Configure Cora SeQuence for SAML 2.0 Authentication

Last Modified on 09/16/2022 4:50 am EDT

Overview

SAML 2.0 is an XML-based protocol that uses security tokens containing assertions to pass information about a principal (usually an end user) between a SAML authority, named an Identity Provider, and a SAML consumer, named a Service Provider. SAML 2.0 enables web-based, cross-domain single sign-on (SSO), which helps reduce the administrative overhead of distributing multiple authentication tokens to the user.

When you use SAML authentication, Cora SeQuence supports single identity provider and service provider initiated sign in and sign out only.

Prerequisites

- Name of identity provider
- Name of service provider
- SingleSignOnServiceUrl
- SingleSignOutServiceUrl
- Binding (POST or Redirect)
- Copy of the idp signing and encryption public certificate
- Claim type that uniquely identifies users (for claims configuration)
- Is the identity provider response signed?
- is the identity provider assertion signed?
- Does the service provider sign the request?
- Publicly signed certificate (if necessary)

For Bindings:

- You can send Authentication Requests using HTTP-REDIRECT or HTTP-POST.
- Consumer Assertion Service endpoint only supports HTTP-POST binding.
- HTTP-Artifact is not supported.

Identity provider and Token-related configuration

- To configure the identity provider for the SAML integration, you can share with the IT team an SAML metadata file from Cora SeQuence. To get the metadata file, browse to in the Cora SeQuence environment and save the result.
- Configure the Consumer Assertion Service endpoint by appending to it with a question mark (?) *binding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST* (i.e. https://administration.pnmsoft.com/authservices/acs?binding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST)
- The token `<subject>` element must contain a `<NameID>` element.

IIS configuration

- Enable only Anonymous Authentication for the Administration site and Flowtime site.
- Enable claims-based authentication in the `web.config` file.

Add Configuration Sections to the web.config File

Add the following section under the `<configuration>` `<configSections>` elements.

```
<configSections>
...
 <sectionGroup name="sequence.engine" type="PNMsoft.Sequence.Configuration.WorkflowEngineConfigurationSectionGroup, PNMsoft.Sequence, Version=8.0.0.0, Culture=neutral, PublicKeyToken=0a1a1b90c1c5dca1">
 ...
 <sectionGroup name="identity" type="PNMsoft.Sequence.Configuration.IdentityConfiguration,PNMsoft.Sequence.IdentityModel.v8, Version=8.0.0.0, Culture=neutral, PublicKeyToken=0a1a1b90c1c5dca1">
   <section name="saml" type="PNMsoft.Sequence.IdentityModel.Configuration.SamlConfigurationSection, PNMsoft.Sequence.IdentityModel.Saml2, Version=8.0.0.0, Culture=neutral, PublicKeyToken=0a1a1b90c1c5dca1"/>
 </sectionGroup>
 ...
 </sectionGroup>
...
 <section name="system.identityModel" type="System.IdentityModel.Configuration.SystemIdentityModelSection, System.IdentityModel, Version=4.0.0.0, Culture=neutral, PublicKeyToken=B77A5C561934E089" />
 <section name="system.identityModel.services" type="System.IdentityModel.Services.Configuration.SystemIdentityModelServicesSection, System.IdentityModel.Services, Version=4.0.0.0, Culture=neutral, PublicKeyToken=B77A5C561934E089" />
...
</configSections>
```

Add HttpModules to the web.config file

Add the following section under the `<system.webServer>` `<modules>` elements. Make sure these modules are list in the list.

```
<system.webServer>
...
  <modules>
    <!-- Other modules already configured in web.config should be here -->
    <add name="SessionAuthenticationModule" type="System.IdentityModel.Services.SessionAuthenticationModule, System.IdentityModel.Services, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" preCondition="managedHandler" />
    <add name="SamlAuthenticationModule" type="PNMsoft.Sequence.IdentityModel.Services.SamlAuthenticationModule, PNMsoft.Sequence.IdentityModel.Saml2, Version=8.0.0.0, Culture=neutral, PublicKeyToken=0a1a1b90c1c5dca1" preCondition="managedHandler" />
  </modules>
...
</system.webServer>
```

Add SAML 2.0 configurations to the web.config file

Add the following section under the `<configuration>` `<sequence.engine>` elements.

```
<identity>
    <saml>
     <serviceProvider name="https://mydomain.com/myapplication"
          description="Sequence Service Provider"
          assertionConsumerServiceUrl="/AuthServices/acs"/>
     <partnerIdentityProviders>
      <add name="https://sts.windows.net/1eb2d3db-45bd-67a8-91a2-3456ab78be9f/"
          description="Azure"
          singleSignOnServiceUrl="https://login.microsoftonline.com/1eb2d3db-45bd-67a8-91a2-3456ab78be9f/saml2
"
          singleLogoutServiceUrl="https://login.microsoftonline.com/1eb2d3db-45bd-67a8-91a2-3456ab78be9f/saml2
"
          singleSignOnServiceBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"/>
     </partnerIdentityProviders>
    </saml>
</identity>
```

SAML Configuration Attributes

Name	Description	Required	Default Value
modulePath	Relative path of SAML authentication endpoint.	Yes	/AuthServices
samlCommandFactoryType	Extension point to enable creation of your own commands to handle sign in and sign out.	Yes	PNMsoft.Sequence.IdentityModel.Commands.CommandFactory, PNMsoft.Sequence.IdentityModel.Saml2, Version=8.0.0.0, Culture=neutral, PublicKeyToken=0a1a1b90c1c5dca1
returnUrl	Absolute URL the user is redirected to after sign in.	No	Cora SeQuence Flowtime URL

serviceProvider Configuration Attributes

Name	Description	Required	Default Value
assertionConsumerServiceUrl	Assertion Consumer Service URL.	Yes	/AuthServices/acs
name	Generally, the application's URL.	Yes	-
description	Service Provider description.	No	-
localCertificateFile	Specifies the X.509 certificate file for this service provider. The certificate file name can be an absolute path or a path relative to the application folder.	No	-

Name	Description	Required	Default Value
localCertificatePassword	Specifies the password with the X.509 certificate file for this service provider. Certificate files (.pfx) that include the private key should be protected by password. Certificate files (.cer) that do not include a private key are not password protected. The certificate password must be kept secure. In a test environment using a test certificate, specifying the password using the LocalCertificatePassword attribute is acceptable. For a production certificate, the password should be stored encrypted in the web.config file. Refer to the LocalCertificatePasswordKey attribute for more details.	No	-
localCertificatePasswordKey	Specifies the web.config file's appSettings key for the certificate file password. For example, if the LocalCertificatePasswordKey attribute value is localCertificatePassword, then under the web.config file's appSettings section, an entry with the name localCertificatePassword is expected, and the entry value is used as the password. By encrypting the appSettings section using the aspnet_regiis utility, the certificate file password is secured.	No	-
localCertificateStoreLocation	Specifies the X.509 certificate store (LocalMachine or CurrentUser).	No	LocalMachine
localCertificateSerialNumber	Specifies the X.509 certificate by serial number for this service provider.	No	-
localCertificateThumbprint	Specifies the X.509 certificate by thumbprint for this service provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-
localCertificateSubject	Specifies the X.509 certificate by subject name for this service provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-

partnerIdentityProviders Configuration Attributes

Name	Description	Required	Default Value
name	Generally, the identity provider's URL or STS.	Yes	-
description	Description of the identity provider.	No	-

Name	Description	Required	Default Value
disableInResponseToCheck	Controls whether to validate that the SAML response token is a response to a request originated by Cora SeQuence. Add this attribute and set it to "true" when IdP-Initiated SSO is required.	No	false
singleSignOnServiceUrl	Specifies the application's single sign-on service URL. SAML authentication requests will be received at this URL.	Yes	-
singleSignOnServiceBinding	Specifies the transport binding to use when sending authentication requests to the partner identity provider's SSO service.	No	urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST
signAuthnRequest	Specifies whether authentication requests sent to the partner identity provider should be signed.	No	false
wantSAMLResponseSigned	Specifies whether the SAML response from the partner identity provider should be signed.	No	false
wantAssertionSigned	Specifies whether the SAML assertion from the partner identity provider should be signed.	No	false
wantAssertionEncrypted	Specifies whether the SAML assertion from the partner identity provider should be encrypted.	No	false
partnerCertificateFile	Specifies the X.509 certificate file for this identity provider. The certificate file name can be an absolute path or a path relative to the application folder.	No	-
partnerCertificateStoreLocation	Specifies the X.509 certificate store (LocalMachine or CurrentUser).	No	LocalMachine

Name	Description	Required	Default Value
partnerCertificateSerialNumber	Specifies the X.509 serial number for this provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-
partnerCertificateThumbprint	Specifies the X.509 certificate by thumbprint for this identity provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-
partnerCertificateSubject	Specifies the X.509 certificate by subject name for this identity provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-
secondaryPartnerCertificateFile	Specifies the X.509 certificate file for this provider. The certificate file name may be an absolute path or a path relative to the application folder.	No	-
secondaryPartnerCertificateStoreLocation	Specifies the X.509 certificate store (LocalMachine or CurrentUser).	No	LocalMachine
secondaryPartnerCertificateSerialNumber	Specifies the X.509 certificate by serial number for this provider. The certificate is retrieved from the local computer’s X.509 certificate store.	No	-
secondaryPartnerCertificateThumbprint	Specifies the X.509 certificate by thumbprint for this provider. The certificate is retrieved from the local computer’s X.509 certificate store.	No	-
secondaryPartnerCertificateSubject	Specifies the X.509 certificate by subject name for this provider. The certificate is retrieved from the local computer’s X.509 certificate store.	No	-

Name	Description	Required	Default Value
tertiaryPartnerCertificateFile	Specifies the X.509 certificate file for this provider. The certificate file name may be an absolute path or a path relative to the application folder.	No	
tertiaryPartnerCertificateStoreLocation	Specifies the X.509 certificate store (LocalMachine or CurrentUser).	No	LocalMachine
tertiaryPartnerCertificateSerialNumber	Specifies the X.509 certificate by serial number for this provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-
tertiaryPartnerCertificateThumbprint	Specifies the X.509 certificate by thumbprint for this provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-
tertiaryPartnerCertificateSubject	Specifies the X.509 certificate by subject name for this provider. The certificate is retrieved from the local computer's X.509 certificate store.	No	-
singleLogoutServiceUrl	Specifies the partner provider's single logout (SLO) service URL. Logout requests will be sent to the SLO service.	Yes	-
singleLogoutServiceBinding	Specifies the transport binding to use when sending logout messages to the partner provider's SLO service.	No	urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect
logoutRequestLifeTime	Specifies the NotOnOrAfter time interval for the logout request. The format is hh:mm:ss.	No	3 minutes
disableOutboundLogout	Specifies whether logout requests sent to the partner provider are not supported.	No	false

Name	Description	Required	Default Value
disableInResponseToCheck	Specifies whether the SAML message's InResponseTo should be checked. This attribute should only be set to true in test environments or to work around limitations in the partner provider.	No	false
signLogoutRequest	Specifies whether logout requests sent to the partner provider should be signed.	No	false
useEmbeddedCertificate	Specifies whether the certificate embedded in the XML signature should be used when verifying the signature. If false then a configured certificate retrieved from the certificate manager is used.	No	false
issuerFormat	Specifies the issuer format to include in SAML messages and assertions sent to the partner provider.	No	Format attribute is not included
digestMethod	Specifies the XML signature digest method.	No	
signatureMethod	Specifies the XML signature method.	No	
keyEncryptionMethod	Specifies the XML encryption key encryption method.	No	
dataEncryptionMethod	Specifies the XML encryption data encryption method.	No	
forceAuthn	**Available with Cora SeQuence V9.3** Forces the user to sign in again via the SAML SSO sign-in page even if the user has a valid session (user has already signed in to the same browser with the IDP). If set to "true," the attribute "forceAuthn=true" is added to the SAML request.	No	false

Supported Bindings

- urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect
- urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

Secondary and Tertiary Certificates

For Partner Identity Providers, you can use secondary and tertiary certificates.

For example:

- SecondaryLocalCertificateFile
- TertiaryLocalCertificateFile

Configure IdentityModel Audience

In the `web.config` file, add the following section under `<configuration>`.

```
<system.identityModel>
  <identityConfiguration>
    <audienceUris>
      <add value="https://mydomain.com/application" />
    </audienceUris>
  </identityConfiguration>
</system.identityModel>
```

Configure the Cora SeQuence Authentication Provider

Add claims authentication under the `<configuration>` `<sequence.engine>` `<authentication>` sections.

```
<claims enabled="true" signoutFromSts="false" loginUrl="~/AuthServices/signin" logoutUrl="~/AuthServices/logou
t" accessDeniedUrl="~/AccessDenied.aspx">
  <IdentityClaims>
    <add claimType="<claim type>" originalIssuer="<token issuer>" authenticationType="<sequence authenticatio
n type" />
  </IdentityClaims>
</claims>
```