SAML Authentication Overview

An overview of the Security Assertion Markup Language (SAML) implementation in Stack Overflow for Teams Enterprise.

Document generated 12/06/2024

PDF VERSION

Tags | **Authentication** | **SSO** | **SAML** |

Applies to: Free Basic Business **Enterprise**

*This documentation is for **Stack Overflow for Teams Enterprise**. Free, Basic, and Business users can access their documentation here. Find your plan.*

Overview

Security Assertion Markup Language (SAML) allows sites to securely share user information for authentication, profile updates and more. Stack Overflow Enterprise (SOE) uses SAML 2.0 for authentication, communicating with trusted identity providers like Okta, Entra ID, OneLogin, and others.

Properly configuring SAML is a detailed process. This guide will give you an overview of how SAML works, as well as specifics on setting up SOE for SAML authentication. You'll also want to follow the SOE technical setup guide specific to your identity provider.

THIS ARTICLE APPLIES TO STACK OVERFLOW FOR TEAMS ENTERPRISE ONLY.
Other Stack Overflow for Teams users should read this article instead. Find your plan.

How SAML works

When a user first accesses Stack Overflow, the site collects their login credentials and sends them to an external identity provider. If successful, user information is sent back to Stack Overflow to authorize their login. If the user is new, Stack Overflow creates a new account before logging them in. This two-way flow of information uses the SAML protocol.

Depending on your identity provider, you may be able to implement optional security features. Outgoing SAML requests can be *signed* with a certificate to verify the identity of the sender. SAML messages can also be *encrypted* with a certificate to protect the user information. Even without these additional features, however, SAML authentication is a highly secure protocol that uses only encrypted connections.

***NOTE**: Stack Overflow Enterprise and your identity provider must use encrypted connections (https://) for all SAML communications. Unencrypted connections (http://) will fail.*

Terminology

Here are some of the terms you'll need to know when setting up SAML for your SOE site.

Term	Meaning
Security Assertion Markup Language (SAML)	A standardized XML format that allows websites to securely share user information.
Identity provider (IdP)	The external service that authenticates users.
Service provider (SP)	The site the user is wanting to use (Stack Overflow Enterprise in this case).
Single sign-sn (SSO)	A SAML implementation that allows users to access multiple sites with one login.
SP-initiated SSO	A login process that starts at the service provider (SOE). SOE collects the login information and uses SAML to send it to the identity provider (IdP) in an authentication request. The IdP verifies the login and uses SAML to return the user info to SOE.
IDP-initiated SSO	A login process that starts at the IdP. The user logs in at the IdP, then clicks through to Stack Overflow. The IdP uses SAML to send verified user info to SOE, which then grants them access.
Authentication request	User login info sent by Stack Overflow to the IdP, in SAML format.
SAML response	SAML data sent back by the IdP in response to the authentication request, regardless of whether the user login succeeded or failed.
SAML assertion	On successful login, the part of the SAML response that includes the unique user ID, other user info, and security-related information.This data *asserts* that the user is who they claim to be.
Key	A string of characters that make up one half of a key pair. A key pair consists of a public and private key, the two of which have a unique, exclusive match.
Public key	The non-secret part of a key pair that can be shared openly.
Private key	The secret part of a key pair that must be kept private.
Certificate	A container that holds keys and information about them (such as expiration date).
Signing certificate	Used to confirm the sender of the SAML data. Can be used to sign both outgoing authentication requests and incoming SAML responses.
Encryption/decryption certificate	Used to encrypt and decrypt data, such as SAML responses from the identity provider.
Certificate thumbprint	A short, unique identifier designed to make it easier for humans to tell certificates apart.

SAML setup process overview

The SAML setup process includes the following steps:

1. Register your SOE site with your IdP.
2. Configure SOE to communicate with your IdP, and vice-versa.
3. Download a certificate from the IdP and upload it into SOE.
4. Test.

When setting up SAML authentication, you'll configure your SOE site and your IdP in a back-and-forth process. We recommend having a browser tab open to each site.

Register SOE with your identity provider

IdPs require users to register each service provider (site). In this case, you'll give the IdP information about your SOE site. Though each IdP is different, here are some common settings you'll need to configure.

Issuer
A unique identifier that tells the IdP which service provider is making the authentication request (your SOE site). Some IdPs automatically create an issuer ID for you.

URL Whitelist
When sending an authentication request to the IdP, SOE includes a redirect (destination) URL for the user to go to after successful authentication. To prevent malicious redirects, most IdPs allow you to specify a whitelist (approved list) of URLs. If an authentication request comes in with a redirect URL that's not on the whitelist, the IdP will not redirect the user there and authentication will fail.

Signing certificate public key
If you've configured SOE to sign outgoing authentication requests (with a private key), you must provide the corresponding public key to the IdP. The IdP will whitelist, validate, and use the public key to verify the request. See the **SAML data examples** section for an example.

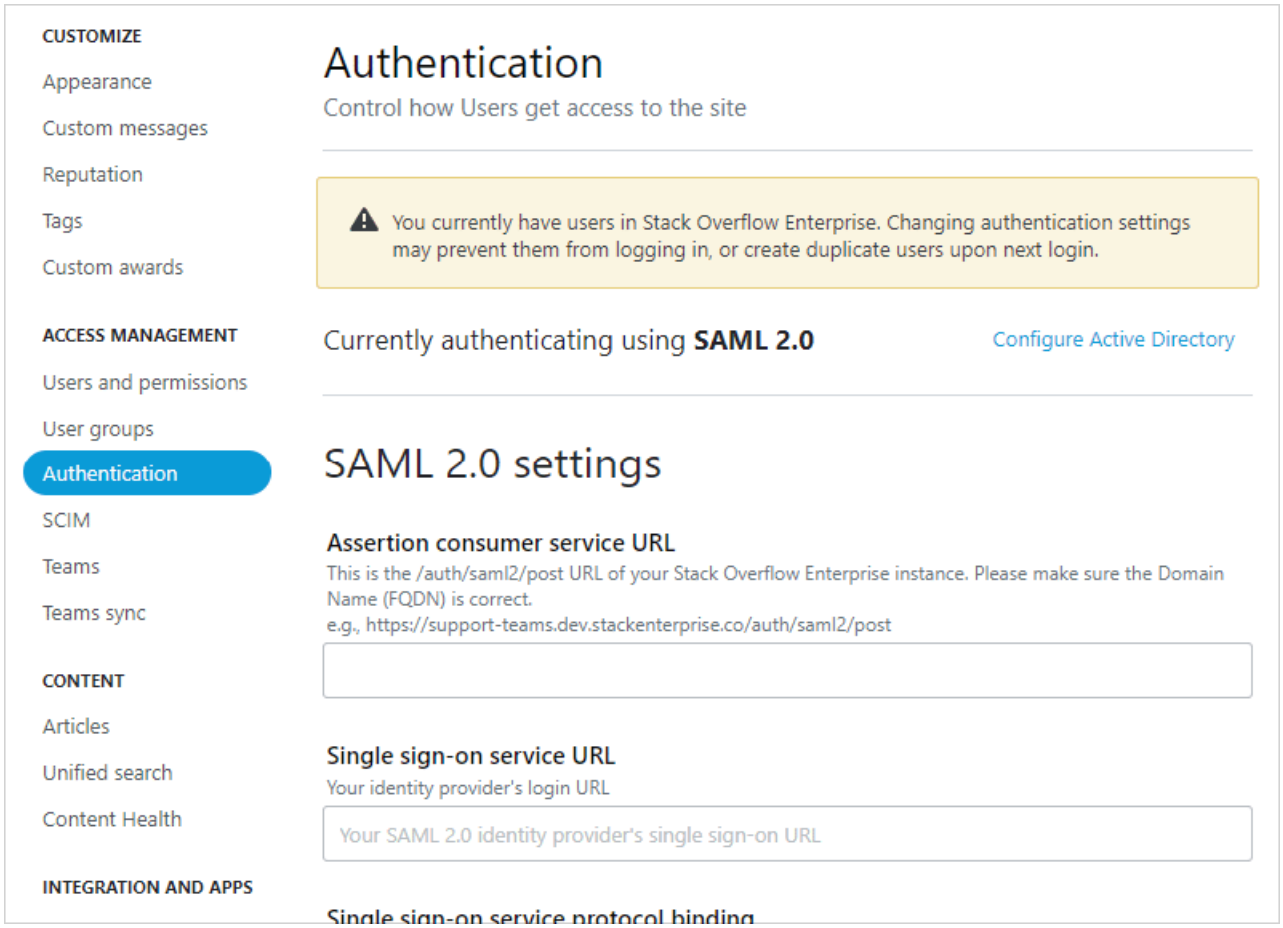
Encryption certificate public key
If you want the IdP to encrypt successful authentication assertions it sends back to SOE, provide the encryption certificate (public key) it should use. See the **SAML data examples** section for an example.

NOTE: *Signed authentication requests and encrypted SAML responses are optional and rarely used by SOE. See the* ***Advanced setup*** *section for more info.*

Configure SAML Authentication on Stack Overflow Enterprise

SOE will interface with many identity providers, each having a slightly different set of parameters and options. You'll need to set SAML options below based on the IdP you use, in consultation with the SOE technical setup guide specific to that IdP.

Access the SAML configuration screen by clicking **Admin settings** on your SOE site, then **Authentication** (under the "Access management" heading). If you see other authentication settings (not SAML 2.0), click **Configure SAML 2.0**. You can also access this screen directly with the URL [your site]/authentication.



SAML 2.0 settings

Assertion consumer service URL
The URL that the IdP will redirect users back to after successful authentication. Set this value to this specific URL on your SOE site: https://[your site]/auth/saml2/post. The URL must use a secure connection (https://).

Single sign-on service URL
The identity provider URL that SOE will send authorization requests to. You'll get this URL from your IdP. The URL must use a secure connection (https://).

Single sign-on service protocol binding
This is the method we use when we send the authentication request to the IdP. Choose the option that ends in **HTTP-Redirect** unless instructed differently by Stack Overflow product support.

HTTP-Redirect binding: include SAML encoding parameter in query string
If your IdP requires authentication requests to include a `SAMLEncoding` parameter, enable this option.

Enforce 80 byte maximum RelayState length?
SOE uses the `RelayState` value as a redirect URL, telling the IdP where to return the user to after successful authentication. If this redirect URL is too long, the IdP may ignore it and instead return the user to your SOE home page. Consult the technical setup guide for your specific IdP for guidance on this setting.

Issuer
This value identifies your SOE site to the IdP. Some IdPs will provide this to you, while others let you choose your own. Consult the technical setup guide for your specific IdP for guidance on this setting.

Audience restriction
This value will be the same as **Issuer** above, unless instructed otherwise by the technical setup guide for your specific IdP.

Use Subject/NameID as user identifier
Every SOE user has a unique identifier. In the SAML response, this is usually `NameID`. Leave this box checked, unless instructed otherwise by the technical setup guide for your specific IdP.

User identifier assertion attribute
If not using the default identifier (**Use Subject/NameID as user identifier** is NOT checked), this field allows you to specify the identifier in the SAML response. Consult the technical setup guide for your specific IdP for guidance on this setting.

If the guide doesn't specify an identifier, you must choose a unique, unchanging identifier. Common choices are Windows SID, Active Directory ObjectGUID, LDAP uid, or some form of unique employee ID. If SOE finds an existing user with the provided ID, it will log them in. If it doesn't find an existing user, SOE will create a new user account with that ID.

NOTE: *You must select a user ID that is both unique and unchanging. User email address, for example, is unique but not unchanging. If you use email address as the unique ID, a new email address entered at the IdP would result in SOE creating a new account for that user.*

Display name assertion
The data element in the SAML assertion that holds the user's full name as displayed on the site. You can set SOE to automatically update this field with SAML data on login. See the **Update user profile on SAML login**) section for more info.

Email address assertion
The data element in the SAML assertion that holds the user's email address. You can set SOE to automatically update this field with SAML data on login. See the **Update user profile on SAML login**) section for more info.

Job Title assertion
The optional data element in the SAML assertion that holds the user's job title. When configured and included in the SAML response, SOE automatically updates this user data on login.

Department assertion
The optional data element in the SAML assertion that holds the user's department. When configured and included in the SAML response, SOE automatically updates this user data on login.

External ID assertion
The optional data element in the SAML assertion that holds the user's external identifier. This data element has no purpose in SOE, and is provided as a convenience for SOE clients to identify their users (such as with a company employee ID).

You can set SOE to display a user's external ID in their profile with the **UserProfile.ShowExternalIdOnProfile** site setting.

Enable importing profile image on user creation
Option to tell SOE to download and store a profile picture when it creates a new user profile.

Profile image URL assertion
If **Enable importing profile image on user creation** is checked above, this tells SOE the name of the data element that holds the user's profile image URL. SOE will download and store the profile picture from this URL when creating a new user account. It will not overwrite the profile picture of an existing user.

Profile image Base64 assertion
If **Enable importing profile image on user creation** is checked above, this tells SOE the name of the data element that holds the user's *Base64-encoded* profile image. This is an alternative to the image link above, and will take precedence if the SAML response contains both values.

NOTE: *Profile pictures should be square and at least 164x164 pixels, similar to Gravatar image size guidelines.*

Disable SP-initiated SAML 2.0 SSO
With a normal SAML authentication flow, SOE collects login information and sends it to the IdP as an authentication request. If this option is checked, SOE will instead prompt the user to log in at their IdP and will not send an authentication request.

NOTE: *This option may be useful if your IdP requires a signed authentication request but SOE is unable to generate one (due to a certificate problem, for example).*

IDP initiated sign on URL Provides users a link to their IdP if **Disable SP-initiated SAML 2.0 SSO** is enabled. If you leave this field blank, users will be told to log in at their IdP but SOE will provide no link.

Automatically login If **Disable SP-initiated SAML 2.0 SSO** is enabled, this redirects the user automatically to their IdP for login (SOE will generate no prompt or link).

Automatic login message If **Automatically login** is enabled, this message displays as the user is being redirected to their IdP for login.

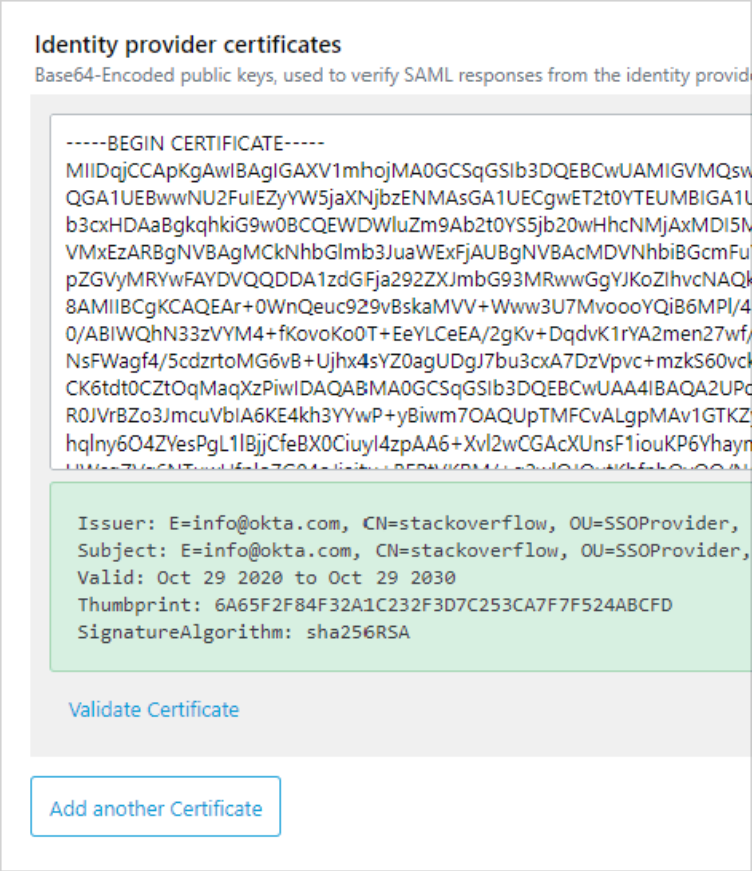
Update certificates from federation metadata URL
For improved security, some IdPs require certificates (public keys) to be refreshed every hour. These IdPs will supply a URL to retrieve new certificates, which you can enter here. SOE will automatically retrieve and install a new certificate every hour.

NOTE: *If **Update certificates from federation metadata URL** is set, you can manually trigger a certificate update at [your site]/enterprise/support/saml-certupdate (admin access required).*

NOTE: *A federation metadata URL is recommended for automated certificate management, making identity provider certificates (below) unnecessary. If you enter a federation metadata URL, the "Identity provider certificates" section of the page will disappear to prevent accidental overwrites that could impact user access to your site. To restore these fields and manually update certificates instead, remove the federation metadata URL.*

Identity provider certificates
SOE requires that your IdP sign every SAML response it sends. The IdP uses its private key to sign the SAML response, then SOE uses the corresponding public key to verify the sender.

Paste the public key certificate provided by your IdP here, including the "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----" boundaries. You can store multiple certificates with **Add another Certificate**, as well as remove ones that have expired or been compromised with **Remove Certificate**.



Click **Validate Certificate** to verify that a certificate is properly formatted. If the certificate is valid, SOE will show a box with information about the certificate. Pay careful attention to the valid date range. If the certificate's valid end date has passed, it will no longer work. Click **Remove Certificate** to remove and replace it.

NOTE: *If a federation metadata URL has been entered (above), the "Identity provider certificates" section of the page will be hidden to prevent accidental overwrites between automatic and manual certificate updates. To restore these fields and manually update certificates instead, remove the federation metadata URL value.*

*__NOTE:__ The certificate used to verify the identity of the IdP response is the **only** certificate required by SOE. Other uses of certificates and keys fall outside typical SAML usage. See the **Advanced setup** section for more info.

Managing certificates is a crucial part of the SAML setup process. See the **Certificates and keys** section for more info.*

Signing configuration

Sign outgoing AuthnRequests
Enable this option if your IdP requires SOE to sign each authentication request with a private key. Signed authentication requests are optional and rarely used by SOE. See the **Advanced setup** section for more info.

Signing certificate for outgoing AuthnRequests
If **Sign outgoing AuthnRequests** is enabled, use this pull-down menu to choose the certificate SOE should use to sign outgoing authentication requests.

Digest method when signing outgoing requests
The signing digest method you choose affects security, and we recommend using SHA-256 or higher. SOE supports (but

doesn't recommend) the less-secure SHA-1 digest method for compatibility with certain IdPs.

How to attach the public key to the signed request
Authentication requests can include the public key for convenience, but not all IdPs support this option. Consult the SAML technical setup guide specific to your IdP for guidance on this setting.

Security settings

Show additional detail if login fails
Display detailed error information to the user if SAML authentication fails.

Force reauthentication
Ask the IdP to always force a new user login (not use a previous authentication session). Not all IdPs honor this.

Verify the `SubjectConfirmation` element on a SAML response
Some identity providers don't send a proper `SubjectConfirmation`. Check this box to verify the parameter.

Additional options

Enable SAML login troubleshooting page
Display a detailed SAML login troubleshooting page at [your site]/enterprise/support/saml-login. This option has security implications, so enable the page only as long as needed. Learn more in the **Troubleshooting** section.

Enable SAML Response logging for troubleshooting
Log all SAML responses to the database for troubleshooting. Enable this option if directed to by Stack Overflow product support.

Ensure KeyInfo element on EncryptedKey
Set this if encrypted SAML responses from your IdP fail to decrypt with "Unable to retrieve the decryption key" error. See the **SAML data examples** section for an example.

Add UTF-8 byte order mark to the EntityDescriptor.xml
If your IdP is rejecting SOE's FederationMetadata.xml file, try enabling this option.

Additional links

The bottom of the settings page has additional links.

If **Enable SAML login troubleshooting page** is enabled, the **Test currently saved SAML configuration** button will take you to the login troubleshooting page ([your site]/enterprise/support/saml-login). Learn more in the **Troubleshooting** section.

Set up additional access rules
Takes you to the **Configure Access Rules for SAML2** page at [your site]/enterprise/support/access-rules. Learn more in the SAML access rules document.

Download SAML 2.0 EntityDescriptor that can be imported into Identity Provider (IdP)
This link downloads a file that certain IdPs will accept for auto-configuration. Learn more in the **Identity provider auto-configuration** section.

Parse SAML 2.0 EntityDescriptor from Identity Provider (IdP) If **Update certificates from federation metadata URL** is set and SOE is failing to update certificates hourly, use this link to troubleshoot the response from your IdP. Learn more in the **Troubleshooting** section.

Certificates and keys

The SAML standard supports the use of certificates for both signing and data encryption. Properly setting up certificates (and the keys they contain) is one of the more challenging aspects of SAML configuration. Remember: you can always reach out to Stack Overflow product support if you need help.

SAML response signing certificate (required)

Because the IdP must sign every response it sends, the SAML response signing certificate is the only *required* certificate. SOE will check the signature against the public keys uploaded to the server. If the response is not signed, or if SOE can't find a matching public key, authentication will fail and SOE will display an error message to the user.

As part of the setup process, your IdP will create a signing certificate with a public key. It's this public key that you'll upload into your SOE SAML settings page.

NOTE: *Even if your IdP provides a URL to automatically update certificates, you should download and save the IdP's public key to SOE for initial setup and testing.*

Authentication request signing certificate (optional)

If you've enabled authentication request signing, SOE will use a private key to sign each outgoing authentication request. The receiving IdP uses the corresponding public key to verify the sender identity. Learn more about encrypted authentication requests in the **Advanced setup** section.

Decryption Certificate (optional)

Some IdPs encrypt the SAML assertion (user data portion) of the SAML response. The IdP uses the certificate's public key to encrypt the data, and SOE uses the private key to decrypt it. Learn more about encrypted SAML assertions in the **Advanced setup** section.

Save SAML settings

When you've completed your SAML setup or update, click **Save settings**. This launches a "Test new authentication settings" pop-up. Click **Submit** to test your authentication settings with a full login flow. If the test fails, you'll remain on the SAML settings page so you can double-check and correct the settings. If the test succeeds, SOE applies your new authentication settings. Logged-in users stay logged in, as all active user sessions remain valid.

This test acts as a safety net to keep invalid authentication settings from locking users out of your site. Until your settings are correct and pass the authentication test, SOE will not apply your changes.

If you need more technical details for your SAML auth flow, use the **Test currently saved SAML configuration** button to view the SAML response, authentication log, and assertions (requires **Enable SAML login troubleshooting page** to be on). Learn more about this feature in the Troubleshooting section of this document.

Advanced setup

The following settings and links are not commonly used, but may be relevant to your specific IdP and SAML configuration.

Authentication request signing

If you've enabled optional authentication request signing, SOE will need access to the private signing key. The location of the signing key depends on your SOE deployment type.

- For **on-premises deployments**, the private key for the signing certificate needs to be accessible in the Windows Certificate Store (Local Computer\Personal\Certificates). The identity running the Internet Information Services (IIS) application pool needs `Read` permission for the private key. If the IIS application pool uses `ApplicationPoolIdentity` (the default), you can add it in the form `IIS AppPool\NameOfTheAppPool`.
- For **hosted deployments**, SOE will use the same certificate for signing and decryption. We generate the private key, and provide the public key to you. If your IdP will be encrypting SAML responses, you'll need to provide this public key to them.

NOTE: *If your IdP requires signed authentication requests, it will need your SOE public key to verify those requests. Please create a new support ticket to request your public key.*

SAML assertion encryption

If you've enable optional SAML assertion encryption at your IdP, SOE will need access to your private decryption key. The location of the private decryption key depends on your SOE deployment type.

- For **on-premises deployments**, the private key for the decryption certificate needs to be accessible in the Windows Certificate Store (Local Computer\Personal\Certificates), and can be different from the signing certificate. The identity running the IIS application pool needs `Read` permission for the private key. If the IIS application pool uses `ApplicationPoolIdentity` (the default), you can add it in the form `IIS AppPool\NameOfTheAppPool`.

NOTE: *For load-balanced deployments, the certificate needs to exist in every server running SOE.*

- For **hosted deployments**, the decryption certificate for inbound SAML responses will be the same as the signing certificate for outbound authentication requests. This is the certificate you selected as **Signing certificate for outgoing AuthnRequests** in the **Signing configuration** area.

NOTE: *SOE uses the* ***Signing certificate for outgoing AuthnRequests*** *for authentication request signing and decrypting encrypted SAML responses. If your IdP is encrypting SAML responses, you'll need to select a certificate for this setting* ***even if you're not signing outgoing authentication requests****.*

Update user profile on SAML login

On successful login, the IdP sends user data back in its SAML response. You can set SOE to automatically update certain user profile fields with this data. If a user changes email address at the IdP, for example, SOE will update the email address in the user profile.

NOTE: *To keep IdP and SOE data in-sync, you must disable the user's ability to change profile fields that the IdP will update. You should then instruct users to change those fields at the IdP instead of in SOE.*

Set fields to update on IdP login

You can enable IdP update of the **Display Name** and **Verified Email** profile fields on successful login. The corresponding site settings are:

- `UserProfile.UpdateDisplayNameFromIdPUponLogin` Display name
  - Visit https://[your_site]/developer/site-settings/edit?name=UserProfile.UpdateDisplayNameFromIdPUponLogin
- `Auth.UpdateVerifiedEmailFromIdPUponLogin` Email address
  - Visit https://[your_site]/developer/site-settings/edit?name=Auth.UpdateVerifiedEmailFromIdPUponLogin

As a site admin, follow this process for each profile field you want the IdP to update:

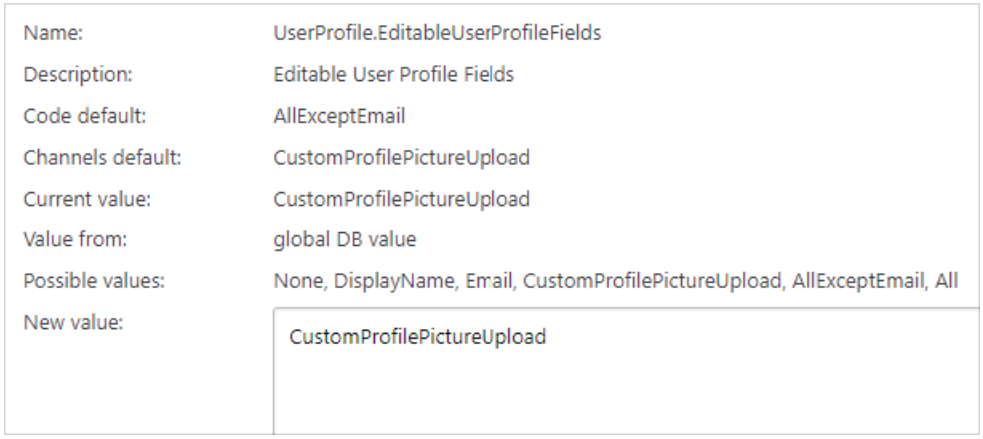
1. Follow the provided link (above) for the site setting you need to change.
2. Select "True" in the **New value** field.
3. Enter a brief reason for the configuration change.
4. Click **Save New Network Default** to save.

NOTE: *Unlike the* ***Display Name*** *and* ***Verified Email*** *profile fields, SOE will never overwrite the* ***Profile Picture*** *of an existing user on subsequent logins. SOE will save a profile picture from the IdP only on initial account creation.*

Disallow user changes

If you allow the IdP to update user profile fields, you need to *disallow* user changes to those same fields. You'll change the `UserProfile.EditableUserProfileFields` setting to the appropriate value from the following options.

- **None** User can't edit any profile fields
- **DisplayName** User can edit display name only
- **Email** User can edit email address only
- **CustomProfilePictureUpload** User can edit profile picture only
- **AllExceptEmail** User can edit all fields but email address
- **All** User can edit all profile fields



Follow these steps to update the `UserProfile.EditableUserProfileFields` field:

1. Visit https://[your_site]/developer/site-settings/edit?name=UserProfile.EditableUserProfileFields.
2. Carefully type the desired value into the **New value** box. *Hint: copy/paste the value from the **Possible values** list to ensure you've entered it correctly.*
3. Enter a reason for the configuration change.
4. Click **Save New Network Default** to save.

Identity provider auto-configuration

Some IdPs will let you upload a data file to automatically configure certain SAML settings. The **SAML 2.0 Settings** page allows you to create and download this data file with information about your SOE site. Follow this process to auto-configure your IdP:

1. Configure SAML settings on the **SAML 2.0 Settings** page.
2. Click **Download SAML 2.0 EntityDescriptor that can be imported into Identity Provider (IdP)** at the bottom of the page.
3. Download the resulting FederationMetadata.xml file and save it to your computer.
4. Upload the data file to your IdP.

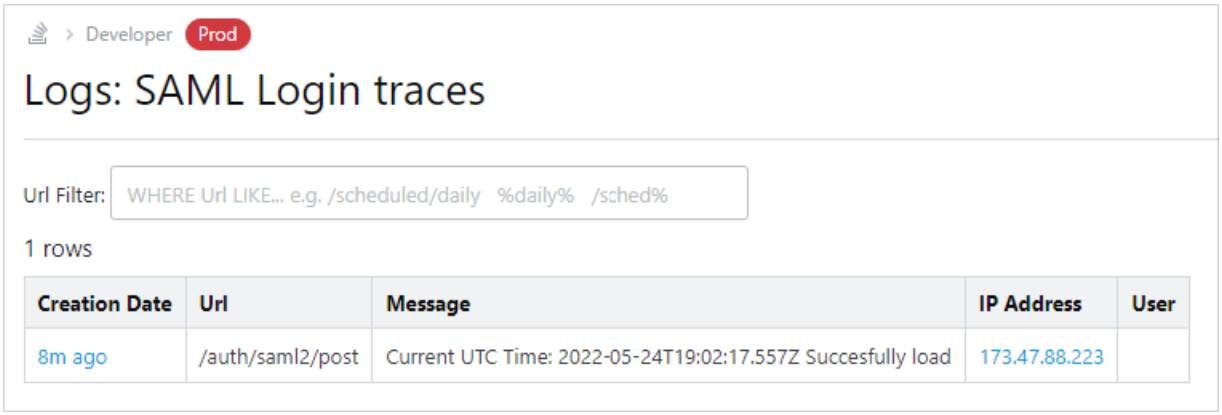
Troubleshooting

Because SAML authentication setup can be tricky, SOE provides several means to troubleshoot authentication problems.

Log SAML responses

You can enable database logging of both successful and unsuccessful SAML authentication responses for quicker troubleshooting. To enable this feature, check **Enable SAML Response logging for troubleshooting** and save the settings.

Site admins can view the stored logs on the developer log page.



- **[your site]/developer/logs/72** The `SamlLoginTrace` table contains the actual SAML authentication logs.
- **[your site]/developer/logs/73** The `SamlTracingStatusChanged` table contains a history of SAML log setting changes.

[your site]/enterprise/support/saml-login

This URL (as well as the **Test currently saved SAML configuration** button) initiates an authentication request to your SAML identity provider and displays the response.

The SAML test page consists of the following four sections.

Base64-encoded SAML response
The raw (Base64-encoded) data, exactly as received from the IdP. If you're requesting help with your SAML setup, include this raw data with the help ticket as an attachment or pasted text.

Successfully parsed SAML Response to XML
The data as parsed in XML format.

Authentication log
A log of the authentication process, including the processing access rules. You'll find a wealth of useful troubleshooting info in this section.

All Attributes in Assertion
The final product of a successful SAML login: the attributes (user data) returned from the IdP.

NOTE: *The SAML test page is meant for initial authentication setup only and should be disabled as soon as your SAML authentication is working. To disable the SAML test page, uncheck the* ***Enable SAML login troubleshooting page*** *box and click* ***Save settings****.*

[your site]/admin/errors

If the SAML test page indicates a certificate error, go to the admin errors page to see if a `CryptographicException` or `InvalidOperationException` error occurred.

Here are the error messages and potential causes:

Keyset does not exist

The IIS Application Pool identity does not have permission to the private key
See above on how to give permission to the IIS Application Pool to the Private Key
Solution: grant the IIS Application Pool identity `Read` permission on the private key. If the IIS Application Pool uses `ApplicationPoolIdentity` (the default), you can add it in the form `IIS AppPool\NameOfTheAppPool`.

The certificate does not have a private key
SOE won't show certificates without a private key in the drop-down list, but it's possible to replace an existing certificate with one that lacks a private key. If a certificate doesn't have a private key, SOE can't use it for signing or decryption.
Solution: add a certificate with both public and private keys.

The parameter is incorrect

During decryption of a SAML assertion, the certificate does not match
This error can occur on a hosted deployment if the IdP uses one certificate for encryption and a different one for signing.
Solution: make sure your IdP is configured to use the same certificate for signing and encryption.

This error can also occur if the **Ensure KeyInfo Element on EncryptedKey** option is checked and the IdP did not send this element. In this case, we try to use the signing certificate to decrypt in both on-premises and hosted deployments.
Solution: uncheck **Ensure KeyInfo Element on EncryptedKey**

Invalid provider type

The certificate was not in the right format
SOE public and private keys must be in the RSA format. If you use a self-signed certificate created through IIS Manager or the PowerShell New-SelfSignedCertificate cmdlet, that certificate will have keys in the CNG format.
Solution: use a certificate created by a certificate authority, with keys in the RSA format.

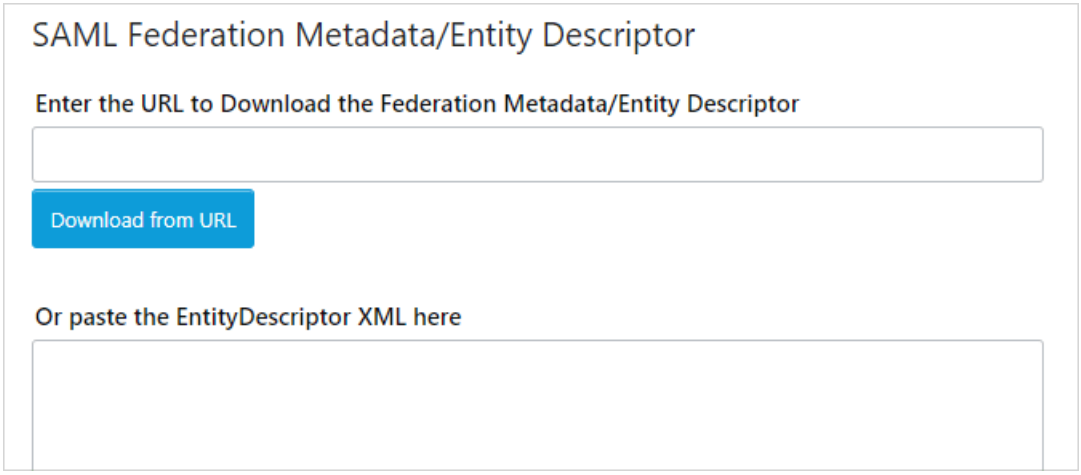
Could not fetch certificate with thumbprint [XXXXXXXXXXXXX]

Certificate required for decryption does not exist in the certificate store
The SAML response from the IdP was encrypted, but SOE couldn't find a certificate to decrypt it.
Solution (on-premises): make sure the certificate exists in the Windows Certificate Store (Local Computer\Personal\Certificates).
Solution (hosted): make sure the IdP uses the same certificate that you've uploaded as the signing certificate.

Automatic certificate updates

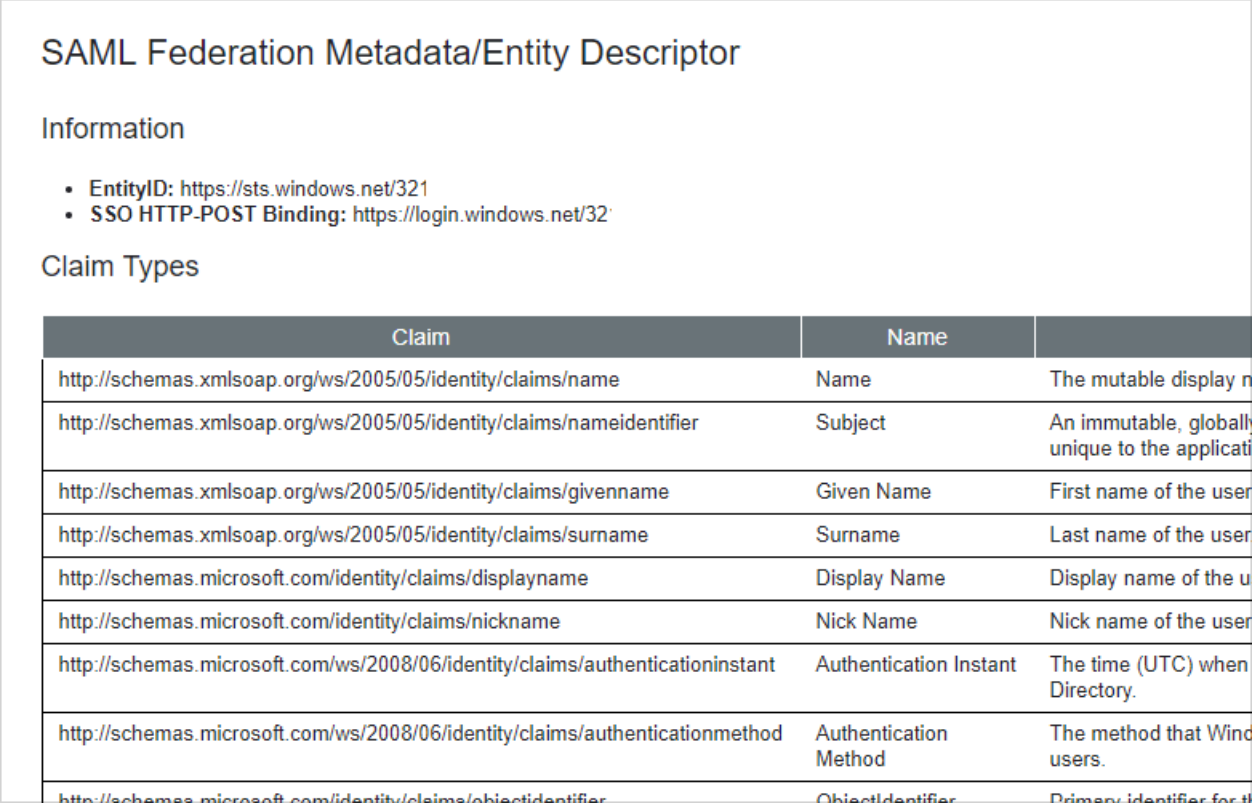
Some IdPs update certificates every hour, providing a link for SOE to download the refreshed certificates automatically. If this process fails, use the **Parse SAML 2.0 EntityDescriptor from Identity Provider (IdP)** link to troubleshoot the problem.

Enter the **Update certificates from federation metadata URL** value to test the full download process. If the URL is working but the resulting file is not, paste the contents of the downloaded FederationMetadata.xml file into the **EntityDescriptor XML** box.



SOE will parse the XML file and display the relevant data, including the authentication request (SSO) URL and signing certificate (public key). You can verify these values against your current SAML settings.

A properly formed FederationMetadata.xml file should look like this:



SAML Federation Metadata/Entity Descriptor

Information

- **EntityID:** https://sts.windows.net/321
- **SSO HTTP-POST Binding:** https://login.windows.net/32

Claim Types

Claim	Name	
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name	Name	The mutable display n
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier	Subject	An immutable, globall unique to the applicati
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname	Given Name	First name of the user
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname	Surname	Last name of the user
http://schemas.microsoft.com/identity/claims/displayname	Display Name	Display name of the u
http://schemas.microsoft.com/identity/claims/nickname	Nick Name	Nick name of the user
http://schemas.microsoft.com/ws/2008/06/identity/claims/authenticationinstant	Authentication Instant	The time (UTC) when Directory.
http://schemas.microsoft.com/ws/2008/06/identity/claims/authenticationmethod	Authentication Method	The method that Wind users.
http://schemas.microsoft.com/identity/claims/objectidentifier	ObjectIdentifier	Primary identifier for th

SAML setup FAQs

- **Q**: Can we access our new SOE site to set up SAML *before* we've configured authentication?
  A: Yes. The SAML setup page is unrestricted when we initially deploy your site. After you configure authentication and create the first "super admin" account, only site admins will be able to access the SAML setup page.
- **Q**: How do we restrict access to our SOE site so only certain users can log in?
  A: The best way to control user access is to set up restrictions at your IdP. SOE also offers support for basic access rules.
- **Q**: Can we manage user roles in Stack Overflow Enterprise through our IdP?
  A: SOE has three user roles: regular user, moderator, and site administrator. You can manage moderator and regular user roles with SCIM 2.0.

SAML data examples

The section below includes examples of SAML data, both outbound authentication requests and inbound SAML responses.

Example of a signed authentication request

This example adds the "Signature" XML Element to the AuthnRequest document

```
<AuthnRequest xmlns="urn:oasis:names:tc:SAML:2.0:protocol"
              xmlns:xsd="http://www.w3.org/2001/XMLSchema"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              ID="soef0140b284224379f68a536af9410d4a6"
              Version="2.0"
              IssueInstant="2017-09-29T06:45:27.3266215Z"
              Destination="https://your-idp/saml-auth/"
              ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
              AssertionConsumerServiceURL="https://your-so-enterprise-url/auth/saml2/post">
  <Issuer xmlns="urn:oasis:names:tc:SAML:2.0:assertion">saml-sp-issuer-identifier</Issuer>
  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
    <SignedInfo>
      <CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
      <SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
      <Reference URI="">
        <Transforms>
          <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
          <Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        </Transforms>
        <DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
        <DigestValue>gyYG98munfSkW1C5FXGT701hey20n12sFkRfPRHyoY4=</DigestValue>
      </Reference>
    </SignedInfo>
    <SignatureValue>q2l[....]I7ErQSVXoxw==</SignatureValue>
  </Signature>
</AuthnRequest>
```

Example of the expected response from the IdP (unimportant parts removed)

```
<?xml version="1.0" encoding="utf-8"?>
<EntityDescriptor ID="_221dada1-5b76-4985-91aa-e79c0f16c7a5" entityID="https://sts.windows.net/321fbc04-5721-4a01-81bb-a9
  <RoleDescriptor xsi:type="fed:SecurityTokenServiceType" protocolSupportEnumeration="http://docs.oasis-open.org/wsfed/fe
    <KeyDescriptor use="signing">
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
          <X509Certificate>MIIDBTCCAe[...]aQ3B</X509Certificate>
        </X509Data>
      </KeyInfo>
    </KeyDescriptor>
    <KeyDescriptor use="signing">
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
          <X509Certificate>MIIDBTCCAe2gAwIBAgIQ[...]meLE3yyC5yGL</X509Certificate>
        </X509Data>
      </KeyInfo>
    </KeyDescriptor>
    <KeyDescriptor use="signing">
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
          <X509Certificate>MIID[...]cu7ItE=</X509Certificate>
        </X509Data>
      </KeyInfo>
    </KeyDescriptor>
  </RoleDescriptor>
  <IDPSSODescriptor protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <KeyDescriptor use="signing">
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
```

```
          <X509Certificate>MIIDBT[...]aQ3B</X509Certificate>
        </X509Data>
```

Example of a signed SAML response

```
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
                Consent="urn:oasis:names:tc:SAML:2.0:consent:unspecified"
                Destination="https://dev.soe.devdomains.org/auth/saml2/post"
                ID="_0b8b0902-aa36-4b04-adc4-b58ac51727a3"
                InResponseTo="soe5b4ae9b4d536b7e076844edc66d53248"
                IssueInstant="2017-10-15T03:26:42.137Z"
                Version="2.0">
  <Issuer xmlns="urn:oasis:names:tc:SAML:2.0:assertion">your-idp-url/idp-issuer-identifier</Issuer>
  <samlp:Status>
    <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"></samlp:StatusCode>
  </samlp:Status>
  <Assertion ID="_b383074d-bebc-45cb-825e-8eabc4cba758"
             IssueInstant="2017-10-15T03:26:42.105Z"
             Version="2.0"
             xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
    <Issuer>your-idp-url/idp-issuer-identifier</Issuer>
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
        <ds:Reference URI="#_b383074d-bebc-45cb-825e-8eabc4cba758">
          <ds:Transforms>
            <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
            <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
          </ds:Transforms>
          <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
          <ds:DigestValue>avZJkvca1MzjMA3NZsAkNOrdFkQyaekX19q6VQricBg=</ds:DigestValue>
        </ds:Reference>
      </ds:SignedInfo>
      <ds:SignatureValue>gHoZ2[...]eMvxhw==</ds:SignatureValue>
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
```

Example of an encrypted SAML response

```
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
                Consent="urn:oasis:names:tc:SAML:2.0:consent:unspecified"
                Destination="https://dev.soe.devdomains.org/auth/saml2/post"
                ID="_0b8b0902-aa36-4b04-adc4-b58ac51727a3"
                InResponseTo="soe5b4ae9b4d536b7e076844edc66d53248"
                IssueInstant="2017-10-15T03:26:42.137Z"
                Version="2.0">
  <Issuer xmlns="urn:oasis:names:tc:SAML:2.0:assertion">your-idp-url/idp-issuer-identifier</Issuer>
  <samlp:Status>
    <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"></samlp:StatusCode>
  </samlp:Status>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
      <ds:Reference URI="#_0b8b0902-aa36-4b04-adc4-b58ac51727a3">
```

```
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
          <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        </ds:Transforms>
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
        <ds:DigestValue>R/wlRbcnpzUXGNc6znNvyTpJrkcXBg3MCbpMChWPkwQ=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>pMV6QVc[...]pDqDQ==</ds:SignatureValue>
    <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
      <X509Data>
        <X509Certificate>MIIC[...]yh6I=</X509Certificate>
      </X509Data>
    </KeyInfo>
  </ds:Signature>
```

Example of an encrypted assertion that needs the Ensure KeyInfo element on EncryptedKey option checked

No `<KeyInfo>` under `<EncryptedKey>`

```
<saml:EncryptedAssertion>
  <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" Type="http://www.w3.org/2001/04/xmlenc#Element">
    <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#tripledes-cbc" />
    <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <xenc:EncryptedKey>
        <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5" />
        <xenc:CipherData>
          <xenc:CipherValue>rSb9[...]2wrw==</xenc:CipherValue>
        </xenc:CipherData>
      </xenc:EncryptedKey>
    </ds:KeyInfo>
    <xenc:CipherData>
      <xenc:CipherValue>83McRPpwN[.....]oHw==</xenc:CipherValue>
    </xenc:CipherData>
  </xenc:EncryptedData>
</saml:EncryptedAssertion>
```

Example of an encrypted assertion that does not need the Ensure KeyInfo element on EncryptedKey option checked

Contains a `<KeyInfo>` under `<EncryptedKey>` telling us which certificate to use for decryption

```
  <EncryptedAssertion xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
    <xenc:EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element" xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
      <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes256-cbc" />
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <e:EncryptedKey xmlns:e="http://www.w3.org/2001/04/xmlenc#">
          <e:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p">
            <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
          </e:EncryptionMethod>
          <KeyInfo>
            <ds:X509Data xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <ds:X509IssuerSerial>
                <ds:X509IssuerName>CN=[...], DC=org</ds:X509IssuerName>
```

```
              <ds:X509SerialNumber>256[...]180</ds:X509SerialNumber>
            </ds:X509IssuerSerial>
          </ds:X509Data>
        </KeyInfo>
        <e:CipherData>
          <e:CipherValue>NAxqIjD[...]2fMesQ==</e:CipherValue>
        </e:CipherData>
      </e:EncryptedKey>
    </KeyInfo>
    <xenc:CipherData>
      <xenc:CipherValue>/O3cyS[...]NMYjpEpl0</xenc:CipherValue>
    </xenc:CipherData>
  </xenc:EncryptedData>
</EncryptedAssertion>
```

If you need further support or have questions, contact your site administrator.