Integrating YuJa Active Learning with Google Apps (SAML)
1. Overview
This document is intended to guide users on how to integrate the YuJa Active Learning Platform as a Service Provider (SP) using Google as the Identity Provider (IDP). Once configured properly, users attempting to access YuJa services will first be redirected to Google, prompting for credentials to login. Once authenticated, the browser will again redirect back to YuJa, logged in as a new or existing user.

2. Setup
Setup involves creating a new SAML app in the Google admin console, then configuring things on the YuJa side by integrating with Google as an IDP, testing and activating the SAML SSO for users of the institution.

**NOTE:** For some steps, <institution> is to be replaced by the wildcard DNS of the institution associated with YuJa. As an example, for "https://hudson.yuja.com", <institution> would be replaced by "hudson".

2.1 Create a New SAML app for YuJa in the Google Admin Console
The steps to create a new SAML app are found at: https://support.google.com/a/answer/6087519?hl=en

Follow the instructions under the following sections: **Set up your own SAML app, Turn on SSO to your new SAML app**, and **Verify SSO between Google Apps and your new SAML app**, referring back to here for specific instructions on certain steps within this document:

Referencing: **Set up your own SAML app**

**Step 5:** Download the IDP metadata. This will be used to configure things on the YuJa side.

**Step 8:** Enter the following information:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity ID</td>
<td>https://&lt;institution&gt;.yuja.com</td>
</tr>
<tr>
<td>Start URL</td>
<td><strong>leave this blank</strong></td>
</tr>
<tr>
<td>Name ID</td>
<td>Basic Information, Primary Email</td>
</tr>
<tr>
<td>Name ID Format</td>
<td>Email</td>
</tr>
</tbody>
</table>

**Step 9:** Check **Signed Response**. This increases security when the SP and IDP are communicating.
**Step 11:** Create four mappings:

<table>
<thead>
<tr>
<th>Application Attribute</th>
<th>Category</th>
<th>User Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>givenname</td>
<td>Basic Information</td>
<td>First Name</td>
</tr>
<tr>
<td>familyname</td>
<td>Basic Information</td>
<td>Last Name</td>
</tr>
<tr>
<td>email</td>
<td>Basic Information</td>
<td>Primary Email</td>
</tr>
<tr>
<td>role</td>
<td>Employee Details</td>
<td>Job Title (or other appropriate field)</td>
</tr>
</tbody>
</table>

This field is used to determine if users are provisioned as students (the default) or are given enhanced privileges (Instructor/IT Manager). The suggested values for this field are IT Manager and Instructor (for users you wish to have IT Manager/Instructor privileges respectively), but you can use existing/custom values (see 2.2 #5 below for a discussion of IT Manager and Instructor mapping).

---

### 2.2 YuJa Platform Side SAML Configuration

1. Go to [https://<institution>.yuja.com](https://<institution>.yuja.com) and login as an IT Manager for your institution.

2. Navigate to the **Institution Management** tab in the **Main Menu**.

3. In the left sidebar, go to **Integrations**.
4. In the dropdown under **Select an API to configure**, select **SSO – Google Apps (SAML)**.

5. Enter the following information:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required?</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google SSO URL</td>
<td>Yes</td>
<td>- The URL used for SSO. This is where YuJa will send AuthnRequest tokens.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Found in Google IDP Metadata under:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;IDPSSODescriptor&gt; → &lt;SingleSignOnService&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>as the &quot;Location&quot; attribute. Note that for YuJa, an HTTP-Redirect binding is used.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- For example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td><a href="https://accounts.google.com/o/saml2/idp?idpid=B05pakw7">https://accounts.google.com/o/saml2/idp?idpid=B05pakw7</a></td>
</tr>
<tr>
<td>Name ID Format</td>
<td>Yes</td>
<td>- The format to be used by the SP and IDP when communicating about a subject.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Found in Google IDP Metadata under:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;IDPSSODescriptor&gt; → &lt;NameIDFormat&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>as the value of that tag. Note that, if available, emailAddress should be prioritized and used.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- For example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</td>
</tr>
<tr>
<td>Feature</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>----------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Remote Logout URL</td>
<td>No</td>
<td>Currently not supported - Leave this value blank.</td>
</tr>
</tbody>
</table>
| Google Signing Certificate Fingerprint       | No       | - The unique fingerprint of the IDP's certificate used when signing SAML responses.  
- See [How to derive the fingerprint of a certificate](#) in the Additional Tools section of this document for more details.  
- For example: 7j2mka9cfe2d09j23eefe01442f6a49d1222391f |
| Given Name Attribute                         | No       | Enter the following value: `givenname`  
- This is the exact value used in the Application Attribute field when creating attribute mappings.                                                                 |
| Family Name Attribute                        | No       | Enter the following value: `familyname`  
- This is the exact value used in the Application Attribute field when creating attribute mappings.                                                                 |
| Email Attribute                              | No       | Enter the following value: `email`  
- This is the exact value used in the Application Attribute field when creating attribute mappings.                                                                 |
| Role Attribute                               | No       | Enter the following value: `role`  
- This is the exact value used in the Application Attribute field when creating attribute mappings.                                                                 |
| IT Manager                                   | No       | - A comma separated list of values can be used  
- If the value received in the Role Attribute matches any of these values, the user will be provisioned as an IT manager.  
- For example: **IT Manager** |
| Instructor                                   | No       | - A comma separated list of values can be used  
- If the value received in the Role Attribute matches any of these values, the user will be provisioned as an instructor.  
- For example: **Instructor** or **Instructor,Teacher,TA** |
| Automatically sync data on user login         | No       | - If checked, whenever a user logs in via Google SAML Apps their basic information will be updated based on the data received in the SAML response. |

*Powering the Connected Classroom*
6. Click **Create**. In the confirmation dialog, click **OK**.

7. If required, you can update the configuration settings if you made a mistake. Simply click **Save** to keep the changes.

8. To test if the configuration was done correctly on both sides, click **Test SAML Login**. This should open a new tab and navigate to Google, prompting for credentials.

9. Login using a valid Google account.
10. After successfully logging in, you should be redirected back to YuJa, logged in as a new user.

**NOTE:** logging in as a new user will log the original account out. Log out of the newly created account and log back in as an IT Manager. Then navigate back to the **Institution Management → Integrations → SSO – Google Apps (SAML)**.

11. Once you have verified that the Google SAML Apps SSO works, you can choose to activate the new authentication scheme for your institution. To do so, click **Activate**, then click **OK**.

**IMPORTANT:** Only activate the new authentication scheme after successfully performing a test login and are ready to make it available for all users in your institution.

### 2.3 Dual Integration with LTI

**Overview**

If your institution has enabled both LMS Integration via LTI and also SSO access, then you have the choice to link the two integrations. We generally recommend this because it mean that irrespective of whether your users login via their LMS or their SSO, they will be presented with the same YuJa account information. In contrast, if Dual Integration with LTI is not setup, a user who uses both their LMS and SSO with YuJa will be provisioned with two separate accounts which in many cases isn’t ideal.

**How It Works**

If your LTI provider within your LMS can be configured to provide YuJa with a unique identifier for the user in the SAML system, it is possible to link the two accounts.

1. Configure your LMS to pass a custom LTI parameter to the YuJa tool called `lis_person_sourcedid` which contains the cross-matching SSO value. This can be an email, employee ID, or other field. You may need to consult your LMS platform’s product documentation on how to set custom LTI parameters. YuJa will make use of this feature to link the two login methods to the same account.

2. Obtain the specific attribute name used in the SAML Response token whose value corresponds to the unique identifier used by the LTI provider (in Step 1 above).
   a. For example, if the unique identifier is the user’s email address, then the linkage attribute would be “email”.
   b. The possible values you can use are specifically those set in the Google Admin Console when configuring the SAML App. They are the names used in the Attribute Mapping step.

3. Enter this value into the **Linkage Attribute** field. Note: This textbox will only appear if your institution has enabled LTI access.
4. Click Save.

5. Now, when logging in for the first time via ADFS (SAML), the YuJa system will search for a link with an LTI account using the value of the linkage attribute. If found, the SAML account will be linked to the existing account. Otherwise, a new account will be provisioned as normal.

All logins past the first one will continue to link to the YuJa account created or found on the first login.

3. Usage
Once both sides have been configured and the SAML SSO has been activated, it is easy to test and see if everything was done properly.

1. Go to the institution’s YuJa domain (i.e. https://<institution>.yuja.com) and press Login. This should redirect the user to the SSO server’s login page.
2. Enter valid credentials and sign in.
3. Once authenticated, the user should be redirected back to YuJa and the login was a success.

4. Additional Tools

4.1 How to Derive the Fingerprint of a Certificate
The fingerprint of the IDP’s certificate is used for additional security purposes when the SP is verifying a SAML response from the IDP. To derive the certificate’s fingerprint, follow the instructions below:

1. In the Google IDP metadata, extract the X509 certificate. This should be located under:
   - <IDPSSODescriptor> → <KeyDescriptor use="signing"> → <KeyInfo> → <X509Data> → <X509Certificate>
2. Once you have the certificate, go to the following website: https://www.samltool.com/fingerprint.php
3. Paste the certificate in the “X509 cert” textbox.
4. Make sure “sha1” is selected as the Algorithm.
5. Click “Calculate Fingerprint”.
6. Copy the “FingerPrint” value generated. This is the value used in the database. Note: The fingerprint should be an array of 20 bytes for “sha1”.
4.2 Useful Chrome Plugin for Debugging SAML Tokens

If you are using Chrome as your web browser, you may want to install a useful SAML plugin at:

https://chrome.google.com/webstore/detail/saml-chrome-panel/paijfdbeoenhembfhkhllainmocckace?hl=en

Once installed, simply open the developer tools in the browser (F12) and click on the SAML tab. Now, when doing an SP-initiated login, the SAML tokens sent by the browser will be shown in detail. This tool can be very useful in debugging SAML requests and responses.