SAML V2.0 Interoperability Demonstration Scenarios, Guidelines & Final Report

February 18, 2005

RSA Conference 2005
February 14-17
San Francisco, CA

Table of Contents

1 Overview

This document describes the demonstration scenarios for the SAML V2.0 interoperability demonstration that will take place at the RSA Conference February 2005. The interop demonstration will show SAML V2.0 capabilities for Web SSO and single logout. In addition, an optional use case may be implemented by vendors to showcase the ID federation capabilities in SAML V2.0. The requirements and implementation guidelines are different for the base and optional interop use cases. 9.2.4 contains the configuration data needed for the interop. 9.3 provides post-interop remarks and conclusions.

The SAML V2.0 Committee Draft 04 specifications will be used as the basis for all issues concerning SAML interoperability.

2 Interop Scenario Websites

Each participant in the RSA2005 SAML interop will have their own DNS domain. The Portal website will be hosted in an additional domain. Within their domain, each participant may host an IdP website and/or an SP website. A participant can choose to host their websites on a single machine or they can be distributed across multiple machines within their DNS domain.

For all providers, there will be a list of idP's and a list of SP's to select from for the base use cases. For those vendors who will support the optional use case scenario, there will be an additional lists of idP's that support the federation use case which will be a subset of the base use case list.

2.1 Identity Provider (idP) Website

The IdP website is the source site in a SAML Web SSO exchange. It includes an authentication authority at which users will log in. The idP website needs to support the following for the base use cases:

- Process an <AuthnRequest> from an SP,
- Initiate an authentication and once complete, allow the user to select an SP to transfer to, sending an unsolicited <Response> to that SP,
- Process a <LogoutRequest> from an SP and propagate it to the other SP's a user is logged into,
- Initiate a logout and propagate it to the SP's a user is logged into.

For the optional use cases, the idP website needs to support:

- Request federation of ID's with an SP
- Federate the ID's and return the proper <Response> to the SP
- Process a <ManageNameIDRequest> to terminate a federation

2.2 Service Provider (SP) Website

The SP website is the destination site in a SAML Web SSO exchange and hosts the participant's demo application. The SP website requires a user to be authenticated before being given access to the application. The SP website must support the following for the base use cases:

- Allow the user to select an idP for authentication,
- Initiate a login to an idP via an <AuthnRequest>,
- Accept an unsolicited <Response> from an idP confirming a user has already logged in,
- Initiate a <LogoutRequest> to an idP to log a user out from all SP's,
- Accept a <LogoutRequest> from an idP to log a user out.
- Accept a redirect from the eAuthentication Portal and process the CSID query parameter to initiate a login to an idP,
- Provide a link to the eAuthentication Portal

For the optional use cases, the SP website needs to support:

- Accept a request to federate ID's
- Authenticate a user to allow federation of ID's
- Initiate a <ManageNameIDRequest> to terminate a federation

By authenticating at an IdP website, a user will be able to access the application at the SP website through the use of a SAML Web SSO exchange without authenticating a second time at the SP website. A single logout will log a user out from all SP's the user is logged into via the associated idP.

Once the SAML Web SSO exchange completes, the demo application at the SP website should personalize the application content based on certain attribute values extracted from SAML Attribute Assertions provided by the IdP. At a minimum, the application should display a small box in some part of the web page with the following information:

- The IdP website (asserting party) that issued the assertion
- The name of the user for whom the assertion was issued
- The name and value of the each of the user attributes supplied in the assertion

If possible, the application should also provide a display of the SAML XML documents that were exchanged during the Web SSO operation.

2.3 GSA eAuthentication Portal

The GSA eAuthentication Portal will be provided by Enspier and will be in a separate domain. The Portal will allow the user to select the idP for the user to authenticate to, which is called a Credential Service in the eAuthentication spec, and the SP to go to, called an Agency Application in the eAuthentication spec. Unlike last year, this scenario will only be initiated directly from the GSA Portal, not from an SP or idP.

3 Overview of Use Cases

The Interop will include both base and optional use cases and scenarios. It is expected that all vendors support the base use cases that apply to the components they will be showing (idP and/or SP). The optional use cases can be supported at each vendor's discretion. All the use cases at the Interop will involve front channel bindings and artifacts will not be used.

3.1 Base Use Cases

The Interop will have four base SAML use cases which will show Web Single Sign-On and Single Logout. This SAML interop event is being sponsored by the GSA's eGov program. In return for their sponsorship, the interop also includes an additional "eAuthentication" use case as defined by the eGov program's eAuthentication initiative.

3.1.1 Web SSO Demonstrations

Two scenarios will be supported to demonstrate Web SSO:

1. The signon is initiated from an SP, in which case:
    a. An idP will be selected,
    b. The SP will redirect to the idP which will authenticate the user,
    c. The idP will respond back to the SP which will show the requested web page
2. The signon is initiated from an idP, in which case:
    a. The idP will authenticate the user,
    b. An SP will be selected,
    c. The user will be sent to the SP along with an unsolicited <Response> and the SP will show the requested web page

3.1.2 Web Single Logout Demonstrations

Two scenarios will be supported to demonstrate Single Logout:

1. The logout is initiated from an idP, in which case the idP will propagate the user logout from all SP sessions the user has using a HTTP Redirect.
2. The logout is initiated from an SP, in which case the SP will submit a <LogoutRequest> to the idP and the idP will propagate the user logout from all SP sessions the user has using a HTTP Redirect.

3.1.3 eAuthentication Demonstration

Two scenarios will be supported to demonstrate the eAuthentication Portal:

1. Initial site selection for an unauthenticated user:
    a. The user goes to the eAuthentication Portal and selects an SP (Agency Application) and an idP (Credential Service),
    b. The Portal redirects the user to the SP with a query parameter indicating the idP selected,
    c. An <AuthnRequest> will be sent to the selected idP via HTTP Redirect,
    d. The SP will redirect to the idP which will authenticate the user,
    e. The idP will respond back to the SP which will show the requested web page
2. After scenario 1, the user selects a subsequent SP to access:
    a. The user clicks a link on the SP and is redirected back to the eAuthentication Portal,
    b. The user selects a second SP to access,
    c. The Portal redirects the user to the subsequent SP with the CSID from the original access.
    d. The first scenario basically repeats itself but without a login request displayed to the user from the idP

3.2 Overview of Optional Use Cases

The optional use cases will focus on identity federation. Two optional use cases and two of the base use cases will be combined to form a single demo scenario. These use cases are not required for any vendor at the Interop. However, if either of these use cases is supported by a vendor, they both must be supported. The optional use cases are:

1. Federating a user between an idP and an SP
2. Defederating a user between an idP and an SP

The demo scenario is very complicated and requires a time commitment from the customer to show and explain. The demo will go as follows:

1. A signon is initiated at an SP that supports the optional use case and the user selects an idP that supports the optional use case
2. The SP sends an <AuthnRequest> to the idP
3. The SP will redirect the user to the idP which authenticates the user. The user enters a special user name that distinguishes it from the base use case user names
4. The idP sets up its part of federation and sends a <Response> to the SP
5. The SP authenticates the user, sets up its part of the federation and then presents the requested web page
6. The user clicks the logout button on the page, triggering the Single Logout base use case
7. The user then logs back into the SP via the idP, triggering the Web SSO base use case
8. The user clicks a defederation button or link on the SP web page which results in a request to the idP to defederate the ID's

4 Interop Base Use Case Scenarios

The use case scenarios described in this section are broken into groups; one group for the "Generic SAML" Web SSO and Single Logout demos and one for the eGov eAuthentication demo. The Generic SAML use cases have two main scenarios each, one when starting at the IdP website and one when starting at the SP website. The eAuthentication use case always starts at the eAuthentication Portal.

4.1 Generic SAML Demo Use Cases

The Generic SAML Demo use cases demonstrate the use of SAML V2.0 as described in the Web SSO Browser Profile of the SAML specification. Artifacts will not be used.

4.1.1 Generic SAML SP-Site-First Scenario

In the Generic SAML SP-Site-First Web SSO scenario, the user initiates the Web SSO demonstration by attempting to directly access a demo application at an SP website. The flow of events for this scenario is as follows:

1. An unauthenticated user at any browser attempts to directly connect to an application at an SP website (e.g. via a browser bookmark).
2. The SP website detects the user is unauthenticated and displays a list of all IdP websites at which the user may log in to gain access to the selected application. For each IdP in the list, a link to an IdP entity ID will be provided.

3. The user selects one of the IdP links and the SP generates a SAML <AuthnRequest> to the selected idP via an HTTP Redirect.
4. The user is required to log in at the IdP website using one of the predefined user accounts.
5. After successful authentication, the IdP website returns a SAML <Response> to the SP along with the EmailAddress, MemberLevel and CommonName attributes for that user. The Response is via an HTTP POST.
6. Upon successful completion of the Web SSO exchange, the user will be granted access to the target application, which will display the user name, subject name and attributes returned in the <Response> from the idP.

4.1.2 Generic SAML IdP-Site-First Scenario

In the Generic SAML IdP-Site-First Web SSO scenario, the user visits the IdP website, logs in, and selects a demo application at an SP website. The flow of events for this scenario is as follows:

1. An unauthenticated user at any browser visits the IdP website.
2. The user is required to log in at the IdP website using one of the predefined user accounts.
3. The IdP website displays a list of the SP resources (URLs) of all participant demo applications for which the IdP shares a SAML profile.
4. The user selects one of the SP resources and the browser is directed to the SP with a <Response> via an HTTP POST.
5. Upon successful completion of the Web SSO exchange, the user will be granted access to the target application which will display the user name, subject name and attributes sent in the <Response> from the idP.

4.2 eGov eAuthentication Use Cases

The eAuthentication use case always involves the eAuthentication Portal. The user selects an SP (known as the Agency Application) at the Portal. If the user is unauthenticated, he also selects an idP (known as the Agency Application). If the user is already authenticated, the idP from the previous authentication is used. The Portal redirects the browser to the SP resource with a query parameter of the idP. At this point, the demo will follow the SP-Site-First scenario, except that the idP will already be selected. Note that this is not how the data flow is specified in the eGov specifications, but this is how it will be supported for this interop.

4.2.1 eGov eAuthentication First Time Access Scenario

The eAuthentication first time use case is always initiated from the eAuthentication Portal. The user selects an idP (known as Credential Service) and SP (known as the Agency Application) at the Portal. The Portal redirects the browser to the SP resource with a query parameter of the idP. At this point, the demo will follow the SP-Site-First scenario, except that the idP will already be selected. The flow for this scenario is as follows:

1. An unauthenticated user accesses the eAuthentication Portal. The Portal page is not protected.

2. The user selects an Agency Application (SP) to access and a Credential Service (idP) to authenticate to.
3. The Portal redirects the browser to the SP resource, passing the idP entity ID (URL) in a query parameter labeled "CSID".
4. The SP gets the idP from the CSID parameter and generates a SAML <AuthnRequest> to the selected idP via an HTTP Redirect.
5. The user is required to log in at the IdP website using one of the predefined user accounts.
6. After successful authentication, the IdP website returns a SAML <Response> to the SP along with the EmailAddress, MemberLevel and CommonName attributes for that user. The Response is via an HTTP POST.
7. Upon successful completion of the Web SSO exchange, the user will be granted access to the target application, which will display the user name, subject name and attributes returned in the <Response> from the idP.

4.2.2 eGov eAuthentication Web SSO Scenario

The eAuthentication Web SSO use case is initiated from an SP web page. The user selects the Portal from a link on the page, and the browser is redirected to the eAuthentication Portal. At the Portal, the user selects an SP (known as the Agency Application). The Portal redirects the browser to the SP resource with a query parameter of the idP that was used to authenticate at the originating SP (probably stored as a cookie to avoid propagating this parameter). The SP will authenticate against the idP specified in the parameter, which should not require the user to explicitly authenticate again. The flow for this scenario is as follows:

1. An authenticated user at an Agency Application (SP) selects the eAuthentication Portal and the browser is redirected there. The Portal page is not protected.
2. The user selects an Agency Application (SP) to access.
3. The Portal retrieves the Credential Service (idP) that was used for the originating SP.
4. The Portal redirects the browser to the SP resource, passing the idP entity ID (URL) in a query parameter labeled "CSID".
5. The SP gets the idP from the CSID parameter and generates a SAML <AuthnRequest> to the selected idP via an HTTP Redirect.
6. The IdP recognizes the user is authenticated and returns a SAML <Response> to the SP along with the EmailAddress, MemberLevel and CommonName attributes for that user. The Response if via an HTTP POST.
7. Upon successful completion of the Web SSO exchange, the user will be granted access to the target application, which will display the user name, subject name and attributes returned in the <Response> from the idP.

4.3 Single Logout SP Initiated Scenario

1. An authenticated user clicks the Logout button on the SP web page.
2. The SP website sends a <LogoutRequest> to the idP the user authenticated to via an HTTP Redirect.

3. The idP website receives the <LogoutRequest>, determines the other SP websites the user is logged into and sends a <LogoutRequest> to those websites via HTTP Redirect.
4. Each SP website returns a <LogoutResponse> after logging the user out.
5. The idP receives <LogoutResponse> messages from the SP websites.
6. The idP returns a <LogoutResponse> to the originating SP website.

At this point, the SP web page may wish to display a status to the user.

4.4 Single Logout idP Initiated Scenario

1. An authenticated user clicks the Logout button on the idP web page.
2. The idP websiter determines the SP websites the user is logged into and sends a <LogoutRequest> to those websites via HTTP Redirect.
3. Each SP website returns a <LogoutResponse> after logging the user out.
4. The idP receives <LogoutResponse> messages from the SP websites.

At this point, the idP web page may wish to display a status to the user.

5 Base Use Case Requirements

5.1 AuthnRequest Requirements

All AuthnRequests MUST be signed. This mean that in the metadata for an idP, **IDPSSODescriptior** MUST have a **WantAuthnRequestsSigned** attribute and it MUST be set to true. Likewise in the metadata for an SP, **SPSSODescriptor** MUST have a **AuthnRequestsSigned** attribute and it MUST be set to true.

The **RequestAbstractType** must have the attributes set as follows:

- ID: NCName string; required by SAML – MUST be sent
- Version: string; required by SAML; value = "2.0" – MUST be sent
- IssueInstant: xs:dateTime; required by SAML – MUST be sent
- Destination: URI; required by SAML when using HTTP Redirect or POST bindings – MUST be sent
- Consent: values from spec – MUST NOT be sent

The **RequestAbstractType** must have the elements set as follows:

- <saml:Issuer>: IssuerType – required for all profiles we're using – MUST be sent
- Format must be omitted or must be set to "urn:…:nameid-format: entity".
- NameQualifier, SPNameQualifier, and SPProvidedID – MUST NOT be sent
- <ds:Signature> - MUST NOT be sent (Note that the redirect binding will sign the URL parameters as defined in the spec, so the message itself should not include a signature)

The <**AuthnRequest>** element must have the attributes set as follows:

- **ForceAuthn** – MUST NOT be sent
- **IsPassive** – MUST NOT be sent
- **AssertionConsumerServiceIndex** – is OPTIONAL, but if sent it MUST be set to 0 (zero)
- **AssertionConsumerServiceURL** – MUST NOT be sent

- **ProtocolBinding** – MUST NOT be sent
- **AttributeConsumingServiceIndex** – MUST NOT be sent
- **ProviderName** - OPTIONAL

The **AuthnRequest** must have the elements set as follows:

- <**saml:Conditions**> – MUST NOT be sent
- <**RequestedAuthnContext**> – MUST NOT be sent
- <**Scoping**> – MUST NOT be sent
- <**saml:Subject**> – MUST NOT be sent
- The <**NameIDPolicy>** element MUST be sent and have the following attributes:
  - a **Format** attribute MUST be sent with the value:
    - urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName
  - an **AllowCreate** attribute is OPTIONAL, but if sent MUST be set to false
  - an **SPNameQualifier** attribute MUST NOT be sent
  - an **SPProvidedID** attribute MUST NOT be sent

5.2 Subject Element

All assertions used for the base use cases in this interop will use SAML **Subject** elements formatted as X.509 Subject Names. Thus,

1. The <**NameID>** element MUST be sent and have a **Format** attribute sent with the value:
   - urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName
2. The value of the <**NameID>** element itself MUST be an X.509 subject name representing the authenticated user. At a minimum, the **NameID** MUST include at least one Relative Distinguished Name with the name **uid**.
3. As defined in the SAML specification, the **Subject** element in each statement of a Web SSO assertion MUST have a **SubjectConfirmation** element. The **SubjectConfirmation** element MUST contain an embedded **Method** element with the value:
   - urn:oasis:names:tc:SAML:2.0:cm:bearer

5.3 Web SSO Response Assertion Requirements

All Web SSO scenarios require the exchange of a SAML Web SSO assertion between the IdP website and the SP website. The **Response** MUST be issued via an HTTP POST. The SAML assertions MUST contain a **Subject** element as defined above. This section defines what the assertions need to contain for this interop.

5.3.1 AuthnStatement

Each Web SSO assertion must contain an **AuthnStatement** element.

1. The element's **AuthnContext** attribute MUST have a value of:
   - **urn:oasis:names:tc:SAML:2.0:ac:classes:Password**
2. The **AuthnStatement** MUST NOT contain a **<SubjectLocality>** element.

5.3.2 AttributeStatement

Each Web SSO assertion must contain an **AttributeStatement** element.

1. The **AttributeStatement** must contain the following **Attribute** elements with the described **Name** attributes:
    a. **MemberLevel**: Used to control web application personalization at the SP website and for display purposes.
    b. **EmailAddress**: Used for display purposes only.
    c. **CommonName**: Used for display purposes only.
2. The **AttributeValue** for each **Name** is not specifically defined, but should be as follows:
    - something reasonable based on the generally accepted meaning of the **Name**.
    - MUST be of type ”xs:string”.
    - MUST be a single value.
3. The **Attribute** of each **Name** MUST have a **NameFormat** of:
    - “urn:oasis:names:tc:SAML:2.0:attrname-format:basic”

5.3.3 Conditions Element

Each assertion must contain a **Conditions** element. The element must contain **NotBefore** and **NotOnOrAfter** attributes that contain appropriate values.

Some effort will be made to synchronize the clocks on the machines involved in the interoperability demonstration. However, assertions that will work with some variation in clock settings must be created. For this interop, IdP websites should create assertions that have the **NotBefore** attribute set to 5 minutes before the current time, and the **NotOnOrAfter** attribute set to 10 minutes beyond the current time.

5.4 idP-Site-First Use Case Requirements

The unsolicited Response sent by the idP in an idP initiated SSO may optionally contain a **RelayState** parameter. If **RelayState** is included, it MUST be a valid URL of a resource on the Service Provider.

5.5 Single Logout Use Case Requirements

Both idP initiated and SP initiated single logout use cases are supported.

The idP used for single logout MUST log the user out from all SP sessions which use that idP.

The **LogoutRequest** MUST contain a **NameID** element that matches the **NameID** as specified in the *Subject Element* section.

The **LogoutRequest** MUST use a HTTP Redirect binding.

NOTE - Single Logout will not be available from the eAuthentication Portal.

5.6 eAuthentication Portal Use Case Requirements

The Entity ID of the idP will be used as the Credential Service (CS) value of the CSID query parameter sent by the Portal to the Agency Application (SP).

The resource name of an SP will be the Agency Application (AA) used by the Portal. An AAID will not be needed for this Interop.

NOTE - Single Logout will not be available from the eAuthentication Portal.

6 Optional Use Case Scenario

The optional use case scenario showcases federation and to a lesser extent, defederation. The scenario also uses portions of the base use cases, but it must be recognized that the Subject NameID's are not the same between the base use cases and the optional use cases. The optional use case scenario is as follow:

1. The user accesses an SP and selects an idP for authentication,
2. The SP sends an <AuthnRequest> to the selected idP via an HTTP Redirect,
3. The idP authenticates the user with one of the predefined optional use case user ID's, recognizes this is an optional use case user
4. The idP sends a <Response> back to the SP with a persistent NameID via HTTP POST,
5. The SP authenticates the user, saves the linked information and displays the requested web page,
6. The user clicks the Logout button,
7. The SP sends a <LogoutRequest> to the idP via HTTP Redirect,
8. The idP sends a <LogoutResponse> back to the SP via HTTP Redirect,
9. The user again accesses the SP,
10. The SP-Site-First base use case scenario then takes place, but entering the special optional use case user ID for authentication that was used in step 3,
11. The SP web page is displayed with the attributes returned from the <Response>,
12. The user clicks the defederation link or button on the SP web page,
13. The SP sends a <ManageNameIDRequest> to the idP requesting termination of the federation,
14. The idP returns a <ManageNameIDResponse> to the SP indicating the federation has been terminated.

7 Optional Use Case Requirements

The optional use case requirements are the same as the base use case requirements, with the exception of the Subject identifier. This is defined below and should be used for all requests and assertions in the optional use cases.

7.1 AuthnRequest Requirements

All AuthnRequests MUST be signed. This mean that in the metadata for an idP, **IDPSSODescriptior** MUST have a **WantAuthnRequestsSigned** attribute and it MUST

be set to true. Likewise in the metadata for an SP, **SPSSODescriptor** MUST have a **AuthnRequestsSigned** attribute and it MUST be set to true.

The **RequestAbstractType** must have the attributes set as follows:

- ID: NCName string; required by SAML – MUST be sent
- Version: string; required by SAML; value = "2.0" – MUST be sent
- IssueInstant: xs:dateTime; required by SAML – MUST be sent
- Destination: URI; required by SAML when using HTTP Redirect or POST bindings – MUST be sent
- Consent: values from spec – MUST NOT be sent

The **RequestAbstractType** must have the elements set as follows:

- <saml:Issuer>: IssuerType – required for all profiles we're using – MUST be sent
- Format must be omitted or must be set to "urn:…:nameid-format: entity".
- NameQualifier, SPNameQualifier, and SPProvidedID – MUST NOT be sent
- <ds:Signature> - MUST NOT be sent (Note that the redirect binding will sign the URL parameters as defined in the spec, so the message itself should not include a signature)

The <**AuthnRequest>** element must have the attributes set as follows:

- **ForceAuthn** – MUST NOT be sent
- **IsPassive** – MUST NOT be sent
- **AssertionConsumerServiceIndex** – is OPTIONAL, but if sent it MUST be set to 0 (zero)
- **AssertionConsumerServiceURL** – MUST NOT be sent
- **ProtocolBinding** – MUST NOT be sent
- **AttributeConsumingServiceIndex** – MUST NOT be sent
- **ProviderName** - OPTIONAL

The **AuthnRequest** must have the elements set as follows:

- <**saml:Conditions**> – MUST NOT be sent
- <**RequestedAuthnContext**> – MUST NOT be sent
- <**Scoping**> – MUST NOT be sent
- <**saml:Subject**> – MUST NOT be sent
- The <**NameIDPolicy>** element MUST be sent and have the following attributes:
  - a **Format** attribute MUST be sent with the value:
    - urn:oasis:names:tc:SAML:2.0:nameid-format:persistent
  - an **AllowCreate** attribute MUST be set to true
  - an **SPNameQualifier** attribute MUST NOT be sent
  - an **SPProvidedID** attribute MUST NOT be sent

7.2 Subject Element

All assertions used for the optional use cases in this interop will use SAML **Subject** elements formatted as Persistent Identifiers. Thus,

1. The <**NameID>** element MUST include a **Format** attribute with the value:
   - urn:oasis:names:tc:SAML:2.0:nameid-format:persistent

2. The value of the <**NameID>** element itself MUST be an opaque ID representing the authenticated user at the idP and SP.
3. As defined in the SAML specification, the **Subject** element in each statement of a Web SSO assertion MUST have a **SubjectConfirmation** element.
    a. The **SubjectConfirmation** element MUST contain an embedded **Method** element with the value:
        - urn:oasis:names:tc:SAML:2.0:cm:bearer

8 Configuration Data Requirements

In order to perform the Web SSO interop scenarios, each vendor will need to provide additional configuration data for the other participants to use. This includes information for accessing demo application web sites, URL's for accessing SAML services, information used in the trust relationship with each participant domain with which they have common Web SSO profile support, etc. This configuration data is defined in section 9.3.

8.1 SP Assertion Consumer Endpoints

Each SP will define a single <IndexedEndpointType> as an assertion consumer as defined in the SAML 2.0 Metadata specification. This single endpoint will have an index of 0.

8.2 URLs

- **eAuthentication Portal URL:** URL of the eAuthentication Portal provided. There will be only one of these for the interop.
- **idP Entity ID:** URL of the idP logon page. There should be zero or one of these per vendor.
- **idP SingleSignOnService:** URL to receive login requests
- **idP SingleLogoutService (request):** URL to receive logout requests
- **idP SingleLogoutService (response):** URL to receive logout responses
- **idP ManageNameIDService (request):** URL to receive defederation requests (optional use case only)
- **idP ManageNameIDService (response):** URL to receive defederation responses (optional use case only)
- **SP Resource:** URL to the demo web application. There may be zero or one of these per vendor.
- **eAuthentication SP Resource:** URL to be directed to from the eAuthentication Portal. This is only needed if it is different than the SP Resource above.
- **SP AssertionConsumerService:** URL to receive Web SSO assertions
- **SP SingleLogoutService (request):** URL to receive logout requests
- **SP SingleLogoutService (response):** URL to receive logout responses
- **SP ManageNameIDService (request):** URL to receive defederation requests (optional use case only)
- **SP ManageNameIDService (response):** URL to receive defederation responses (optional use case only)

8.3 Identifier Requirements

Each participant needs to provide the following values to the other participants:

- **idP Entity ID:** The entity ID for the idP for those vendors hosting idP functionality .
- **SP Entity ID:** The entity ID for the SP for those vendors hosting SP functionality.
- **SP Resource Names:** One or more URL's that are required for those vendors hosting SP functionality.

8.4 Base Use Case User Account Requirements

The Web SSO interop base use case scenarios rely on each IdP website having a common set of use accounts which all demo participants will be able to easily remember. The password for all uses will be "saml2005". The accounts associated with the users for the **base use case scenarios** are as follows:

1. Alice (uid=alice)
2. Bob (uid=bob)
3. Charlie (uid=Charlie)

When used in the Web SSO interop scenarios, these accounts must be mapped to and from SAML Subjects. As defined earlier, the SAML Subjects must have NameID element values in an X.509 subject name format. Interop participants may need to perform identity mapping of these user accounts to obtain the X.509 Subject Names to be used in the SAML Subjects. This mapping can be performed using any algorithm that the participant wishes to use. For this interop it is required that the X.509 Subject Names begin with the uid RDN. For example, user "bob" should have a NameID value beginning with "uid=bob". The DN may be as short as that but it may also contain more "<name>=<value>" RDN pairs separated by commas.

Each account must have attributes named EmailAddress, CommonName and MemberLevel. The value of those attributes should be a basic string type, but are not explicitly defined for this Interop.

8.5 Optional Use Case User Account Requirements

The Web SSO Interop optional use case scenario relies on each participating IdP website having a common set of use accounts which all demo participants will be able to easily remember. The accounts associated with the users for the **optional use case scenarios** will be based on the individual vendor names who support the optional scenario. The general format will be <vendor>user1 through <vendor>user5 (ie. rsauser1). The names will be all lowercase. The password for all users will be "saml2005". The list of participating vendors are as follows:

1. Entrust (ie. entrustuser1)
2. NTT (ie. nttuser1)
3. OpenNetwork (ie. ontuser1)
4. Oracle (idP only) (ie. oracleuser1)
5. RSA (ie. rsauser1)
6. Sun (ie. sunuser1)

7. Symlabs (ie. symuser1)
8. Trustgenix (ie. tguser1)

Each account must have attributes named EmailAddress, CommonName and MemberLevel. The value of those attributes should be a basic string type, but are not explicitly defined for this Interop. It is recommended these values be different for each user per vendor (ie. rsauser1 would be different than rsauser2).

8.6 Digital Signing and Encryption Requirements

Individual elements will not be encrypted for this Interop. All transactions will be encrypted using SSL 3.0. Since front channel bindings will be used for all profiles, each message MUST be digitally signed as specified in the SAML 2.0 Profiles. The mechanism used to correlate the signing certificate to the SAML party sending the message is left up to the participants.

The assertions in <Response> messages will be signed as defined in the SAML 2.0 Profiles, however the entire <Response> will not be signed.

Following [XMLSig], a <KeyInfo> element is not required in a signature. You must not require a KeyInfo element to be present to successfully process a signature.

9 General Guidelines and Requirements

9.1 Supported Web Browsers

The following Web browsers should be supported for use in all scenarios:

- Microsoft Internet Explorer 6.0
- Mozilla Firefox 1.0

9.2 PKI Considerations

SAML uses digital certificates to secure inter-component communication channels via SSL and to digitally sign SAML assertions and protocol messages. While many SSL and digital signing (DSIG) implementations are lenient in their enforcement of PKIX digital certificate recommendations, we would like to ensure that the certificates used in the interoperability demonstration conform to those recommendations as closely as possible. Using standard SSL and DSIG certificates will reduce the amount of time spent debugging certificate-related problems.

For more information regarding X.509 digital certificates and their extensions, please refer to the Internet X.509 Public Key Infrastructure Certificate and CRL Profile (IETF RFC 2459) located at:

- http://www.ietf.org/rfc/rfc2459.txt

9.2.1 Root Certificate Authority

In order to ease the PKI configuration for all participants, the interop demo will make use of a single Certificate Authority for generation of all SSL and DSIG certificates. DataPower will be providing a CA for this purpose. Individual participants will need to create PEM-based certificate request files that can be emailed to the CA administrator. The administrator will issue the appropriate certificates and provide them back to the requestors. The CA's trusted root certificate will be distributed to all participants and will need to be installed in all browsers and systems in order for the generated end-entity certificates to be used.

To facilitate the exchange of these certificates, the CA administrator will package all certificates into a ZIP file for distribution to the participants. The certs will be distributed as PEM files and/or PKCS #7 files.

9.2.2 Certificate Authority Certificates

CA Certificates will contain the following extensions:

- AuthorityKeyIdentifier
- SubjectKeyIdentifier
- BasicConstraints*
    - cA:true
- KeyUsage*
    - keyCertSign
    - cRLSign

* This extension will be marked critical.

They will also contain the following extension to ensure compatibility with Netscape:

- NetscapeCertType
    - sslCA
    - smimeCA
    - objectSigningCA

9.2.3 SSL Server Certificates

SSL server certificates will contain the following extensions:

- AuthorityKeyIdentifier
- SubjectKeyIdentifier
- KeyUsage*
    - keyEncipherment
- SubjectAltName
    - dNSName

* This extension will be marked critical.

They will also contain the following extension to ensure compatibility with Netscape:

- NetscapeCertType
    - sslServer

9.2.4 Digital Signing Certificates

Digital signing certificates will contain the following extensions:

- AuthorityKeyIdentifier
- SubjectKeyIdentifier
- KeyUsage*
    - digitalSignature

* This extension will be marked critical.

Appendix A.Configuration Data

A.1.*Network Configuration*

All systems used in the Interop demonstration will be on a private network. Internet connections will not be available to the demonstration systems during the actual show. During setup, we will try to provide an Internet connection for vendors to use in the booth, with the understanding this may have to be shared among vendors.

Each participant will be assigned their own network DNS domain and IP subnet. The network will rely on static IP addresses in the 192.168.x.y subnet, where x is unique for each interop participant and y refers to the participant's systems in that subnet. The assigned network subnets are shown in the table below. Since vendors may use a single system to support the entire demo or may distribute demo components among multiple systems, the number of address in use within each subnet may vary. For consistency, it is requested that all application websites be hosted on a system with the IP address 192.168.x.1.

Each interop participant will be provided with an etc/hosts file configured with the host names and IP addresses defined below. All participants must have their IP configurations set to use a subnet mask of 255.255.0.0 and a gateway of 192.168.1.254.

Participant	IP addresses	DNS names	Comments
eGov/Enspier	192.168.10.1	Eauthentication.gsa.gov	eAuthentication Portal
Computer Associates	192.168.20.2	Smdemo1.netegrity.com	idP (base only)
	192.168.20.1	Smdemo2.netegrity.com	SP (base only)
DataPower	192.168.30.1	Mysp.datapower.com	SP (base only)
Entrust	192.168.40.2	Idp.entrust.com	idP
	192.168.40.1	Sp.entrust.com	SP
NTT	192.168.50.2	d-idp.liberty-iop.org	idP
	192.168.50.1	d-sp.liberty-iop.org	SP
OpenNetwork	192.168.60.1	www.summittrust.com	idP
	192.168.60.1	www.summittrust.com	SP
Oracle	192.168.70.2	Idp.oracle.com	idP
RSA Security	192.168.80.2	Twain.rsa.com	idP
	192.168.80.1	Jackson.rsa.com	SP
Sun	192.168.90.2	Idp.sun.com	idP
	192.168.90.1	Sp.sun.com	SP
Symlabs	192.168.100.2	idp.symlabs.com	idP
	192.168.100.1	Sp.symlabs.com	SP
Trustgenix	192.168.110.2	Idp-saml2.trustgenix.com	idP
	192.168.110.1	Sp-saml2.trustgenix.com	SP

A.2. *SAML Configuration Data*

This section contains the URL's and Issuer Strings for each interop participant. (As of February 4, 2005)

- ***eGov/Enspier***

Portal URL	http://eauthentication.gsa.gov

- ***Computer Associates***

IdP Entity ID	http://smdemo1.netegrity.com/idp
idP Demo App URL (resource)	http://smdemo1.netegrity.com/index.html
idP SingleSignonService URL	http://smdemo1.netegrity.com/affwebservices/public/saml2sso
idP SingleLogoutService request URL	http://smdemo1.netegrity.com/affwebservices/public/saml2sso
Idp SingleLogoutService response URL	http://smdemo1.netegrity.com/affwebservices/public/saml2slo
idP ManageNameIDService request URL	N/A
idP ManageNameIDService response URL	N/A
SP Entity ID	http://smdemo2.netegrity.com/sp
SP Demo App URL (resource)	http://samldemo2.netegrity.com/saml2demo/NETElectronics/index.html
SP AssertionConsumerService URL	http://smdemo2.netegrity.com/affwebservices/public/saml2assertionconsumer
SP SingleLogoutService request URL	http://smdemo2.netegrity.com/affwebservices/public/saml2slo
SP SingleLogoutService response URL	http://smdemo2.netegrity.com/affwebservices/public/saml2slo
SP ManageNameIDService request URL	N/A
SP ManageNameIDService response URL	N/A

- ***DataPower***

SP Entity ID	http://mysp.datapower.com
SP Demo App URL (resource)	http://192.168.30.1:8080/login.html
SP AssertionConsumerService URL	http://mysp.datapower.com:4000/rsademo/ResponseHandler
SP SingleLogoutService request URL	http://mysp.datapower.com:4000/rsademo/MyLogout
SP SingleLogoutService response URL	http://mysp.datapower.com:4000/rsademo/LoggedOut
SP ManageNameIDService request URL	N/A
SP ManageNameIDService response URL	N/A

- ***Entrust***

IdP Entity ID	https://idp.entrust.com
idP Demo App URL (resource)	https://idp.entrust.com/GetAccess/Login
idP SingleSignOnService URL	https://idp.entrust.com/GetAccess/Saml/SSO/RedirectRequest
idP SingleLogoutService request URL	https://idp.entrust.com/GetAccess/Saml/SLO/RedirectRequest
Idp SingleLogoutService response URL	https://idp.entrust.com/GetAccess/Saml/SLO/RedirectResponse
idP ManageNameIDService request URL	https://idp.entrust.com/GetAccess/Saml/MNI/RedirectRequest
idP ManageNameIDService response URL	https://idp.entrust.com/GetAccess/Saml/MNI/RedirectResponse
SP Entity ID	https://sp.entrust.com
SP Demo App URL (resource)	https://sp.entrust.com/GetAccess/ResourceList
SP AssertionConsumerService URL	https://sp.entrust.com/GetAccess/Saml/SSO/PostResponse
SP SingleLogoutService request URL	https://sp.entrust.com/GetAccess/Saml/SLO/RedirectRequest
SP SingleLogoutService response URL	https://sp.entrust.com/GetAccess/Saml/SLO/RedirectResponse
SP ManageNameIDService request URL	https://sp.entrust.com/GetAccess/Saml/MNI/RedirectRequest
SP ManageNameIDService response URL	https://sp.entrust.com/GetAccess/Saml/MNI/RedirectResponse

- ***NTT***

IdP Entity ID	https://d-idp.liberty-iop.org:8443/idp/saml20
idP Demo App URL (resource)	https://d-idp.liberty-iop.org:8443/idp/en/login.jsp
idP SingleSignOnService URL	https://d-idp.liberty-iop.org:8443/idp/authn_redirect_saml20
idP SingleLogoutService request URL	https://d-idp.liberty-iop.org:8443/idp/logoutreq_post_saml20
Idp SingleLogoutService response URL	https://d-idp.liberty-iop.org:8443/idp/logoutres_post_saml20
idP ManageNameIDService request URL	https://d-idp.liberty-iop.org:8443/idp/managereq_redirect_saml20
idP ManageNameIDService response URL	https://d-idp.liberty-iop.org:8443/idp/manageres_redirect_saml20
SP Entity ID	https://d-sp.liberty-iop.org:8443/sp2/saml20
SP Demo App URL (resource)	https://d-sp.liberty-iop.org:8443/sp2/en/index.jsp
SP AssertionConsumerService URL	https://d-sp.liberty-iop.org:8443/sp2/asscon_redirect_saml20
SP SingleLogoutService request URL	https://d-sp.liberty-iop.org:8443/sp2/logout_post_saml20
SP SingleLogoutService response URL	https://d-sp.liberty-iop.org:8443/sp2/logoutres_post_saml20
SP ManageNameIDService request URL	https://d-sp.liberty-iop.org:8443/sp2/services/soap
SP ManageNameIDService response URL	https://d-sp.liberty-iop.org:8443/sp2/managenameidres_redirect_saml20

- ***OpenNetwork***

IdP Entity ID	urn:summittrust.com
idP Demo App URL (resource)	http://www.summittrust.com/uidp/signon
idP SingleSignOnService URL	http://www.summittrust.com/saml2authnservice/authnservice.aspx
idP SingleLogoutService request URL	https://www.summittrust.com/saml2sloservice/sloservice.aspx
Idp SingleLogoutService response URL	https://www.summittrust.com/saml2sloservice/sloservice.aspx
idP ManageNameIDService request URL	
idP ManageNameIDService response URL	
SP Entity ID	urn:summittrust.com
SP Demo App URL (resource)	http://www.summittrust.com/portal
SP AssertionConsumerService URL	http://www.summittrust.com/saml2acs/acs.aspx
SP SingleLogoutService request URL	https://www.summittrust.com/saml2sloservice/sloservice.aspx
SP SingleLogoutService response URL	https://www.summittrust.com/saml2sloservice/sloservice.aspx
SP ManageNameIDService request URL	
SP ManageNameIDService response URL	

- ***Oracle***

IdP Entity ID	https://idp.oracle.com:4443
idP Demo App URL (resource)	https://idp.oracle.com:8443/federation/sso
idP SingleSignOnService URL	https://idp.oracle.com:4443/federation/sso/authnv20
idP SingleLogoutService request URL	https://idp.oracle.com:4443/federation/sso/sloreqv2x
Idp SingleLogoutService response URL	https://idp.oracle.com:4443/federation/sso/sloreturn2x
idP ManageNameIDService request URL	https://idp.oracle.com:4443/federation/sso/mnireq
idP ManageNameIDService response URL	https://idp.oracle.com:4443/federation/sso/mnireturn

- ***RSA Security***

IdP Entity ID	https://rsa.com
idP Demo App URL (resource)	https://twain.rsa.com:7002/sso/SSO?SPEntityID=xxx&TARGET=yyy
idP SingleSignOnService URL	https://twain.rsa.com:7002/sso/SSO
idP SingleLogoutService request URL	https://twain.rsa.com:7002/slo/logout/AP
Idp SingleLogoutService response URL	https://twain.rsa.com:7002/slo/return/AP
idP ManageNameIDService request URL	
idP ManageNameIDService response URL	
SP Entity ID	https://rsa.com
SP Demo App URL (resource)	http://jackson.rsa.com/protectedpage.html
SP AssertionConsumerService URL	https://twain.rsa.com:7002/sso/ACS
SP SingleLogoutService request URL	https://twain.rsa.com:7002/slo/logout/RP
SP SingleLogoutService response URL	https://twain.rsa.com:7002/slo/return/RP
SP ManageNameIDService request URL	
SP ManageNameIDService response URL	

- ***Sun***

IdP Entity ID	http://idp.sun.com
idP Demo App URL (resource)	http://idp.sun.com/idp/index.jsp
idP SingleSignOnService URL	http://idp.sun.com/amserver/ssos
idP SingleLogoutService request URL	http://idp.sun.com/amserver/slos
Idp SingleLogoutService response URL	http://idp.sun.com/amserver/slos
idP ManageNameIDService request URL	http://idp.sun.com/amserver/mnids
idP ManageNameIDService response URL	http://idp.sun.com/amserver/mnids
SP Entity ID	http://sp.sun.com
SP Demo App URL (resource)	http://sp.sun.com/sp/index.jsp
SP AssertionConsumerService URL	http://sp.sun.com/amserver/acs
SP SingleLogoutService request URL	http://sp.sun.com/amserver/slos
SP SingleLogoutService response URL	http://sp.sun.com/amserver/slos
SP ManageNameIDService request URL	http://sp.sun.com/amserver/mnids
SP ManageNameIDService response URL	http://sp.sun.com/amserver/mnids

- ***Symlabs***

IdP Entity ID	https://idp.symlabs.com:8681/meta.xml
idP Demo App URL (resource)	https://idp.symlabs.com:8681/N
idP SingleSignOnService URL	https://idp.symlabs.com:8681/F
idP SingleLogoutService request URL	https://idp.symlabs.com:8681/X
Idp SingleLogoutService response URL	https://idp.symlabs.com:8681/XR
idP ManageNameIDService request URL	https://idp.symlabs.com:8681/J
idP ManageNameIDService response URL	https://idp.symlabs.com:8681/JR
SP Entity ID	https://sp.symlabs.com:8643/meta.xml
SP Demo App URL (resource)	https://sp.symlabs.com:8643/E
SP AssertionConsumerService URL	https://sp.symlabs.com:8643/P
SP SingleLogoutService request URL	https://sp.symlabs.com:8643/K
SP SingleLogoutService response URL	https://sp.symlabs.com:8643/KR
SP ManageNameIDService request URL	https://sp.symlabs.com:8643/I
SP ManageNameIDService response URL	https://sp.symlabs.com:8643/IR

- ***Trustgenix***

IdP Entity ID	https://idp-saml2.trustgenix.com:8443/tfs
idP Demo App URL (resource)	https://idp-saml2.trustgenix.com:8443/idp-demo
idP SingleSignOnService URL	https://idp-saml2.trustgenix.com:8443/tfs/IDPSSO_SAML20
idP SingleLogoutService request URL	https://idp-saml2.trustgenix.com:8443/tfs/IDPSLO_SAML20
Idp SingleLogoutService response URL	https://idp-saml2.trustgenix.com:8443/tfs/IDPSLO_SAML20
idP ManageNameIDService request URL	https://idp-saml2.trustgenix.com:8443/tfs/IDPMNI_SAML20
idP ManageNameIDService response URL	https://idp-saml2.trustgenix.com:8443/tfs/IDPMNI_SAML20
SP Entity ID	https://sp-saml2.trustgenix.com:8443/tfs
SP Demo App URL (resource)	https://sp-saml2.trustgenix.com:8443/sp-demo/index.jsp
SP AssertionConsumerService URL	https://sp-saml2.trustgenix.com:8443/tfs/SPSSO_SAML20
SP SingleLogoutService request URL	https://sp-saml2.trustgenix.com:8443/tfs/SPSLO_SAML20
SP SingleLogoutService response URL	https://sp-saml2.trustgenix.com:8443/tfs/SPSLO_SAML20
SP ManageNameIDService request URL	https://sp-saml2.trustgenix.com:8443/tfs/SPMNI_SAML20
SP ManageNameIDService response URL	https://sp-saml2.trustgenix.com:8443/tfs/SPMNI_SAML20

Appendix B.Interop Summary

The SAML 2.0 Interop at the RSA Conference was definitely successful. Eleven vendors eventually presented demonstrations of interoperability. By the opening of the Interop booth on Wednesday, February 16th, 390 out of 405 base use case tests were successfully completed and 108 out of 112 optional use case tests, more than 96% in each instance. Some of the incomplete tests were successfully performed before the end of the show, raising the success rates even further. The dry run, conducted on February 2nd through the 4th at the GSA certification lab in Washington DC, proved its value immediately when over 70% of the use cases were successfully tested by the end of the first day of setup at the Interop.

The Interop was very well received by the press and customers. A highpoint was a press conference just before the booth opened on Wednesday, when Georgia Marsh, Deputy Program Manager for the General Services Administration (GSA), said the government was extremely interested in this technology and they would create a marketplace for it by making it a factor in approving products for use by other agencies. The booth was well attended during its two day run.

A number of factors were responsible for the success of the Interop. Limiting all the use cases to front channel bindings with HTTP Redirect and Post eliminated the need for client certificates and simplified configuration. The acceptance of a modified data flow for the GSA eAuthentication scenario minimized changes vendors needed to perform to support it. Having a weekly call with all the vendors brought a number of issues to light and allowed them to be resolved quickly. The most important factor, however, was having the dry run prior to the actual Interop and the fact that all the vendors attended, not just a subset of them. The involvement of all of the vendors during the planning and dry run testing showed a commitment that everyone benefited from.