Example Social to SAML Mappings

Overview

This page gives some examples of how attributes which are asserted by social identity providers (via both OAuth and OpenID) could be mapped to MACE-Dir/SAML attributes.

Examples

> ⚠️ **Caution**
>
> In most cases, it still needs to be verified whether the value for `eduPersonTargetedID` is unique for a given person and service.

Facebook Mappings

Facebook supports a proprietary SSO protocol built on top of OAuth.

eduPerson Attribute	Facebook Attribute	Example Value	Notes
`givenName`	`facebook.first_name`	**Lucas**	
`sn`	`facebook.last_name`	**Rockwell**	
`displayName`	`facebook.name`	**Lucas Rockwell**	See `cn` below, as well.
`cn`	`facebook_cn`	**Lucas Rockwell**	Duplicate of `displayName`.
`mail`	`facebook.email`	**lr@lucasrockwell.com**	
`uid`	`facebook.username`	**lucasrockwell**	Can be blank, and a user can change this once for the lifetime of their account.
`eduPersonPrincipalName`	`facebook_user`	**lucasrockwell@facebook.com**	
`eduPersonTargetedID`	`facebook_targetedID`	*http://facebook.com!12...71*	

Google Mappings

Google supports two standard SSO protocols: OpenID 2.0 and OpenID Connect. The latter is an emerging IETF standard profile of OAuth2.

OpenID 2.0

eduPerson Attribute	Google Attribute	Example Value	Notes
`givenName`	`http://axschema.org/namePerson/first`	**Lucas**	
`sn`	`http://axschema.org/namePerson/last`	**Rockwell**	
`displayName`			Google does not provide `displayName`
`cn`			Google does not provide `cn`
`mail`	`openid.sreg.email`	**lucasrockwell@gmail.com**	
`uid`			Google does not provide `uid`
`eduPersonPrincipalName`	`http://axschema.org/contact/email`	**lucasrockwell@gmail.com**	Using http://axschema.org/contact/email for `ePPN` works for Google, but perhaps not other OpenID providers.
`eduPersonTargetedID`	Private Personal Identifier (PPID)	**	An opaque, per-SP identifier, just like ePTID

OpenID Connect

eduPerson Attribute	Google Attribute	Example Value	Notes
givenName			
sn			
displayName			
cn			
mail			
uid			
eduPersonPrincipalName			
eduPersonTargetedID			

LinkedIn Mappings

LinkedIn supports a proprietary SSO protocol built on top of OAuth.

eduPerson Attribute	LinkedIn Attribute	Example Value	Notes
givenName	linkedin.firstName	**Lucas**	
sn	linkedin.lastName	**Rockwell**	
displayName			LinkedIn does not provide displayName
cn			LinkedIn does not provide cn
mail			LinkedIn does not provide mail
uid	linkedin.id	**Y...r**	
eduPersonPrincipalName	linkedin_user	**Y...r@linkedin.com**	Local part is the same value as linkedin.id
eduPersonTargetedID	linkedin_targetedID	*http://linkedin.com!Y...r*	Unique value is the same value as linkedin.id

Twitter Mappings

Twitter supports a proprietary SSO protocol built on top of OAuth.

eduPerson Attribute	Twitter Attribute	Example Value	Notes
givenName			Twitter does not provide givenName
sn			Twitter does not provide sn
displayName	twitter.name	**Lucas Rockwell**	
cn	twitter.name	**Lucas Rockwell**	
mail			Twitter does not provide mail
uid	twitter.screen_name	**lucasrockwell**	
eduPersonPrincipalName	twitter_screen_n_realm	**lucasrockwell@twitter.com**	
eduPersonTargetedID	twitter_targetedID	*http://twitter.com!1...5*	

Windows Live Mappings

Windows Live supports a proprietary SSO protocol built on top of OAuth.

eduPerson Attribute	Windows Live Attribute	Example Value	Notes
givenName	windowslive.FirstName	**Lucas**	
sn	windowslive.LastName	**Rockwell**	
displayName			Windows Live does not provide displayName
cn			Windows Live does not provide cn
mail	windowslive_mail	**lr@lucasrockwell.com**	This is not necessarily an address @hotmail.com.
uid	windowslive_uid	**fd...89**	
eduPersonPrincipalName	windowslive_user	**fd...89@windowslive.com**	Local part is the same value as windowslive_uid
eduPersonTargetedID	windowslive_targetedID	*http://windowslive.com!fd...89*	Unique value is the same value as windowslive_uid

Attribute Matrix

The matrix below lists various attributes and which providers supply those attributes. **Note: This table is not complete.**

Provider	First Name	Last Name	Transient Email*	Persistent Email	Human-readable Unique ID	Machine-readable Unique ID	SP-specific ID
Facebook	✅	✅	✅		✅	(Have not verified this yet.)	(Have not verified this yet.)
Google OpenID Connect	✅	✅		❓	✅ (Email...)	✅ (Appears user can only look it up if Google+ is enabled for the account.)	
Google OpenID 2.0	✅	✅		✅	✅ (Email...)		✅ (The OpenID can either be set for the SP realm, or the domain realm, so only SP-specific if you ask Google to do that for you.)
LinkedIn	✅	✅	✅		✅ (Only if enabled via the Public Profile Settings page, however, a user can change this at will.)		✅
Twitter	✅	✅			✅	(Have not verified this yet.)	(Have not verified this yet.)
Windows Live	✅	✅	✅		⚠️ (Email, but there is more than one, so perhaps not...)	(Have not verified this yet.)	(Have not verified this yet.)

ⓘ Notes

* Unless the email address ends in the domain of the provider, then the likelihood that the the user can change at their whim is pretty high. This is great if you are using email as email, i.e., you want to actually know the user's email address. On the other hand, this can have very significant impacts on your service if you are trying to use email as the basis for eduPersonPrincipalName.