S

# SAML: How Single Sign-On Works and How to Secure It

SAML is an XML-based standard for exchanging authentication and attribute information between an identity provider and a service provider.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
3 min

CONTENTS

## In this guide

01 SAML roles and terms

02 SP-initiated SSO flow

03 What the service provider must validate

04 Signing, encryption, and TLS

05 Operational security checklist

06 SAML versus OAuth and OpenID Connect

07 Source

**Quick answer:** Security Assertion Markup Language (SAML) is an XML-based standard used to exchange authentication and attribute information between an identity provider (IdP) and a service provider (SP). In web single sign-on, the IdP authenticates the user and

sends a signed SAML response that the SP must validate before creating a session.

## 01 SAML roles and terms

- **Principal:** the user whose identity is being asserted.
- **Identity provider:** authenticates the user and issues assertions.
- **Service provider:** the application that relies on the assertion and grants a local session.
- **Assertion:** XML containing statements about authentication, subject, attributes, conditions, and intended audience.
- **Metadata:** configuration describing entity identifiers, endpoints, supported bindings, and signing or encryption certificates.

SAML normally lets the application rely on the IdP without receiving the user's password. The application still creates and protects its own session after validation.

## 02 SP-initiated SSO flow

1. The user visits the service provider and requests a protected resource.
2. The SP creates a SAML authentication request and redirects the browser to the IdP.
3. The IdP authenticates the user using its configured methods and policies.
4. The IdP returns a SAML response through the browser to the SP's assertion consumer service.
5. The SP validates the response and assertion, maps the subject and attributes to an authorized account, and creates a session.

IdP-initiated SSO begins at the identity provider without a corresponding SP request. It can be supported, but loses the strong request-response correlation provided by `InResponseTo` and therefore requires careful replay and destination protections.

## 03 What the service provider must validate

- The XML signature using an explicitly trusted IdP certificate and an allowed algorithm.
- The issuer, destination, recipient, audience, and subject confirmation method.
- `NotBefore` and `NotOnOrAfter` with small, monitored clock tolerance.
- `InResponseTo` for SP-initiated flows and one-time use or replay detection.
- The exact signed assertion or response element used for authorization, preventing XML signature-wrapping mistakes.
- Expected NameID and attribute formats, with authorization based on stable controlled values rather than a display name.

Use a maintained SAML library and secure XML parser. Disable external entities and unsafe transformations. Do not implement signature handling with ad hoc string or DOM searches.

## 04 Signing, encryption, and TLS

A digital signature protects integrity and authenticates the issuer when the key is trusted. Encryption hides assertion contents from the browser and intermediaries but does not replace signature validation. TLS protects transport and endpoints. Requirements depend on the binding and threat model, but browser-delivered assertions must have integrity protection.

## 05 Operational security checklist

1. Exchange metadata through an authenticated process and pin trust to expected entity IDs and certificates.
2. Plan certificate rotation with an overlap period, monitoring, and rollback. Never disable signature validation to fix an expired certificate.
3. Require phishing-resistant MFA and conditional access at the IdP for sensitive applications.
4. Send only required attributes, map groups narrowly, and test removal of access when users change roles or leave.

5. Protect IdP administrator accounts, signing keys, and metadata publishing; compromise can affect every connected SP.
6. Log authentication request IDs, issuer, subject mapping, application, outcome, and relevant validation failures without storing unnecessary assertion secrets.

## 06 SAML versus OAuth and OpenID Connect

SAML is widely used for enterprise browser SSO. OAuth 2.0 is primarily an authorization framework for delegated API access, while OpenID Connect adds identity authentication on top of OAuth 2.0. They are not interchangeable merely because all can participate in a sign-in experience. Choose based on application architecture, client type, ecosystem, and security requirements.

## 07 Source

Protocol roles, flows, assertions, and security considerations are described in the OASIS SAML 2.0 Technical Overview.


**NEED MORE HELP?**

### We are here when the guide is not enough.

Visit the Gridinsoft Support Center for current product guidance or submit a support request with the details of your case.

**Open this guide online at support.gridinsoft.com**