SAML – Security Assertion Markup Language

# What is SAML

SAML is an acronym used to describe the Security Assertion Markup Language (SAML). Its primary role in online security is that it enables you to access multiple web applications using one set of login credentials. It works by passing authentication information in a particular format between two parties, usually an identity provider (idP) and a web application.

Example of Identity Provider (idP) is

* auth0: <https://auth0.com/>
* okta: <https://developer.okta.com/docs/concepts/saml/>

Google and Microsoft etc. are SAML Service Providers, they accept SAML tokens from Identity Providers, they also accept their own authentication service,

# How does SAML work

SAML works by exchanging user information, such as logins, authentication state, identifiers, and other relevant attributes between the identity and service provider. As a result, it simplifies and secures the authentication process as the user only needs to log in once with a single set of authentication credentials. So, when the user tries to access a site, the identity provider passes the SAML authentication to the service provider, who then grants the user entry. Let's illustrate this concept with a real-world analogy.

Organizations often need to confirm your identity before granting you access. A good case is the airline industry. Before you board an aircraft, the airline needs to confirm you are who you say you are to ensure the security of other passengers. So, they verify your identity with some form of government-issued picture identification. Once they confirm that your name on your identity matches the name on your airline ticket, they then allow you to board the aircraft.

In the example below, the government is the identity provider, and the airline is the service provider. Your government-issued identification is the SAML assertion. When you apply for a government ID, you usually need to complete a form, have your picture taken, and in some circumstances, your fingerprints as well. The government (service provider) then stores these identifying attributes in their database and issues you with a physical ID associated with your identity. In the airline example, when you arrive at the gate, the airline (service provider) checks your ID (SAML) assertion. The airline accepts your ID as it contains your details, and the identity card or passport passes scrutiny as a valid document. After successful authentication, the airline then allows you to board the aircraft.

A picture containing graphical user interface

Description automatically generated

In step 7 above, once SP sends Security Context (like End User’s Driver’s License or the claim on the Claim Based Identity system) back to the browser, the browser should send it to the SP in each API call and the SP should check and verify it each time.

Reference: <https://en.wikipedia.org/wiki/Claims-based_identity>

Diagram

Description automatically generated

# How SAML work again

SAML SSO works by transferring the user’s identity from one place (the identity provider) to another (the service provider). This is done through an exchange of digitally signed XML documents.

Consider the following scenario: A user is logged into a system that acts as an identity provider. The user wants to log in to a remote application, such as a support or accounting application (the service provider). The following happens:

* The user accesses the remote application using a link on an intranet, a bookmark, or similar and the application loads.
* The application identifies the user’s origin (by application subdomain, user IP address, or similar) and redirects the user back to the identity provider, asking for authentication. This is the authentication request.
* The user either has an existing active browser session with the identity provider or establishes one by logging into the identity provider.
* The identity provider builds the authentication response in the form of an XML-document containing the user’s username or email address, signs it using an X.509 certificate, and posts this information to the service provider.
* The service provider, which already knows the identity provider and has a certificate fingerprint, retrieves the authentication response and validates it using the certificate fingerprint.
* The identity of the user is established and the user is provided with app access.

SAML SSO Flow

The diagram below illustrates the single sign-on flow for service provider-initiated SSO, i.e. when an application triggers SSO.

Diagram, schematic

Description automatically generated

Diagram of SAML SSO flow

Identity provider-initiated SSO is similar and consists of only the bottom half of the flow.

# What is SAML SSO?

SAML Single Sign-On is a mechanism that leverages SAML allowing users to log on to multiple web applications after logging into the identity provider. As the user only has to log in once, SAML SSO provides a faster, seamless user experience.

SAML SSO is easy to use and more secure from a user perspective as they only need to remember one set of user credentials. It also provides fast and seamless access to a site as every application they access does not prompt them to enter a username and password. Instead, the user logs into the identity provider and then accesses the relevant web application by clicking on its icon or navigating to the site via its URL.

SAML SSO also offers other benefits in addition to an enhanced user experience. It improves productivity for both the user and the Help Desk. Users do not need to waste time logging into multiple web applications with a unique set of credentials for each one. Consequently, they do not inundate the Help Desk with password reset requests, freeing the service team to attend to other service-related issues.

In addition to increased user satisfaction and improved productivity, SAML SSO also helps reduce costs. For example, Help Desks need to manage fewer calls. Instead of building a local authentication implementation for their solution, they can subscribe to an identity provider, reducing the labor cost of building and maintaining it internally.

Diagram

Description automatically generated

# OAuth vs SAML

OAuth and SAML are both protocols we use for allowing access. However, the primary difference between the two is that we use SAML for authentication and OAuth for authorization.

If we revisit the airline analogy, the passenger's ID is the SAML assertion, and the ticket the OAuth token. The airline uses the ID to verify the passenger’s identity before allowing them to board the aircraft. However, once the passengers are on the plane, the flight attendants use the ticket to confirm the passengers' status and entitlement. For example, they may have a first-class ticket giving them access to seats and amenities not accessible by passengers in economy.

# SAML Example

SAML uses a claims-based authentication workflow. First, when a user tries to access a site, the service provider asks the identity provider to authenticate the user. Then, the service provider uses the SAML assertion issued by the identity provider to grant the user access. Let's illustrate the workflow with an example.

* The user opens their browser and navigates to the service provider's web application, which uses an identity provider for authentication.
* The web application responds with a SAML request.
* The browser passes SAML request to the identity provider.
* The identity provider parses the SAML request.
* The identity provider authenticates the user by prompting for a username and password or some other authentication factor. NOTE: The identity provider will skip this step if the user is already authenticated.
* The identity provider generates the SAML response and returns it to the user's browser.
* The browser sends the generated SAML response to the service provider's web application which verifies it.
* If the verification succeeds, the web application grants the user access.

So as long as the user has a valid SAML Assertion then it’s authenticated.

# What encryption does SAML use?

The IdP encrypts the SAML assertion with a random symmetric key which in turn is encrypted with the SP's public key. The SP uses its private key to decrypt the symmetric key which in turn is used to decrypt the SAML assertion.

1. Encrypting the SAML assertion is optional. In most situations it isn't encrypted and privacy is provided at the transport layer using HTTPS.

2. It's an extra level of security that's enabled if the SAML assertion contains particularly sensitive user information or the environment dictates the need. HTTPS should always be used so SAML assertion encryption is on top of the security provided at the transport layer. If there are intermediate network nodes, the HTTPS traffic may be decrypted. The SAML assertion will remain encrypted from IdP through to SP regardless of any intermediate network nodes.