How to configure SAML SSO integration with JupiterOne

Single Sign On is supported via a custom authentication client configured within a JupiterOne account. This feature is available to all enterprise customers upon request. To request SSO integration to be enabled for your account, please open a support ticket or contact your technical account manager.

Supported Features

- **SP-initiated SSO**

  Service Provider Initiated (SP-initiated) SSO means when SAML authentication is initiated by the Service Provider (SP). This is triggered when the end user tries to access a resource in JupiterOne or login directly to the JupiterOne account.

- **JIT (Just In Time) Provisioning**

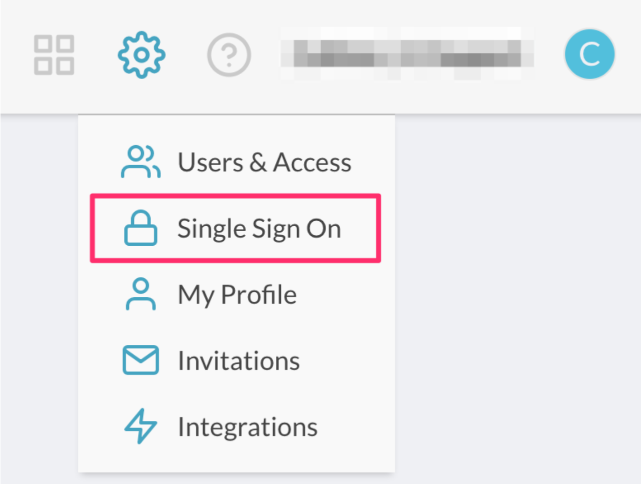
  Users are created/updated on the fly using the SAML attributes sent as part of the SAML response coming from the Identity Provider (IdP). The user is created during initial login to JupiterOne and updated during subsequent logins.

IdP-initiated SSO is currently unsupported due to a limitation of Amazon Cognito.

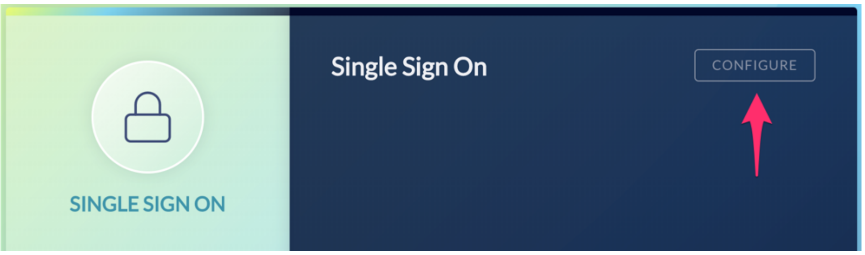
Configuration Steps

1. Log in to your JupiterOne account – your user must be a member of the *Administrators* group.

2. Go to the **Single Sign On** setup from the configurations menu.

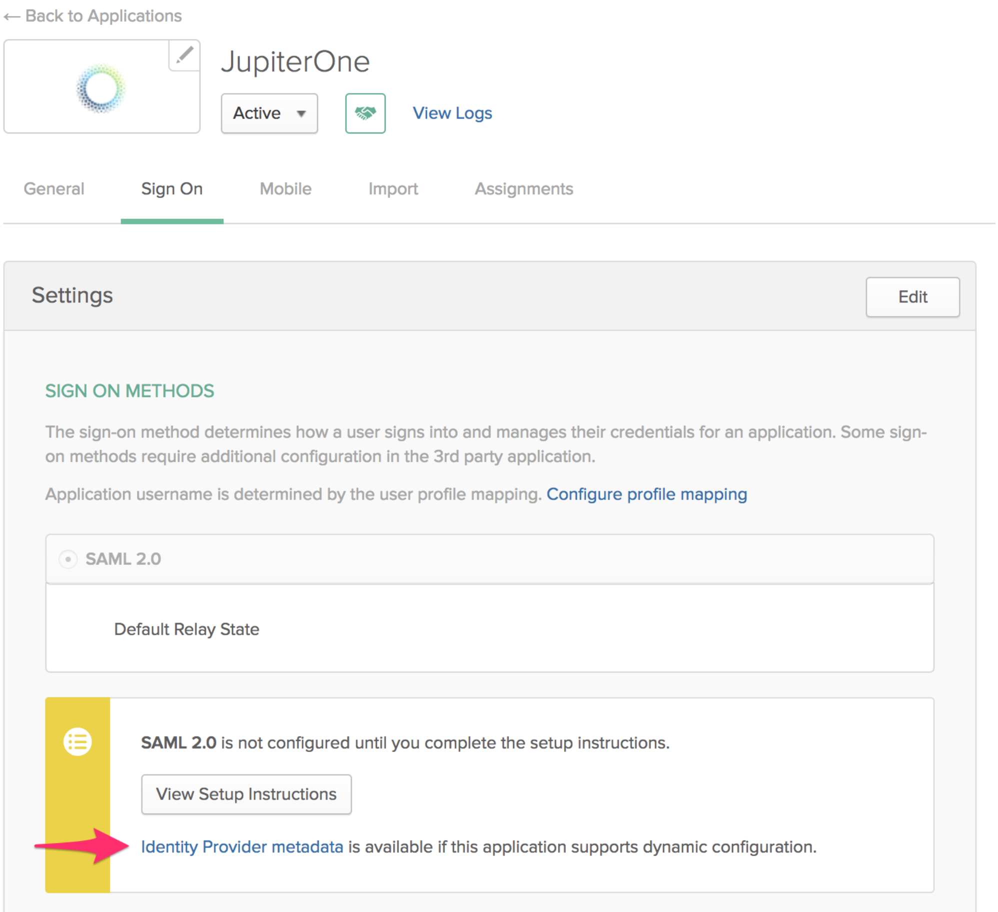


3. Click on **Configure**.



4. In the client configuration screen, copy the following two variables to be used when adding JupiterOne as an application in your SAML IdP account:
   - **Single Sign On URL**
   - **Audience URI (SP Entity ID)**
5. In your IdP Account, add a new SAML Application and name it "JupiterOne".
   - Copy/paste the previous two variable values in the SAML settings.
   - Use the same **Single Sign On URL** string value for **Recipient URL** and **Destination URL.**
   - Leave the **Default Relay State** blank.
   - Select *EmailAddress* for **Name ID Format**.
   - Select *Email* or *Username* for **Application Username**.
   - See next section for details on **Attribute Mappings**.
6. Complete setup of the SAML application within your IdP account, and copy the **Identity Provider Metadata** link.

   In Okta, this link can be found on the **Sign On** tab of the application, under **View Setup Instructions**, as shown below:



The sign-on method determines how a user signs into and manages their credentials for an application. Some sign-on methods require additional configuration in the 3rd party application.

Application username is determined by the user profile mapping. Configure profile mapping

○ SAML 2.0

Default Relay State

SAML 2.0 is not configured until you complete the setup instructions.

Identity Provider metadata is available if this application supports dynamic configuration.

7. Go back to **JupiterOne Auth Client Settings** screen, paste the above link to the **SAML Metadata Document URL** field.

8. Enter a **Client Name**, such as “Okta”, and save.

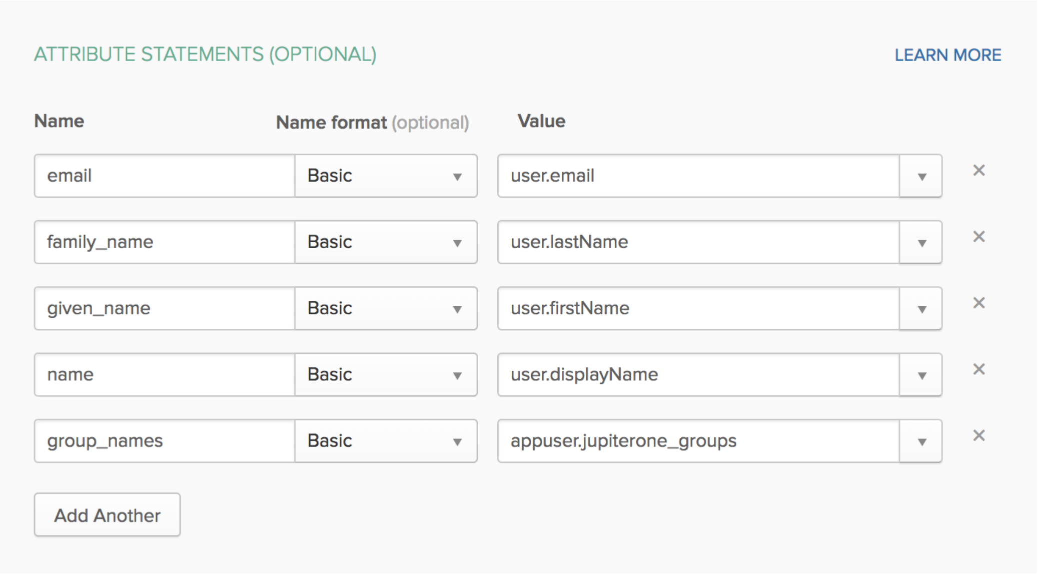
You are all set. Next time you access your JupiterOne account via the vanity URL (e.g. https://your_company.apps.us.jupiterone.io), you should be redirected to your SAML IdP for authentication.

Attribute Mappings

The following attribute mappings are supported:

- `email`: User’s email address
- `family_name`: User’s last name
- `given_name`: User’s first name
- `name`: User’s display name
- `group_names`: Dynamically assigns user to specified groups within JupiterOne. Use a comma to separate multiple group names (without spaces). Users without `group_names` mapping are assigned to the **Users** group within your JupiterOne account by default.

Here’s an example of attribute mapping configuration in Okta:

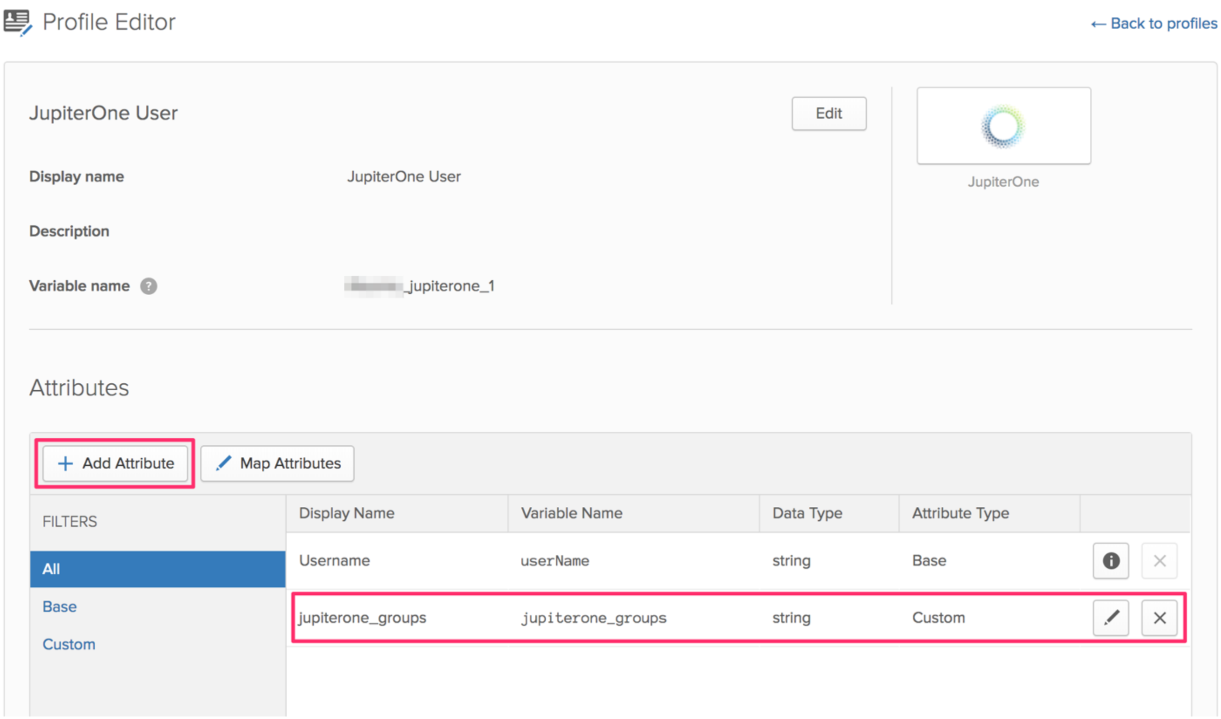


ATTRIBUTE STATEMENTS (OPTIONAL)

Name	Name format (optional)	Value
email	Basic	user.email
family_name	Basic	user.lastName
given_name	Basic	user.firstName
name	Basic	user.displayName
group_names	Basic	appuser.jupiterone_groups

We highly recommend adding a custom *group attribute* to the JupiterOne app profile in your IdP account (e.g. Okta). This is typically added using the **Profile Editor** for the app. You can name the attribute something like `jupiterone_groups`.

Below is an example within Okta:



Profile Editor

JupiterOne User

Display name: JupiterOne User

Description

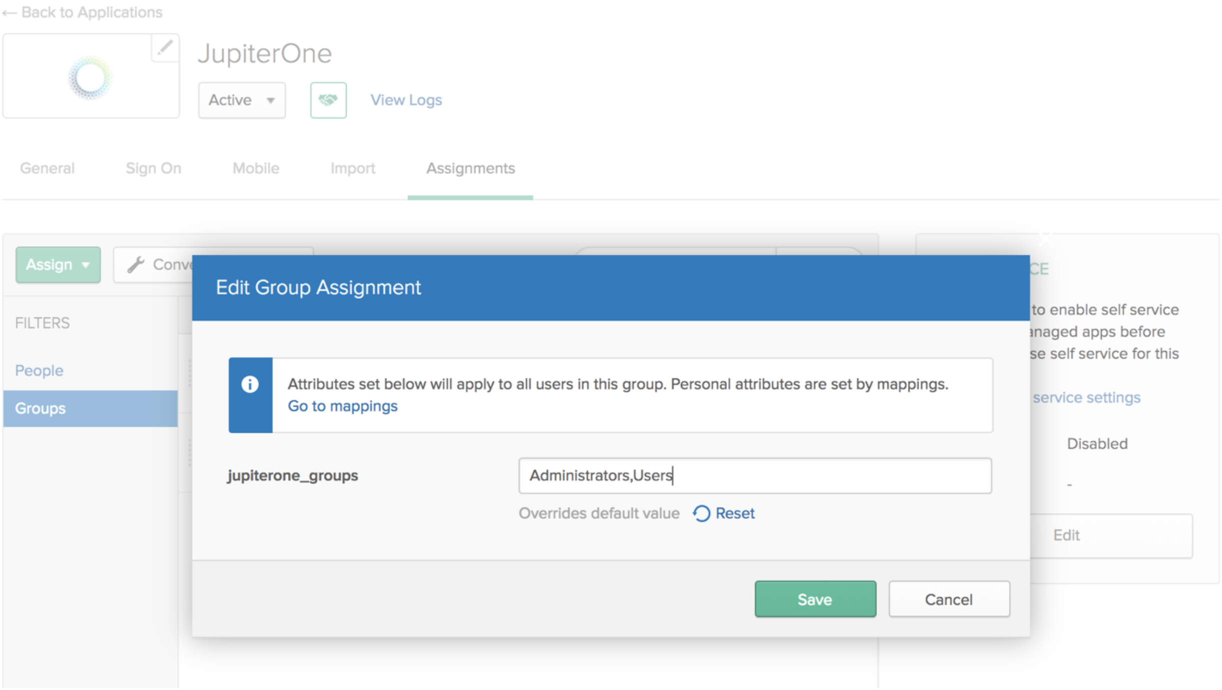
Variable name: [redacted]_jupiterone_1

Attributes

Display Name	Variable Name	Data Type	Attribute Type
Username	userName	string	Base
jupiterone_groups	jupiterone_groups	string	Custom

You can then use this custom app attribute to assign group memberships to your users based on their IdP group assignments. The actual value for the attribute is typically configured on the group(s) assigned to the app.

Below is an example within Okta:



Edit Group Assignment

Attributes set below will apply to all users in this group. Personal attributes are set by mappings.
Go to mappings

jupiterone_groups: Administrators,Users

Overrides default value

Note that provisioning users with `group_names` attribute mapping is *OPTIONAL*. Users without `group_names` mapping are assigned to the **Users** group within your JupiterOne account by default.

Removing Users

When you unassign / remove a user from the JupiterOne app within your IdP, the user will no longer be able to log in to your JupiterOne account because the authentication happens with your IdP. However, the user memberships will remain in the Groups. You can manually remove them from the groups within JupiterOne.

–