IdP key rollover

Introduction

The term key rollover refers to a process whereby one key is systematically replaced by another key in SAML metadata. Since SAML entities (and therefore SAML metadata) are distributed, key rollover must be deliberate, so as not to break the key operations of a relying party.

The general process of rolling over a metadata key in an IdP without causing unnecessary downtime is as follows. The following describes the process in detail for Shibboleth IdPs. The information can be used as inspiration for other IdP implementations.

1. Create a new key pair for signing metadata
2. Add the new KeyDescriptor to your metadata
3. Send the new metadata to Operations and wait for the newly updated metadata to propagate throughout the Federation
4. Reconfigure the IdP software to use the new key (instead of the old key) as the signing key
5. Remove the old KeyDescriptor from your metadata and send the new metadata to Operations

If you need to replace the front end SSL/TLS key, you do not need to send anything to Operations.

Shibboleth IdP metadata key roller suitable for both Apache HTTP or Apache Tomcat as front end.

The following is designed to work on in a Linux environment. Read carefully before proceeding so that you understand the process. Several restarts of tomcat and/or apache are required. Backup your entire Shibboleth installation (and snapshot your server if possible) before proceeding.

NOTE WELL: The following instructions are pretty much untested at present and assumes that you are running Shibboleth IdP 2.3 or later.

Change into the IdP distribution directory, shibboleth-identityprovider-VERSION. *This is the directory you created when you installed or last updated the IdP*. Run the following using the appropriate user that can write to your Shibboleth installation

```
export IdPCertLifetime=3
```

```
./install.sh renew-cert
```

Respond to the prompts appropriately. The new private key, long lived certificate, and keystore files will be generated with the file name suffix '.new'.

Backup your metadata file (adjust the paths to reflect your own installation)

```
sudo cp /opt/shibboleth-idp/metadata/idp-metadata.xml /opt/shibboleth-idp/metadata/idp-metadata.xml.bak.0
```

Paste the new certificate in, omitting the -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----- lines. Do this two times, once in the IDPSSODescriptor and once in the AttributeAuthorityDescriptor element.

```
<KeyDescriptor>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>

      </ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</KeyDescriptor>
```

Restart your IdP, Send the metadata file to operations@swamid.se. Wait for operations to add the metadata into the appropriate stream.

Once the stream has updated (allow 24 hours), switch the metadata used by the IdP. Backup the old files.

```
sudo mv /opt/shibboleth-idp/credentials/idp.key /opt/shibboleth-idp/credentials/idp.key.bak.0
sudo mv /opt/shibboleth-idp/credentials/idp.crt /opt/shibboleth-idp/credentials/idp.crt.bak.0
sudo mv /opt/shibboleth-idp/credentials/idp.jks /opt/shibboleth-idp/credentials/idp.jks.bak.0
```

Activate the new files

```
sudo mv /opt/shibboleth-idp/credentials/idp.key.new /opt/shibboleth-idp/credentials/idp.key
sudo mv /opt/shibboleth-idp/credentials/idp.crt.new /opt/shibboleth-idp/credentials/idp.crt
sudo mv /opt/shibboleth-idp/credentials/idp.jks.new /opt/shibboleth-idp/credentials/idp.jks
```

If you use tomcat alone, make sure it can access the idp.jks file. If you have an Apache front, check the VirtualHost configuration for port 8443 can access the key and crt files. Update your Tomcat/Apache configuration if you have changed the passphrase. Restart Apache and/or Tomcat and test. If everything works, you can remove the old certificate from the IdP's metadata and re-submit the metadata to operations@swamid.se so that the old certificate is removed from the stream.