

Téléchargement de Repo d'Ubuntu 16.04 à l'appliance locale de Repo utilisant le cryptage cloudrepo.key SHA256

Contenu

[Conditions requises](#)

Introduction

Ce document décrit comment au sync une appliance locale de Repo avec repo.cliqrtech.com afin d'obtenir le repo Ubuntu1604 téléchargé à l'appliance de Repo utilisant le chiffrement à clé SHA256.

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Interface de Linux
- Environnement de virtual machine
- Appliance de Repo

Composants utilisés

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

- Version 4.8.1.1 de CloudCenter
- Appliance de Repo

Informations générales

CloudCenter a commencé prendre en charge Ubuntu16.04 démarrant la release 4.8.1.1 mais le repo Ubuntu16.04 n'est pas disponible dans l'appliance de Repo de CloudCenter. Par conséquent, afin de télécharger le repo d'Ubuntu 16.04, l'appliance locale de Repo doit syncer avec le repo.cliqrtech.com

Problème

Si les syncs locaux d'appareils de Repo avec `repo.cliqrtech.com`, il ne pourront pas obtenir correctement le repo d'Ubuntu 16.04 parce que le cryptage `cloudrepo.key` est SHA1 et Ubuntu 16.04 exige au moins un SHA256.

Solution

Une nouvelle clé de GnuPrivacyGuard (GPG) doit être créée utilisant SHA256 afin d'avoir `cloudrepo.key` qui téléchargera efficacement le repo d'Ubuntu 16.04 à l'appliance locale de Repo.

Étape 1. Assurez-vous que `repo.cliqrtech.com` est le référentiel principal.

```
/usr/bin/repo_config_wizard.sh
```

Étape 2. Dans le répertoire local de `/tmp` d'appareils de Repo, copiez et collez le script suivant et nommez-le `recreate_gpg_key.sh` [le script est relié en cette TZ].

```
#!/bin/bash

REPO_DIR='/repo'

# Move gnupg folder to recreate keys
if [[ -d '/home/repo/.gnupg' ]]; then
    su repo -c "mv -f /home/repo/.gnupg /home/repo/gnupg_bkp"
fi

# Create gpg.conf file and add sha256 algorithm to it
gpg_conf_file='/home/repo/.gnupg/gpg.conf'
su repo -c "gpg --list-keys"
if [[ ! -f ${gpg_conf_file} ]]; then
    echo "Gpg.conf file not created. Failing in gpg install/configure"
    exit 1
fi
echo cert-digest-algo SHA256 >> ${gpg_conf_file}
echo digest-algo SHA256 >> ${gpg_conf_file}

gpg_txt='/tmp/gpg.txt'

cat << EOF > ${gpg_txt}
%echo Generating a basic OpenPGP key
Key-Type: RSA
Key-Length: 4096
Name-Real: CloudRepo
Name-Comment: GPG key for Cloud Repo
Name-Email: foo@foo.bar
Expire-Date: 0
Passphrase: cloudrepo
# Do a commit here, so that we can later print "done" :-)
%commit
%echo done
EOF

su repo -c "gpg --batch --gen-key ${gpg_txt}"
```

```

rm -f ${gpg_txt}

# Remove cloudrepo key before recreating it
cloud_repo_key=${REPO_DIR}/cloudrepo.key'
if [[ -f ${cloud_repo_key} ]]; then
    rm -f ${cloud_repo_key}
fi
gpg_key_val=$(su repo -c "gpg --list-keys" | grep "^pub" | tail -n 1 | awk '{print $2}' |
cut -d '/' -f 2 )

gpg_tmp_file='/tmp/gpg.tmp'
echo cloudrepo > ${gpg_tmp_file}
su repo -c "cat ${gpg_tmp_file} | gpg --no-tty --batch --passphrase-fd 0 --output
${cloud_repo_key} --armor --export ${gpg_key_val}"
rm -f ${gpg_tmp_file}

# Run rebuild metadata script to recreate ubuntu1604 metadata with the new key
touch /repo/debRepo/ubuntu1604/amd64/binary/tmp_file
touch /repo/debRepo/ubuntu1404/amd64/binary/tmp_file
touch /repo/debRepo/ubuntu1204/amd64/binary/tmp_file
su repo -c "/repo/scripts/rebuild_repo_metadata.sh"
rm -f /repo/debRepo/ubuntu1604/amd64/binary/tmp_file
rm -f /repo/debRepo/ubuntu1404/amd64/binary/tmp_file
rm -f /repo/debRepo/ubuntu1204/amd64/binary/tmp_file

exit 0

```

Étape 3. Change l'autorisation de recreate_gpg_key.sh.

```
chmod 755 recreate_gpg_key.sh
```

Étape 4. Exécutez recreate_gpg_key.sh comme racine.

```
./recreate_gpg_key.sh
```

Étape 5. Confirmez que le repo d'Ubuntu 16.04 a été ajouté à l'appliance locale de Repo.

```
ls /repo/debRepo/
```