Configurer le pousser SCP des logins ESA de messagerie

Contenu

Introduction

Ce document décrit comment installer et configurer le pousser sécurisé de copie (SCP) de la messagerie se connecte (ou d'autres types de log) d'une appliance de sécurité du courrier électronique de Cisco (ESA) à un serveur externe de Syslog.

Informations générales

Un administrateur peut recevoir des notifications d'erreur déclarant que des logs ne peuvent pas être poussés utilisant le SCP, ou il peut y avoir des journaux des erreurs énonçant la non-concordance de clés.

Conditions préalables

Sur le serveur de Syslog que l'ESA des fichiers journal SCP à :

1. Assurez que le répertoire à utiliser est disponible.
2. Passez en revue « /etc/ssh/sshd_config » pour les configurations d'AuthorizedKeysFile. Ceci indique le SSH recevoir des authorized_keys et regarder dans le répertoire home de l'utilisateur pour la piqûre de key_name écrite dans .ssh/authorized_keys classez :
   ```
   AuthorizedKeysFile      %h/.ssh/authorized_keys
   ```
3. Vérifiez les autorisations du répertoire d'être utilisé. Vous pouvez devoir apporter des modifications d'autorisations : Des autorisations sur « $HOME » est placées à 755.Des autorisations sur « $HOME/.ssh » est placées à 755.Des autorisations sur « $HOME/.ssh/authorized_keys » est placées à 600.

Restrictions de niveau et autorisations de fichier sur UNIX/Linux

Il y a trois types de restrictions d'accès :

```
Permission Action chmod option=====================================read (view) r or 4write (edit) w or 2execute (execute) x or 1
```

Il y a également trois types de restrictions d'utilisateur :

```
User ls output=================owner -rwx------group ----rwx---other -------rwx
```

Autorisations de répertoire/répertoire :

```
Permission Action chmod option=============================================================read (view contents: i.e., ls command) r or 4write (create or remove files from dir) w or 2execute (cd into directory) x or 1
```

Notation numérique :

Une autre méthode pour représenter des autorisations de Linux est une notation octale comme affichée par `stat - c %a`. Cette notation se compose au moins de trois chiffres. Chacun des trois chiffres de droite représente un composant différent des autorisations : propriétaire, groupe, et d'autres.

Chacun de ces chiffres est la somme de ses bits composants dans le système de chiffre binaire :

```
Symbolic Notation Octal Notation English=============================================================---------- 0000 no permissions---x--x--x 0111 execute--w--w--w- 0222 write--wx-wx-wx 0333 write & execute-r--r--r-- 0444 read-r-xr-xr-x 0555 read & execute-rw-rw-rw- 0666 read & write-rwxrwxrwx 0777 read. write & execute
```

Pour l'étape #3, la recommandation de placer le répertoire $HOME à 755 serait : 7=rwx `5=r-x 5=r-x`

Ceci signifie que le répertoire a les autorisations par défaut `- rwxr-xr-x` (représenté dans la notation octale en tant que 0755).

Configurer le pousser SCP des logins ESA de messagerie

1. Exécutez le **logconfig de** commande CLI.
2. Sélectionnez l'option **nouvelle**.
3. Choisissez le type de fichier journal pour cet abonnement, ce sera "1" pour des logs de messagerie des textes d'IronPort, ou n'importe quels autres types de fichier journal de votre choix.
4. Écrivez le nom pour le fichier journal.
5. Sélectionnez le niveau approprié de log. Typiquement vous devriez sélectionner "3" pour informationnel, ou tout autre niveau de log de votre choix.
6. Une fois incité « choisissez la méthode pour récupérer les logs », sélectionnez "3" pour le **pousser SCP**.
7. Entrez dans l'adresse IP ou l'adresse Internet de DN pour livrer les logs à.
8. Entrez dans le port pour se connecter à sur le serveur distant.
9. Écrivez le répertoire sur le serveur distant pour placer des logs.
10. Entrez dans un nom du fichier pour l'utiliser pour des fichiers journal.
11. Configurez, si dû, les identifiants uniques basés sur l'étude des systèmes comme

$hostname, *$serialnumber* s'ajouter au nom du fichier de log.

12. Placez le maximum filesize avant de transférer.
13. Configurez le renversement basé sur temps des fichiers journal, si c'est approprié.
14. Une fois demandé « faites vous veulent activer vérifier de clé de hôte ? », écrivez « Y ».
15. Vous êtes alors présenté « placez s'il vous plaît le ssh key suivant dans votre fichier d'authorized_keys de sorte que les fichiers journal puissent être téléchargés. »
16. Copiez cette clé, car vous devrez mettre le ssh key dans votre fichier de « authorized_keys » sur le serveur de Syslog. Collez la clé donnée du logconfig au fichier $HOME/.ssh/authorized_keys sur le serveur de Syslog.
17. De l'ESA, exécutez la **validation de** commande CLI pour sauvegarder et commettre des modifications de configuration.

La configuration du log peut également faire du GUI : **Abonnements d'administration système > de log**

> Remarque: Veuillez passer en revue le chapitre se connectant du guide utilisateur ESA pour les détails et les informations supplémentaires complets.

Confirmation

Hostkeyconfig

Exécutez le **logconfig >** le **hostkeyconfig de** commande. Vous devriez voir une entrée pour le serveur de Syslog configuré répertorié en tant que « ssh-DSS » avec un semblable principal abrégé à la clé fournie pendant la configuration.

```
myesa.local > logconfig
...
[]> hostkeyconfig

Currently installed host keys:
1. 172.16.1.100 ssh-dss AAAAB3NzaC1kc3MAAACBAMUqUBGztO0T...OutUns+DY=
```

Logs système

Les logs système enregistrent ce qui suit : démarrez les informations, des alertes virtuelles d'expiration de la licence d'appareils, les informations d'état de DN, et commentez des utilisateurs tapés utilisant la commande de validation. Les logs système sont utiles pour dépanner l'état de base de l'appliance.

Exécuter les **system_logs de queue de** commande du CLI te fournira un aspect vivant à l'état du système.

Vous pouvez également choisir le **rollovernow de** commande CLI et sélectionner le numéro associé au fichier journal. Vous verrez ceci le fichier journal SCP à votre serveur de Syslog dans les system_logs :

```
myesa.local > tail system_logs

Press Ctrl-C to stop.
Thu Jan 5 11:26:02 2017 Info: Push success for subscription mail_logs: Log
mail_logs.myesa.local.@20170105T112502.s pushed via SCP to remote host 172.16.1.100:22
```

Dépannage avancé

S'il y a les questions continues avec la Connectivité au serveur de Syslog, de l'hôte local et ssh d'utilisation, exécutez le « ssh testuser@hostname - v » pour tester l'accès client en mode bavard. Ceci peut dépannage d'aide afficher où la connexion de ssh ne réussit pas.

```
$ ssh testuser@172.16.1.100 -v
OpenSSH_7.3p1, LibreSSL 2.4.1
debug1: Reading configuration data /Users/testuser/.ssh/config
debug1: /Users/testuser/.ssh/config line 16: Applying options for *
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 20: Applying options for *
debug1: Connecting to 172.16.1.100 [172.16.1.100] port 22.
debug1: Connection established.
debug1: identity file /Users/testuser/.ssh/id_rsa type 1
debug1: key_load_public: No such file or directory
debug1: identity file /Users/testuser/.ssh/id_rsa-cert type -1
debug1: identity file /Users/testuser/.ssh/id_dsa type 2
debug1: key_load_public: No such file or directory
debug1: identity file /Users/testuser/.ssh/id_dsa-cert type -1
debug1: key_load_public: No such file or directory
debug1: identity file /Users/testuser/.ssh/id_ecdsa type -1
debug1: key_load_public: No such file or directory
debug1: identity file /Users/testuser/.ssh/id_ecdsa-cert type -1
debug1: key_load_public: No such file or directory
debug1: identity file /Users/testuser/.ssh/id_ed25519 type -1
debug1: key_load_public: No such file or directory
debug1: identity file /Users/testuser/.ssh/id_ed25519-cert type -1
debug1: Enabling compatibility mode for protocol 2.0
debug1: Local version string SSH-2.0-OpenSSH_7.3
debug1: Remote protocol version 2.0, remote software version OpenSSH_6.6.1p1 Ubuntu-2ubuntu2.8
debug1: match: OpenSSH_6.6.1p1 Ubuntu-2ubuntu2.8 pat OpenSSH_6.6.1* compat 0x04000000
debug1: Authenticating to 172.16.1.100:22 as 'testuser'
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: algorithm: curve25519-sha256@libssh.org
debug1: kex: host key algorithm: ssh-dss
debug1: kex: server->client cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression:
zlib@openssh.com
debug1: kex: client->server cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression:
zlib@openssh.com
debug1: expecting SSH2_MSG_KEX_ECDH_REPLY
debug1: Server host key: ssh-dss SHA256:c+YpkZsQyUwi3tkIVJFXHAstwlkdewO1G0s7P2khV7U
debug1: Host '172.16.1.100' is known and matches the DSA host key.
debug1: Found key in /Users/testuser/.ssh/known_hosts:5
debug1: rekey after 134217728 blocks
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug1: rekey after 134217728 blocks
debug1: SSH2_MSG_NEWKEYS received
debug1: Skipping ssh-dss key /Users/testuser/.ssh/id_dsa - not in PubkeyAcceptedKeyTypes
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug1: Authentications that can continue: publickey,password
debug1: Next authentication method: publickey
debug1: Offering RSA public key: /Users/testuser/.ssh/id_rsa
debug1: Authentications that can continue: publickey,password
debug1: Trying private key: /Users/testuser/.ssh/id_ecdsa
debug1: Trying private key: /Users/testuser/.ssh/id_ed25519
debug1: Next authentication method: password
```

```
testuser@172.16.1.100's password: <<< ENTER USER PASSWORD TO LOG-IN >>>
debug1: Enabling compression at level 6.
debug1: Authentication succeeded (password).
Authenticated to 172.16.1.100 ([172.16.1.100]:22).
debug1: channel 0: new [client-session]
debug1: Requesting no-more-sessions@openssh.com
debug1: Entering interactive session.
debug1: pledge: exec
debug1: No xauth program.
Warning: untrusted X11 forwarding setup failed: xauth key data not generated
debug1: Requesting authentication agent forwarding.
debug1: Sending environment.
debug1: Sending env LANG = en_US.UTF-8
debug1: Sending env LC_CTYPE = en_US.UTF-8
```