SAML Support for PPO

January 2015

Contents

Why SAML Support for PPO

Support in PPO for Security Assertion Mark-up Language (SAML) is important for a number of reasons. SAML support enables us to implement a single sign-on mechanism that will allow our clients' users to authenticate against their enterprise systems, instead of using our normal PPO security. This gives their network administrators better control over users access to PPO, as well as the added peace of mind that user identities are maintained from within the enterprise. SAML is an internet standard that has been widely adopted by major software vendors including SAP, Microsoft, IBM and Oracle. These vendor products typically act in the role of identity providers, service providers, middleware and discovery services.

Introduction to SAML

Security Assertion Mark-up Language (SAML) was introduced as an internet open standard to allow for the implementation of browser based single sign-on (SSO) mechanisms. It is an XML-based protocol used to securely exchange user information between service providers (SP) and identity providers (IdP), typically for log on and log off operations. For PPO's implementation, the service provider initiated log-in pattern was used. The diagram in Fig. 1 below illustrates the flow of information during the process.

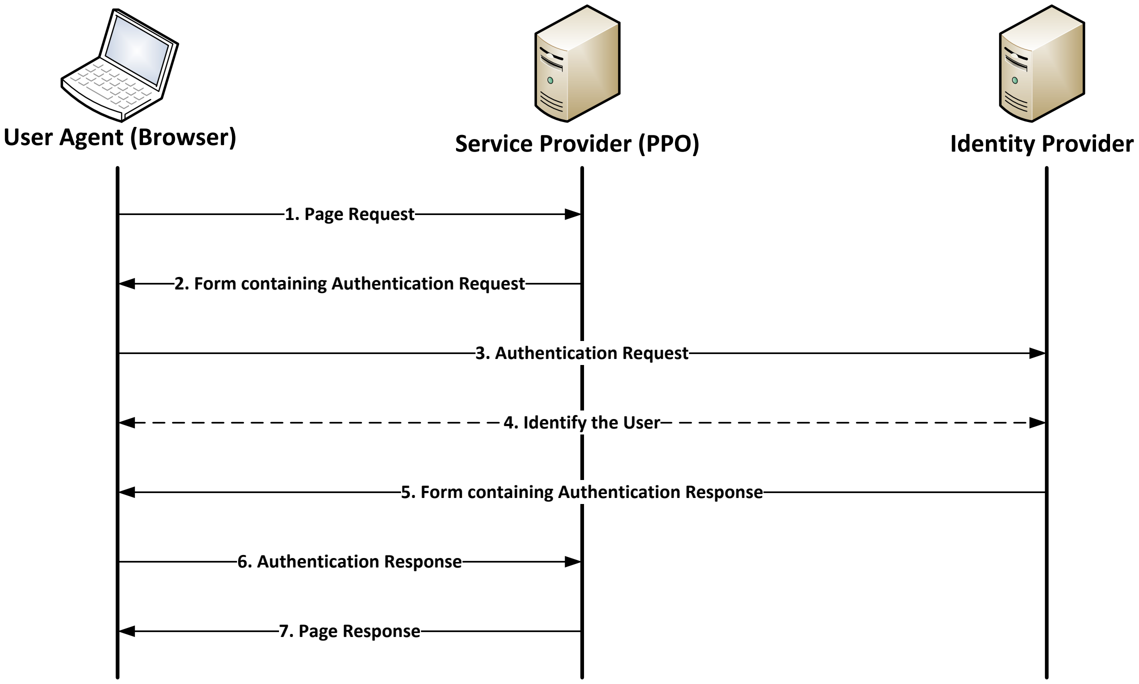

User Agent (Browser)
Service Provider (PPO)
Identity Provider
1. Page Request
2. Form containing Authentication Request
3. Authentication Request
4. Identify the User
5. Form containing Authentication Response
6. Authentication Response
7. Page Response

Figure 1 – Service provider initiated SSO

1. A user makes a page request to PPO. PPO discovers that the user is not authenticated and initiates the SSO process.

2. PPO responds to the user's browser with a hidden form containing an authentication request to be forwarded to the IdP.

```
<form method="post" action="https://idp.example.org/SAML2/SSO/POST">

  <input type="hidden" name="SAMLRequest" value="[Encoded authentication request]" />

  <input type="hidden" name="RelayState" value="[Reference to the resource requested]" />

  <input type="submit" value="Submit" />

</form>
```

Figure 2 – Sample hidden form

3. The user's browser then automatically posts the hidden form containing the authentication request to the identity provider's publically available endpoint.

```
<samlp:AuthnRequest

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

  Version="2.0"

  ID="_17bbce34-e607-4294-a856-a9ba4403abf8"

  IssueInstant="2014-03-16T12:56:01.49Z"

  Destination="[Identity provider's endpoint]"

  ForceAuthn="false"

  IsPassive="false">

  <saml:Issuer>www.ppolive.com</saml:Issuer>

  <samlp:NameIDPolicy AllowCreate="true" />

</samlp:AuthnRequest>
```

Figure 3 – Sample authentication request

4. The identity provider receives the authentication request and proceeds if it is satisfied that it originates from a trusted source. The mechanism for determining the user's identity could differ from one IdP to the next. Typically the user will receive a logon form as response.

5. Once authenticated, the identity provider replies to the browser with an authentication response.

6. The browser automatically posts the authentication response back to PPO. PPO parses the response received. The identity contained within the assertion in the response is used to identify the user in PPO and establish a session.

```
<samlp:Response
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  Version="2.0"
  ID="_4d2f3736-2f27-4398-a35c-3cc41466dfcf"
  IssueInstant="2014-03-16T12:56:22.143Z"
  Destination="https://www.ppolive.com/instance/samlauth.aspx"
  Consent="urn:oasis:names:tc:SAML:2.0:consent:unspecified" >
  <saml:Issuer>[Identity provider's endpoint]</saml:Issuer>
  <samlp:Status>
    <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success" />
  </samlp:Status>
  <saml:EncryptedAssertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
    <!-- Encrypted assertion here -->
  </saml:EncryptedAssertion>
</samlp:Response>
```

Figure 4 – Sample response

```
    <saml:Assertion

      xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"

      Version="2.0"

      ID="_f5da046c-3c39-46e5-953b-3dca125d67b8"

      IssueInstant="2014-03-16T12:56:22.141Z">

      <saml:Issuer>[Identity provider's endpoint]</saml:Issuer>

      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">

        <!-- Signature -->

      </ds:Signature>

      <saml:Subject>

        <saml:NameID>[User's identity here]</saml:NameID>

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

          <saml:SubjectConfirmationData NotOnOrAfter="2014-03-16T13:01:22.143Z"
Recipient="https://www.ppolive.com/instance/samlauth.aspx" />

        </saml:SubjectConfirmation>

      </saml:Subject>

      <saml:Conditions NotBefore="2014-03-16T12:56:20.495Z" NotOnOrAfter="2014-03-16T13:56:20.495Z">

        <saml:AudienceRestriction>

          <saml:Audience>https://www.ppolive.com/instance</saml:Audience>

        </saml:AudienceRestriction>

      </saml:Conditions>

      <saml:AuthnStatement AuthnInstant="2014-03-16T08:40:37.717Z" SessionIndex="_f5da046c-3c39-46e5-
953b-3dca125d67b8">

        <saml:AuthnContext>

          <saml:AuthnContextClassRef>urn:federation:authentication:windows</saml:AuthnContextClassRef>

        </saml:AuthnContext>

      </saml:AuthnStatement>

    </saml:Assertion>
```

Figure 5 – Sample decrypted assertion contained in response

7. PPO responds with the page that was originally requested by the user.

All traffic is encrypted and communication between the browser and PPO is secure over HTTPS.

PPO Implementation

ComponentSpace SAML v2.0 for .NET

To avoid having to code a complete SAML 2.0 API, we have decided to use a well-known third party .NET component instead. ComponentSpace supplies a .NET API that supports all the features, protocols and bindings required by PPO.

This API composes all messages sent to the identity provider, as well as decomposing responses. It also handles signatures within messages and the encryption and decryption thereof.

SAML Security mode

In order to seamlessly replace the default authentication mechanism of PPO with that of a single sign-on solution, we had to introduce a new security mode. The security mode dictates PPO’s behaviour in certain circumstances. For details, refer to *‘What can a user expect?*

The default security mode for PPO is configured in our root application configuration.

```
<configuration>

  <appSettings>

    <add key="SECURITYMODE" value="4"/>

  </appSettings>

</configuration>
```

Figure 6 – Web.config configuration

In order to change the security mode for an individual instance of PPO to an alternative, the instance’s web.config file needs to be changed as stipulated in *‘How to configure PPO for SAML integration’*.

In conjunction with this security mode, two other configuration settings have been added to refer to the login and logout endpoints pre-arranged with the identity provider. Refer to *‘How to configure PPO for SAML integration’* for details.

PPO.Web.Security.Saml

Code that we have introduced to utilise the ComponentSpace SAML API can be found in the namespace PPO.Web.Security.Saml. Classes have been added to handle authentication and logout requests, and responses from the identity provider.

SamlAuth.aspx

SamlAuth.aspx is an active server page that serves as the target endpoint in PPO.

The URL to this page is https://www.ppolive.com/[instancename]/samlauth.aspx.

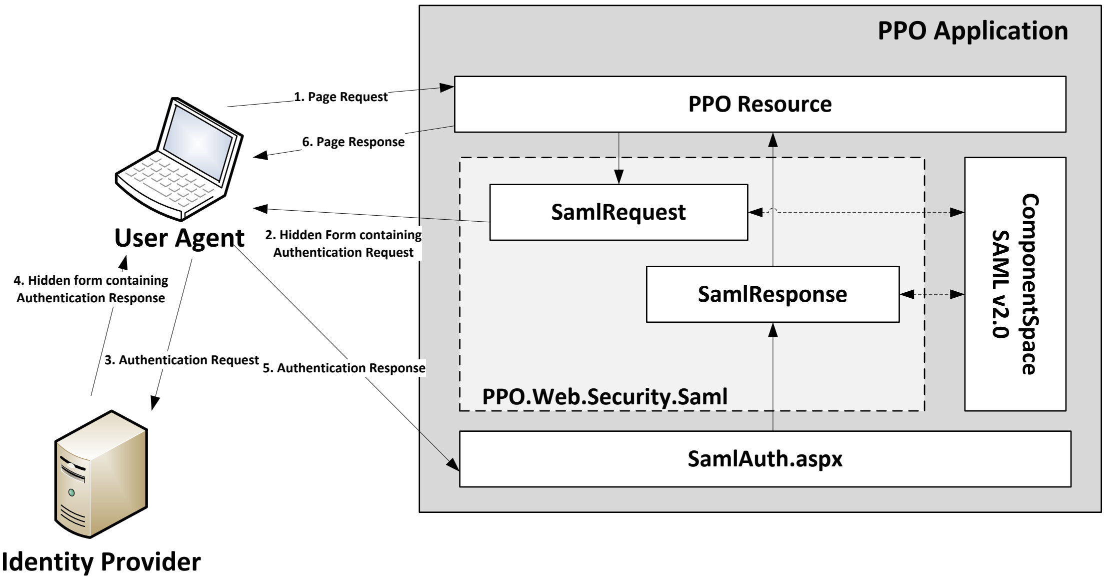


Figure 7 – PPO.Web.Security.Saml in context

How to configure PPO for SAML integration

Setup a trust relationship

Before any integration can take place between PPO and an identity provider, a trust relationship is configured between both parties. A public key for PPO's certificate is supplied to the identity provider along with XML metadata that includes some additional information about the service provider.

```
    <md:EntityDescriptor
      entityID="https://www.ppolive.com/instance"
      xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata">
      <md:SPSSODescriptor
          WantAssertionsSigned="true"
          protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
          <md:KeyDescriptor>
              <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
                  <ds:X509Data>
                      <ds:X509Certificate>[Encrypted public key here]</ds:X509Certificate>
                  </ds:X509Data>
              </ds:KeyInfo>
          </md:KeyDescriptor>
          <md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameidformat:unspecified</md:NameIDFormat>
          <md:AssertionConsumerService
            index="1"
            Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
            Location="https://www.ppolive.com/instance/samlauth.aspx"/>
      </md:SPSSODescriptor>
      <md:Organization>
          <md:OrganizationName xml:lang="en">PPO</md:OrganizationName>
          <md:OrganizationDisplayName xml:lang="en">Project Portfolio
Office</md:OrganizationDisplayName>
          <md:OrganizationURL xml:lang="en">www.projectportfoliooffice.com</md:OrganizationURL>
      </md:Organization>
      <md:ContactPerson contactType="technical">
          <md:GivenName>James</md:GivenName>
          <md:SurName>Hekma</md:SurName>
          <md:EmailAddress>jimmy@projectportfoliooffice.com</md:EmailAddress>
      </md:ContactPerson>
    </md:EntityDescriptor>
```

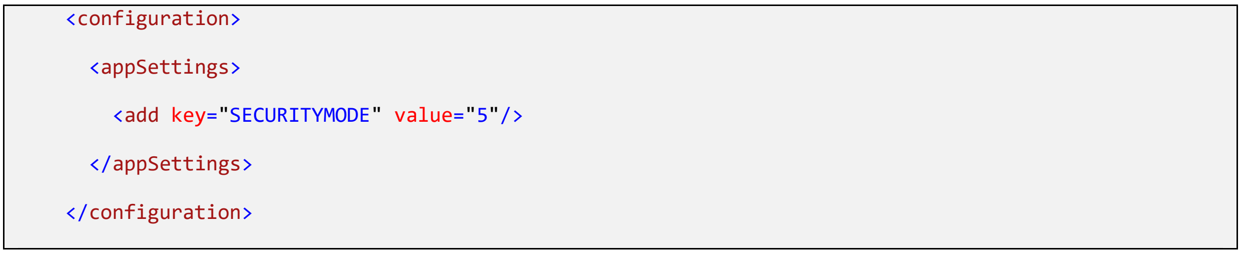
Figure 8 – SAML metadata

Important things to point out in the metadata:

- KeyDescriptor – contains the public key of the service provider's certificate
- NameIDFormat – indicates what name identifier format the service supports
- AssertionConsumerService Location – the pre-arranged endpoint of the service provider's service

PPO Configuration

PPO supports two security modes - Default and SAML. In order to activate the SAML security mode for a PPO instance, the following setting is required in the instance's web.config:



```
<configuration>
  <appSettings>
    <add key="SECURITYMODE" value="5"/>
  </appSettings>
</configuration>
```

Figure 9 – Instance web.config

The endpoints to the identity provider are configured in an instance's configuration settings located in its database:

- SINGLESIGNONSERVICEURL – contains the URL to the pre-arranged endpoint of the identity provider's login service
- SINGLESIGNOUTSERVICEURL – contains the URL to the pre-arranged endpoint of the identity provider's logout service

PPO Cross-site authentication

For the SAML security mode, access is not allowed to the normal PPO login page (login.aspx). For cross-site authentication purposes, an alternative logon page has been created, that excludes the 'forgot password' feature (xsite.aspx).

Troubleshooting

Setting up SAML integration for a PPO instance requires close cooperation with the identity provider's administrators.

The following are some errors that could be expected when the IdP responds back to PPO:

Message	Cause	Action
Your identity could not be matched against a user of our system [Identity: joe.bloggs@gmail.com]	No user exist in PPO that matches the identity indicated	Add a PPO user with a user name that matches the specified identity.
User marked as inactive [User name: joebloggs@gmail.com]	The user that matches the identity is marked as inactive in PPO.	Mark the specified PPO user as active
The SAML assertion doesn't contain an identifier	Response was received successfully but it does not contain an identifier	If this is the case for all users, it might indicate a problem with the configuration on the IdP end. Look at the configured NameIDFormat. If this only applies to individual cases, it might indicate that the information for this user is incomplete on the IdP end
Failed to receive SAML response by HTTP post	Thrown when receiving the response from the IdP fails	Re-try. When the problem persists, contact the IdP administrator
The SAML assertion signature failed to verify.	The signature used for encrypting the response failed the verification process	Indicates that the IdP has incorrectly signed the SAML response message

What can a user expect?

There are a number of changes that a user can expect when a PPO instance is configured for SAML authentication.

Logon

When user makes a request to a PPO page via their browser and is unauthenticated, the default login mechanism will be replaced with an alternative mechanism, usually in the form of a logon page similar to that of PPO.

Some users belonging to large enterprises and federated systems might use their normal way of authentication like for their other systems they are already accustomed to.

Accessing the default login page (login.aspx) is not allowed. Users are rather redirected to their home page. This will also be the case for users that have previously bookmarked this page.

Log Out

The default behaviour for the Log Out button on the PPO menu bar, is to clear the user's session and to redirect to the login page.

With SAML authentication configured, the user's session is still cleared, but the user is directed to the IdP portal as well where further clean-up procedures could be required.

No more passwords

Seeing that logon credentials are being maintained by the identity provider, there is no need to do so in PPO. Password related functionality that is disabled, includes:

- The exclusion of temporary passwords from welcome emails
- The ability to reset user's passwords
- The ability for a user to change their own password

SOAP web service integration

Seeing that SAML is browser based, this authentication scheme cannot be used when consuming PPO's SOAP web service as it requires a user to supply credentials through a web browser and integration is usually performed by back-end processes.

Authentication is however still possible with a PPO's default authentication scheme and a user account specifically configured for this purpose. Assistance is required from the PPO technical team to set this up as it requires switching between authentication schemes with some downtime as a result. This could be disruptive to active users and coordination between the customer and PPO support staff is therefore essential.

Please contact PPO support if you require assistance with this.

Microsoft Project Add-In

SAML is constrained to web browsers accessing web based applications such as PPO since it relies on a mechanism of redirects to allow the user to authenticate against the Identity Provider.

The SAML standard also does not enforce any particular authentication mechanism, so the Identity Provider is free to implement any of a variety of mechanisms such as forms based authentication, Windows integrated security or token based authentication.

What this means is that it is not possible for a non-web based application such as the MSP Add-In to implement SAML authentication in the normal way and therefore the add-in currently does not work on PPO instances that use Single-Sign-On.

The industry has recognised this problem and various efforts are currently under way to allow SAML to work seamlessly on all platforms. Some possible solutions include using a combination of OAuth and SAML as well as using SAML back-channel communication. We are carefully watching progress in this area and will consider implementing a solution for the add-in as soon as we believe that a workable solution is available.

The work-around used for calling SOAP web-services is not practical in this case since it would require users to maintain dual identities (a SAML identity as well as a PPO identity) which would defeat the purpose of single-sign-on.

References

SAML 2.0 – Wikipedia - http://en.wikipedia.org/wiki/SAML_2.0
ComponentSpace SAML v2.0 - http://www.componentspace.com/SAMLv20.aspx