oVirt - PKI

Alon Bar-Lev

Red Hat

2012-10-17

Ovirt PKI

- Back-end purposes
  - Application Server TLS/SSL (Server identification)
  - VDSM authentication (Client authentication)
  - SSH authentication (PK) (Client authentication)
- Host purposes
  - VDSM TLS/SSL (Server identification)
- Trust Anchor
  - Internal CA

PKI Artifacts - Back-end

- /etc/pki/ovirt-engine/
    - ca.pem – Trust anchor
    - private
        - ca.pem – CA private key
    - keys
        - engine_id_rsa – K1 to be used for ssh authentication
        - engine.ssh.key.txt – public(K1)
    - certs
        - ca.der – Trust anchor
        - engine.[cd]er – certificate(K1)
        - *.pem – issued certificates
    - .truststore – Java trust anchor
    - .keystore – Java material: certificate(K1), K1
    - database.txt, serial.txt – Standard OpenSSL CA
- Http
    - http://<server>/ca.crt
    - http://<server>/engine.ssh.key.txt

PKI Artifacts - Host

- /etc/pki
  - vdsm
    - keys - for VDSM and libvirtd
      - vdsmkey.pem – K2
    - certs – for VDSM and libvirtd
      - cacert.pem – Trust Anchor
      - vdsmcert.pem – certificate(K2)
    - libvirt-spice – for spice TLS/SSL
      - ca-cert.pem -> ../certs/cacert.pem
      - server-cert.pem -> ../certs/vdsmcert.pem
      - server-key.pem -> ../keys/vdsmkey.pem
  - libvirt – for libvirt
    - clientcert.pem -> ../vdsm/certs/vdsmcert.pem
    - private/clientkey.pem -> ../../vdsm/keys/vdsmkey.pem
  - CA – for libvirt
    - cacert.pem -> ../vdsm/certs/cacert.pem
- /root/.ssh

PKI Sequences

- Installation
  - Create CA
  - Generate engine key
  - Extract SSH key out of engine key
  - Setup Apache/JBoss with engine key
  - Logs are in installation logs
- Enrollment
  - Sign certificate requests of VDSM.
  - Logs are in engine logs.

CA Certificate

```
Serial Number: 1 (0x1)
Issuer: C=US, O=tlv.redhat.com, CN=CA-dhcp-1-191.tlv.redhat.com.69783
Validity
    Not Before: Oct  9 17:45:15 2012
    Not After : Oct  8 17:45:16 2022 GMT
Subject: C=US, O=tlv.redhat.com, CN=CA-dhcp-1-191.tlv.redhat.com.69783
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
        Public-Key: (2048 bit)
X509v3 extensions:
    X509v3 Subject Key Identifier:
        15:7B:BD:02:3D:15:04:F1:82:46:51:D5:87:AB:67:DF:AF:75:B2:30
    Authority Information Access:
        CA Issuers - URI:http://dhcp-1-191.tlv.redhat.com:80/ca.crt
    X509v3 Authority Key Identifier:
        keyid:15:7B:BD:02:3D:15:04:F1:82:46:51:D5:87:AB:67:DF:AF:75:B2:30
        DirName:/C=US/O=tlv.redhat.com/CN=CA-dhcp-1-191.tlv.redhat.com.69783
        serial:01
    X509v3 Basic Constraints: critical
        CA:TRUE
    X509v3 Key Usage: critical
        Certificate Sign, CRL Sign
```

Engine Certificate

```
Serial Number: 2 (0x2)
Signature Algorithm: sha1WithRSAEncryption
Issuer: C=US, O=tlv.redhat.com, CN=CA-dhcp-1-191.tlv.redhat.com.69783
Validity
    Not Before: Oct  9 17:45:15 2012
    Not After : Sep 14 17:45:17 2017 GMT
Subject: C=US, O=tlv.redhat.com, CN=dhcp-1-191.tlv.redhat.com
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
        Public-Key: (1024 bit)
X509v3 extensions:
    X509v3 Subject Key Identifier:
        FD:14:1B:31:A7:DF:FC:0B:88:08:92:73:82:5B:55:0B:E0:4D:96:67
    Authority Information Access:
        CA Issuers - URI:http://dhcp-1-191.tlv.redhat.com:80/ca.crt
    X509v3 Authority Key Identifier:
        keyid:15:7B:BD:02:3D:15:04:F1:82:46:51:D5:87:AB:67:DF:AF:75:B2:30
        DirName:/C=US/O=tlv.redhat.com/CN=CA-dhcp-1-191.tlv.redhat.com.69783
        serial:01
    X509v3 Basic Constraints:
        CA:FALSE
    X509v3 Key Usage: critical
        Digital Signature, Key Encipherment
    X509v3 Extended Key Usage: critical
        TLS Web Server Authentication, TLS Web Client Authentication
```

VDSM Certificate

```
Serial Number: 3 (0x3)
Signature Algorithm: sha1WithRSAEncryption
Issuer: C=US, O=tlv.redhat.com, CN=CA-dhcp-1-191.tlv.redhat.com.69783
Validity
    Not Before: Oct  9 18:15:56 2012
    Not After : Oct  9 18:15:57 2017 GMT
Subject: O=tlv.redhat.com, CN=10.35.1.114
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
        Public-Key: (2048 bit)
```

Useful commands

- Display certificate

  -
    ```
    openssl x509 -in @CERT@ -text
    ```
- Display key

  -
    ```
    openssl rsa -in @KEY@
    ```
- Manipulate Java Key Store

  -
    ```
    keytool -help
    ```

Current Implementation Issues

- Implementation does not take professional services into account. No human interface.
- Implementation assumes single trust anchor.
- Implementation uses same key and certificate for both client authentication and server authentication.
- No separation between user visible PKI (web) and internal PKI (ssh, VDSM).
- Implementation uses Java proprietary formats.
- No [supported] ability to replace CA with different implementation.

PKI: Glance to 3.2

- Use separate keys for web TLS/SSL and engine authentication.
- Support separate trust anchor for TLS/SSL.
- Drop the Java proprietary formats in favor of standard PKCS#12 format.

PKI

- Questions?