02 May 2023
Version 1.0

Setting up Azure AD SAML SSO in CitizenHub

Luke Wallis

NETCALL

Contents

This guide is based on the usage of the **SAML** *identity standard*

Integrating a Liberty Create application with an Azure Active Directory identity service is a process that is performed within the *Build Studio* of the application. A new *SSO Provider* needs to be created that will take login requests and act accordingly in Create. To do this navigate to the *Security* section in *Build Studio* and click on **Add SSO Identity Provider** in the top right.

A certain amount of setup will be required within your Azure AD configuration application, this will involve setting up the Liberty Create application and defining the attributes and claims that will be sent as part of a login request. Please see your Azure AD documentation for instructions on how to achieve this.

1 SSO Provider Basics

The *Basics* tab on the *SSO Provider* contains various options and settings, this guide will not contain a definitive definition of each – this can be viewed here if needed. For an Azure AD integration that is handling your internal users (i.e. not customers), the following settings are recommended:

Property	Value	Notes
Name	Azure AD SSO	
Display Name	Azure AD SSO	
Restricted by environment	False	
Enabled	True	
User object	User	
Roles that can use this provider	Admin Agent Build Administrator	
Auto-create unknown users	True	Setting this as 'true' will result in users being created when they don't yet exist in the system
Auto-created users' role	Agent	This setting assumes that the majority of users being created will need to be agents
Allow 2FA bypass	False	

2 Provider Details

The *Provider Details* tab allows you to define the structure of the requests that will be received when a user attempts to login via the SSO. The **Provider Realm** should be set to *External* and the **Identity standard** should be set as *SAML 2.0*.

The **IDP Metadata** will need to be downloaded from your Azure AD portal; this can normally be downloaded by clicking ‘download’ next to the *Federation Metadata XML* entry in the *SAML Signing Certificate* section under your app configuration. Please consult your Azure AD documentation if this is not the case.

If the **Dynamic to Environment** property has been set to *true* then choose the environment that will use the SSO provider. The downloaded *IDP Metadata* file can be uploaded by clicking on the *Choose a file* button labelled ‘Metadata’. The SSO provider can now be saved.

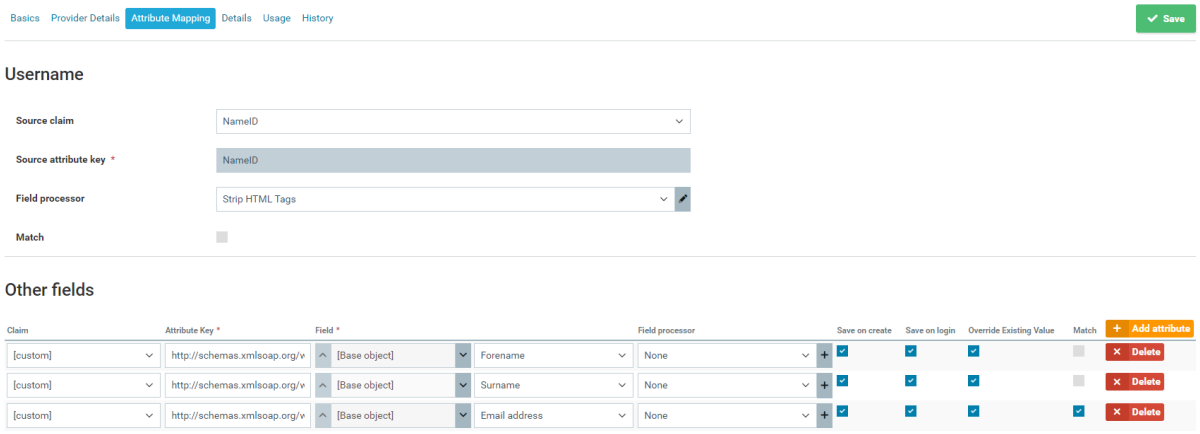
3 Attribute Mapping

This section provides guidance on how the attributes sent across as part of the login request will be handled. The claims that are included are defined in the Azure portal and therefore can vary depending on your requirements, the following instructions are based on the default configuration of an Azure app.

On the *Attribute Mapping* page, the Source claim selected should be *NameID* and the *Field processor* applied should be ‘Strip HTML Tags’. If you would like to use the *NameID* (generally a unique string that is used to identify one of your users for THIS application) to match a user by username then tick the ‘match’ option.

Additional mapping can be defined in the **Other fields** section, which will allow you to save information passed in the claims to properties of the User record (and its related records if needed). The value selected in the *Claim* field should match the value displayed under *Claim Name* in your Azure portal. If the claim name does not exist in the drop-down box then choose *[custom]* and type the *Claim Name* in the *Attribute Key* input then select where you would like to save the value.

The following configuration takes a simple set of attributes (NameID, forename, surname and email address) and uses the email address to match to an existing user:



Note that the *Attribute Key* values are actually URLs – in this case, the *Claim Names* that are sent across by Azure are these URLs. In order to inspect a login request as it comes in, enable detective logging in your app and then attempt a login via your SSO.
You should see an entry in the detective with the text **SSO Login attributes supplied:** - click on this entry to show the full request from your SSO. This will show you each of the claim values as well as the keys and can be used to configure the *Attribute Mappings* of your SSO Provider.