How to Download Ubuntu 16.04 Repo to Local Repo Appliance with cloudrepo.key SHA256 Encryption

Contents

Introduction

This document describes how to synchronize a local Repo Appliance with the repo.cliqrtech.com in order to download Ubuntu 16.04 Repo to the Repo Appliance with the use of cloudrepo.key SHA256 encryption.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Linux Interface
- Virtual Machine Environment
- Repo Appliance

Components Used

The information in this document is based on these software versions:

- CloudCenter version 4.8.1.1
- Repo Appliance

Background Information

CloudCenter started to support Ubuntu 16.04 with the release of 4.8.1.1. However, the Ubuntu 16.04 repo is not available in the CloudCenter Repo Appliance.

Therefore, if you want to download the Ubuntu 16.04 repo, synchronize the local Repo Appliance with repo.ciqrtech.com.

Problem

Synchronization of Ubuntu 16.04 Repo Download to Local Repo Appliance

If the local Repo Appliance synchronizes up with the repo.cliqrtech.com, it is not able to properly obtain the Ubuntu 16.04 repo because of the cloudrepo.key encryption is SHA1 and Ubuntu 16.04 requires at least a SHA256.

Solution

In order to have cloudrepo.key, a new GnuPrivacyGuard (GPG) key is to be created with the use of SHA256, which effectively downloads the Ubuntu 16.04 repo to the local Repo Appliance.

Step 1. Ensure that repo.cliqrtech.com is the master repository.

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

Step 2. In the local Repo Appliance /tmp directory, copy and paste this script and name it **recreate_gpg_key.sh** (script is attached to this Techzone article).

```
#!/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 algorith 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
```

Step 3.Change the **recreate_gpg_key.sh** permission.

```
chmod 755 recreate_gpg_key.sh
```

Step 4. Execute **recreate_gpg_key.sh** as root.

```
 ./recreate_gpg_key.sh
```

Step 5. Confirm that Ubuntu 16.04 repo is added to the local Repo Appliance.

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