Authentication

- LDAP Active Directory
- OpenID Connect

LDAP Active Directory

This guide assumes you already have a forest/domain configured and that your environment is properly secured.

This is a basic configuration. You may want to set additional options or permissions in your forest/domain.

See also Delegated Authentication for single sign-on (SSO) integration.

Setup

To enable authentication with LDAP and Active Directory, the following needs to be done:

- Configure secure LDAP in your domain.
- Create a user and optionally an UO to use for LDAP authentication.

Configure Your EMS Server

- Set up an Element Cloud Enterprise server.
- Click the Integrations tab.
- Select LDAP from the list of available Advanced Authentication methods.
- Set the following configuration parameters:

```
Bind URI: ldaps://ldap.example.com:636

Base: OU=matrix,DC=example,DC=com

Bind DN: CN=emsadmin,CN=Users,DC=example,DD=com

Bind Password: supersecret

UID: SamAccountName

Display Name: See below

Email: mail
```

- For Display Name, you have a few options based on your preference. For example:
  - displayName
  - GivenName
  - Name
  - sn
- For a full list, open PowerShell on your domain controller and enter

```
Import-Module ActiveDirectory
Get-ADUser test_user -Properties *
```

- Save your LDAP settings and wait for your EMS server to reprovision.
- Authentication in Element should now be working. If not, please look in the logs for your firewall or domain controllers or contact EMS support from our support form

OpenID Connect

Your homeserver can be configured to authenticate its users with an OpenID Connect provider. Here we list the most popular providers and how to configure them.

- Authentik
- Gitea
- GitHub
- GitLab
- Google
- Okta

See also Delegated Authentication for single sign-on (SSO) integration .

Authentik

- Create a new `OAuth2/OpenID Provider` provider
- Name: can be anything
- Authentication flow: `default-authentication-flow`
- Authorization flow: `default-provider-authentication-explicit-consent`
- Client type: Confidential
- Take note of the Client ID and Client Secret
- Redirect URIs/Origins (RegEx): `https://my-host.ems.host/_synapse/client/oidc/callback`. Adapt the URL to match your homeserver's address. You must use your `.ems.host` domain, even if your server uses Custom DNS.
- Signing Key: `authentik Self-signed Certificate`
- Create an application using the provider you just created. Take note of the Slug

In the Element Matrix Services configuration form

- Preset: `Custom`
- Issuer: `https://your-authentik-instance.com/application/o/the-slug-from-above/` (you can also find this URL on the provider as `OpenID Configuration Issuer`)
- Client ID and Secret: Values from above
- Discover endpoints: Enable

- Scopes: `openid,profile,email`
- Subject claim: `sub`
- Username attribute: `preferred_username`
- Display name attribute: `name`

Gitea

- Create a new `OAuth2 Application` on https://your-gitea-instance.com/user/settings/applications
- Choose a name for you and your users to recognize
- Set `Redirect URIs` to `https://my-host.ems.host/_synapse/client/oidc/callback`. Adapt the URL to match your homeserver's address. You must use your `.ems.host` domain, even if your server uses Custom DNS.
- Confidential Client: enable

In the Element Matrix Services configuration form

- Preset: `Custom`
- Issuer: `https://your-gitea-instance.com/`
- Client ID and Secret: Values given by Gitea OAuth2 settings
- Discover endpoints: Enable
- Scopes: `openid,profile`
- Subject claim: leave empty
- Username attribute: `name`
- Display name attribute: `name`'

GitHub

For detailed information, read GitHub's guide on OpenID .

1. Create a new application on GitHub.com .
2. Choose a name for you and your users to recognize.
3. Choose a homepage URL. You can pick any URL. If your company maintains a guide on how to use Matrix, this would be most helpful.

4. The Authorization callback URL needs to be `https://my-host.ems.host`. Adapt the URL to match your homeserver's address. You must use your `.ems.host` domain, even if your server uses Custom DNS.
5. Save and note the client ID and client secret. Those are needed when adding the OpenID Connect integration in our interface.

In the Element Matrix Services configuration form

Use the preset `GitHub` for a simplified form or use `Custom` with the following values:

- Issuer must be `https://github.com/`
- Use the client id and secret from above.
- Discover must be turned off.
- Authorization URI must be `https://github.com/login/oauth/authorize`.
- Token URI must be `https://github.com/login/oauth/authorize`.
- User Info URI must be `https://api.github.com/user`.
- JWKS URI is not required, because the scope `profile` will be requested.
- The scopes should be `openid,profile,read:user`.
- Subject Claim must be `id`.
- Username attribute should be `login`.
- The display name can be `name` (GitHub's display name) or `login` (GitHub's user handle).

GitLab

For detailed information, read GitLab's guide on OpenID .

1. Create a new application on GitLab.com.
2. Choose a name for you and your users to recognize.
3. Choose a homepage URL. You can pick any URL. If your company maintains a guide on how to use Matrix, this would be most helpful.
4. The Redirect URL needs to be `https://my-host.ems.host/_synapse/client/oidc/callback`. Adapt the URL to match your homeserver's address. You must use your `.ems.host` domain, even if your server uses Custom DNS.
5. Check the scopes `read_user`, `openid` and `profile`.
6. Save and note the client ID and client secret. Those are needed when adding the OpenID Connect integration in our interface.

To connect your own GitLab instance, simply adapt the URL path.

In the Element Matrix Services configuration form

- Issuer must be `https://gitlab.com/` or the URL of your GitLab instance.
- Use the client id and secret from above.
- Discover must be turned on.
- The scopes should be `openid,profile,read_user`.
- Leave Subject Claim empty.
- Username attribute should be `nickname`.
- Display name attribute can be `name` (GitLab's display name) or `nickname` (GitLab's user handle).

Google

For detailed information, read Google's guide on OpenID .

1. Create a new application on Google .
2. Click `Create credentials` and `OAuth client ID`.
3. Select the application type `Web application`.
4. Choose a name for you and your users to recognize.
5. Add an authorized redirect URI with your homeserver URL, like `https://my-host.ems.host/_synapse/client/oidc/callback`. You must use your `.ems.host` domain, even if your server uses Custom DNS.
6. Save and note the client ID and client secret. Those are needed when adding the OpenID Connect integration in our interface.

In the Element Matrix Services configuration form

Use the preset `Google` for a simplified form or use `Custom` with the following values:

- Issuer must be `https://accounts.google.com/`.
- Use the client id and secret from above.
- Discover must be turned on.
- The scopes should be `openid,profile,email,name`.
- Leave Subject Claim empty.
- Username attribute can be `email`. This means your Matrix addresses will include the server domain of the user's e-mail address.

- Display name attribute can be `name`.

If you want shorter usernames and are not worried about username collisions within your domain, please consider using SAML2 to authenticate with Google.

Okta

For detailed information, read Okta's guide onOpenID.

1. Create a new App. Sign-in method `OIDC - OpenID Connect` and Application type `Web Application`.
2. Choose a name for you and your users to recognize.
3. Sign-in redirect URIs: `https://my-host.ems.host/_synapse/client/oidc/callback`. Adapt the URL to match your homeserver's address. You must use your `.ems.host` domain, even if your server uses Custom DNS.
4. Sign-out redirect URIs: `https://my-host.ems.host/_synapse/client/oidc/backchannel_logout`. Adapt the URL to match your homeserver's address. You must use your `.ems.host` domain, even if your server uses Custom DNS.

In the Element Matrix Services configuration form

- Choose Preset Custom.
- Issuer: `https://your-domain.okta.com`.
- Client ID: Your client ID from the Okta admin panel.
- Client secret: Your client secret from the Okta admin panel.
- Scopes: `openid,profile`.
- Leave Subject Claim empty.
- Username attribute: See below
- Display name attribute: for example, `given_name family_name`

Username attributes

This refers to the user's localpart in their Matrix ID (`@localpart:example.com`). The data provided in a minimally configured Okta user is not ideal for integration with EMS. Below are some possible configuration suggestions. All examples below use the Matrix server domain `example.com`.

Available values for username and display name are `email` (you must include `email` in Scopes), `phone_number` (you must include `phone` in Scopes), `address`, `name`, `family_name`, `given_name`, `middle_name`, `nickname`, `preferred_username`, `profile`, `picture`, `website`, `gender`, `birthdate`, `zoneinfo`, `locale`, and `updated_at`. (List updated June 8, 2023. See this document for updated information. Available options are listed in the table under

"Scopes" and after the "profile" bullet under "Scope values").

Make sure all users that will be using your EMS server have the selected attributes set.

Option 1: Username attribute `email`. This will use the user's entire email address as their localpart. Including the domain. It will also be encoded to be compatible with Matrix. For example, email `jane@example.com` will become `jane=40example.com:example.com`. To use the email, you must also include `email` in Scopes.

Option 1b: We can add some logic to your OIDC config to exclude the email domain. Contact support for further details.

Option 2: Username attributes: `name`. This will use the user's full name from Okta. Note that spaces (and other special characters) are not supported in Matrix localparts. For example, spaces will be encoded as `=20`. (I.e., `Jane Doe` becomes `@jane=20doe:example.com`.

Option 2b: We can add some logic to replace spaces with for example underscore. Contact us for details.

Option 3: By default, usernames in Okta must be an email. But, if you have changed this behavior, you can set Username attributes to `preferred_username` to use the username.

Note, the attribute you choose for localparts does not have to be unique. But if you, for example, set Username attributes to `given_name`, the first Jane who sign in to your EMS server will become `@jane:example.com` and the second Jane becomes `@jane1:example.com`.

Please contact EMS Support at https://ems.element.io/support to discuss your options.