ISO and RPM Image Verification

Introduction

Starting from Cisco IoT FND 4.9.0, you can verify the ISO and RPM images before the installation or upgrade of IoT FND.

For more information, refer to How to Run the Signature Verification Program, on page 2.

Table 1: IoT FND ISO Image Zip File Contents (iot-fnd-<release>-<build number>-signed)

Zip File Contents	Description
1. iot-fnd-<release>-<build number>.iso	Cisco provided image for which signature is to be verified.
2. iot-fnd-<release>-<build number>.iso.signature	Signature generated for the image.
3. FND_RPM_SIGN-CCO_RELEASE.pem	Cisco signed x.509 end-entity certificate contains the public key that is used to verify the signature. This certificate is chained to Cisco root CA and sub CA posted on https://www.cisco.com/security/pki/
4. cisco_x509_verify_release.py	Signature verification program. After downloading the image, its digital signature, and the x.509 certificate, this program is used to verify the 3-tier x.509 certificate chain and the signature. Certificate chain validation is done by verifying the authenticity of end-entity using Cisco-sourced sub CA and root CA (which the script downloads from Cisco).
5. cisco_x509_verify_release.py.signature	Signature generated for the script cisco_x509_verify_release.py.
6. cisco_openpgp_verify_release.py	Signature verification program for verifying the Open-pgp Complaint Public Key against x.509 end-entity certificate.

Zip File Contents	Description
7. cisco_openpgp_verify_release.py.signature	Signature generated for the script cisco_openpgp_verify_release.py.
8. FND-rel-binary.gpg	Open-pgp public key is used for verification of the signed RPM.
9. FND-rel-ascii.gpg	Open-pgp public key is used for verification of the signed RPM.

How to Run the Signature Verification Program

Prerequisites:

- Python 2.7.x
- OpenSSL
- Verification scripts running on customer-premises need internet connectivity to reach Cisco to download root and sub-CA certs

To run a signature verification program:

Step 1 Unzip the file `iot-fnd-<release>-<build number>-signed.zip` and `cd` to the folder `iot-fnd-<release>-<build number>-signed`.

Step 2 Extract the public key from the public cert:

```
openssl x509 -pubkey -noout -in FND_RPM_SIGN-CCO_RELEASE.pem > FND-EE-cert.pubkey
```

Expected Result:

```
FND-EE-cert.pubkey is created under the same folder
```

Step 3 Verify the verification scripts using the public key and signature files.

a) `openssl dgst -sha512 -verify FND-EE-cert.pubkey -signature cisco_x509_verify_release.py.signature cisco_x509_verify_release.py`

Expected Result:

```
Verified OK
```

b) `openssl dgst -sha512 -verify FND-EE-cert.pubkey -signature cisco_openpgp_verify_release.py.signature cisco_openpgp_verify_release.py`

Expected Result:

```
Verified OK
```

Step 4 Verify the ISO file.

```
./cisco_x509_verify_release.py -e FND_RPM_SIGN-CCO_RELEASE.pem -s
iot-fnd-<release>-<build number>.iso.signature -i iot-fnd-<release>-<build
number>.iso -v dgst -sha512
```

Expected Result:

```
Retrieving CA certificate from http://www.cisco.com/security/pki/certs/crcam2.cer ...

Successfully retrieved and verified crcam2.cer.

Retrieving SubCA certificate from http://www.cisco.com/security/pki/certs/innerspace.cer ...

Successfully retrieved and verified innerspace.cer.

Successfully verified root, subca and end-entity certificate chain.

Successfully fetched a public key from FND_RPM_SIGN-CCO_RELEASE.pem.

Successfully verified the signature of iot-fnd-<release>-<build number>.iso using
FND_RPM_SIGN-CCO_RELEASE.pem
```

Step 5 Install the ISO image file.

```
 cd /mnt

mkdir iso

mount -t iso9660 -o loop <path>/iot-fnd-<release>-<build number>.iso /mnt/iso

mkdir /tmp/ISO

cp -pRf /mnt/iso /tmp/ISO

umount /mnt/iso/
```

Step 6 Verify if the delivered binary and ASCII keys have matching fingerprints.

a) `gpg FND-rel-binary.gpg`

Expected Result:

```
pub  2048R/F7D5ED29 2017-01-01 identity-name (FND.rel) identity-name@cisco.com
```

b) `gpg FND-rel-ascii.gpg`

Expected Result:

```
pub 2048R/F7D5ED29 2017-01-01 identity-name (FND.rel) identity-name@cisco.com
```

Step 7 Verify the binary GPG key against the EE cert.

```
./cisco_openpgp_verify_release.py -e FND_RPM_SIGN-CCO_RELEASE.pem -G
FND-rel-binary.gpg
```

Expected Result:

```
Downloading CA certificate from http://www.cisco.com/security/pki/certs/crcam2.cer ...

Successfully downloaded crcam2.cer.

Downloading SubCA certificate from http://www.cisco.com/security/pki/certs/innerspace.cer ...

Successfully downloaded innerspace.cer.

Successfully verified Cisco root, subca and end-entity certificate chain.

Successfully fetched a public key from FND_RPM_SIGN-CCO_RELEASE.pem.

Successfully authenticated FND-rel-binary.gpg key using Cisco X.509 certificate trust chain.
```

Step 8 Verify the RPM Signature using the GPG ASCII key.

```
sudo rpm --import FND-rel-ascii.gpg
```

```
rpm -K /tmp/ISO/iso/cgms-<release>-<build number>.x86_64.rpm
```

Expected Result:

```
/tmp/ISO/iso/cgms-<release>-<build number>.x86_64.rpm: rsa sha1 (md5) pgp md5 OK
```

Step 9 Repeat Step 8 for all the RPMs. Once the RPM is verified, you can install or upgrade the RPM.