

# Transferência do Repo de Ubuntu 16.04 ao dispositivo local do Repo usando a criptografia cloudrepo.key SHA256

## Índice

[Requisitos](#)

## Introdução

Este documento descreve como à sincronização um dispositivo local do Repo com repo.cliqrtech.com a fim obter o repo Ubuntu1604 transferido ao dispositivo do Repo usando a criptografia chave SHA256.

## Pré-requisitos

### Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Relação de Linux
- Ambiente da máquina virtual
- Dispositivo do Repo

## Componentes Utilizados

As informações neste documento são baseadas nestas versões de software:

- Versão 4.8.1.1 de CloudCenter
- Dispositivo do Repo

## Informações de Apoio

CloudCenter começou apoiar Ubuntu16.04 que começa a liberação 4.8.1.1 mas o repo Ubuntu16.04 não está disponível no dispositivo do Repo de CloudCenter. Conseqüentemente, a fim transferir o repo de Ubuntu 16.04, o dispositivo local do Repo precisa de ser sincronizado acima com o repo.cliqrtech.com

# Problema

Se as sincronizações locais do dispositivo do Repo acima com `repo.cliqrtech.com`, ele não poderão obter corretamente o repo de Ubuntu 16.04 porque a criptografia `cloudrepo.key` é SHA1 e Ubuntu 16.04 exigem pelo menos um SHA256.

## Solução

Uma chave nova de GnuPrivacyGuard (GPG) precisa de ser criada usando SHA256 a fim ter `cloudrepo.key` que transferirá eficazmente o repo de Ubuntu 16.04 ao dispositivo local do Repo.

Etapa 1. Certifique-se de que `repo.cliqrtech.com` é o repositório mestre.

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

Etapa 2. No diretório local de `/tmp` do dispositivo do Repo, copie e cole o seguinte script e nomeie-o [script is attached to this TZ] de `recreate_gpg_key.sh`.

```
#!/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

```

**Etapa 3.**Change a permissão de recreate\_gpg\_key.sh.

```
chmod 755 recreate_gpg_key.sh
```

**Etapa 4.** Execute recreate\_gpg\_key.sh como a raiz.

```
./recreate_gpg_key.sh
```

**Etapa 5.** Confirme que o repo de Ubuntu 16.04 esteve adicionado ao dispositivo local do Repo.

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