Metadata Validation Changes in Skolfederation

New metadata validation requirements based on the federation Technical Profile 1.0.0 are being enforced in Federationsadmin.

Summary of Planned and Enforced Sections

Section	Title	Applies to	Implemented
2.1.1	lang	Identity Providers Service Providers	TBD
2.1.7 / 3.1.5	SAML endpoints	Identity Providers Service Providers	April 9, 2026
2.1.10 / 3.1.8	ContactPerson	Identity Providers Service Providers	April 9, 2026
2.1.3	`errorURL`	Identity Providers	June 16, 2025
2.1.6	SAML certificates (signing)	Identity Providers	June 16, 2025

3.1.4	SAML certificates (encryption)	Service Providers	June 16, 2025
3.1.6	RequestedAttributes	Service Providers	June 16, 2025

Section 2.1.1 - lang

Implemented: TBD

Description

This requirement ensures that human-readable elements in metadata include language information to improve usability and interoperability in multilingual environments.

Requirement

Requirement

All human-readable elements that support language tagging **MUST** include the `xml:lang` attribute.

The following rules apply:

- Language values **MUST** follow ISO 639-1 (e.g. `sv`, `en`)
- Both Swedish (`sv`) and English (`en`) **MUST** be present
- If a language is used in one metadata element, it **MUST** be used consistently across all metadata elements that support `xml:lang`
    - Exception: `md:RegistrationPolicy`
- The same language value **MUST NOT** appear more than once per element
    - Exception: `mdui:Logo`

This includes (but is not limited to):

- `mdui:DisplayName`
- `mdui:Description`
- `md:ServiceName`
- `md:ServiceDescription`

Examples

```
<mdui:DisplayName xml:lang="en">Example Organization</mdui:DisplayName>
<mdui:DisplayName xml:lang="sv">Exempelorganisation</mdui:DisplayName>
<mdui:Description xml:lang="en">Example Organization - more text...</mdui:Description>
<mdui:Description xml:lang="sv">Exempelorganisation - mer text...</mdui:Description>
```

Action

Ensure that:

- Both `sv` and `en` are always present
- Language usage is consistent across all relevant elements
- Only valid ISO 639-1 language codes are used
- No duplicate `xml:lang` values exist per element (except for `Logo`)

2.1.7 / 3.1.5 - SAML endpoints

Implemented: TBD

Description

This requirement enforces correctness and security of SAML endpoint definitions to ensure proper federation interoperability.

Requirement

All SAML endpoints **MUST**:

- use HTTPS (`https://`)
- be valid and well-formed URLs
- not contain localhost, IP addresses, or non-public domains

Applicable endpoints include, but are not limited to:

- `SingleSignOnService`
- `SingleLogoutService`
- `AssertionConsumerService`

For Service Providers, the following additional requirement applies:

- an `AssertionConsumerService` **MUST NOT** use the binding `urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect`

Examples

Valid

```
<md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="https://sp.example.se/acs"
index="0"/>
```

Invalid – ACS with HTTP-Redirect

```
<md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
Location="https://sp.example.se/acs"
index="0"/>
```

Valid

```
<md:SingleSignOnService
 Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
Location="https://idp.example.se/sso"/>
```

Action

Verify that all endpoint URLs:

- use HTTPS
- are publicly reachable
- follow correct URL syntax

For Service Providers, also verify that:

- no `AssertionConsumerService` uses the HTTP-Redirect binding
- `AssertionConsumerService` endpoints use an allowed binding, such as HTTP-POST

2.1.10 / 3.1.8 - ContactPerson

Implemented: TBD

Description

This requirement ensures that federation operators and participants can reliably contact responsible functions for operational, technical, and support-related matters, while complying with personal data protection legislation.

Requirement

Contact information **MUST NOT** refer to a natural person.

The following requirements apply for both Identity Providers and Relying Parties:

- An entity **MUST** include exactly one `ContactPerson` of each of the following types:
  - `administrative`
  - `technical`
  - `support`
- There **MUST NOT** be more than one `ContactPerson` per contact type
- Each `ContactPerson` **MUST** include:
  - `EmailAddress`
- The `EmailAddress`:
  - **MUST** start with `mailto:`
  - **MUST** refer to a functional mailbox (not a personal email)

Guidance

- The `administrative` contact is used for governance and federation-related matters
- The `technical` contact is used for technical issues and integration
- The `support` contact is used for end-user and non-technical support

Examples

Valid

```
<md:ContactPerson contactType="administrative">
<md:EmailAddress>mailto:admin@example.se</md:EmailAddress>
</md:ContactPerson>

<md:ContactPerson contactType="technical">
<md:EmailAddress>mailto:tech@example.se</md:EmailAddress>
</md:ContactPerson>

<md:ContactPerson contactType="support">
<md:EmailAddress>mailto:support@example.se</md:EmailAddress>
</md:ContactPerson>
```

Invalid – missing required contact types

```
<md:ContactPerson contactType="technical">
<md:EmailAddress>mailto:tech@example.se</md:EmailAddress>
</md:ContactPerson>
```

Invalid – multiple contacts of same type

```
<md:ContactPerson contactType="technical">
<md:EmailAddress>mailto:tech1@example.se</md:EmailAddress>
</md:ContactPerson>

<md:ContactPerson contactType="technical">
<md:EmailAddress>mailto:tech2@example.se</md:EmailAddress>
</md:ContactPerson>
```

Invalid – personal contact

```
<md:ContactPerson contactType="technical">
<md:EmailAddress>mailto:firstname.lastname@example.se</md:EmailAddress>
</md:ContactPerson>
```

Action

Ensure that:

- Exactly one `ContactPerson` exists for each required type (`administrative`, `technical`, `support`)
- No duplicate contact types exist
- All email addresses:
    - use the `mailto:` scheme
    - point to functional (role-based) mailboxes
- No personal data is used in contact information

Section 2.1.3 – `errorURL`

Implemented: 16 June 2025

Description

The `errorURL` is a metadata element in an Identity Provider (IdP) configuration that points to a web page intended to help users troubleshoot login problems. When a user encounters an issue during authentication, a Relying Party (e.g. a Service Provider) may redirect the user to this URL for guidance or support. Including a valid and accessible `errorURL` enhances the user experience and aligns with SAML best practices.

Requirement

An Identity Provider **MUST** include an `errorURL` element in its metadata.

> *"A Relying Party may use the errorURL of an Identity Provider to assist users in resolving login issues."*

IdPs **SHOULD** follow the SAML V2.0 Metadata Deployment Profile for errorURL.

Examples

Example not supporting Metadata Deployment Profile for errorURL

```
<md:IDPSSODescriptor errorURL="https://example.com/error.html">
```

Example supporting Metadata Deployment Profile for errorURL, with the required and optional placeholders

```
<md:IDPSSODescriptor errorURL="https://example.com/ERRORURL_CODE?
ts=ERRORURL_TS&rp=ERRORURL_RP&tid=ERRORURL_TID&ctx=ERRORURL_CTX">
```

Action

Ensure your IdP metadata contains a reachable `errorURL`.

Generic errorURL

A generic errorURL is provided by Skolfederation as an example and fallback. More info here.

Section 2.1.6 – SAML certificates (signing)

Implemented: 16 June 2025

Description

A signing certificate is a critical part of an Identity Provider's SAML metadata. It ensures that SAML assertions and metadata can be cryptographically validated by relying parties. The certificate is included via a `<KeyDescriptor>` element, either explicitly marked with `use="signing"` or with no `use` attribute at all.

Requirement

An Identity Provider **MUST** include at least one signing certificate.

> *"A KeyDescriptor element with no `use` attribute or one set to `signing`."*

Examples

Example with use attribute set to signing:

```
<md:KeyDescriptor use="signing">
         <ds:KeyInfo>
                <ds:X509Data>
                        <ds:X509Certificate>
                                <example-certificate-contents>
                        </ds:X509Certificate>
                </ds:X509Data>
        </ds:KeyInfo>
</md:KeyDescriptor>
```

Example with no use attribute set:

```
<md:KeyDescriptor>
        <ds:KeyInfo>
                <ds:X509Data>
                        <ds:X509Certificate>
                                <example-certificate-contents>
                        </ds:X509Certificate>
                </ds:X509Data>
        </ds:KeyInfo>
</md:KeyDescriptor>
```

Action

Verify that a valid signing certificate is present in your metadata.

Section 3.1.4 – SAML certificates (encryption)

Implemented: 16 June 2025

Description

An encryption certificate is required in a Service Provider's SAML metadata to allow Identity Providers to encrypt assertions. This certificate must be included using a `<KeyDescriptor>` element, either explicitly marked with `use="encryption"` or with no `use` attribute (which implies general-purpose use, including encryption).

Requirement

A Service Provider **MUST** include at least one encryption certificate.

> *"A KeyDescriptor element with no `use` attribute or one set to `encryption`."*

Examples

Example with use attribute set to encryption:

```
<md:KeyDescriptor use="encryption">
         <ds:KeyInfo>
                <ds:X509Data>
                        <ds:X509Certificate>
                                <example-certificate-contents>
                        </ds:X509Certificate>
                </ds:X509Data>
        </ds:KeyInfo>
</md:KeyDescriptor>
```

Example with no use attribute set:

```
<md:KeyDescriptor>
        <ds:KeyInfo>
                <ds:X509Data>
                        <ds:X509Certificate>
                                <example-certificate-contents>
                        </ds:X509Certificate>
                </ds:X509Data>
        </ds:KeyInfo>
</md:KeyDescriptor>
```

Action

Ensure your SP metadata includes a valid encryption certificate.

Section 3.1.6 – Requested Attributes (SP)

Implemented: 16 June 2025

Requirement

A Service Provider **MUST** include at least one `AttributeConsumingService` element.

Each `AttributeConsumingService` **MUST** contain:

- A `ServiceName` element with an `xml:lang` attribute.
- A `ServiceDescription` element with an `xml:lang` attribute.
- At least one `RequestedAttribute`.

Each `RequestedAttribute` **MUST** include:

- A `Name` attribute.
- A `FriendlyName` attribute.
- A `NameFormat` attribute set to `urn:oasis:names:tc:SAML:2.0:attrname-format:uri`.

It is **strongly recommended** to use attribute from the federation's attribute profile for interoperability purposes.

If an attribute from the attribute profile is used, the `FriendlyName` **MUST** exactly match the name defined in the profile.

Examples

```
<AttributeConsumingService index="1">
        <ServiceName xml:lang="en">Demo Service</ServiceName>
        <ServiceDescription xml:lang="en">Used for testing login functionality</ServiceDescription>
        <RequestedAttribute Name="urn:oid:1.2.752.29.4.13" FriendlyName="norEduPersonNIN" NameFormat="urn:oasis:
names:tc:SAML:2.0:attrname-format:uri" />
</AttributeConsumingService>
```

Action

Add `RequestedAttribute` definitions that match the federation profile. Remember to use `xml:lang` for language tagging.

Enforcement Notes

The validator enforces these rules **only when metadata is uploaded or updated**. Existing metadata is unaffected unless resubmitted.

Need Help?

For validation help, you can:

- Use the external metadata validator,
- Review the technical profile,
- Contact your technical expertise for information how these rules affect your implementation,
- Contact us for any clarification: info@skolfederation.se