Surveillance de Cisco ESA avec SNMP

Table des matières

Introduction

Conditions préalables

- Exigences
- Composants utilisés

Configuration du protocole SNMP

Configuration et interrogation du client SNMP sous Linux

- Installer les outils SNMP
- Charger les fichiers MIB
  - Utilisation d'un OID pour surveiller l'utilisation du processeur
- Activer les noms symboliques
- Exécuter des requêtes SNMP
  - Messages de file d'attente
  - Utilisation du processeur
  - Tableau Expiration de la clé de licence

Différence entre OID numériques et noms symboliques

- C'est pareil ?

Informations connexes

Introduction

Ce document décrit comment surveiller Cisco Secure Email Gateway à l'aide de SNMP, y compris la structure MIB, l'utilisation de l'OID et les requêtes pratiques.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Connaissances de base du protocole SNMP
- Accès à l'appliance Cisco ESA
- Connaissance de la ligne de commande Linux
- Cisco ESA avec service SNMP activé
- Client SNMP installé (comme les outils Net-SNMP)
- Fichiers MIB IronPort disponibles et chargés

- Chaîne de communauté ou identifiants SNMP v3

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Passerelle de messagerie sécurisée Cisco (ESA)
- Client Linux avec outils Net-SNMP
- Fichiers MIB : IRONPORT-SMI.txt, ASYNCOS-MAIL-MIB.txt

Configuration du protocole SNMP

La configuration SNMP sur ESA est effectuée via CLI. Afin d'activer SNMP sur Cisco ESA, accédez à la CLI et exécutez snmpconfig.

La configuration par défaut implique :

- Activation du service SNMP
- Choix de l'interface et du port de gestion (généralement 161)
- Activation de SNMPv3 (sécurité par défaut : authPriv avec SHA et AES)
- Définition des phrases secrètes d'authentification et de confidentialité
- Activation de SNMPv1/v2c, en spécifiant la chaîne de communauté (par exemple, ironport)
- Définition des réseaux IPv4 autorisés pour les requêtes SNMP
- Configuration de la version de déroutement SNMP et de l'adresse IP cible de déroutement
- Définition de l'emplacement du système et des coordonnées

Après avoir activé SNMP, vous pouvez voir un résumé semblable à ceci :

```
esa1.ironport.com> snmpconfig

Current SNMP settings:
Listening on interface "Management"



      port 161.
SNMP v3: Enabled. Security level: authPriv
Authentication Protocol: SHA
Encryption Protocol: AES
SNMP v1/v2: Enabled, accepting requests from subnet
```

```
      , .
SNMP v1/v2 Community String: ironport
Trap version: V3
Trap target:



Location: esxi data center
System Contact: ciscoros soc
```

Une fois le protocole SNMP activé et configuré, le matériel est prêt à accepter les requêtes SNMP provenant des adresses IP source autorisées.

Configuration et interrogation du client SNMP sous Linux

Pour cet exemple, un serveur Debian a été utilisé. Notez que les étapes d'installation peuvent varier en fonction de votre gestionnaire de package de distribution.

Installer les outils SNMP

```
sudo apt-get install snmp snmp-mibs-downloader
```

Vérifiez que snmpwalk binary est installé.

```
diegoher@debian-server:/usr/share/snmp/mibs$ snmpwalk --version
NET-SNMP version: 5.9
```

Charger les fichiers MIB

Placez les fichiers MIB IronPort dans le dossier /usr/share/snmp/mibs.

```
root@debian-server:/usr/share/snmp/mibs# pwd
/usr/share/snmp/mibs
root@debian-server:/usr/share/snmp/mibs# ls
ASYNCOS-MAIL-MIB.txt  IRONPORT-SMI.txt        NET-SNMP-EXAMPLES-MIB.txt  NET-SNMP-PASS-MIB.txt  UCD-DEMO-MIB.txt    UCD-IPFWACC-MIB.txt
iana                  LM-SENSORS-MIB.txt      NET-SNMP-EXTEND-MIB.txt    NET-SNMP-TC.txt        UCD-DISKIO-MIB.txt  UCD-SNMP-MIB.txt
ietf                  NET-SNMP-AGENT-MIB.txt  NET-SNMP-MIB.txt           NET-SNMP-VACM-MIB.txt  UCD-DLMOD-MIB.txt
```

debian-server oids

Remarque : Les fichiers MIB se trouvent dans l'article SNMP partagé à la fin de ce document.

Utilisation d'un OID pour surveiller l'utilisation du processeur

Cette commande interroge l'ESA sur son utilisation actuelle du CPU. L'OID pointe directement vers la métrique de CPU définie dans la MIB. Le résultat affiche une valeur, telle que INTEGER : 37, indiquant une utilisation CPU du périphérique de 37 %. Cela permet aux administrateurs de surveiller les performances des périphériques en temps réel et d'intervenir si l'utilisation dépasse les limites acceptables.

```
snmpwalk -v2c -c ironport

      .1.3.6.1.4.1.15497.1.1.1.2
```

L'utilisation d'OID dans les commandes SNMP fournit un accès direct à des mesures spécifiques pour une surveillance et un dépannage efficaces.

Activer les noms symboliques

```
export MIBS=ALL
```

Si vous définissez export MIBS=ALL, les outils SNMP utilisent des noms lisibles par l'utilisateur définis dans les fichiers MIB au lieu de longs OID numériques. Cela facilite l'écriture, la compréhension et le dépannage des requêtes, car vous pouvez faire référence à des objets par des noms significatifs comme workQueueMessages plutôt que par des séquences de nombres.

Exécuter des requêtes SNMP

Utilisez snmpwalk pour interroger ESA sur des mesures clés. Les requêtes SNMP vous permettent de récupérer des données d'état et de performances en temps réel à partir de votre ESA Cisco. En utilisant des noms symboliques, vous pouvez facilement surveiller des objets spécifiques tels que l'état de la file d'attente, l'expiration de la licence et l'utilisation du matériel sans avoir besoin de référencer des OID numériques complexes.

Messages de file d'attente

```
diegoher@debian-server:/usr/share/snmp/mibs$ snmpwalk -v2c -c ironport
```

```
      workQueueMessages
ASYNCOS-MAIL-MIB::workQueueMessages.0 = Gauge32: 0
```

Ce résultat montre qu'il n'y a actuellement aucun message dans la file d'attente de travail ESA. La valeur représente le nombre d'e-mails en attente de traitement en temps réel.

Utilisation du processeur

```
diegoher@debian-server:/usr/share/snmp/mibs$ snmpwalk -v2c -c ironport
```

Cela indique que le CPU de ESA est actuellement à 37 % d'utilisation. La valeur vous donne un aperçu de la charge de traitement de l'appliance au moment où la requête a été exécutée.

Tableau Expiration de la clé de licence

```
diegoher@debian-server:/usr/share/snmp/mibs$ snmpwalk -v2c -c ironport
```

```
       keyExpirationTable
ASYNCOS-MAIL-MIB::keyExpirationIndex.1 = INTEGER: 1
ASYNCOS-MAIL-MIB::keyExpirationIndex.2 = INTEGER: 2
ASYNCOS-MAIL-MIB::keyExpirationIndex.3 = INTEGER: 3
ASYNCOS-MAIL-MIB::keyExpirationIndex.4 = INTEGER: 4
ASYNCOS-MAIL-MIB::keyExpirationIndex.5 = INTEGER: 5
```

```
ASYNCOS-MAIL-MIB::keyExpirationIndex.6 = INTEGER: 6
ASYNCOS-MAIL-MIB::keyExpirationIndex.7 = INTEGER: 7
ASYNCOS-MAIL-MIB::keyExpirationIndex.8 = INTEGER: 8
ASYNCOS-MAIL-MIB::keyDescription.1 = STRING: Bounce Verification
ASYNCOS-MAIL-MIB::keyDescription.2 = STRING: Data Loss Prevention
ASYNCOS-MAIL-MIB::keyDescription.3 = STRING: External Threat Feeds
ASYNCOS-MAIL-MIB::keyDescription.4 = STRING: Incoming Mail Handling
ASYNCOS-MAIL-MIB::keyDescription.5 = STRING: IronPort Anti-Spam
ASYNCOS-MAIL-MIB::keyDescription.6 = STRING: IronPort Email Encryption
ASYNCOS-MAIL-MIB::keyDescription.7 = STRING: Outbreak Filters
ASYNCOS-MAIL-MIB::keyDescription.8 = STRING: Sophos Anti-Virus
ASYNCOS-MAIL-MIB::keyIsPerpetual.1 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keyIsPerpetual.2 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keyIsPerpetual.3 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keyIsPerpetual.4 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keyIsPerpetual.5 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keyIsPerpetual.6 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keyIsPerpetual.7 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keyIsPerpetual.8 = INTEGER: true(1)
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.1 = Gauge32: 0
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.2 = Gauge32: 0
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.3 = Gauge32: 0
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.4 = Gauge32: 0
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.5 = Gauge32: 0
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.6 = Gauge32: 0
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.7 = Gauge32: 0
ASYNCOS-MAIL-MIB::keySecondsUntilExpire.8 = Gauge32: 0
```

- keyExpirationIndex.X : Chaque index représente une clé de fonction unique installée sur Cisco ESA.
- keyDescription.X : Fournit le nom ou la description de chaque clé de fonction, telle que « Vérification de renvoi », « Prévention de perte de données », « Antispam IronPort » et « Antivirus Sophos ».
- keyIsPerpetual.X : Indique si la licence de chaque fonction est permanente. La valeur true (1) signifie que la licence n'expire pas.
- keySecondsUntilExpire.X : Indique le nombre de secondes restantes avant l'expiration de la licence. La valeur 0 confirme que la licence est permanente ou a déjà expiré.



```
[]> summary

Feature Name                                                    License Authorization Status
-------------------------------------------------------------------------------------------------
Email Security Appliance Anti-Spam License                      In Compliance
Email Security Appliance Outbreak Filters                       In Compliance
Email Security Appliance Graymail Safe-unsubscribe              Not requested
Email Security Appliance External Threat Feeds                  In Compliance
Email Security Appliance Advanced Malware Protection Reputation Not requested
Mail Handling                                                   In Compliance
Email Security Appliance Sophos Anti-Malware                    In Compliance
Email Security Appliance PXE Encryption                         In Compliance
Email Security Appliance Advanced Malware Protection            Not requested
Email Security Appliance McAfee Anti-Malware                    Not requested
Email Security Appliance Intelligent Multi-Scan                 Not requested
Email Security Appliance Image Analyzer                         Not requested
Email Security Appliance Bounce Verification                    In Compliance
Email Security Appliance Data Loss Prevention                   In Compliance
```

exemple de licence

Ce résultat confirme les clés de fonction actuelles de l'appliance, leurs descriptions et l'état de la licence. Toutes les licences répertoriées sont perpétuelles, comme indiqué par keyIsPerpetual et keySecondsUntilExpire. Ces informations permettent de s'assurer que les fonctions de sécurité essentielles restent actives et valides sur votre Cisco ESA.

Différence entre OID numériques et noms symboliques

OID numériques :

- Ils sont universels et fonctionnent toujours, même si les fichiers MIB ne sont pas chargés sur le système.
- Exemple : 2.1.3.6.1.4.1.15497.1.1.1.2
- Ils sont moins lisibles et peuvent être difficiles à mémoriser.

Noms symboliques :

- Il s'agit de noms conviviaux définis dans les fichiers MIB, tels que perCentCPUUtilization.
- Elles facilitent l'écriture et la compréhension des commandes.
- Ils nécessitent le chargement correct des fichiers MIB et la configuration de la variable d'environnement MIBS.
- Exemple : snmpwalk -v2c -c ironport 10.31.124.165 perCENTtilization.

C'est pareil ?

Les deux méthodes interrogent la même métrique et produisent des résultats identiques, mais les noms symboliques sont plus pratiques et lisibles par l'homme, tandis que les OID numériques sont plus fiables dans les environnements où les fichiers MIB ne peuvent pas être présents ou chargés.

Informations connexes

- Surveillance de l'état du système via SNMP
- Assistance technique de Cisco et téléchargements

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.