Key rollover on Shibboleth SP

This page describes the process of certificate rollover for Shibboleth Identity Providers. The procedure described below allows replacing certificates without any service disruptions.

Shibboleth SP 3 default installation have both an Encryption and a Signing certificate. Older installations might have one certificate used for both Encryption and Signing.

Step 0 : Create new certificate

shib-keygen creates by default a 3072 bits key. To get 4096 bits you have to manually edit the file `/usr/sbin/shib-keygen` and replace

```
[req]
prompt=no
default_bits=3072
encrypt_key=no
default_md=sha256
```

with

```
[req]
prompt=no
default_bits=4096
encrypt_key=no
default_md=sha256
```

To generate a new keypair and self-signed certificate for the IdP, run the following commands as root user:

```
sudo -s

cd /etc/shibboleth/certs

# Save old encrytion cert
mv sp-encrypt-cert.pem sp-encrypt-cert-old.pem
mv sp-encrypt-key.pem sp-encrypt-key-old.pem

# Create new certs
# Signing will be activated later but we need it to update Metadata
shib-keygen -n sp-signing-new
# Encryption will be activated directly
shib-keygen -n sp-encrypt
```

With the above commands a new certificates and private keys are generated inside the `/etc/shibboleth/certs` directory.

Step 1 : Add key to Shibboleth

Edit `/etc/shibboleth/shibboleth2.xml` and add part for old encryption key. Needed during rollover until all IdP:s have picked up the new key.

Before	After

`<CredentialResolver type="File" use="signing"`
`  key="certs/sp-signing-key.pem"`
`  certificate="certs/sp-signing-cert.pem"/>`
 	
`<CredentialResolver type="File" use="encryption"`
`  key="certs/sp-encrypt-key.pem"`
`  certificate="certs/sp-encrypt-cert.pem"/>`
	`<CredentialResolver type="File" use="signing"`
	`  key="certs/sp-signing-key.pem"`
	`  certificate="certs/sp-signing-cert.pem"/>`
	
	`<CredentialResolver type="File" use="encryption"`
	`  key="certs/sp-encrypt-key.pem"`
	`  certificate="certs/sp-encrypt-cert.pem"/>`
	
	`<CredentialResolver type="File" use="encryption"`
	`  key="certs/sp-encrypt-key-old.pem"`
	`  certificate="certs/sp-encrypt-cert-old.pem"/>`

Test config and if no problems appears restart service

```
sudo -s

# Test config
/usr/sbin/shibd -tc /etc/shibboleth/shibboleth2.xml

service shibd restart
```

Now the SP supports both new and old encryption certs for incoming traffic but still uses old signing-key for signing outgoing.

Step 2 : Upload new Metadata

> **https://sp.exaple.com/Shibboleth.sso/Metadata is NOT correct**
>
> Note that the metadata is generated based on the config in shibboleth2.xml as is not what we want to publish. The generated is now showing new and old encryption + old signing, we want new encryption + new and old signing.

First we need to update our XML and replace the encryption certificate and add the new signing certificate.

Download the XML from metadata.swamid.se and edit

Replace	With

`<md:KeyDescriptor use="encryption">`
`  <ds:KeyInfo>`
`  <ds:X509Data>`
`  <ds:X509Certificate>Old cert</ds:X509Certificate>`
`  </ds:X509Data>`
`  </ds:KeyInfo>`
`</md:KeyDescriptor>`
`<md:KeyDescriptor use="signing">`
`  <ds:KeyInfo>`
`  <ds:X509Data>`
`  <ds:X509Certificate>Old cert</ds:X509Certificate>`
`  </ds:X509Data>`
`  </ds:KeyInfo>`
`</md:KeyDescriptor>`
	`<md:KeyDescriptor use="encryption">`
	`  <ds:KeyInfo>`
	`  <ds:X509Data>`
	`  <ds:X509Certificate>New cert</ds:X509Certificate>`
	`  </ds:X509Data>`
	`  </ds:KeyInfo>`
	`</md:KeyDescriptor>`
	`<md:KeyDescriptor use="signing">`
	`  <ds:KeyInfo>`
	`  <ds:X509Data>`
	`  <ds:X509Certificate>New cert</ds:X509Certificate>`
	`  </ds:X509Data>`
	`  </ds:KeyInfo>`
	`</md:KeyDescriptor>`
	`<md:KeyDescriptor use="signing">`
	`  <ds:KeyInfo>`
	`  <ds:X509Data>`
	`  <ds:X509Certificate>Old cert</ds:X509Certificate>`
	`  </ds:X509Data>`
	`  </ds:KeyInfo>`
	`</md:KeyDescriptor>`

- Upload the XML to metadata.swamid.se/admin.
- Use "Merge missing from published" to copy over all EntityCategory's and MDUI information from the old Entity if not already in the XML-file.
- Request publication.
- Wait until you get confirmation of publication and then for at least 8 h more (recommended 24 h if in SWAMID and 48 h in eduGAIN) for all entities to pick up the new cert/key.

Step 3 : Switch signing cert

Run the following commands as root user:

```
sudo -s

cd /etc/shibboleth/certs

# Save old signing cert
mv sp-signing-cert.pem sp-signing-cert-old.pem
mv sp-signing-key.pem sp-signing-key-old.pem

# Swap in new signing cert
mv sp-signing-new-cert.pem sp-signing-cert.pem
mv sp-signing-new-key.pem sp-signing-key.pem
```

Test config and if no problems appears restart service

```
sudo -s

# Test config
/usr/sbin/shibd -tc /etc/shibboleth/shibboleth2.xml

service shibd restart
```

Step 4 : Upload new Metadata again

Now we need update our XML and remove the old signing certificate.

Replace	With

`<md:KeyDescriptor use="encryption">`
`  <ds:KeyInfo>`
`  <ds:X509Data>`
`  <ds:X509Certificate>New cert</ds:X509Certificate>`
`  </ds:X509Data>`
`  </ds:KeyInfo>`
`</md:KeyDescriptor>`
`<md:KeyDescriptor use="signing">`
`  <ds:KeyInfo>`
`  <ds:X509Data>`
`  <ds:X509Certificate>New cert</ds:X509Certificate>`
`  </ds:X509Data>`
`  </ds:KeyInfo>`
`</md:KeyDescriptor>`
`<md:KeyDescriptor use="signing">`
`  <ds:KeyInfo>`
`  <ds:X509Data>`
`  <ds:X509Certificate>Old cert</ds:X509Certificate>`
`  </ds:X509Data>`
`  </ds:KeyInfo>`
`</md:KeyDescriptor>`
	`<md:KeyDescriptor use="encryption">`
	`  <ds:KeyInfo>`
	`  <ds:X509Data>`
	`  <ds:X509Certificate>New cert</ds:X509Certificate>`
	`  </ds:X509Data>`
	`  </ds:KeyInfo>`
	`</md:KeyDescriptor>`
	`<md:KeyDescriptor use="signing">`
	`  <ds:KeyInfo>`
	`  <ds:X509Data>`
	`  <ds:X509Certificate>New cert</ds:X509Certificate>`
	`  </ds:X509Data>`
	`  </ds:KeyInfo>`
	`</md:KeyDescriptor>`

- Upload the XML to metadata.swamid.se/admin.
- Use "Merge missing from published" to copy over all EntityCategory's and MDUI information from the old Entity if not already in the XML-file.
- Request publication.

Step 5 : Disable / remove key from software.

Edit `/etc/shibboleth/shibboleth2.xml` to remove old encryption key.

Before	After

`<CredentialResolver type="File" use="signing"`
`key="certs/sp-signing-key.pem"`
`certificate="certs/sp-signing-cert.pem"/>`
 	
`<CredentialResolver type="File" use="encryption"`
`key="certs/sp-encrypt-key.pem"`
`certificate="certs/sp-encrypt-cert.pem"/>`
 	
`<CredentialResolver type="File" use="encryption"`
`key="certs/sp-encrypt-key-old.pem"`
`certificate="certs/sp-encrypt-cert-old.pem"/>`
	`<CredentialResolver type="File" use="signing"`
	`  key="certs/sp-signing-key.pem"`
	`  certificate="certs/sp-signing-cert.pem"/>`
	
	`<CredentialResolver type="File" use="encryption"`
	`  key="certs/sp-encrypt-key.pem"`
	`  certificate="certs/sp-encrypt-cert.pem"/>`

Test config and if no problems appears restart service

```
sudo -s

# Test config
/usr/sbin/shibd -tc /etc/shibboleth/shibboleth2.xml

service shibd restart

# When we are sure everyting works we can remove the old files.
cd /etc/shibboleth/certs
rm sp-signing-cert-old.pem sp-signing-key-old.pem sp-encrypt-key-old.pem sp-encrypt-cert-old.pem
```