The Complete Guide To **Security Assertion Markup Language (SAML)**

Everything You Need To Know About SAML In One Place

strongdm

Table of Contents

What is SAML? 3
History of SAML 5
How Does SAML Work? 5
Examples of SAML 6
Importance of SAML 7
Benefits of SAML 8
Risks of SAML 9
SAML vs. Popular Alternatives 9
SAML Implementation Tutorial 10
SAML Use Cases 10
SAML Authentication with strongDM 11
Enjoy More Security and Less Friction 12

Summary

In this article, we will take a big-picture look at **Security Assertion Markup Language (SAML)** and how it compares with other online security protocols. You will discover the fundamentals of SAML, including what SAML is, what its associated benefits and risks are, and why SAML is the most popular identity federation standard in use today. By the end of this article, you'll have a clearer understanding of how SAML authentication works, the key pain points SAML addresses, and the ways enterprises can use SAML for single sign-on (SSO) to give their employees and partners seamless, secure access to business-critical applications.

What Is SAML?

SAML is a popular online security protocol that verifies a user's identity and privileges. It enables single sign-on (SSO), allowing users to access multiple web-based resources across multiple domains using only one set of login credentials.

SAML stands for **Security Assertion Markup Language**. SAML is an open standard used for authentication. It provides single sign-on across multiple domains, allowing users to authenticate only once. Users gain access to multiple resources on different systems by supplying proof that the authenticating system successfully authenticated them.

SAML is the most widely adopted federated identity standard for authentication. It works by passing a SAML token (called an assertion) containing identifying user information between the authenticating system and a system on a different domain that offers a resource. Typically, the resource is a web- or cloud-based application. Resources can be internal to an organization, externally hosted, or delivered as a service.

What Is SAML Authentication?

SAML authentication is the process of verifying the user's identity and security credentials. A user's credentials specify who the user is. At a minimum, a user's credentials must include a username and password. Depending on the level of protection desired, organizations may require additional security strategies, such as:

- Two-factor authentication (2FA) or multifactor authentication (MFA)
- An identifying image chosen by the user
- A challenge test, such as CAPTCHA, which can distinguish between a human response and machine input
- Biometrics, such as a fingerprint or retinal scan

SAML also supports authorization, which defines a user's privileges. The set of privileges assigned to an individual user typically depend on the user's role or job responsibilities. SAML authorization tells the authenticating system what type of access each user is allowed to have. SAML simplifies this process by designating an identity provider (IdP) as a single point of authentication and authorization. The identity provider has authority to grant or deny access to each user, depending on the user's identifying credentials.

SAML SSO

Although SAML covers federation, identity management, and single sign-on (SSO), its most common use in modern practice is SSO. By allowing users to access multiple applications using only one set of login credentials, SAML SSO eliminates the need to keep track of a jumbled assortment of username and password combinations. Requiring users to remember only one username and password provides a simpler, more streamlined user experience. It also makes it less likely that users will forget their passwords, use the same password for multiple applications, or choose passwords that are weak and easy to guess.

SAML SSO improves security by centralizing authentication and authorization, making it unnecessary to store a

separate set of user credentials for each individual application. It shifts the responsibility of storing sensitive information to the system that is best equipped to manage many layers of security—a smart strategy that reduces risk. In addition, this approach lowers support costs by reducing the number of Help Desk calls needed to assist users who have lost or forgotten their passwords.

It's important to note that SAML is not the same as SSO. SAML is an XML-based computer language that facilitates single sign-on. SSO is an umbrella term for any of several methods, including SAML, OpenID Connect, and OAuth, that lets you use one set of login credentials, such as a username and password, to log into multiple applications.

SAML Provider

SAML facilitates the exchange of user identity data between two types of SAML providers:

- **Identity provider** (IdP)—A SAML authority that centralizes user identity data and provides a single point of secure authentication. The IdP can be an in-house identity and access management (IAM) system or a hosted authentication SAML service provider, such as Google Apps.
- **Service provider** (SP)—A SAML consumer that offers a resource to users. Typically, that resource is a web-based application or a paid subscription service, such as a customer relationship management (CRM) platform.

SAML Assertion

A SAML assertion is a packet of information (also known as an XML document) that contains all the information necessary to confirm a user's identity, including the source of the assertion, a timestamp indicating when the assertion was issued, and the conditions that make the assertion valid. SAML defines three different types of assertion statements:

- **Authentication**— An authentication assertion affirms that a specific identity provider authenticated a specific user at a specific time.
- **Attribute**—An attribute is an identifying detail associated with a specific user. Examples of attributes include data such as the user's first name, last name, email address, phone number, X.509 public certificate file, and so on.
- **Authorization decision**—The authorization decision informs whether a specific user has been allowed or denied access to the requested resource. Typically, a SAML Policy Decision Point (PDP) issues this type of assertion when a user requests access to a resource.

A typical SAML assertion comprises a single authentication statement and an optional single attribute statement; however, in certain cases, a SAML response can contain multiple assertions.

SAML 2.0

SAML 2.0 is an XML-based authentication protocol for identity federation that provides seamless single sign-on access to Business-to-Business (B2B) and Business-to-Employee (B2E) applications. SAML 2.0 facilitates the exchange of user identity data across multiple security domains. These domains may be separate organizations or divisions within an enterprise.

- Widely adopted since its introduction in 2005, SAML 2.0 is a mature standard used primarily for enterprise and government applications.

History Of SAML

The origins of SAML trace back to 2002, when the **Organization for the Advancement of Structured Information Standards (OASIS)** adopted SAML 1.0 as an open standard. SAML 1.0 defined the earliest XML framework for exchanging authentication and authorization information. Version 1.1 immediately followed, arriving in 2003. Several years later, SAML 1.1 converged with two other standards: the Liberty Alliance Identity Federation Framework **(ID-FF)** and **Shibboleth**. ID-FF described a circle of trust and Shibboleth contributed proprietary extensions.

Thus, SAML 2.0 represents the convergence of SAML 1.1, ID-FF, and Shibboleth. While SAML 1.0 and 1.1 are similar, the differences between versions 1.1 and 2.0 are substantial. OASIS ratified SAML 2.0 in March 2005, replacing SAML 1.1. To date, SAML 2.0 remains the latest (and final) version.

How Does SAML Work?

SAML single sign-on authentication works by facilitating the exchange of user identity data between three parties:

- **User**–A human (for example, an enterprise employee) who needs to authenticate into an organization's network in order to gain access to online resources
- **Identity provider**—An in-house identity and access management (IAM) system or a third-party, cloud-based platform that centrally manages user identity information and makes authentication decisions
- **Service provider**—An SSO-based resource or service (typically, a web- or cloud-based application) that the user wants to access

With SAML SSO, users do not log into applications directly. Rather, they log into an SSO platform instead. When a user authenticates successfully, SAML gives that user access to multiple resources across multiple domains. All the SSO-based applications the user has permission to access are available from one dashboard, enabling the user to enjoy a "click-and-work" desktop environment.

When a user enters their login credentials, the service provider sends a SAML request to the identity provider. The identity provider performs SAML-based authentication to verify the user and generates a digitally signed and encrypted SAML assertion that represents the user's identity and permissions. The identity provider then sends the SAML assertion to the service provider.

Because a trust relationship between the service provider and the identity provider already exists, the service provider allows the user to access the requested resource or service. To grant access to resources, the service provider uses the identity provider's response to create and configure a session for the user.

SAML Components

Fundamentally, SAML defines the standardized markup language used to encode the data that is shared between the parties involved in the SAML process. Besides assertions, SAML includes four additional components. These comprise the set of associated transport protocols, bindings, profiles, and flows used to transfer SAML assertions between relying parties. SAML defines a relying party as any system entity that receives and accepts authentication information from another system entity.

Here are the four additional components of SAML explained in more detail:

- **Protocols**—Specially formatted XML messages that define how various entities request security data and how they respond to SAML requests. SAML supports two transport protocols: hypertext transfer protocol secure (HTTPS) and simple object access protocol (SOAP).
- **Bindings**—Formats that define how the transport protocols transfer messages. For instance, HTTP Redirect Binding defines how the system transports SAML messages using HTTP redirect messages, whereas SOAP binding defines how SAML messages are to be transported within SOAP messages.
- **Profiles**—Formats that determine how SAML assertions, protocols, and bindings are to be bundled together to enable interoperability in specific federated applications.

Flows—Processes that run when a user tries to access an SSO-enabled application from a browser. SAML supports two types of flows: IdP-initiated SSO and **SP-initiated SSO**. In IdP-initiated SSO, the identity provider authenticates the user then redirects them to the service provider. In SP-initiated SSO, the service provider redirects the user to the identity provider. After successful authentication, the identity provider redirects the user to the service provider.

Examples Of SAML

This section outlines two typical SAML authentication flow scenarios. The first SAML example is IdP-initiated SSO and the second is SP-initiated SSO.

SAML Authentication Using IdP-Initiated SSO

Enterprise workforce SSO solutions commonly use IdP-initiated SSO. Here is a SAML authentication example that illustrates how IdP-initiated SSO works:

- 1. The user enters their login credentials to gain access to their dashboard, which displays a list of icons. Each icon represents a company resource that the user has permission to access. The resource may be an internal service or an external web- or cloud-based application.
- 2. When the user requests access to a resource by clicking on an icon, the SAML identity provider generates an XML-based SAML assertion that includes the user's identity and any other relevant attributes that need to be communicated to the service provider.
- 3. The identity provider then signs the assertion. The signature contains the private key component of the public/private key pair that identifies the trusted partnership between the identity provider and the service provider.
- 4. Next, the identity provider takes one of the following actions:
    a. Sends the SAML assertion to the service provider from the user's browser
    b. Sends the service provider a reference to the SAML assertion, which the service provider then uses to retrieve the assertion

- 5.. Upon receiving the SAML assertion, the service provider validates the signature. The service provider uses the public key contained within the signature to verify that the SAML assertion came from the trusted identity provider, ensuring that the data contained within the assertion is valid.
- 6. The service provider then extracts the user's identity and any relevant attributes from the SAML assertion.
- 7. If this process is successful, the service provider grants the user access to the application.

SAML Authentication Using SP-Initiated SSO

SP-initiated SSO occurs when a user attempts to access a resource on the service provider's website before the service provider has authenticated the user. Here is an example of SAML authentication flow using SP-initiated SSO:

- 1. When the user tries to access a resource, the service provider recognizes that the user has not yet established an active session and redirects the user to the identity provider for authentication by sending a SAML request. Optionally, the service provider may also provide the address of the web page (URL) the user attempted to access to the identity provider.
  Note: If the service provider utilizes multiple identity providers for SAML SSO, it must first determine which identity provider supports that specific user. To make that determination, the service provider can do one of the following:
    a. Ask the user to supply their email address, and then use the email domain to identify the appropriate identity provider.
    b. Display a list of supported identity providers and ask the user to choose the appropriate one.
    c. Check the resource URL to determine whether it is unique to one identity provider.
    d. Use a cookie to determine the appropriate identity provider. In this case, the service provider would need to have placed the cookie in the user's browser the first time the user signed onto the identity provider's website. The service provider could then use that same cookie to process subsequent sign-on requests.
- 2. The identity provider authenticates the user. After creating the SAML assertion (as in the IdP-initiated example above), the identity provider redirects the user to the service provider. If the service provider sent a URL to the identity provider in step 1, the identity provider also returns the URL to the service provider.
- 3. Upon receiving the SAML assertion, the service provider validates the signature. As in the previous example, the service provider uses the public key contained within the signature to verify that the SAML assertion came from the trusted identity provider, ensuring that the data in the assertion is valid.
- 4. The service provider then extracts the user's identity and any relevant attributes from the SAML assertion.
- 5. If authentication is successful, the service provider grants the user access to the application.

Importance Of SAML

Even though newer open standards, such as Open Authorization (OAuth 2.0) and OpenID Connect (OIDC), are gaining in popularity, SAML is still going strong after 20 years. A broad range of software and services provide SAML integration, including identity providers, service providers, discovery services, enhanced client or proxy (ECP) clients, metadata services, and identity broker services. SAML plays a critical role in simplifying communication between the various connect parties.

Its widespread use and growing adoption are proof that SAML continues to be relevant, even as less mature technologies gain ground. SAML is an integral component of myriad SSO implementations. Virtually all large enterprises rely on SAML SSO to enable seamless, secure SAML login to multiple applications or services using only one set of sign-on credentials. SAML allows organizations to reduce their security risk by centralizing the authentication process and sharing user identity data across multiple domains.

Significantly, SAML adoption in the SaaS industry is drifting toward 100%. Enterprises and governments are major consumers of SaaS solutions. Because **SAML is so widely used among large organizations**, SAML integration is a key requirement for these customers. Therefore, SaaS vendors must offer SAML-compatible solutions in order to win high-value contracts.

Benefits Of SAML

SAML is one of the oldest identity federation standards available today, offering a rich set of features and scalability. Its maturity and field-proven reliability make SAML an ideal security solution for large enterprises. Included among its many benefits are

- **Interoperability**—Because SAML is an open standard, it makes interoperability possible between diverse systems.
- **Improved user experience**—Users need to sign on only once to gain access to multiple web applications. SAML SSO allows for faster authentication and reduces Help Desk calls for password resets because users have fewer login credentials to remember.
- **Increased security**—SAML centralizes authentication with an identity provider, which then grants users access to various web applications. Because identity providers specialize in providing authentication, they offer more comprehensive security technologies that protect against hackers by requiring additional verification steps, such as MFA.
- **Platform neutrality**—SAML decouples the security framework from platform architectures and vendor-specific implementations. It supports service-oriented architecture (SOA) by making security less dependent on application logic.
- **Loose directory coupling**—SAML requires no user data to be maintained, nor is data synchronization required between directories.
- **Lower administrative costs**—The identity provider maintains account information across multiple services, reducing costs for service providers.
- **Risk transference**—SAML places the burden of storing sensitive information on the identity provider, which specializes in providing authentication and has the time and resources to implement more robust security measures.

Because SAML is based on XML, it is extremely flexible—but also complex and potentially tricky to implement. SAML's superior flexibility has led other federated identity standards to adopt certain elements of SAML. Another major advantage of SAML is its interoperability. SAML facilitates communication between different systems, devices, and applications, enabling them to exchange authentication information. SAML's interoperability makes it a preferred standard for enterprises and web- and cloud-based providers, particularly software-as-a-service (SaaS) solutions.

Risks Of SAML

SAML continues to enjoy a high rate of adoption, even though it is less secure than newer protocols, such as Open Authorization (OAuth 2.0) and OpenID Connect (OIDC). Despite its drawbacks, most enterprises and governments still view SAML as the preferred choice for SSO and online security, primarily because it centralizes authentication and provides a streamlined user login experience. However, SAML carries several inherent risks and design flaws, including:

- **IdP-initiated SSO dangers**—IdP-initiated SSO provides weaker assurances than SP-initiated SSO, leaving organizations open to man-in-the-middle attacks. An attacker could steal the SAML assertion and use it to log in to the service provider, gaining unauthorized access to a user's account. Alternatively, an attacker could replace a SAML assertion with a different one, forcing the user to log in as the attacker.
- **Insecure signatures**—SAML uses signatures based on computed values. This is an insecure practice that renders SAML inherently insecure by design because hackers can exploit any flaws, differences, or ambiguities in a computed value. The more complex a computation is, the greater the risk.
- **Extensive attack surface**—In SAML, there are many points where an unauthorized user could access a system and manipulate or extract data. **The more attack vectors a surface has, the more difficult it is to protect.** Contributing to this vulnerability is the fact that SAML is based on XML, a verbose meta-language that lacks semantics and hides structure—qualities that make XML hard to form, read, and parse.
- **Implementation errors**—Many developers lack the knowledge required to implement SAML. Pressure to deliver services under tight deadlines increases the risk of missing potential issues, leaving systems and organizations vulnerable to set-up errors.

Despite these flaws, SAML experts agree that most risks can be mitigated if companies have the proper tools and expertise to implement SAML SSO.

SAML Vs. Popular Alternatives

In this section, we'll examine three open-standard protocols used by modern businesses, OAuth, OIDC, and LDAP, and compare them to SAML differences between.

SAML Vs. OAuth

SAML SSO is an authentication protocol that also provides authorization by passing a SAML assertion between the identity provider and the service provider. Open Authorization (OAuth) provides authorization only and does not support SSO. OAuth provides secure delegated access, allowing an application to take certain actions or access certain resources from a server on a user's behalf. Instead of requiring the user to share their login credentials, OAuth allows the identity provider to issue tokens to third-party applications upon the user's request.

SAML Vs. OIDC

SAML is a long-trusted authentication protocol that enables users to access multiple web applications using a

single set of login credentials. Much newer than SAML, OpenID Connect (OIDC) is an authentication protocol that verifies the identity of a user who is trying to connect to a mobile or single-page web application through a secure server, such as HTTPS. Notably, OIDC uses lightweight, compact JSON Web Tokens (JWTs) that include a digital signature, whereas SAML uses XML, which is verbose and more complex.

Although SAML continues to be favored by large enterprises, not all companies need or want SAML. Those seeking an alternative to SAML SSO are pairing OIDC with OAuth. This solution offers a slightly different feature set than SAML, however, and cannot replace SAML in every use case, Nonetheless, pairing OIDC with OAuth is a viable option for many implementations.

SAML Vs. LDAP

Both SAML and Lightweight Directory Access Protocol (LDAP) support authentication, but their uses cases are completely different. SAML is a standard used for identity federation services, such as SSO. Organizations use SAML to verify users' identities and grant access to multiple web applications across domains. In contrast, LDAP is a standard used for communicating with in-house directory services databases. It allows organizations to verify users' identities and grant access to on-premises servers, applications, and even some devices.

SAML Implementation Tutorial

To implement SAML, you will need

- An account with a SAML-compatible identity provider
- An account with a SAML-compatible service provider, such as Salesforce, Slack, or Zoom

In general, the steps to implement SAML are as follows:

- 1. Configure your chosen service provider for the application you want to set up.
- 2. Configure SSO for your desired application.
- 3. Enable external authentication for the users who are to be allowed application access.
- 4. Test the configuration. When the identity provider authenticates you, it notifies the service provider. The service provider verifies the response. If valid, the service provider grants you access to the application.

SAML Use Cases

In this section, we'll explore how various types of organizations can use SAML SSO. There are three typical SAML use cases. Let's examine each one.

Enterprise Use Cases

Large enterprises usually utilize an in-house identity and access management (IAM) system to authenticate users and secure the organization's internal applications. Therefore, enterprise SAML use cases usually entail sharing user identity data between the company's existing IAM system and web applications. Examples of common enterprise use cases include

- Outbound Internet SSO to provide access to web-based, cloud, and other business-related applications
- Inbound Internet SSO to provide access to internal web applications
- Internal web SSO to provide centralized access and facilitate secure sharing of user identity data between the organization and its subsidiaries

Small And Medium-Sized Business (SMB) Use Cases

Because many SMBs don't want to invest in an in-house IAM, they typically rely on a hosted SAML provider that offers pre-integrated SSO with hundreds of popular cloud applications. One such service, Google Apps SSO, lets users sign in to all their enterprise cloud applications using their managed Google account credentials.

Service Provider Use Cases

Many service providers use SAML to offer their customers Internet SSO. Examples of common service provider use cases include:

- Inbound connections from customers wishing to enable Internet SSO for their users who require access to the service provider's applications. In this scenario, the customer acts as the IdP.
- Inbound Internet SSO connections from a hosted SAML provider, such as Google Apps
- Outbound connections to other service providers that allow users to gain access to third-party services without supplying additional login credentials. These types of connections require the secure sharing of user identity data via SAML.

SAML Authentication With StrongDM

strongDM provides a SAML server that enables organizations to connect individual users or services to the resources they require, regardless of their location. With our People-First Access Platform, enterprises get to control who can gain access to their infrastructure and specify exactly how much access each user may have.

strongDM provides the ability to integrate with identity providers to centralize infrastructure management and automate user and group provisioning with a single source of truth. You have the option to store credentials securely on our platform or **use your third-party secrets manager**. With strongDM, you can designate access controls based on roles or user attributes. And **onboarding and offboarding employees is easy**—All it takes is a few clicks to grant or revoke access to resources such as databases, servers, clusters, web applications, and clouds.

Implementing SAML can be challenging because XML is so complex. It is easy to overlook arcane details, unwittingly leaving an application vulnerable to attacks that could compromise security or user privacy. By partnering with strongDM, enterprises can offer their employees a customized SSO experience that is both seamless and secure. Each user needs to enter only one set of login credentials to gain access to multiple applications, all of which are conveniently displayed on their personal dashboard.

Enjoy More Security And Less Friction With StrongDM

SAML is the most mature of the main federated identity protocols available today. With its rich feature set and an exemplary performance record that spans over 20 years, SAML has earned a reputation as the authentication protocol of choice for large enterprises and governments.

By gaining a deeper understanding of SAML SSO, enterprise leaders can make informed decisions about how to give employees, customers, vendors, and partners seamless, secure access to business-critical infrastructure. Optimizing user authentication is a key step in eliminating complicated login procedures, mitigating security gaps that leave systems vulnerable to malicious attacks, and reducing the costs associated with technical support.

Want to learn more about using SAML integration to control access to your enterprise resources? Get a no-BS demo of strongDM today.

strongDM's infrastructure access platform gives every business secure access controls in a way folks love to use. Trusted by the Fortune 500 to fast-growing businesses like Peloton, SoFi, Chime, Yext, and Better, strongDM gives businesses the control and visibility they need at the speed they want with one platform that works for every environment. strongDM is intentionally distributed. Head to **www.strongdm.com** to learn more.