervolgens lokaal te casten voor verdere validaties.

Verificatie voor Websocket verbindingen

Websocket verbindingen kunnen niet worden geauthentiseerd met de standaard autorisatie header, omdat aangepaste headers niet worden ondersteund door native websocket implementaties in de browser. Verificatieprotocollen op toepassingsniveau, waarbij de verificatieinformatie in de lading de oprichting van een websocket niet verhindert, en dus kunnen kwaadaardige entiteiten DOS- of DDOS-aanvallen maken door talloze verbindingen te maken om het systeem te overweldigen.

Om deze mogelijkheid te beperken, beschikken de nginx omgekeerde proxy-configuraties over specifieke controles die het mogelijk maken dat websocket-verbindingen alleen worden geaccepteerd van die IP-adressen die met succes een geauthentiseerd REST-verzoek hebben ingediend voordat de websocket-verbinding is opgezet. Dit betekent dat klanten die proberen een websocket verbinding te creëren voordat een REST-aanvraag wordt ingediend, nu een fout in de vergunning krijgen en geen ondersteund gebruiksscenario is.

Strijdkrachten tegen aanvallen

Finesse 12.6 ES02 authenticatie scripts verhinderen actief brute force aanvallen die kunnen worden gebruikt om het gebruikerswachtwoord te raden. Dit doet u door het IP-adres te blokkeren dat gebruikt wordt om toegang te krijgen tot de service, na een bepaald aantal mislukte pogingen in een korte tijd. Deze verzoeken worden verworpen door een **fout van 418 klanten**. De details van de geblokkeerde IP-adressen kunnen worden geraadpleegd in de bestanden **<nginx-install-folder>/logs/blocking.log** en **<nginx-install-folder>/logs/error.log.**

Het aantal mislukte verzoeken, het tijdsinterval en de blokkeringsduur zijn Configureerbaar. Configuraties zijn aanwezig in het **<nginx-install-folder>/conf/conf.d/maps.conf** bestand.

```
## These two constants indicate five auth failures from a client can be allowed in thirty
seconds.
## if  the threshold is crossed,client ip will be blocked.
map $host $auth_failure_threshold_for_lock {
    ## Must-change Replace below two parameters as per requirement
    default 5 ;
}

map $host $auth_failure_counting_window_secs {
    ## Must-change Replace below two parameters as per requirement
    default 30;
}

## This indicates duration of blocking a client to avoid brute force attack
map $host $ip_blocking_duration {
    ## Must-change Replace below parameter as per requirement
    default 1800;
}
```

Vastlegging

```
grep -r "IP is already blocked." error.log
========================================

2021/10/29 19:21:00 [error] 943068#943068: *43 [lua] block_unauthorized_users.lua:53:
10.70.235.30 :: IP is already blocked..., client: 10.70.235.30, server: saproxy.cisco.com,
request: "GET /finesse/api/SystemInfo?nocache=1635591686497 HTTP/2.0", host:
"saproxy.cisco.com:8445", referrer:
"https://saproxy.cisco.com:8445/desktop/container/?locale=en_US"
```

```
tail -f blocking.log
====================

2021/10/29 17:30:59 [error] 939738#939738: *1857 [lua] content_by_lua(rest_cache:189 2:
[10.70.235.30] will be blocked for [ 30 minutes ] for exceeding retry limit., client:
10.70.235.30, server: saproxy.cisco.com, request: "GET /finesse/api/SystemInfo HTTP/1.1", host:
"saproxy.cisco.com:8445"
```

Aanbevolen wordt om klanten te integreren met Fail2Ban of vergelijkbaar om het verbod aan de IPtable/firewallregels toe te voegen.

Valideren van statische URL's

Alle geldige eindpunten die op een niet-geauthenticeerde manier toegankelijk zijn, worden actief bijgehouden in de ES02 scripts.

Verzoeken naar deze niet-geauthentiseerde paden worden actief verworpen, als een ongeldige URI wordt gevraagd, zonder deze verzoeken naar de upstream server te verzenden.

cache van CORS-koppen

Wanneer het eerste optieverzoek succesvol is, **worden** de **responskop-**headers **access-control-allow-headers, access-control-allow-Originator**, **access-control-allow-methods, access-control-pose-headers**, en **access-control-allow-geloofsbrieven** bij de proxy gedurende vijf minuten gecached. Deze kopregels worden gecached voor elke respectievelijke upstream server.

Configureren

Dit document beschrijft de configuratie van Neginx als de omgekeerde proxy die moet worden gebruikt om FineReader VPN-Minder toegang in te schakelen. De UCCE oplossing component, proxy en OS versies die worden gebruikt om de verstrekte instructies te controleren zijn geleverd. De desbetreffende instructies moeten worden aangepast aan het besturingssysteem/de proxy van uw keuze.

- Gebruikte NGINX-versie - OpenResty 1.19.9.1
- Gebruikt besturingssysteem voor configuratie - CentOS 8.0

  Opmerking: De beschreven configuratie van NGINX kan worden gedownload van de downloadpagina van Finse release 12.6(1)ES2-software.

OpenReader installeren als omgekeerde proxy in DMZ

In deze sectie worden de installatiestappen van de OpenResty-gebaseerde proxy beschreven. De omgekeerde proxy is doorgaans geconfigureerd als een specifiek apparaat in de gedemilitariseerde zone van het netwerk (DMZ), zoals in het implementatieschema dat eerder is vermeld.

1. Installeer het **besturingssysteem van uw keuze** met de vereiste hardwarespecificatie. Kernel en IPv4 parameter-tweaks kunnen afhankelijk van het geselecteerde besturingssysteem verschillen en gebruikers worden geadviseerd deze aspecten opnieuw te controleren als de gekozen OS-versie anders is.

2. Configuratie van twee netwerkinterfaces. Voor de toegang van het publiek van de internetklanten is één interface vereist en een andere om met de servers in het interne netwerk te communiceren.
3. Installeer OpenResty.

Alle vlammen van Nginx kunnen voor dit doel worden gebruikt, zolang ze gebaseerd zijn op Nginx 1.19+ en Lua ondersteunen:

- Neginx Plus
- Nginx Open Source (NGinx open source moet samen met OpenResty-gebaseerde Lua-modules worden gecompileerd zodat deze kan worden gebruikt)
- OpenResty
- GetPageSpeed-extraheren

Opmerking: De configuratie die is meegeleverd, is getest met OpenResty 1.19 en zal naar verwachting met andere distributies werken met slechts kleine updates, indien aanwezig.

Installatie openen

1. Installeer OpenResty. Zie OpenResty Linux-pakketten. Als onderdeel van de installatie OpenResty wordt NGINX op deze locatie geïnstalleerd en voegt u het OpenResty-pad toe aan de ***PATH***-variabele door in het bestand ***~/.bashrc*** toe te voegen.
   ```
   export PATH=/usr/local/openresty/bin:$PATH
   ```
2. Begin / stop Nginx. Voer om Neginx te starten **openresty**.Voer de volgende handelingen in om Nginx te stoppen **openresty -s stop**.

Neginx configureren

De configuratie wordt uitgelegd voor een op OpenResty gebaseerde installatie. De standaardinstellingen voor OpenResty zijn:

- <nginx-install-folder> = /usr/local/openingsresty/nginx
- <Open-installatie-directory> = /usr/lokaal/openlijk

1. Download en haal het bestand uit de Finesse release 12.6(1)ES2 software download pagina (12.6-ES02-reverse-proxy-Configuratie.zip) die de omgekeerde proxy-configuratie voor Nginx bevat.
2. Kopieer nginx.conf, nginx/conf.d, en nginx/html/ van de geëxtraheerde map voor omgekeerde proxy-configuratie naar **<nginx-install-folder>/conf**, **<nginx-install-folder>/conf/conf.d/** en **<nginx-install-folder>/html/** respectievelijk.
3. Kopieer de **nginx/lua** folder van de afgeleide omgekeerde proxy configuratie folder binnen de **<nginx-install-folder>.**
4. Kopieer de inhoud van **lualib** naar **<Open-resty-install-folder>/lualib/resty**.
5. Configureer logrotatie door het **nginx/logrotate/**proxy-bestand te kopiëren naar de **<nginx-install-folder>/LOgrotate/**map. Wijzig de inhoud van het bestand in de juiste logbestanden als de standaardinstellingen van Nginx niet worden gebruikt.
6. Nginx moet worden uitgevoerd met een speciale niet-bevoorrechte servicekening, die moet worden vergrendeld en een ongeldig shell moet hebben (of zoals van toepassing voor het gekozen OS).

7. Vind de "**must-change**" string in de bestanden onder de geëxtraheerde mappen met de naam **html** en **conf.d** en vervang de aangegeven waarden door de juiste items.
8. Zorg ervoor dat alle verplichte vervangingen gedaan worden, die met de **moet-veranderen** commentaren in de configuratiebestanden worden beschreven.
9. Zorg ervoor dat de cachedirectories die voor CUIC en Finesse zijn ingesteld, worden aangemaakt onder **<nginx-install-folder>/cache** samen met deze tijdelijke bestanden. <nginx-install-folder>/cache/client_temp<nginx-install-folder>/cache/proxy_temp

Opmerking: De aangeboden configuratie is bedoeld voor een proefinstallatie in 2000 en moet op passende wijze worden uitgebreid voor een grotere inzet.

De NGINX-cache configureren

Standaard worden de proxy cache paden opgeslagen in het bestandsysteem. We raden aan om ze te veranderen in in in geheugen schijven door een cache locatie te creëren zoals hier wordt getoond.

1. Maken gidsen voor de verschillende volmacht cache paden onder /home. Als voorbeeld, moeten deze gidsen worden gemaakt voor de primaire Finse. Voor de secundaire Finesse- en CUIC-servers moeten dezelfde stappen worden gevolgd.
```
mkdir -p /home/primaryFinesse/rest
mkdir -p /home/primaryFinesse/desktop
mkdir -p /home/primaryFinesse/shindig
mkdir -p /home/primaryFinesse/openfire
mkdir -p /home/primaryCUIC/cuic
mkdir -p /home/primaryCUIC/cuicdoc
mkdir -p /home/client_temp
mkdir -p /home/proxy_temp
echo "tmpfs /home/primaryFinesse/rest tmpfs
size=1510M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/desktop tmpfs
size=20M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/shindig tmpfs
size=500M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/openfire tmpfs
size=10M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryCUIC/cuic tmpfs
size=100M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryCUIC/cuicdoc tmpfs
size=100M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/client_temp tmpfs
size=2048M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/proxy_temp tmpfs
size=2048M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab
```
Opmerking: Vergroot de client en proxy_temp caches met 1 GB voor elk nieuw Finse cluster dat aan de configuratie is toegevoegd.
2. Monteer de nieuwe steunpunten met de opdracht **mount -av**.
3. Bevestig dat het bestandssysteem de nieuwe steunpunten met de **df -h** uit.
4. Verander de proxy_cache_path locaties in de Finesse en CUIC cache configuratiebestanden. Bijvoorbeeld, om de paden voor de Finse primaire taak te veranderen, ga naar **<nginx-install-folder>conf/conf.d/finesse/caches** en verander de bestaande cache locatie**/usr/local/openresty/nginx/cache/finesse25/**naar de nieuwe bestandsindeling **/home/primaire finesse.**
```
##Must-change /usr/local/openresty/nginx/cache/ location would change depending on folder
extraction ##
Nginx config file to cache the desktop/shindig and notification service related static
```

```
files.
proxy_cache_path /home/primaryFinesse/desktop levels=1:2 use_temp_path=on
keys_zone=desktop_cache_primary:10m max_size=15m
inactive=3y use_temp_path=off; proxy_cache_path /home/primaryFinesse/shindig levels=1:2
use_temp_path=on keys_zone=shindig_cache_primary:10m
max_size=500m inactive=3y use_temp_path=off; proxy_cache_path /home/primaryFinesse/openfire
levels=1:2 use_temp_path=on
keys_zone=openfire_cache_primary:10m max_size=10m inactive=3y use_temp_path=off;
proxy_cache_path /home/primaryFinesse/rest
levels=1:2 use_temp_path=on keys_zone=rest_cache:10m max_size=1500m inactive=40m
use_temp_path=off;
```

5. Volg dezelfde stappen voor de Finse secondaire en CUIC servers.

Opmerking: Zorg ervoor dat de som van alle TCP-schijfinstellingen die in alle vorige stappen zijn gemaakt, is toegevoegd aan de definitieve geheugengrootte voor de implementatie, aangezien deze schijven zijn geconfigureerd om als schijven in de toepassing te zien en zoveel geheugenruimte verbruiken.

SSL-certificaten configureren

Gebruik zelfgetekende certificaten - testimplementaties

Zelfondertekende certificaten mogen alleen worden gebruikt totdat de omgekeerde volmacht klaar is om in productie te worden genomen. Gebruik bij een productie-installatie alleen een door de certificeringsinstantie (CA) ondertekend certificaat.

1. Nginx-certificaten genereren voor inhoud van SSL-map. Voordat u certificaten genereert, moet u een map maken die **ssl** (**ssl**) wordt genoemd onder **USR/Local/openresty/nginx**. Aangezien u twee hostnamen (dezelfde proxy server) gebruikt om toegang te krijgen tot Finesse knooppunt1 en Finesse knooppunt2, moet u twee certificaten genereren met de hulp van deze opdrachten (een voor **<reproxy_primaire_fqdn>** en een ander voor **<reseproxy_secondair_fqdn>**). **sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /usr/local/openresty/nginx/ssl/nginx.key -out /usr/local/openresty/nginx/ssl/nginx.crt** (Geef host name op als: <reversiproxy_primair_fqdn>)**sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /usr/local/openresty/nginx/ssl/nginxnode2.key -out /usr/local/openresty/nginx/ssl/nginxnode2.crt** (Geef host name op als :<reseproxy_secondair_fqdn>)Zorg ervoor dat het certificeringspad **etc/nginx/ssl/nginx.crt** en **/usr/local/openresty/nginx/ssl/nginxnode2.crt** is, aangezien deze al zijn ingesteld in Finse NGinx-configuratiebestanden.
2. Verander de toestemming van de particuliere sleutel **400 (r—)**.
3. Configureer de firewall/iptafels bij omgekeerde proxy om de communicatie van de firewall aan te passen aan de poorten waaraan de NGINX-server is geconfigureerd om te luisteren.
4. Voeg het IP adres en hostname van Finesse, IDS en CUIC toe onder de ingang **/etc/hosts** op de omgekeerde proxy server.
5. Raadpleeg de handleiding voor oplossingsfuncties voor de configuraties die op de component servers moeten worden uitgevoerd om de Nginx host als een omgekeerde proxy te configureren.

Opmerking: De aangeboden configuratie is bedoeld voor een proefinstallatie in 2000 en moet op passende wijze worden uitgebreid voor een grotere inzet.

Gebruik CA-ondertekend certificaat - productie-implementaties

Een CA-ondertekend certificaat kan met deze stappen op de omgekeerde proxy worden geïnstalleerd:

1. Generate the certificaatsignaalaanvraag (CSR). Om de CSR en privé sleutel te genereren, moet u **openssl req -new -newkey rsa:4096 -keyout nginx.key -out nginx.csr** nadat u inlogt bij de proxy. Volg de prompt en geef de details op. Dit genereert de CSR (nginx.csr in het voorbeeld) en de private RSA-toets (nginx.key in het voorbeeld) van kracht 4096 bits.Bijvoorbeeld:

```
[root@reverseproxyhost.companyname.com ssl]# openssl req -new -newkey rsa:4096 -keyout
nginx.key -out nginx.csr
Generating a RSA private key
.....+++++
.....
.....+++++
writing new private key to 'nginx.key'
Enter PEM pass phrase:passphrase
Verifying - Enter PEM pass phrase:passphrase
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:CA
Locality Name (eg, city) [Default City]:Orange County
Organization Name (eg, company) [Default Company Ltd]:CompanyName
Organizational Unit Name (eg, section) []:BusinessUnit
Common Name (eg, your name or your server's hostname)
[]:reverseproxyhostname.companydomain.comEmail Address []:john.doe@comapnydomain.com

Please enter the following 'extra' attributes to be sent with your certificate request A
challenge password []:challengePWD
An optional company name []:CompanyName
```

   Schrijf de PEM pass-zin op, omdat deze gebruikt zal worden om de privésleutel tijdens de implementatie te decrypteren.
2. Ontvang het ondertekende certificaat van de CA. Stuur de CSR naar de certificeringsinstantie en verdien het ondertekende certificaat.
3. Hiermee implementeert u het certificaat en de toets. Decrypteer de sleutel die eerder als deel van de eerste stap met de **openssl rsa -in nginx.key -out nginx_decrypted.key** uit. Plaats het CA-ondertekende certificaat en de gedecrypteerde sleutel in een map (/usr/local/openresty/nginx/ssl in het voorbeeld) in de omgekeerde proxy-machine. SSL-configuraties bijwerken of toevoegen in verband met het certificaat in de NGINX-configuraties.

```
server {
    server_name <proxy-name>;
    listen 8445 ssl reuseport http2;
    listen [::]:8445 ssl reuseport http2;
    .....
    ssl_certificate /usr/local/openresty/nginx/ssl/ca_signed_cert.crt;
    ssl_certificate_key /usr/local/openresty/nginx/ssl/nginx_decrypted.key;
    .....
}
```

4. Configureer de rechten voor de certificaten. Voer in **chmod 400 /usr/local/openresty/nginx/ssl/ca_signed_cert.crt** en **chmod 400 /usr/local/openresty/nginx/ssl/nginx_decrypted.key**, zodat het certificaat alleen-lezen toestemming

heeft en beperkt is tot de eigenaar.
5. Opnieuw laden van Nginx.

Gebruik Custom Diffie-Hellman parameter

Maak een aangepaste Diffie-Hellman parameter met deze opdrachten:

- `openssl dhparam -out /usr/local/openresty/nginx/ssl/dhparam.pem 2048`
- `chmod 400 /usr/local/openresty/nginx/ssl/dhparam.pem`

Wijzig de serverconfiguratie om de nieuwe parameters te gebruiken:

```
server {
    .....
    ssl_dhparam /usr/local/openresty/nginx/ssl/dhparam.pem;
    .....
}
```

Zorg ervoor dat OCSP-stapeling is ingeschakeld - controle van herroeping van certificaat

> **Opmerking:** Om dit mogelijk te maken, moet de server een CA ondertekend certificaat gebruiken en moet de server toegang hebben tot de CA die het certificaat heeft ondertekend.

Voeg dit toe aan de configuratie:

```
server {
    .....
    ssl_stapling on;
    ssl_stapling_verify on;
    .....
}
```

NGINX-configuratie

Het standaard configuratie bestand van Neginx (**/usr/local/openresty/nginx/conf/nginx.conf**) moet worden aangepast om deze items te bevatten om beveiliging af te dwingen en prestaties te leveren. Deze inhoud moet worden gebruikt om het standaardconfiguratiebestand te wijzigen dat door de NGINX-installatie gemaakt is.

```
# Increasing number of worker processes will not increase the processing the request. The number
of worker process will be same as number of cores
# in system CPU. Nginx provides "auto" option to automate this, which will spawn one worker for
each CPU core.
worker_processes  auto;

# Process id file location
pid /usr/local/openresty/nginx/logs/nginx.pid;

# Binds each worker process to a separate CPU
worker_cpu_affinity auto;
```

```
#Defines the scheduling priority for worker processes. This should be calculated by "nice"
command. In our proxy set up the value is 0
worker_priority 0;


error_log  /usr/local/openresty/nginx/logs/error.log info;

#user root root;

# current limit on the maximum number of open files by worker processes, keeping 10 times of
worker_connections

worker_rlimit_nofile 102400;

events {
    multi_accept on;


    # Sets the maximum number of simultaneous connections that can be opened by a worker
process.
    # This should not be more the current limit on the maximum number of open files i.e. hard
limit of the maximum number of open files for the user (ulimit -Hn)
    # The appropriate setting depends on the size of the server and the nature of the traffic,
and can be discovered through testing.
    worker_connections  10240;
    #debug_connection 10.78.95.21
}


http {

    include       mime.types;


    default_type  text/plain;


    ## Must-change Change with DNS resolver ip in deployment
    resolver 192.168.1.3;


    ## Must-change change lua package path to load lua libraries
    lua_package_path
"/usr/local/openresty/lualib/resty/?.lua;/usr/local/openresty/nginx/lua/?.lua;;"


    ## Must-change change proxy_temp folder as per cache directory configurations
    proxy_temp_path /usr/local/openresty/nginx/cache/proxy_temp 1 2 ;
    ## Must-change change client_temp folder as per cache directory configurations
    client_body_temp_path /usr/local/openresty/nginx/cache/client_temp 1 2 ;


    lua_shared_dict userlist 50m;
    lua_shared_dict credentialsstore 100m;
    lua_shared_dict userscount 100k;
    lua_shared_dict clientstorage 100m;
    lua_shared_dict blockingresources 100m;
    lua_shared_dict tokencache_saproxy 10M;
    lua_shared_dict tokencache_saproxy125 10M;
    lua_shared_dict ipstore 10m;
    lua_shared_dict desktopurllist 10m;
    lua_shared_dict desktopurlcount 100k;
    lua_shared_dict thirdpartygadgeturllist 10m;
    lua_shared_dict thirdpartygadgeturlcount 100k;
```

```
    lua_shared_dict corsheadersstore 100k;


    init_worker_by_lua_block {
        local UsersListManager = require('users_list_manager')
        local UnauthenticatedDesktopResourcesManager =
require("unauthenticated_desktopresources_manager")
        local UnauthenticatedResourcesManager =
require("unauthenticated_thirdpartyresources_manager")
        --  Must-change Replace saproxy.cisco.com with reverseproxy fqdn

        if  ngx.worker.id() == 0 then
            UsersListManager.getUserList("saproxy.cisco.com",
"https://saproxy.cisco.com:8445/finesse/api/Users")
            UnauthenticatedDesktopResourcesManager.getDesktopResources("saproxy.cisco.com",
"https://saproxy.cisco.com:8445/desktop/api/urls?type=desktop")
            UnauthenticatedResourcesManager.getThirdPartyGadgetResources("saproxy.cisco.com",
"https://saproxy.cisco.com:8445/desktop/api/urls?type=3rdParty")         end     }     include
conf.d/*.conf;     sendfile        on;     tcp_nopush     on;     server_names_hash_bucket_size
512;
```

Omgekeerde proxy-poort configureren

Standaard luistert de configuratie van Nginx naar poort 8445 voor Finse verzoeken. In een tijd kan slechts één poort worden geactiveerd van een omgekeerde volmacht om Finse verzoeken te steunen, bijvoorbeeld 8445. Als poort 443 moet worden ondersteund, moet u het **<nginx-install-folder>conf/conf.d/finesse.conf** bewerken om luisteren op 443 mogelijk te maken en om luisteren op 8445 uit te schakelen.

Configuratie van wederzijdse TLS-verificatie tussen omgekeerde proxy en upstream componenten

In ES02 worden standaard alle upstream componenten geconfigureerd om alle hosts die als deel van toegestane hosts zijn toegevoegd, te valideren door **gebruik te maken van** omgekeerde **proxy-hosts de** opdracht **<proxy-host>toe te voegen**. Om succesvolle communicatie te hebben tussen omgekeerde proxy-hosts en upstream componenten (Finesse/IDS/CUIC/Livedata) moeten reverse proxy-certificaten worden geüpload naar de tomcat-trust store van alle upstream componenten. Knop opnieuw starten als de certificaten zijn geüpload.

De validatie van het upstream servercertificaat door omgekeerde proxy is standaard optioneel en uitgeschakeld. Als u volledige TLS wederzijdse authenticatie wilt bereiken tussen reverse proxy en upstream hosts, moet deze configuratie niet worden gereageerd op de bestanden **sl.conf** en **sl2.conf**.

```
#Enforce upstream server certificate validation at proxy ->
#this is not mandated as per CIS buit definitely adds to security.
#It requires the administrator to upload all upstream server certificates to the proxy
certificate store
#Must-Change Uncomment below lines IF need to enforce upstream server certificate validation at
proxy
#proxy_ssl_verify on;
#proxy_ssl_trusted_certificate /usr/local/openresty/nginx/ssl/finesse25.crt;
```

Het **proxy_ssl_vertrouwde_certificate** bestand moet de upstream certificatenreeks bevatten die is aaneengeschakeld.

Wissen

De reverse proxy cache kan worden gewist uit.

Standaardrichtsnoeren

In deze sectie worden kort de standaardrichtlijnen beschreven die moeten worden opgevolgd wanneer u Nginx als proxyserver instelt.

Deze richtlijnen zijn afgeleid van het Center for Internet Security. Zie voor meer informatie over elk richtsnoer hetzelfde.

1. Het wordt altijd aanbevolen de nieuwste stabiele OpenResty- en OpenSSL-versie te gebruiken.
2. Aanbevolen wordt om Nginx in een afzonderlijke diskmontage te installeren.
3. De NGINX procesid moet eigendom zijn van de wortelgebruiker (of, zoals van toepassing, voor gekozen OS) en moet toestemming **644 (rw—)** of strenger hebben.
4. Nginx moet verzoeken om onbekende hosts blokkeren. Zorg ervoor dat elk serverblok de server_name richtlijn expliciet definieert. Om te verifiëren, zoek alle serverblokken in de **nginx.conf** en **nginx/conf.d** folder en controleer of alle serverblokken de server_name bevatten.
5. Nginx moet alleen luisteren in de toegestane havens. Zoek alle serverblokken in de directory **nginx.conf** en **nginx/conf.d** en controleer of er naar richtlijnen wordt geluisterd om te controleren of alleen de toegestane havens open zijn om te luisteren.
6. Aangezien Cisco Finesse HTTP niet ondersteunt, wordt het aanbevolen om ook de HTTP-poort van de proxy-server te blokkeren.
7. Het Nginx SSL-protocol moet TLS 1.2 zijn. Ondersteuning voor legacy SSL-protocollen moet worden verwijderd. Het moet ook zwakke SSL-telefoons uitschakelen.
8. Aanbevolen wordt om Nginx-fout- en toegangslogbestanden naar de afstandsbediening te sturen.
9. Het wordt geadviseerd om de **mod_security** module te installeren die als een web applicatie firewall werkt. Zie de Security handleiding voor meer informatie. Merk op dat Nginx load niet is geverifieerd binnen de module **mod_security**.

Toewijzing-bestand configureren

De omgekeerde proxy-toepassing van het bureaublad vereist een mapping-bestand om de lijst van extern zichtbare hostname/poortcombinaties te configureren en hun mapping naar de eigenlijke servernamen en -poorten die worden gebruikt door de Finse, IDS- en CUIC-servers. Dit mapping-bestand dat op interne servers is ingesteld, is de sleutelconfiguratie waarmee de klanten die via het internet zijn aangesloten, kunnen worden omgeleid naar de vereiste hosts en poorten die op het internet worden gebruikt.

Het mapping-bestand moet worden ingezet op een webserver die toegankelijk is voor de component-servers en de URI ervan moet worden geconfigureerd zodat de implementatie kan werken. Aanbevolen wordt om het mapping-bestand te configureren met behulp van een speciale webserver die binnen het netwerk beschikbaar is. Als een dergelijke server niet beschikbaar is, kan de omgekeerde proxy worden gebruikt in plaats daarvan, op grond waarvan de proxy toegankelijk moet zijn binnen het netwerk en bovendien het risico inhoudt dat de informatie wordt overgebracht naar externe klanten die ongeoorloofde toegang tot de DMZ kunnen verlenen. In de volgende paragraaf wordt beschreven hoe dit kan worden verwezenlijkt.

Raadpleeg de functiehandleiding voor de exacte stappen om het mapping-bestand URI op alle componentservers te configureren en voor meer informatie over hoe u de mapping-bestandsgegevens maakt.

Omgekeerde proxy als tapeserver gebruiken

Deze stappen worden alleen vereist als de omgekeerde proxy ook wordt gebruikt als de host van het proxy-mapping-bestand.

1. Configureer de omgekeerde proxy-hostname in de domeincontroller die door de Finse/CUIC- en IDS-hosts wordt gebruikt, zodat het IP-adres kan worden opgelost.
2. Upload de gegenereerde Nginx ondertekende certificaten op beide knooppunten onder groot computervertrouwen en start de server opnieuw.
3. update de **must-change** waardes in **<NGINX_HOME>/html/proxymap.txt**.
4. NGinx-configuraties opnieuw laden met de **nginx -s reload** uit.
5. Bevestig dat het configuratiebestand vanaf een andere netwerkhost toegankelijk is met behulp van het **curl** uit.

CentOS 8 Kernel Hardening

Als het gekozen besturingssysteem CentOS 8 is, wordt aanbevolen om de fijnafstemming van de kern te laten plaatsvinden met behulp van deze systeemconfiguraties voor installaties die een speciale server gebruiken om de proxy te host.

```
## Configurations for kernel hardening - CentOS8. The file path is /etc/sysctl.conf
## Note that the commented configurations denote that CentOS 8's default value matches
## the recommended/tested value, and are not security related configurations.

# Avoid a smurf attack
net.ipv4.icmp_echo_ignore_broadcasts = 1

# Turn on protection for bad icmp error messages
net.ipv4.icmp_ignore_bogus_error_responses = 1

# Turn on syncookies for SYN flood attack protection
net.ipv4.tcp_syncookies = 1

# Turn on and log spoofed, source routed, and redirect packets
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.default.log_martians = 1

# Turn off routing
net.ipv4.ip_forward = 0
net.ipv4.conf.all.forwarding = 0
net.ipv6.conf.all.forwarding = 0

net.ipv4.conf.all.mc_forwarding = 0
net.ipv6.conf.all.mc_forwarding = 0

# Block routed packets
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.default.accept_source_route = 0
net.ipv6.conf.all.accept_source_route = 0
net.ipv6.conf.default.accept_source_route = 0
```

```
# Block ICMP redirects
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv6.conf.all.accept_redirects = 0
net.ipv6.conf.default.accept_redirects = 0
net.ipv4.conf.all.secure_redirects = 0
net.ipv4.conf.default.secure_redirects = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0

# Filter routing packets with inward-outward path mismatch(reverse path filtering)
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1

# Router solicitations & advertisements related.
net.ipv6.conf.default.router_solicitations = 0
net.ipv6.conf.default.accept_ra_rtr_pref = 0
net.ipv6.conf.default.accept_ra_pinfo = 0
net.ipv6.conf.default.accept_ra_defrtr = 0
net.ipv6.conf.default.autoconf = 0
net.ipv6.conf.default.dad_transmits = 0
net.ipv6.conf.default.max_addresses = 1
net.ipv6.conf.all.accept_ra = 0
net.ipv6.conf.default.accept_ra = 0

# Backlog - increased from default 1000 to 5000.
net.core.netdev_max_backlog = 5000

# Setting syn/syn-ack retries to zero, so that they don't stay in the queue.
net.ipv4.tcp_syn_retries = 0
net.ipv4.tcp_synack_retries = 0

# Max tcp listen backlog. Setting it to 511 to match nginx config
net.core.somaxconn = 511

# Reduce the duration of connections held in TIME_WAIT(seconds)
net.ipv4.tcp_fin_timeout = 6

# Maximum resources allotted
# fs.file-max = 2019273
# kernel.pid_max = 4194304
# net.ipv4.ip_local_port_range = 32768 60999

# TCP window size tuning
# net.ipv4.tcp_window_scaling = 1
# net.core.rmem_default = 212992
# net.core.rmem_max = 212992
# net.ipv4.tcp_rmem = 4096 87380 6291456
# net.ipv4.udp_rmem_min = 4096
# net.core.wmem_default = 212992
# net.core.wmem_max = 212992
# net.ipv4.tcp_wmem = 4096 16384 4194304
# net.ipv4.udp_wmem_min = 4096
# vm.lowmem_reserve_ratio = 256 256 32 0 0
# net.ipv4.tcp_mem = 236373 315167 472746

# Randomize virtual address space
kernel.randomize_va_space = 2

# Congestion control
# net.core.default_qdisc = fq_codel
# net.ipv4.tcp_congestion_control = cubic

# Disable SysReq
```

```
kernel.sysrq = 0

# Controls the maximum size of a message, in bytes
kernel.msgmnb = 65536

# Controls the default maximum size of a message queue
kernel.msgmax = 65536

# Controls the eagerness of the kernel to swap.
vm.swappiness = 1
```

De computer wordt opnieuw opgestart nadat u de aanbevolen wijzigingen hebt aangebracht.

IP-tafels versleutelen

IPtafels is een toepassing waarmee een systeembeheerder de IPv4- en IPv6-tabellen, ketens en regels kan configureren die worden geleverd door de Linux-kernelfirewall.

Deze IPtafellinnen zijn geconfigureerd om de proxy-toepassing te beveiligen tegen aanvallen met brute kracht door de toegang te beperken in de Linux-kernelfirewall.

De commentaren in de configuratie geven aan welke dienst met behulp van de regels aan tariefbeperkingen is onderworpen.

> **Opmerking:** Als beheerders een andere poort gebruiken of de toegang tot meerdere servers uitbreiden met dezelfde poorten, moet de juiste grootte worden gemaakt voor deze poorten op basis van deze getallen.

```
## Configuration for iptables service
## The file path is /etc/sysconfig/iptables
## Make a note for must-change values to be replaced.
## Restart of the iptable service is required after applying following rules
*filter :INPUT ACCEPT [0:0] :FORWARD ACCEPT [0:0] :OUTPUT ACCEPT [0:0] # Ensure loopback traffic
is configured -A INPUT -i lo -j ACCEPT -A OUTPUT -o lo -j ACCEPT -A INPUT -s 127.0.0.0/8 -j DROP
# Ensure ping openeded only for the particular source and blocked for rest # Must-Change:
Replace the x.x.x.x with valid ip address -A INPUT -p ICMP --icmp-type 8 -s x.x.x.x -j ACCEPT #
Ensure outbound and established connections are configured -A INPUT -p tcp -m state --state
RELATED,ESTABLISHED -j ACCEPT -A OUTPUT -p tcp -m state --state NEW,RELATED,ESTABLISHED -j
ACCEPT # Block ssh for external interface # Must-Change: Replace the ens224 with valid ethernet
interface -A INPUT -p tcp -i ens224 --dport 22 -j DROP # Open inbound ssh(tcp port 22)
connections -A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT # Configuration for
finesse 8445 port -A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m connlimit --
connlimit-above 10 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1
-j LOG --log-prefix " Connections to 8445 exceeded connlimit " -A INPUT -p tcp -m tcp --dport
8445 --tcp-flags SYN SYN -m connlimit --connlimit-above 10 --connlimit-mask 32 --connlimit-saddr
-j DROP -A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto
6/sec --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8445_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -
j LOG --log-prefix " Exceeded 8445 hashlimit " -A INPUT -p tcp -m tcp --dport 8445 --tcp-flags
SYN SYN -j DROP # Configuration for IdS 8553 port -A INPUT -p tcp -m tcp --dport 8553 --tcp-
flags SYN SYN -m connlimit --connlimit-above 6 --connlimit-mask 32 --connlimit-saddr -m limit --
limit 1/min --limit-burst 1 -j LOG --log-prefix " IdS connection limit exceeded" -A INPUT -p tcp
-m tcp --dport 8553 --tcp-flags SYN SYN -m connlimit --connlimit-above 6 --connlimit-mask 32 --
connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m hashlimit --
hashlimit-upto 2/sec --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name
TCP_8553_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m limit --limit
1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 8553 hashlimit " -A INPUT -p tcp -m tcp --
dport 8553 --tcp-flags SYN SYN -j DROP # Configuration for IdP 443 port -A INPUT -p tcp -m tcp -
```

-dport 443 --tcp-flags SYN SYN -m connlimit --connlimit-above 8 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " IdP connection limit exceeded" -A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m connlimit --connlimit-above 8 --connlimit-mask 32 --connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 4/sec --hashlimit-burst 6 --hashlimit-mode srcip,dstport --hashlimit-name TCP_443_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 443 hashlimit " -A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -j DROP # Must-Change: A2A file transfer has not been considered for below IMNP configuration. # For A2A for support, these configuration must be recalculated to cater different file transfer scenarios. # Configuration for IMNP 5280 port -A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " IMNP connection limit exceeded" -A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30 --connlimit-mask 32 --connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 20/sec --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_5280_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 5280 hashlimit " -A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -j DROP # Configuration for IMNP 15280 port -A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " IMNP connection limit exceeded" -A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30 --connlimit-mask 32 --connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 20/sec --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_15280_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 15280 hashlimit " -A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -j DROP # Configuration for IMNP 25280 port -A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " IMNP connection limit exceeded" -A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30 --connlimit-mask 32 --connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 20/sec --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_25280_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 25280 hashlimit " -A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -j DROP # Configuration for CUIC 8444 port -A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m connlimit --connlimit-above 6 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " CUIC connection limit exceeded" -A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m connlimit --connlimit-above 6 --connlimit-mask 32 --connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8444_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 8444 hashlimit " -A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -j DROP # Configuration for CUIC 8447 port -A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m connlimit --connlimit-above 6 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " CUIC connection limit exceeded" -A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m connlimit --connlimit-above 6 --connlimit-mask 32 --connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8447_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 8447 hashlimit " -A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -j DROP # Configuration for LiveData 12005 port -A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m connlimit --connlimit-above 10 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " LD connection limit exceeded" -A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m connlimit --connlimit-above 10 --connlimit-mask 32 --connlimit-saddr -j DROP -A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name TCP_12005_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 12005 hashlimit " -A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -j DROP # Configuration for LiveData 12008 port -A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m connlimit --connlimit-above 10 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " LD connection limit exceeded" -A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m connlimit --connlimit-above 10 --connlimit-mask 32 --connlimit-

```
saddr -j DROP -A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name TCP_12008_DOS -j ACCEPT -A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded 12008 hashlimit " -A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -j DROP # Block all other ports -A INPUT -j REJECT --reject-with icmp-host-prohibited -A FORWARD -j REJECT --reject-with icmp-host-prohibited COMMIT
```

Deze regels zouden direct kunnen worden toegepast door de **machine/etc/sysfig/iptafels** handmatig te bewerken of de configuratie in een bestand op te slaan, zoals **iptafels.conf** en cat **iptafels** uit te voeren **iptafels.conf >>/etc/sysfig/iptafels** om de regels toe te passen.

U moet de IPtafellinnen opnieuw opstarten nadat u de regels hebt toegepast. Voer in **systemctl restart iptables** om de IPtafellinnen opnieuw te starten.

Clientverbindingen beperken

Naast de vorige configuratie van IP-tabellen, worden installaties die het adresbereik van klanten kennen die de proxy gebruiken, aanbevolen om deze kennis te gebruiken om de toegangsregels voor proxy veilig te stellen. Dit kan grote voordelen opleveren als het erop aankomt de volmacht te verzekeren van kwaadaardige netwerken die vaak worden gecreëerd in het adressengebied van het IP van landen die meer lakse regels hebben op het gebied van online veiligheid. Het is daarom sterk aanbevolen om de IP-adresbereik te beperken tot op land/staat of ISP gebaseerde IP-bereik als u zeker bent van de toegangspatronen.

Clientverbindingen blokkeren

Het is ook nuttig om te weten hoe te om een specifiek bereik van adressen te blokkeren wanneer een aanval van een IP adres of een bereik van IP adressen wordt geïdentificeerd die moet worden gemaakt. In dergelijke gevallen kunnen de verzoeken van die IP-adressen worden geblokkeerd met **tegenstrijdige** regels.

Blok onderscheiden IP-adressen

Om meerdere verschillende IP-adressen te blokkeren, voegt u een regel toe aan het **IPTables** configuratiebestand voor elk IP-adres.

Bijvoorbeeld, om adressen te blokkeren **192.0.2.3** en **192.0.2.4**, ga in:

```
iptables -A INPUT -s 192.0.2.3 -j DROP iptables -A INPUT -s 192.0.2.4 - j DROP.
```

Een bereik van IP-adressen blokkeren

Blokkeer meerdere IP-adressen in een bereik en voeg één regel toe aan het **IPTables** configuratiebestand met het IP-bereik.

Bijvoorbeeld, om adressen van 192.0.2.3 tot 192.0.2.35 te blokkeren, ga in:

```
iptables -A INPUT -m iprange --src-range 192.0.2.3-192.0.2.35 -j DROP.
```

Alle IP-adressen in een subnetwerk blokkeren

Blokkeer alle IP-adressen in een volledig subnetwerk door één enkele lijn aan het

configuratiebestand **IPTables** toe te voegen met het gebruik van de klasse inter-domein routingnotatie voor het IP-adresbereik. Bijvoorbeeld, om alle klasse **C adressen** te blokkeren, ga in:

```
iptables -A INPUT -s 192.0.0.0/16 -j DROP.
```

SELinux

SELinux is een platform security raamwerk dat geïntegreerd is als een verbetering in de Linux-OS. De procedure om beleid SELinux te installeren en toe te voegen om OpenResty uit te voeren zoals de omgekeerde proxy hierna wordt geleverd.

1. Stop het proces met het **openresty -s stop** uit.
2. Infinx-server configureren en starten/stoppen met **systemctl** Opdracht zodat tijdens het opstarten van het proces OpenResty automatisch wordt gestart. Voer deze opdrachten in als basisgebruiker. Ga naar **/gebruiker/lib/systeem/systeem**.Open bestand met de naam **openresty.service**.Update de inhoud van het bestand volgens de locatie **van het PIDF-bestand**.

   ```
   [Unit]
   Description=The OpenResty Application Platform
   After=syslog.target network-online.target remote-fs.target nss-lookup.target
   Wants=network-online.target

   [Service]
   Type=forking
   PIDFile=/usr/local/openresty/nginx/logs/nginx.pid
   ExecStartPre=/usr/local/openresty/nginx/sbin/nginx -t
   ExecStart=/usr/local/openresty/nginx/sbin/nginx
   ExecReload=/bin/kill -s HUP $MAINPID
   ExecStop=/bin/kill -s QUIT $MAINPID
   PrivateTmp=true

   [Install]
   WantedBy=multi-user.target
   ```

   Als basisgebruiker voert u het programma in **sudo systemctl enable openresty**.Start / Stop de OpenResty-service met de **systemctl start openresty / systemctl stop openresty** deze opdracht uitvoeren en er zeker van zijn dat het proces start / stopt als basisgebruiker.

1. **Selinux installeren** Standaard worden alleen enkele SELinux-pakketten in CentOs geïnstalleerd.Het **beleidskeuzes** en de bijbehorende afhankelijkheden moeten worden geïnstalleerd om het SELinux-beleid te kunnen verwezenlijken.Voer deze opdracht in om **de** gewenste **waterpas te** installeren

   ```
   yum install policycoreutils-devel
   ```

   Zorg ervoor dat na het installeren van de verpakking **sepolicy** commando werkt.

   ```
   usage: sepolicy [-h] [-P POLICY]

                   {booleans,communicate,generate,gui,interface,manpage,network,transition}
                   ...

   SELinux Policy Inspection Tool
   ```

2. **Een nieuwe Linux-gebruiker en -kaart maken met de linux-gebruiker**Voer in **semanage login -l** om de mapping tussen Linux-gebruikers en SELinux te bekijken.

   ```
   [root@loadproxy-cisco-com ~]# semanage login -l

   Login Name           SELinux User         MLS/MCS Range        Service
   ```

```
__default__          unconfined_u         s0-s0:c0.c1023       *                *
root                 unconfined_u         s0-s0:c0.c1023       *
```

Als root maakt u een nieuwe Linux-gebruiker (**nginx-**gebruiker) die is toegewezen aan de SELinux **user_u-**gebruiker.

```
useradd -Z user_u nginxuser
[root@loadproxy-cisco-com ~]# passwd nginxuser
Changing password for user nginxuser.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

Om de mapping tussen **nginxuser** en **user_u** te bekijken, voert u deze opdracht als wortel in:

```
[root@loadproxy-cisco-com ~]# semanage login -l

Login Name           SELinux User         MLS/MCS Range        Service

__default__          unconfined_u         s0-s0:c0.c1023       *
nginxuser            user_u               s0                   *
root                 unconfined_u         s0-s0:c0.c1023       *
```

SELinux **__default__**login door standaard in kaart gebracht aan de SELinux **unbegrensde_u** gebruiker. Deze opdracht is vereist om **user_u** standaard te beperken tot deze opdracht:

```
semanage login -m -s user_u -r s0 __default__
```

Om te controleren of de opdracht goed werkte, voert u **semanage login -l**. Het moet deze output produceren:

Wijzig nginx.conf en voer veranderingseigendom voor nginxuser uit.Voer in **chown -R nginxuser:nginxuser** * in de **<OpenPresence-install-folder>**folder.Wijzig het **nginx.conf-**bestand aan om nginxuser als de gebruiker op te nemen voor het uitvoeren van werknemersprocessen.

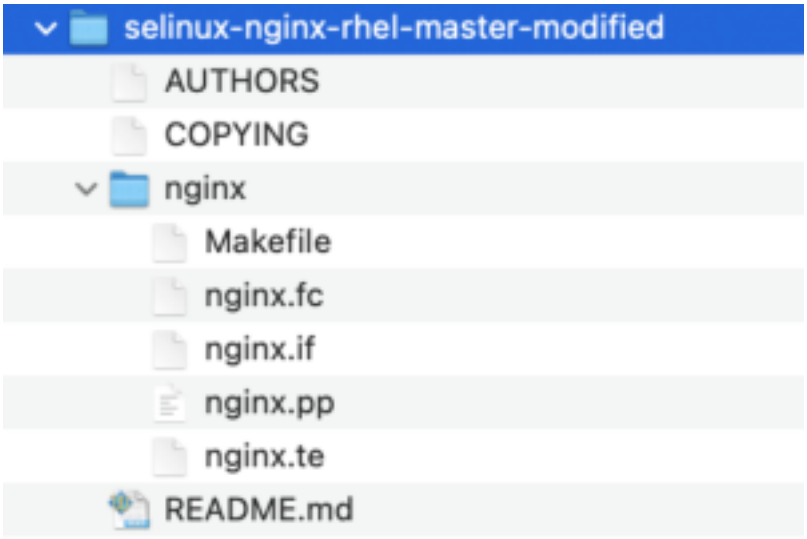
```
........

user nginxuser nginxuser;

..........
```

Schrijf het SELinux-beleid voor NGINX

1. In plaats van een nieuw standaard douanebeleid te genereren voor Nginx met de **sepolicy generate --init /usr/bin/nginx** commando , het verdient de voorkeur te beginnen met een bestaand beleid .
2. De bestanden **nginx.fc** (File Context file) en **nginx.te** (Type Encapsulation file) die u met de meegeleverde URL hebt gedownload, zijn aangepast om het omgekeerde proxy-gebruik te passen.
3. Deze aangepaste versie kan als referentie worden gebruikt, aangezien deze voor het specifieke geval van gebruik is vastgesteld.
4. Download het bestand **selinux-nginx-rhel-master-gemodificeerd.tar** van de downloadpagina van Finesse 12.6 ES02-software voor het bestand.



5. Pak het .tar-bestand uit en navigeer naar de **nginx**-map erin.
6. Open het **.fc-**bestand en controleer de gewenste bestandspaden van nginx installateur, cache en pid-bestand.
7. Stel de configuratie samen met de **make** uit.
8. Het **nginx.pp**-bestand wordt gegenereerd.
9. Het beleid laden met de **semodule** uit.
   ```
   semodule -i nginx.pp
   ```
10. Ga naar **/wortel** en maak een leeg bestand met de naam **touch /.autorelabel**.
11. Herstart het systeem.
12. Typ deze opdracht om te controleren of het beleid is geladen.
    ```
    semodule --list-modules=full
    ```
    ```
    [root@loadproxy-cisco-com ~]# semodule --list-modules=full
    400 nginx                 pp
    200 container             pp
    200 flatpak               pp
    100 abrt                  pp
    100 accountsd             pp
    100 acct                  pp
    100 afs                   pp
    100 aiccu                 pp
    100 aide                  pp
    100 ajaxterm              pp
    100 alsa                  pp
    ```
13. Neginx moet zonder schending draaien. (De schendingen zijn beschikbaar in **var/log/messen** en **/var/log/audit/audit.log**).
14. Typ deze opdracht om de status van Nginx te controleren.
    ```
    ps -aefZ | grep nginx
    ```
    ```
    [root@loadproxy-cisco-com ~]# ps -aefZ |grep nginx
    system_u:system_r:nginx_t:s0    root        1686       1  0 16:14 ?        00:00:00 nginx: master process /usr/bin/nginx
    system_u:system_r:nginx_t:s0    nginxus+    1687    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1688    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1689    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1690    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1691    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1692    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1693    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1694    1686  0 16:14 ?        00:00:00 nginx: worker process
    system_u:system_r:nginx_t:s0    nginxus+    1695    1686  0 16:14 ?        00:00:00 nginx: cache manager process
    unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 2543 2252  0 16:17 pts/0 00:00:00 grep --color=auto nginx
    ```
15. Nu zou de Finesse agent/Supervisor desktop toegankelijk moeten zijn.

Verifiëren

Gebruik dit gedeelte om te bevestigen dat de configuratie correct werkt.

Finesse

1. Verzoek https://<reseproxy:port>/finesse/api/SystemInfo. uit de DMZ en controleer of ze bereikbaar zijn.
2. Controleer **<host>** waarden in zowel **<PrimeNode>** als **<secondairNode>** zijn geldige reverse proxy-hostnamen. Het moet geen Finesse hostname zijn.

CUIC- en bewegende gegevens

1. Als Finse hostname in de respons wordt gezien in plaats van omgekeerde proxy-hostname, valideren de proxy-mapping-configuraties en toegestane hosts worden correct toegevoegd in Finse servers zoals beschreven in de sectie "Population Network Translation Data" van "VPN-Minder toegang tot finesse desktop" in de Finse 12.6 UCCE functiegids.
2. Als LiveData-gadgets correct worden geladen in Finse desktop zijn de CUIC- en LiveData-proxy-configuraties goed.
3. Om CUIC en LiveData configuratie te valideren moet u HTTP-aanvragen indienen bij deze URL's vanuit de DMZ en zien of ze bereikbaar zijn. https://<reproxy:cuic_port>/cuic/rest/onhttps://<reproxy:ldweb_port>/livedata/securityhttps://<reproxy:ldsocketio_port>/security

IDs

Voer de volgende stappen uit om de configuratie van IDs te valideren:

1. Meld u aan bij de IDSAdmin-interface op **https://<ids_LAN_host:ids_port>:853/disadmin** van het LAN omdat de admin-interface niet via omgekeerde proxy is blootgesteld.
2. Kies **Instellingen > Vertrouwen van IDs**.
3. Bevestig dat het knooppunt van de proxy-clusteruitgever is opgenomen op de Download SP-metagegevenspagina en klik op **Volgende**.
4. Bevestig dat de IDP proxy correct wordt weergegeven indien geconfigureerd op de pagina met IDP-metagegevens uploaden en klik op **Volgende**.
5. Start test SSO via alle proxy-clusterknooppunten van de Test SSO-pagina en bevestig alle resultaten. Dit vereist client machine connectiviteit om volmachtknopen om te keren.

Prestaties

De gegevensanalyse van de meest equivalente prestatieopname, gemaakt met het betreffende gereedschap, is beschikbaar op de downloadpagina van Finesse release 12.6(1)ES2-software (**load_result.zip**). De gegevens vertegenwoordigen de status van de proxy voor desktop- en supervisor-activiteiten, op een steekproefsgewijze 2000 UCCE-inzet met behulp van SSO-logins en CUIC LD-rapporten, zoals ingesteld in de standaardlay-out voor 2000 gebruikers gedurende een periode van acht uur. Kan worden gebruikt om de computer-, schijf- en netwerkvereisten voor een installatie af te leiden met NGinx op vergelijkbare hardware.

Problemen oplossen

SELinux

1. Als Nginx standaard niet gestart is of het bureaublad Finesse Agent niet toegankelijk is, stelt u SELinux in op **permissieve** modus met deze opdracht:
   ```
   setenforce 0
   ```
2. Probeer het progamma opnieuw te starten met de **systemctl restart nginx** uit.
3. De inbreuken zijn te vinden in **/var/log/messen** en **/var/log/audit/audit.log**.
4. Het is vereist om .te bestand te regenereren met regels om deze schendingen door een van deze opdrachten aan te pakken:
   ```
   cat /var/log/audit/audit.log | audit2allow -m nginx1 > nginx1.te. # this will create nginx1.te file
                  or
   ausearch -c 'nginx' --raw | audit2allow -M  my-nginx  # this will create my-nginx.te file
   ```
5. Update het oorspronkelijke **nginx.te**-bestand in de **selinux-nginx-rhel-master-aangepaste/nginx-**map met nieuw gegenereerde soepregels.
6. datzelfde met de **make** uit.
7. Het nginx.pp-bestand wordt opnieuw gegenereerd.
8. Laad het beleid met moduleopdracht.
   ```
   semodule -i nginx.pp
   ```
9. Make SELinux om deze opdracht **af** te **dwingen**:
   ```
   setenforce
   ```
10. Herstart het systeem.
11. Herhaal deze procedure tot de gewenste schendingen zijn vastgesteld.