+++ title = “SAML Authentication” description = “Grafana SAML Authentication” keywords = [“grafana”, “saml”, “documentation”, “saml-auth”] aliases = [“/docs/grafana/latest/auth/saml/”] weight = 900 +++

SAML authentication

SAML authentication integration allows your Grafana users to log in by using an external SAML 2.0 Identity Provider (IdP). To enable this, Grafana becomes a Service Provider (SP) in the authentication flow, interacting with the IdP to exchange user information.

The SAML single sign-on (SSO) standard is varied and flexible. Our implementation contains a subset of features needed to provide a smooth authentication experience into Grafana.

> Only available in Grafana Enterprise v6.3+. If you encounter any problems with our implementation, please don’t hesitate to contact us.

Supported SAML

Grafana supports the following SAML 2.0 bindings:

- From the Service Provider (SP) to the Identity Provider (IdP):
  - `HTTP-POST` binding
  - `HTTP-Redirect` binding
- From the Identity Provider (IdP) to the Service Provider (SP):
  - `HTTP-POST` binding

In terms of security:

- Grafana supports signed and encrypted assertions.
- Grafana does not support signed or encrypted requests.

In terms of initiation, Grafana supports:

- SP-initiated requests
- IdP-initiated requests

By default, SP-initiated requests are enabled. For instructions on how to enable IdP-initiated logins, see https://grafana.com/docs/grafana/latest/enterprise/saml/#idp-initiated-single-sign-on-sso.

Set up SAML authentication

The table below describes all SAML configuration options. Continue reading below for details on specific options. Like any other Grafana configuration, you can apply these options as [environment variables]({{< relref

"../administration/configuration.md#configure-with-environment-variables" >}}).

Setting	Required	Description	Default
`enabled`	No	Whether SAML authentication is allowed	`false`
`single_logout`	No	Whether SAML Single Logout enabled	`false`
`allow_idp_initiated`	No	Whether SAML IdP-initiated login is allowed	`false`
`certificate` or `certificate_path`	Yes	Base64-encoded string or Path for the SP X.509 certificate	
`private_key` or `private_key_path`	Yes	Base64-encoded string or Path for the SP private key	
`signature_algorithm`	No	Signature algorithm used for signing requests to the IdP. Supported values are rsa-sha1, rsa-sha256, rsa-sha512.	
`idp_metadata`, `idp_metadata_path`, or `idp_metadata_url`	Yes	Base64-encoded string, Path or URL for the IdP SAML metadata XML	
`max_issue_delay`	No	Duration, since the IdP issued a response and the SP is allowed to process it	`90s`
`metadata_valid_duration`	No	Duration, for how long the SP metadata is valid	`48h`
`relay_state`	No	Relay state for IdP-initiated login. Should match relay state configured in IdP	
`assertion_attribute_name`	No	Friendly name or name of the attribute within the SAML assertion to use as the user name	`displayName`
`assertion_attribute_login`	No	Friendly name or name of the attribute within the SAML assertion to use as the user login handle	`mail`
`assertion_attribute_email`	No	Friendly name or name of the attribute within the SAML assertion to use as the user email	`mail`
`assertion_attribute_groups`	No	Friendly name or name of the attribute within the SAML assertion to use as the user groups	
`assertion_attribute_role`	No	Friendly name or name of the attribute within the SAML assertion to use as the user roles	
`assertion_attribute_org`	No	Friendly name or name of the attribute within the SAML assertion to use as the user organization	
`allowed_organizations`	No	List of comma- or space-separated organizations. User should be a member of at least one organization to log in.	
`org_mapping`	No	List of comma- or space-separated Organization:OrgId:Role mappings. Organization can be `*` meaning "All users". Role is optional and can have the following values: `Viewer`, `Editor` or `Admin`.	

Setting	Required	Description	Default
`role_values_editor`	No	List of comma- or space-separated roles which will be mapped into the Editor role	
`role_values_admin`	No	List of comma- or space-separated roles which will be mapped into the Admin role	
`role_values_grafana_admin`	No	List of comma- or space-separated roles which will be mapped into the Grafana Admin (Super Admin) role	

Enable SAML authentication

To use the SAML integration, in the `auth.saml` section of in the Grafana custom configuration file, set `enabled` to `true`.

Refer to [Configuration]({{< relref "../administration/configuration.md" >}}) for more information about configuring Grafana.

Certificate and private key

The SAML SSO standard uses asymmetric encryption to exchange information between the SP (Grafana) and the IdP. To perform such encryption, you need a public part and a private part. In this case, the X.509 certificate provides the public part, while the private key provides the private part.

Grafana supports two ways of specifying both the `certificate` and `private_key`.

- Without a suffix (`certificate` or `private_key`), the configuration assumes you've supplied the base64-encoded file contents.
- With the `_path` suffix (`certificate_path` or `private_key_path`), then Grafana treats the value entered as a file path and attempts to read the file from the file system.

You can only use one form of each configuration option. Using multiple forms, such as both `certificate` and `certificate_path`, results in an error.

Signature algorithm

> Only available in Grafana v7.3+

The SAML standard recommends using a digital signature for some types of messages, like authentication or logout requests. If the `signature_algorithm` option is configured, Grafana will put a digital signature into SAML requests. Supported signature types are `rsa-sha1`, `rsa-sha256`, `rsa-sha512`. This option should match your IdP configuration, otherwise, signature validation will fail. Grafana uses key and certificate configured with `private_key` and `certificate` options for signing SAML requests.

IdP metadata

You also need to define the public part of the IdP for message verification. The SAML IdP metadata XML defines where and how Grafana exchanges user information.

Grafana supports three ways of specifying the IdP metadata.

- Without a suffix `idp_metadata`, Grafana assumes base64-encoded XML file contents.
- With the `_path` suffix, Grafana assumes a file path and attempts to read the file from the file system.
- With the `_url` suffix, Grafana assumes a URL and attempts to load the metadata from the given location.

Maximum issue delay

Prevents SAML response replay attacks and internal clock skews between the SP (Grafana) and the IdP. You can set a maximum amount of time between the IdP issuing a response and the SP (Grafana) processing it.

The configuration options is specified as a duration, such as `max_issue_delay = 90s` or `max_issue_delay = 1h`.

Metadata valid duration

SP metadata is likely to expire at some point, perhaps due to a certificate rotation or change of location binding. Grafana allows you to specify for how long the metadata should be valid. Leveraging the `validUntil` field, you can tell consumers until when your metadata is going to be valid. The duration is computed by adding the duration to the current time.

The configuration option is specified as a duration, such as `metadata_valid_duration = 48h`.

Identity provider (IdP) registration

For the SAML integration to work correctly, you need to make the IdP aware of the SP.

The integration provides two key endpoints as part of Grafana:

- The `/saml/metadata` endpoint, which contains the SP metadata. You can either download and upload it manually, or you make the IdP request it directly from the endpoint. Some providers name it Identifier or Entity ID.
- The `/saml/acs` endpoint, which is intended to receive the ACS (Assertion Customer Service) callback. Some providers name it SSO URL or Reply URL.

IdP-initiated Single Sign-On (SSO)

Only available in Grafana v7.3+

By default, Grafana allows only service provider (SP) initiated logins (when the user logs in with SAML via Grafana's login page). If you want users to log in into Grafana directly from your identity provider (IdP), set the `allow_idp_initiated` configuration option to `true` and configure `relay_state` with the same value specified in the IdP configuration.

IdP-initiated SSO has some security risks, so make sure you understand the risks before enabling this feature. When using IdP-initiated SSO, Grafana receives unsolicited SAML requests and can't verify that login flow was started by the user. This makes it hard to detect whether SAML message has been stolen or replaced. Because of this, IdP-initiated SSO is vulnerable to login cross-site request forgery (CSRF) and man in the middle (MITM) attacks. We do not recommend using IdP-initiated SSO and keeping it disabled whenever possible.

Single logout

Only available in Grafana v7.3+

SAML's single logout feature allows users to log out from all applications associated with the current IdP session established via SAML SSO. If the `single_logout` option is set to `true` and a user logs out, Grafana requests IdP to end the user session which in turn triggers logout from all other applications the user is logged into using the same IdP session (applications should support single logout). Conversely, if another application connected to the same IdP logs out using single logout, Grafana receives a logout request from IdP and ends the user session.

Assertion mapping

During the SAML SSO authentication flow, Grafana receives the ACS callback. The callback contains all the relevant information of the user under authentication embedded in the SAML response. Grafana parses the response to create (or update) the user within its internal database.

For Grafana to map the user information, it looks at the individual attributes within the assertion. You can think of these attributes as Key/Value pairs (although, they contain more information than that).

Grafana provides configuration options that let you modify which keys to look at for these values. The data we need to create the user in Grafana is Name, Login handle, and email.

Configure team sync

Team sync support for SAML only available in Grafana v7.0+

To use SAML Team sync, set [`assertion_attribute_groups`]({{< relref "./enterprise-configuration.md#assertion-attribute-groups" >}}) to the attribute name where you store user groups. Then Grafana will use attribute values extracted from SAML assertion to add user into the groups with the same name configured on the External group sync tab.

[Learn more about Team Sync]({{< relref "../enterprise/team-sync.md" >}})

Configure role sync

> Only available in Grafana v7.0+

Role sync allows you to map user roles from an identity provider to Grafana. To enable role sync, configure role attribute and possible values for the Editor, Admin, and Grafana Admin roles. For more information about user roles, refer to [About users and permissions]({{< relref "../administration/manage-users-and-permissions/about-users-and-permissions.md" >}}).

1. In the configuration file, set [`assertion_attribute_role`]({{< relref "./enterprise-configuration.md#assertion-attribute-role" >}}) option to the attribute name where the role information will be extracted from.
2. Set the [`role_values_editor`]({{< relref "./enterprise-configuration.md#role-values-editor" >}}) option to the values mapped to the `Editor` role.
3. Set the [`role_values_admin`]({{< relref "./enterprise-configuration.md#role-values-admin" >}}) option to the values mapped to the organization `Admin` role.
4. Set the [`role_values_grafana_admin`]({{< relref "./enterprise-configuration.md#role-values-grafana-admin" >}}) option to the values mapped to the `Grafana Admin` role.

If a user role doesn't match any of configured values, then the `Viewer` role will be assigned.

Refer to [About users and permissions]({{< relref "../administration/manage-users-and-permissions/about-users-and-permissions.md" >}}) for more information about roles and permissions in Grafana.

Example configuration:

```
[auth.saml]
assertion_attribute_role = role
role_values_editor = editor, developer
role_values_admin = admin, operator
role_values_grafana_admin = superadmin
```

Important: When role sync is configured, any changes of user roles and organization membership made manually in Grafana will be overwritten on next user login. Assign user organizations and roles in the IdP instead.

Configure organization mapping

Only available in Grafana v7.0+

Organization mapping allows you to assign users to particular organization in Grafana depending on attribute value obtained from identity provider.

1. In configuration file, set [`assertion_attribute_org`]({{< relref "./enterprise-configuration.md#assertion-attribute-org" >}}) to the attribute name you store organization info in. This attribute can be an array if you want a user to be in multiple organizations.
2. Set [`org_mapping`]({{< relref "./enterprise-configuration.md#org-mapping" >}}) option to the comma-separated list of `Organization:OrgId` pairs to map organization from IdP to Grafana organization specified by id. If you want users to have different roles in multiple organizations, you can set this option to a comma-separated list of `Organization:OrgId:Role` mappings.

For example, use following configuration to assign users from `Engineering` organization to the Grafana organization with id `2` as Editor and users from `Sales` - to the org with id `3` as Admin, based on `Org` assertion attribute value:

```
[auth.saml]
assertion_attribute_org = Org
org_mapping = Engineering:2:Editor, Sales:3:Admin
```

You can specify multiple organizations both for the IdP and Grafana:

- `org_mapping = Engineering:2, Sales:2` to map users from `Engineering` and `Sales` to `2` in Grafana.
- `org_mapping = Engineering:2, Engineering:3` to assign `Engineering` to both `2` and `3` in Grafana.

You can use `*` as an Organization if you want all your users to be in some organizations with a default role:

- `org_mapping = *:2:Editor` to map all users to `2` in Grafana as Editors.

Configure allowed organizations

Only available in Grafana v7.0+

With the [`allowed_organizations`]({{< relref "./enterprise-configuration.md#allowed-organizations" >}}) option you can specify a list of organizations where the user must be a member of at least one of them to be able to log in to Grafana.

Example SAML configuration

```
[auth.saml]
enabled = true
certificate_path = "/path/to/certificate.cert"
```

```
private_key_path = "/path/to/private_key.pem"
idp_metadata_path = "/my/metadata.xml"
max_issue_delay = 90s
metadata_valid_duration = 48h
assertion_attribute_name = displayName
assertion_attribute_login = mail
assertion_attribute_email = mail

assertion_attribute_groups = Group
assertion_attribute_role = Role
assertion_attribute_org = Org
role_values_editor = editor, developer
role_values_admin = admin, operator
role_values_grafana_admin = superadmin
org_mapping = Engineering:2:Editor, Engineering:3:Viewer, Sales:3:Editor, *:1:Editor
allowed_organizations = Engineering, Sales
```

Set up SAML with Okta

This guide will follow you through the steps of configuring SAML authentication in Grafana with Okta. You need to be an admin in your Okta organization to access Admin Console and create SAML integration. You also need permissions to edit Grafana config file and restart Grafana server.

Configure the SAML integration in Okta

To configure SAML integration with Okta, create integration inside the Okta organization first.

1. Log in to the Okta portal.
2. Go to the Admin Console in your Okta organization by clicking **Admin** in the upper-right corner. If you are in the Developer Console, then click **Developer Console** in the upper-left corner and then click **Classic UI** to switch over to the Admin Console.
3. In the Admin Console, navigate to **Applications** > **Applications**.
4. Click **Add Application**.
5. Click **Create New App** to start the Application Integration Wizard.
6. Choose **Web** as a platform.
7. Select **SAML 2.0** in the Sign on method section.
8. Click **Create**.
9. On the **General Settings** tab, enter a name for your Grafana integration. You can also upload a logo.

10. On the **Configure SAML** tab, enter the SAML information related to your Grafana instance:
    - In the **Single sign on URL** field, use the `/saml/acs` endpoint URL of your Grafana instance, for example, `https://grafana.example.com/saml/acs`.
    - In the **Audience URI (SP Entity ID)** field, use the `/saml/metadata` endpoint URL, for example, `https://grafana.example.com/saml/metadata`.
    - Leave the default values for **Name ID format** and **Application username**.
    - In the **ATTRIBUTE STATEMENTS (OPTIONAL)** section, enter the SAML attributes to be shared with Grafana, for example:

Attribute name (in Grafana)	Value (in Okta profile)
Login	`user.login`
Email	`user.email`
DisplayName	`user.firstName + " " + user.lastName`

    - In the **GROUP ATTRIBUTE STATEMENTS (OPTIONAL)** section, enter a group attribute name (for example, `Group`) and set filter to `Matches regex .*` to return all user groups.
11. Click **Next**.
12. On the final Feedback tab, fill out the form and then click **Finish**.

Edit SAML options in the Grafana config file

Once the application is created, configure Grafana to use it for SAML authentication. Refer to [Configuration]({{< relref "../administration/configuration.md" >}}) to get more information about how to configure Grafana.

1. In the `[auth.saml]` section in the Grafana configuration file, set [`enabled`]({{< relref "./enterprise-configuration.md#enabled" >}}) to `true`.
2. Configure the certificate and private key({{< relref "#certificate-and-private-key" >}}).
3. On the Okta application page where you have been redirected after application created, navigate to the **Sign On** tab and find **Identity Provider metadata** link in the **Settings** section.
4. Set the [`idp_metadata_url`]({{< relref "./enterprise-configuration.md#idp-metadata-url" >}}) to the URL obtained from the previous step. The URL should look like `https://<your-org-id>.okta.com/app/<application-id>/sso/saml/metadata`.
5. Set the following options to the attribute names configured at the **step 10** of the SAML integration setup. You can find this attributes on the **General** tab of the application page (**ATTRIBUTE STATEMENTS**

and **GROUP ATTRIBUTE STATEMENTS** in the **SAML Settings** section).

- [`assertion_attribute_login`]({{< relref "./enterprise-configuration.md#assertion-attribute-login" >}})
- [`assertion_attribute_email`]({{< relref "./enterprise-configuration.md#assertion-attribute-email" >}})
- [`assertion_attribute_name`]({{< relref "./enterprise-configuration.md#assertion-attribute-name" >}})
- [`assertion_attribute_groups`]({{< relref "./enterprise-configuration.md#assertion-attribute-groups" >}})

6. Save the configuration file and and then restart the Grafana server.

When you are finished, the Grafana configuration might look like this example:

```
[server]
root_url = https://grafana.example.com

[auth.saml]
enabled = true
private_key_path = "/path/to/private_key.pem"
certificate_path = "/path/to/certificate.cert"
idp_metadata_url = "https://my-org.okta.com/app/my-application/sso/saml/metadata"
assertion_attribute_name = DisplayName
assertion_attribute_login = Login
assertion_attribute_email = Email
assertion_attribute_groups = Group
```

Troubleshoot SAML authentication

To troubleshoot and get more log information, enable SAML debug logging in the configuration file. Refer to [Configuration]({{< relref "../administration/configuration.md#filters" >}}) for more information.

```
[log]
filters = saml.auth:debug
```