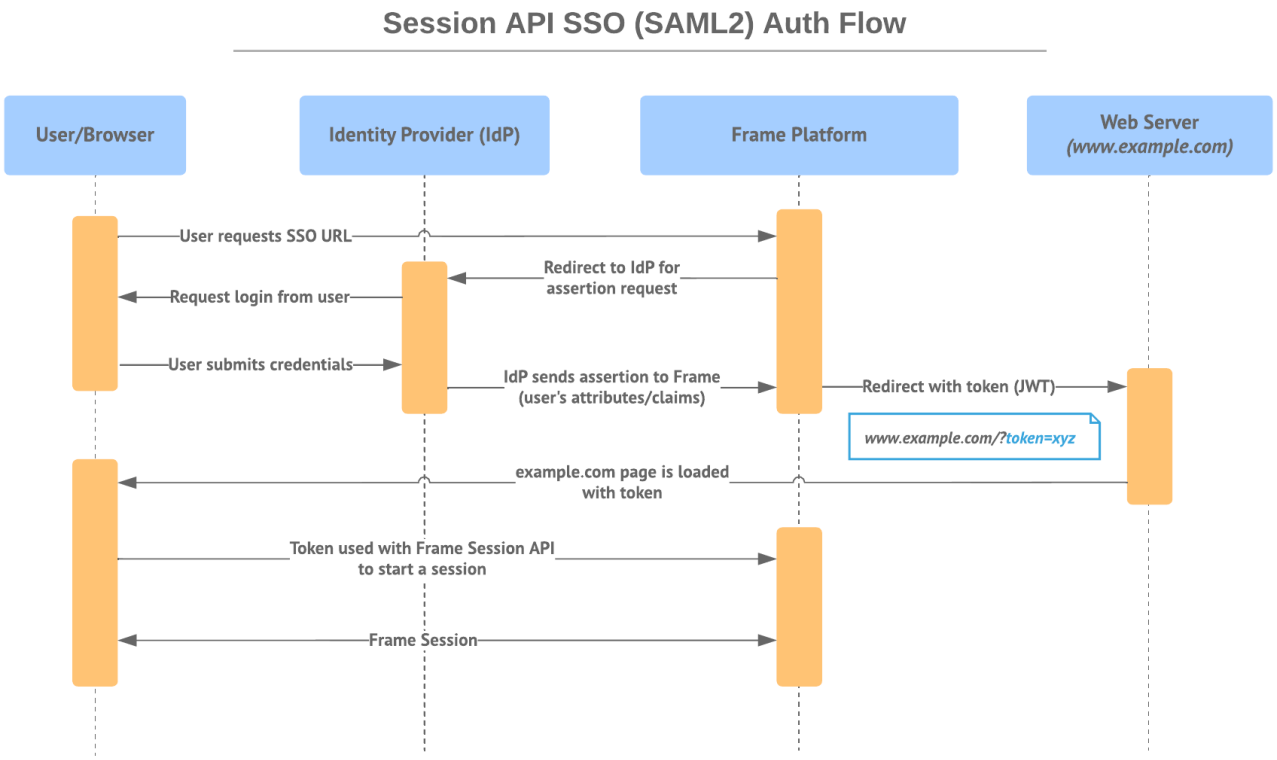
Acquire Tokens through SSO (SAML2)

Through SAML2 integrations with Frame, you can leverage existing SSO workflows to retrieve tokens for your users. Here's a general overview of what that process looks like:



To get started, there are three key items needed when using SSO workflows to authenticate your users:

1. You need to build a SSO URL for your SAML2 integrations. *This URL is used to kick-off the authentication process to retrieve a token for your users.*
2. To use the token with our Session API, your web application should expect *and capture* the **token search query parameter** to be present in the URL.
3. A **SAML2 relying party** is required for each domain name (e.g. example.com) you wish provide authenticated redirects to (e.g. `acme.com` or `localhost:3000`).
   :::info
   Adding relying parties for your SAML2 integrations is currently a manual process. Please create a support ticket.

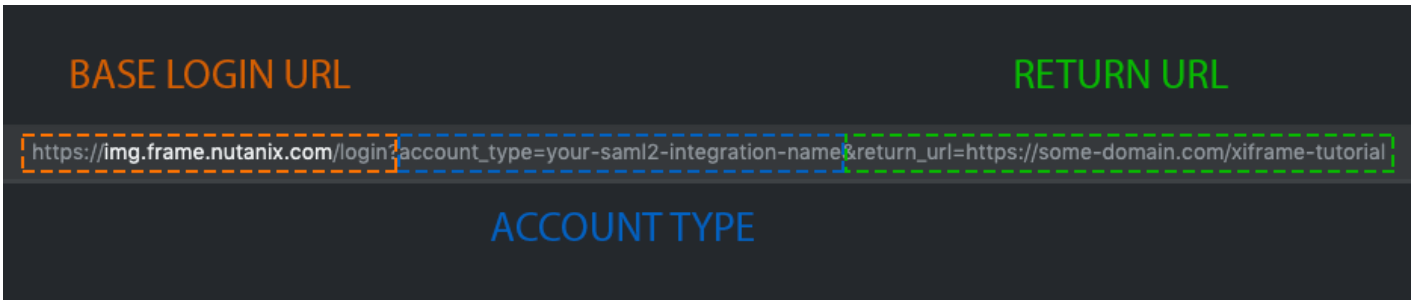
:::

Building a SSO URL

Using an existing SAML2 integration with Frame, you can construct an SSO link to authenticate your users and return them to the URI that is hosting your Frame Session API.

```
https://img.frame.nutanix.com/login?account_type=saml2-tutorial-example&return_url=https://example.com/frame-tutorial
```

To illustrate, here's a breakdown of the SSO URL:



These three URL components tell us where to send users to login, and where to redirect their browser to (with a token) after they've successfully signed in.

URL Search Query Param	Description
Base Login URL	This URL points to the Frame environment you'd like to authenticate with. For most Frame customers, this will be `https://img.frame.nutanix.com/login?`
`account_type`	This represents your SAML2 integration name. This tells our system where to redirect the user to login. This value is case-sensitive and must match the **exact name** of your SAML2 provider.
`return_url`	The URL you'd like to redirect your users back to after authenticating. When redirecting after a successful login, this URL is accompanied by a JWT in the URL as a search query parameter.

Capturing the token from the Return URL

When successfully authenticating with a SSO workflow, the browser will be redirected to the *return_url* with a token *search query parameter*.

For example, if a user successfully logs in using this SSO URL:

`https://img.frame.nutanix.com/login?account_type=acme-saml2&return_url=https://example.com/frame-example`

Frame will redirect the user's browser back to the `return_url` with a token (JWT) appended as a **search query parameter**:

`https://example.com/frame-example?token=xyz`

Using Javascript, you can easily capture the value for use with the Session API like this:

```
 // assign URL Search params to `params`
 const params = new URLSearchParams(document.location.search);

 // At this point, you can use the token with the Frame Session API
 const token = params.get('token');

 // Session API Parameters
 const terminalOptions = {
   serviceUrl: "https://cpanel-backend-prod.frame.nutanix.com/api/graphql",
   terminalConfigId: "38cb4f1c-a019-4163-9f1d-168b59fb5062.0f3a542b-884e-4edc-aa5f-65380597061",
   token: token
 };

 let terminal = await createInstance(terminalOptions);

 // Optional: you can store the token somewhere for re-use, such as localStorage.
 localStorage.token = token

 // Optional: Clear the token from the URL
 params.delete('token');
 window.history.replaceState({}, document.title, document.location.pathname + params.toString());
```

Revision #3
Created 1 October 2025 04:55:24
Updated 14 January 2026 10:38:24 by Nikola Savic