SSO SOLUTION DESIGN

Feb 2018

M

RCoA
Royal College of Anaesthetists

INTRODUCTION

The purpose of this document is to provide a high level description of the implementation of Single Sign On as part of the Royal College of Anaesthetists digital engagement platform.

Document Scope

This document does not seek to provide a detailed technical design instead it presents a solution design that details broad integration patterns and approaches and then specifically describes how these would apply to a number of RCoA applications.

Assumptions and Dependencies

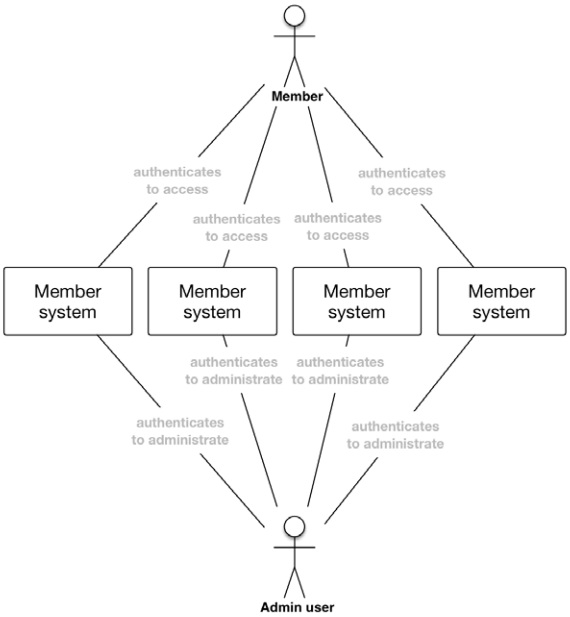
Assumptions

This document is primarily focussed on describing integrations with the applications that will form the core part of the Royal College of Anaesthetists digital platform. Some of these applications are currently in place, but some are still subject to procurement decisions. In addition, although a recommendation has been made about the choice of an SSO platform, at the time of writing this choice has not been finalised, so where possible we've described implementation approaches that would work with a range of SSO platforms.

HIGH-LEVEL SOLUTION DESIGN

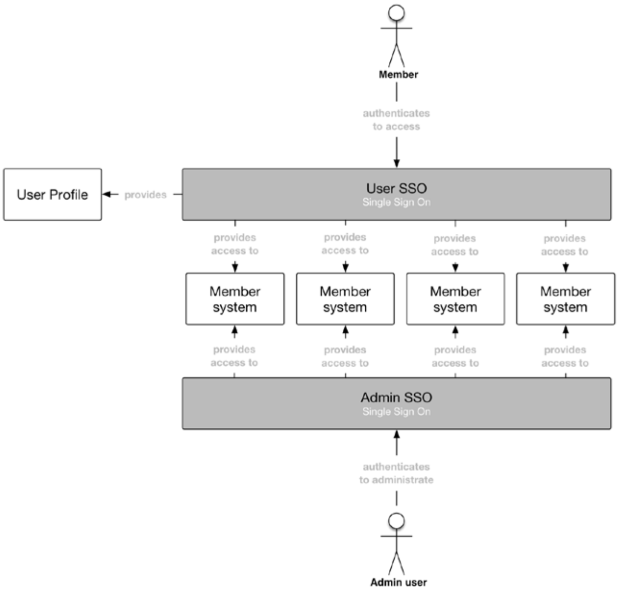
Current state

The current state architecture can be described simply as a number of member systems (examples include exam and events management platforms) each with their own user directory and authentication subsystem. This means that both Admin users and Members need to keep track of a separate set of user credentials for accessing each member system.



Future state

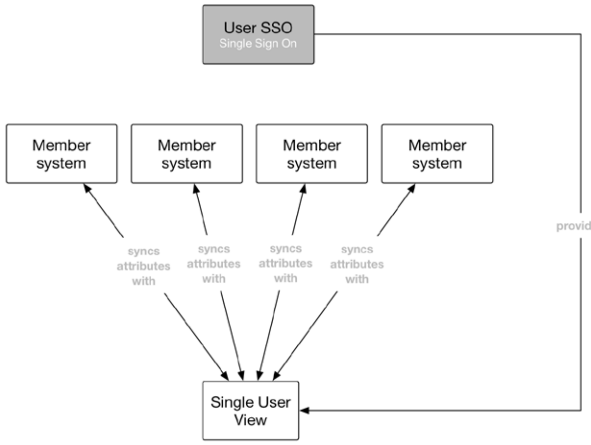
The architecture detailed below introduces SSO with two separate roles. The first role (described as Admin SSO) provides centralised authentication for users needing to administrate the various member systems. As is typical with most SSO these users now only need to remember one set of credentials for logging in to the member systems. The second role (described as User SSO) provides centralised authentication and authorisation for user needing to access the member systems as an end user. Finally the User SSO provides a User Profile or Single User View. This is simply a centralised user directory that synchronises appropriate user attributes from the associated member systems.



The diagram above describes the architecture that provides single sign on both for administrative users of the member systems and for the members themselves.

Single User View

A single user view in this context is simply a centralised user directory that becomes the source of truth for user data collected, stored and managed as part of the RCoA digital engagement platform. Pertinent user data from external member systems is synchronised to the central user directory so that it is available for querying and further synchronising with other applications such as CRM



The diagram above describes how the Single User View is provided by the directory service provided by the SSO product and how it interacts with the various member systems it is connected to, detailing two way attribute synchronising

Centralised User Directory

SSO platforms typically provide a central user directory that provides the storage and management for the user account that is then subsequently linked to all the users other accounts. Both Okta and Azure AD provide the following capabilities as part of its user directory offering:

- Storing rich profiles of user attributes
- Customising and extend user and app profiles with custom attributes
- Bi-directionally map and move attributes from user directory to applications

These capabilities enable the following:

- Synchronising user profile information across cloud applications, on-premise directory systems and applications.
- Provisioning of application user accounts with rich profile information such as roles, managers, geo-locations and other attributes that aid in configuring complex authentication and authorization rules.
- Collecting, importing and storing any type of user attribute, including externally defined custom attributes.

Essentially this describes a centralised view of the user with associated attributes that hold data that is synchronised from external applications. For example an exams management system might hold some information about the qualifications a member has. With a centralised user directory this information can be synchronised to the central user profile and used by other applications within the digital platform to determine, for example whether or not to display links to signing up for particular exams.

Centralised login and authentication

In addition to providing a centralised view of the user, SSO products provide a central point of login, account management and authentication.

As part of providing that capability, the SSO product supports interacting with the authentication subsystems of 3rd party applications (in our case member systems) in the following ways

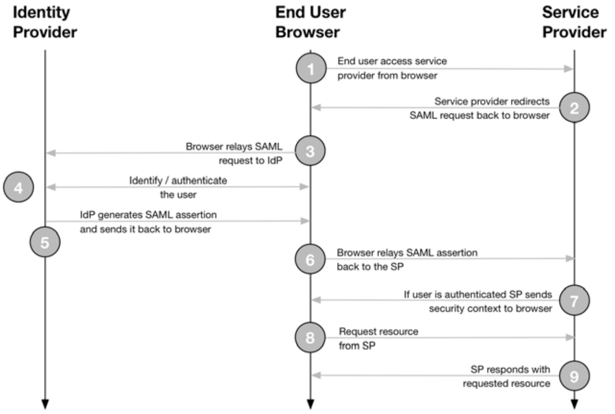
Authentication methods

SAML: The SAML specification defines three roles: the principal (typically a user), the identity provider (IdP), and the service provider (SP). In the use case addressed by SAML, the principal requests a service from the service provider. The service provider requests and obtains an authentication assertion from the identity provider. On the basis of this assertion, the service provider can make an access control decision – in other words it can decide whether to perform some service for the connected principal.

Before delivering the authentication assertion to the SP, the IdP may request some information from the principal – such as a user name and password – in order to authenticate the principal. SAML specifies the assertions between the three parties: in particular, the messages that assert identity that are passed from the IdP to the SP. In SAML, one identity provider may provide SAML assertions to many service providers. Similarly, one SP may rely on and trust assertions from many independent IdPs.

In this scenario, the role of the IdP is played by our SSO product.

SAML is mostly used as a web-based authentication mechanism in as much as it relies on using the browser agent to broker the authentication flow. At a high-level, the authentication flow of SAML looks like this:


Identity Provider
End User Browser
Service Provider
1
End user access service provider from browser
2
Service provider redirects SAML request back to browser
3
Browser relays SAML request to IdP
4
Identify / authenticate the user
5
IdP generates SAML assertion and sends it back to browser
6
Browser relays SAML assertion back to the SP
7
If user is authenticated SP sends security context to browser
8
Request resource from SP
9
SP responds with requested resource

OpenID Connect: OpenID Connect 1.0 is a simple identity layer on top of the OAuth 2.0 protocol. It allows Clients to verify the identity of the End-User based on the authentication performed by an Authorization Server, as well as to obtain basic profile information about the End-User in an interoperable and REST-like manner.

OpenID Connect allows clients of all types, including Web-based, mobile, and JavaScript clients, to request and receive information about authenticated sessions and end-users. The specification suite is extensible, allowing participants to use optional features such as encryption of identity data, discovery of OpenID Providers, and session management, when it makes sense for them.

SWA: For web applications that do not provide support for federated single sign-on Okta has developed an integration method called Secure Web Authentication (SWA). When SWA is enabled on an application, end users see an additional link below the application icon on their Okta home page, and through this link users can set and update their credential in the secure store for that application only. The credential is stored in an encrypted format using strong AES encryption combined with a customer specific private key. When a user subsequently clicks the application icon, Okta securely posts the username/password to the app login page over SSL and the user is automatically logged in. SWA can optionally be made even easier for end users; admins can require the username and password that is used for SWA-based apps to be the same as that user's Okta credentials, removing one more step for end users (they are no longer prompted for the initial password entry).

Password-based single sign-on: is Azure ADs version of SWA and enables secure application password storage and replay using a web browser extension or mobile app. This leverages the existing sign-in process provided by the application, but enables an administrator to manage the passwords and does not require the user to know the password. It is worth noting that this integration does require the installation of a browser extension unlike Okta's SWA

Login pages

This is one capability where the two SSO vary a little. Okta provides a pre-built login widget which can be fully styled, whereas Azure AD provides a more limited list of customisations that can be made to the sign in, and password reset pages. However both provide APIs that can be access from javascript which means that providing functionality that displays the current logged in user and links to log in / out etc can be implemented in way that allows this information to be displayed across any of the member services that allow the injection of javascript into the pages that they render. Details of how to use the Okta login widget can be found here:

- https://developer.okta.com/code/javascript/okta_sign-in_widget

And details of the customisation options for Azure AD login and password reset pages can be found here:

- https://docs.microsoft.com/en-us/azure/active-directory/customize-branding
- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-passwords-customize

Provisioning user accounts

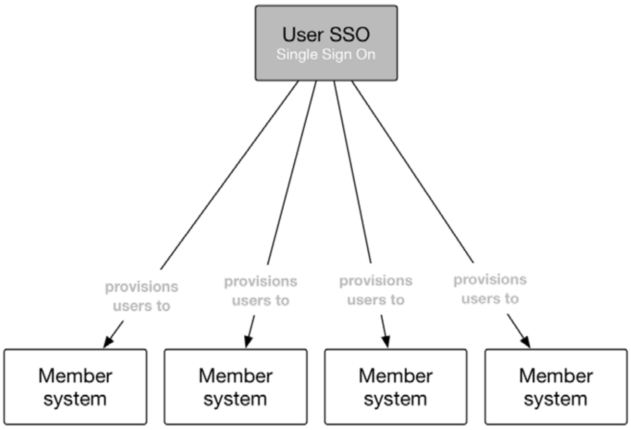
An important part of the capabilities provided by an SSO product as that of user provisioning. In this model, although a user might have a user account in 3rd party system, they don't take any repsonsibility for managing that account. The SSO product takes resposnibility for creating and removing user accounts and managing credentials. In order for the SSO product to be able to do this the 3rd party system needs to provide a way for the SSO product to assume this responsibility. This typically takes place over user management APIs provided by the 3rd party system.

The provisioning features provided by the SSO products include the provisioning of accounts for new users, deprovisioning accounts for deactivated users, and synchronizing user attributes across multiple directories.

These provisioning features provide the capability to manage user accounts automatically within applications. Provisioning and deprovisioning are bi-directional, so accounts can be created inside an application and imported into the SSO product or added to the SSO product and then pushed to corresponding applications.

Using provisioning allows for some powerful advantages such as

- Bulk user import (from several sources)
- The ability to natively create, read, and update users in the SSO product
- Password synchronization / password push (across multiple directories)



Out of the box provisioning methods

The SSO products both support provisioning and deprovisioning for any on-premises web app that have a web services API that is available to using a publicly addressable connection. The SSO product makes calls to that app's web service to create new user accounts, update attributes, and deactivate users as needed based on the user assignment rules configured in the SSO product.

Custom provisioning methods

For applications that don't expose native user management APIs that can be integrated with SSO frameworks, implementors can implement an integration application using a specification called The System for Cross-domain Identity Management (SCIM). SCIM based integrations work with an applications native user management system, sometimes this is through interacting with a user datastore directly and sometimes through interaction with the webforms that control the management of users in the underlying system. More information about SCIM can be found here:

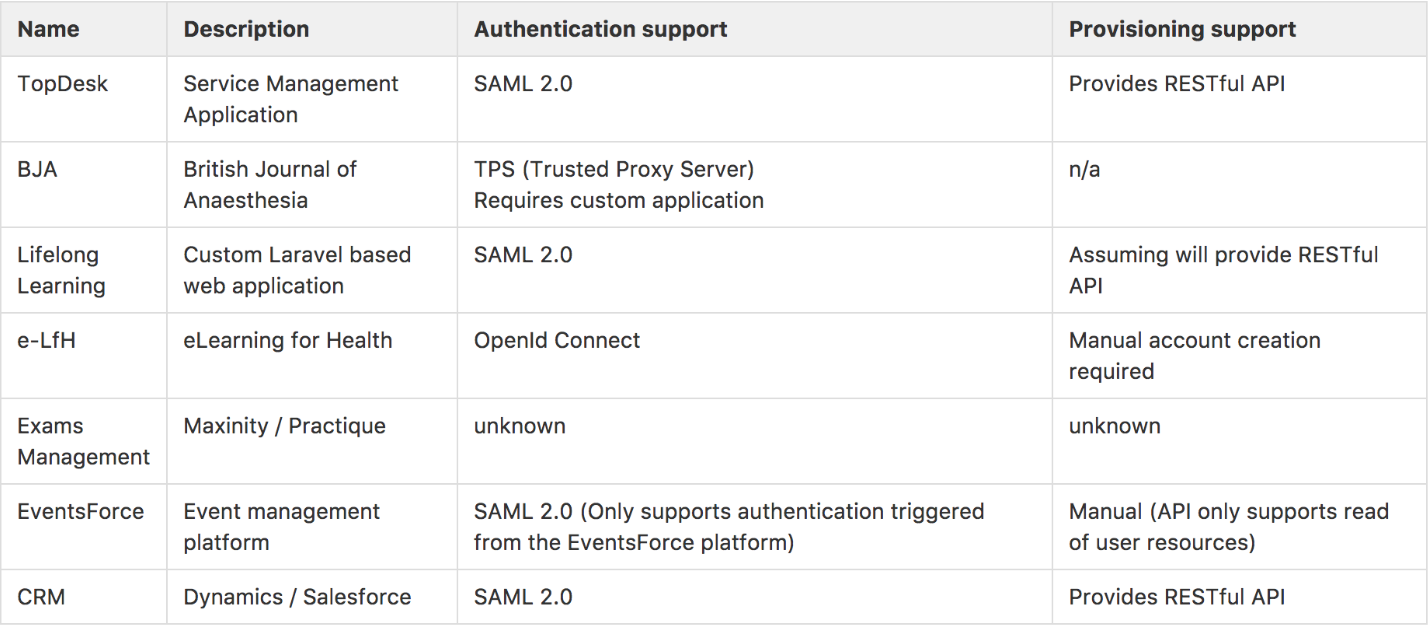
http://www.simplecloud.info/

Integration with RCoA Applications

Overview

This section of the design document describes how integrations could take place between the chosen SSO product and applications currently deployed or in the process of being deployed by the Royal College of Anaesthetists. For each application we've looked at whether it supports the standard forms of exchanging authentication information such as SAML and OpenId Connect or whether it requires a custom integration. In addition we've looked at the level of support each application provides for automated user provisioning.

The following table describes the range of applications assessed as part of this design document.



Name	Description	Authentication support	Provisioning support
TopDesk	Service Management Application	SAML 2.0	Provides RESTful API
BJA	British Journal of Anaesthesia	TPS (Trusted Proxy Server) Requires custom application	n/a
Lifelong Learning	Custom Laravel based web application	SAML 2.0	Assuming will provide RESTful API
e-LfH	eLearning for Health	OpenId Connect	Manual account creation required
Exams Management	Maxinity / Practique	unknown	unknown
EventsForce	Event management platform	SAML 2.0 (Only supports authentication triggered from the EventsForce platform)	Manual (API only supports read of user resources)
CRM	Dynamics / Salesforce	SAML 2.0	Provides RESTful API

The following sections provide further detail for how each application should be integrated.

Service Management

TopDesk is a SaaS service management system used by the Royal College of Anaesthetists.

Provisioning user accounts

TopDesk supports provisioning from both Okta and Microsoft Azure AD using its API, details of the pertinent API can be found here:

https://developers.topdesk.com/documentation/index.html#api-Person

Authenticating users

TopDesk supports SAML based authentication from both Okta and Microsoft Azure AD as described in the following documentation:

- http://saml-doc.okta.com/SAML_Docs/How-to-Configure-SAML-2.0-for-TOPdesk.html
- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-topdesk-secure-tutorial

Journal

The British Journal of Anaesthesia (BJA) is made available to members of the Royal College of Anaesthetists over an integration provided by Elsevier called a Trusted Proxy Server. The TPS provides a way of validating that requests for a particular journal have come from a trusted partner by applying encrypted parameters to a url requesting access to a journal.

Unfortunately this method of providing interaction with an external service isn't directly compatible with either of the SSO products in consideration.

In order to provide access to the BJA in a way that is integrated with SSO requires an additional application that:

- Sits in between the SSO platform and the TPS
- Uses SSO to authenticate RCoA members
- Generates valid TPS tokens
- Redirects users to the correct journal article

The application itself needs only to exist as a simple RESTful API the accepts a request for accessing a particular journal article and returns an authenticated url by which the user can access the journal article.

For example a request for a BJA article using the subdomain bjanaesthesia.rcoa.org would ensure that the user was authenticated with SSO, a token would be generated and then the user would be redirected to the article on the bjanaesthesia.org domain with the attached encrypted token providing access.

GET → https://bjanaesthesia.rcoa.org/article/S0007-0912(18)30008-4/fulltext → Authenticate using SSO → Generate Token → 301 Redirect to → http://bjanaesthesia.org/article/S0007-0912(18)30008-4/fulltext?TPSTOKEN=encryptedtoken

The application itself needs only to exist as a simple RESTful API the accepts a request for accessing a particular journal article and returns an authenticated url by which the user can access the journal article.

For example a request for a BJA article using the subdomain bjanaesthesia.rcoa.org would ensure that the user was authenticated with SSO, a token would be generated and then the user would be redirected to the article on the bjanaesthesia.org domain with the attached encrypted token providing access.

Lifelong Learning

Provisioning user accounts

Assuming that this custom application provides APIs for managing users / permissions the following functionality in Azure AD should be used for automating the provisioning of user accounts in the life long learning application.

- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-app-provisioning
- https://support.okta.com/help/Documentation/Knowledge_Article/Provisioning-and-Deprovisioning-572354290

Authenticating users

Authenticating users would take place using a standard SAML integration as described in the following articles

- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-custom-apps
- https://developer.okta.com/standards/SAML/setting_up_a_saml_application_in_okta

eLearning for Health

- eLearning for Healthcare (e-LfH) operate a Hub which allows users to access their learning content and various bespoke e-LfH applications and features.

Provisioning user accounts

e-LfH doesn't provide the ability to provision user accounts directly so the creation of account will need to be handled on the e-LfH portal directly. Once the account is created however it can be linked to the users SSO account.

Authenticating users

The Hub supports the use of OpenId Connect and so can be integrated with Okta and Azure AD using the following instructions

- https://support.okta.com/help/Documentation/Knowledge_Article/Using-the-App-Integration-Wizard-1111708899#OIDCWizard
- https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-protocols-openid-connect-code

Exams Management

The following applications were described as potential purchases in order to manage college Exams. Unfortunately insufficient information about the systems, the APIs they provide and their ability to integrate with SSO products was available at the time this document was generated.

- https://www.fry-it.com/practique
- https://www.maxinity.co.uk/

Events Management

EventsForce is an events management platform and is the current front runner to provide event management for RCoA.

Provisioning user accounts

EventsForce doesn't provide the ability to provision users from the SSO platform, user data however can be synchronised with the Single User View using the EventsForce API described here

- https://eventsforce.docs.apiary.io/

Authenticating users

EventsForce provides the ability to configure integration with SSO platforms using a SAML integration described here:

- https://eventsforce.zendesk.com/hc/en-gb/community/posts/217715846-Eventsforce-Single-Sign-On-SSO-using-SAML-2-0

This integration affects using EventsForce in a number of ways dependant on the role of the user.

Event Administrators - are able to access the platform having signed in to the SSO platform

Event Website Users - are able to register for an event without having to enter an email address, username and password in EventsForce. In addition these users are able to amend a registration without having to sign in to EventsForce. Finally users are able to visit pages that describe a private event without having to sign in to EventsForce

CRM

Provisioning user accounts

Both Okta and Azure AD provide provisioning for Salesforce

- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-salesforce-provisioning-tutorial

Azure AD provides the native authentication for MS Dynamics Online

Authenticating users

Both Okta and Azure AD provide out of the box SAML based integrations with Salesforce and MS Dynamics

Appendix A: References

Authentication

- http://saml-doc.okta.com/SAML_Docs/How-to-Configure-SAML-2.0-for-TOPdesk.html
- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-topdesk-secure-tutorial
- https://eventsforce.zendesk.com/hc/en-gb/community/posts/217715846-Eventsforce-Single-Sign-On-SSO-using-SAML-2-0
- https://support.okta.com/help/Documentation/Knowledge_Article/Using-the-App-Integration-Wizard-1111708899#OIDCWizard
- https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-protocols-openid-connect-code
- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-custom-apps
- https://developer.okta.com/standards/SAML/setting_up_a_saml_application_in_okta

Provisioning

- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-app-provisioning
- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-saas-salesforce-provisioning-tutorial
- https://support.okta.com/help/Documentation/Knowledge_Article/Provisioning-and-Deprovisioning-572354290
- https://eventsforce.docs.apiary.io/
- http://www.simplecloud.info/
- https://developers.topdesk.com/documentation/index.html#api-Person

Login customisation

- https://developer.okta.com/code/javascript/okta_sign-in_widget
- https://docs.microsoft.com/en-us/azure/active-directory/customize-branding
- https://docs.microsoft.com/en-us/azure/active-directory/active-directory-passwords-customize

Appendix B: Abbreviations, Acronyms and Definitions

- **OpenID Connect (OIDC) 3** - is an authentication protocol, based on the OAuth 2.0 family of specifications. It handles authentication via JSON Web Tokens (JWTs) delivered via the OAuth 2.0 protocol. OpenID Connect is a fairly recent protocol, with version 1.0 of the framework being adopted in 2014.
- **Security Assertion Markup Language (SAML)** - An authentication and authorization standard commonly found in the enterprise, SAML differs from Open ID in that it does not dynamically discover and accept authentication from new identity providers. The IdPs that a service wants to trust must be specified and hard-coded into each login event. Typically used to give the users of a corporate network access to a specific 3rd party service—for instance, so you don’t have to sign in again when you click a link to Salesforce on your company’s intranet.
- **The System for Cross-domain Identity Management (SCIM)** specification is designed to make managing user identities in cloud-based applications and services easier
- **Single Sign On (SSO)** - A subset of federated identity management, a means through which authentication and interoperability can be achieved in a federated system.
- **A Service Provider (SP)** is the entity providing the service – typically in the form of an application
- **An Identity Provider (IdP**) is the entity providing the identities, including the ability to authenticate a user. The Identity Provider typically also contains the user profile – additional information about the user such as first name, last name, job code, phone number, address, etc. Depending on the application, some service providers may require a very simple profile (username, email), while others may require a richer set of user data (job code, department, address, location, manager, etc).
- **A SAML Request**, also known as an authentication request, is generated by the Service Provider to “request” an authentication.
- **A SAML Response** is generated by the Identity Provider. It contains the actual assertion of the authenticated user. In addition, a SAML Response may contain additional information, such as user profile information and group/role information, depending on what the Service Provider can support.

- **A Service Provider Initiated (SP-initiated)** login describes the SAML login flow when initiated by the Service Provider. This is typically triggered when the end user tries to access a resource or login directly on the Service Provider side, such as when the browser tries to access a protected resource on the Service Provider side.
- **An Identity Provider Initiated (IDP-initiated**) login describes the SAML login flow initiated by the Identity Provider. Instead of the SAML flow being triggered by a redirection from the Service Provider, in this flow the Identity Provider initiates a SAML Response that is redirected to the Service Provider to assert the user's identity.
- **A Trusted Proxy Server (TPS)** provides Society publishing partners with seamless, full access to otherwise protected journal content on the Elsevier site.