**SAML Flow**

Many organizations implement a Single Sign On (SSO) service, allowing users to log in to multiple applications using a single set of credentials. Security Assertion Markup Language (SAML) is one of the ways to implement SSO. SAML is an XML-based framework used for authentication and authorization between a Service Provider (SP) and an Identity Provider (IDP). In SAML, authentication information is exchanged via digitally signed XML documents, ensuring integrity while enabling seamless authentication.

The steps below show an example of SAML authentication.

1. The user accesses the application in the browser.
2. Next, the application redirects the user to the identity provider, requesting authentication.

**Identity Provider (IDP)**: This is an entity providing the ability to authenticate a user. An Identity Provider also holds details about the user, such as first name, last name, email and mobile number, etc.

1. Once the user authenticates, the identity provider builds the authentication response in the form of an XML document containing the user’s username or email address. This response is called a SAML Assertion. It signs it using an X.509 certificate and posts this information to the service provider.

**SAML Assertion**: A SAML Assertion is an XML document containing user information that the identity provider sends to the service provider. There are three types of SAML Assertion:

**Authentication**: This asserts to the service provider that the user is indeed authenticated with the identity provider at a particular time using a specific authentication method. It may also contain other information about the user in the form of an authentication statement.

**Attribute**: This passes the SAML attributes to the service provider. SAML attributes are specific pieces of data that provide information about the user.

**Authorization Decision**: This assertion states if the user is authorized to use a particular service or if the identity provider denied their request due to failing authentication or authorization.

1. The service provider, which already knows the identity provider and has a certificate, retrieves the authentication response and validates the certificate fingerprint.

**Service Provider**: The application that the user wants to access.

1. The identity of the user is established and the user is granted access to the application.

A diagram of a service provider

Description automatically generatedThe document below shows a raw SAML request.

Code: xml

<?xml version="1.0"?>

<samlp:AuthnRequest

xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"

AssertionConsumerServiceURL="https://ship-inlanefreight.com/sso/SAML2/POST"

Destination="https://ship-inlanefreight.com/idp/profile/SAML2/Redirect/SSO"

ID="\_cdae718238ba9c207a35cc7c70b046a0"

IssueInstant="2019-03-12T20:54:58Z"

ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"

Version="2.0">

<saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">https://shiip-inlanefreight.com/shipservice</saml:Issuer>

<samlp:NameIDPolicy AllowCreate="1"/>

</samlp:AuthnRequest>

Let's examine some of the key attributes.

**AssertionConsumerServiceURL**: This identifies where the IDP should send the SAML Response after authentication.

**Destination**: Indicates the address to which the request should be sent (IDP).

**ProtocolBinding**: This typically accompanies the AssertionConsumerServiceURL attribute and defines the mechanism by which SAML protocol messages will be transmitted.

**saml:Issuer**: Identifies the entity that generated the request message.

A raw SAML response looks like the following.

Code: xml

<?xml version="1.0" encoding="UTF-8"?>

<samlp:Response Destination="https://shibdemo-sp1.test.edu/Shibboleth.sso/SAML2/POST" ID="\_2af3ff4a06aa82058f0eaa8ae7866541" InResponseTo="\_cdae718238ba9c207a35cc7c70b046a0" IssueInstant="2019-03-12T20:54:54.061Z" Version="2.0" xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol">

<saml:Issuer Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity" xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">https://ship-inlanefreight.com/idp/shipservice</saml: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/2000/09/xmldsig#rsa-sha1"/>

<ds:Reference URI="#\_2af3ff4a06aa82058f0eaa8ae7866541">

<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/2000/09/xmldsig#sha1"/>

<ds:DigestValue>Se+WwXd5r44J56LauTz/wnP3jWg=</ds:DigestValue>

</ds:Reference>

</ds:SignedInfo>

<ds:SignatureValue>--snip--</ds:SignatureValue>

<ds:KeyInfo>

<ds:X509Data>

<ds:X509Certificate>--snip--</ds:X509Certificate>

</ds:X509Data>

</ds:KeyInfo>

</ds:Signature>

<samlp:Status>

<samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>

</samlp:Status>

<saml:Assertion xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ID="\_e0acf8ced7e2cafc7c65b2c097842486e0838d76e0" IssueInstant="2019-03-13T22:44:33Z" Version="2.0">

<saml:Issuer>https://ship-inlanefreight.com/idp/shipservice</saml: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/2000/09/xmldsig#rsa-sha1"/>

<ds:Reference URI="#\_e0acf8ced7e2cafc7c65b2c097842486e0838d76e0">

<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/2000/09/xmldsig#sha1"/>

<ds:DigestValue>kDAb3x6EFvA9VblqwbIFcCnLQvo=</ds:DigestValue>

</ds:Reference>

</ds:SignedInfo>

<ds:SignatureValue>--snip--

</ds:SignatureValue>

<ds:KeyInfo>

<ds:X509Data>

<ds:X509Certificate>--snip--</ds:X509Certificate>

</ds:X509Data>

</ds:KeyInfo>

</ds:Signature>

<saml:Subject>

<saml:NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient" SPNameQualifier="https://shibdemo-sp1.test.edu/shibboleth">\_29b7a1a396d841b09fcf2b0bd8ce88fed6ad70e1a7</saml:NameID>

<saml:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">

<saml:SubjectConfirmationData InResponseTo="\_cdae718238ba9c207a35cc7c70b046a0" NotOnOrAfter="2019-03-13T22:49:33Z" Recipient="https://ship-inlanefreight.com/sso/SAML2/POST"/>

</saml:SubjectConfirmation>

</saml:Subject>

<saml:Conditions NotBefore="2019-03-13T22:44:03Z" NotOnOrAfter="2019-03-13T22:49:33Z">

<saml:AudienceRestriction>

<saml:Audience>https://ship-inlanefreight.com/idp/shipservice</saml:Audience>

</saml:AudienceRestriction>

</saml:Conditions>

<saml:AuthnStatement AuthnInstant="2019-03-13T22:44:33Z" SessionIndex="\_a52c3c1242663b44b706523f0a2ada454eb997e40a" SessionNotOnOrAfter="2019-03-14T06:44:33Z">

<saml:AuthnContext>

<saml:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:Password</saml:AuthnContextClassRef>

</saml:AuthnContext>

</saml:AuthnStatement>

<saml:AttributeStatement>

<saml:Attribute Name="uid" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

<saml:AttributeValue xsi:type="xs:string">John</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute Name="mail" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

<saml:AttributeValue xsi:type="xs:string">John@inlanefreight.com</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute Name="first\_name" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

<saml:AttributeValue xsi:type="xs:string">John</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute Name="last\_name" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

<saml:AttributeValue xsi:type="xs:string">Gabreil</saml:AttributeValue>

</saml:Attribute>

</saml:AttributeStatement>

</saml:Assertion>

</samlp:Response>

Let's examine some essential attributes of a SAML response.

**ds:Signature**: This is an XML Signature that protects the integrity of and authenticates the issuer of the assertion. The SAML assertion may also be signed but it doesn’t have to be. The example above contains two ds:Signature elements. The reason is that one is the message’s signature, while the other is the Assertion’s signature.

**saml:Assertion**: Contains information about the user’s identity and potentially other user attributes.

**saml:Subject**: Specifies the principal that is the subject of all of the statements in the assertion.

**saml:StatusCode**: A code representing the status of the activity carried out in response to the corresponding request.

**saml:Conditions**: This specifies conditions such as the time an Assertion is valid for and that the Assertion is addressed to a particular Service Provider.

**saml:AuthnStatement**: States that the IDP authenticated the Subject of the Assertion.

**saml:AttributeStatement**: Contains attributes that describe the Subject of the Assertion.