

Authentication Section

A detailed look at Delegated Authentication options available and setup examples.

Config:



In the Authentication section you will find options to configure settings specific to Authentication. Regardless of if you are using the Matrix Authentication Server, or have enabled Legacy Auth, the settings on this page will remain the same.

However please note, MAS does not support delegated authentication with SAML or GroupSync - if you wish to enable either of these you will need to return to the Host section and enable Legacy Auth.

All settings configured via the UI in this section will be saved to your `deployment.yml`, with the contents of secrets being saved to `secrets.yml`. You will find specific configuration examples in each section.

Config Example

- `deployment.yml`

```
metadata:
```

```
annotations:
```

```
ui.element.io/layer: |
```

```
components:
```

```
synapse:
```

```
spec:
```

```
components:
```

```
synapse:
```

config:

delegatedAuth:

- secrets.yml

kind: Secret

metadata:

name: synapse

namespace: element-onprem

data:

By default, if you do not change any settings on this page, defaults will be added to your configuration file/s (see example below).

Config Example

- deployment.yml

metadata:

annotations:

ui.element.io/layer: |

components:

spec:

synapse:

config:

delegatedAuth:

localPasswordDatabase:

enableRegistration: false # Note, if you deploy without any authentication methods enabled, the installer will default to Local Accounts.

- secrets.yml

apiVersion: v1

kind: Secret

metadata:

data:

ldapBindPassword: examplePassword

User Profiles

User Profiles

User profiles permissions

☒ Allow Avatar Change

Allow users to change their avatars themselves

Default

☒ Allow Display Name Change

Allow users to change their display names themselves

Default

☒ Allow Email Change

Allow users to change their emails themselves

Default

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

userProfiles:

allowAvatarChange: true # Not present if left as default

allowDisplayNameChange: true # Not present if left as default


allowEmailChange: true # Not present if left as default

The User Profiles section provides some self-explanatory config options to adjust what changes users are allowed to make to their User Profile, such as changing their Display Name. You may wish to restrict this if you'd prefer to delegate the setting of these values to the associated Identity Provider.

OIDC

You can add and configure one, or multiple, OIDC providers - to do so you will need to click the [Add OIDC](#) / [Add more OIDC](#) button found after toggling on the ODIC section:

OIDC



The list of OIDC Providers

[Add OIDC](#)

Once an OIDC provider is added, you can remove any providers by clicking the rubbish bin icon found to the left of the provider.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

-

IdP Name

IdP Name *

The display name of the Identity Provider (IDP).

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

idpName: example_name # Required

IdP ID

IdP ID *

01JEB56FGVQTA4T27VAR0WDSW1

Edited

The unique identifier for the Identity Provider (IDP).

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

idpId: 01JDS2WKNYTQS21GFAKM9AKD9R # Required

IdP Brand

IdP Brand

An optional styling hint for clients.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

idpBrand: example_brand

Issuer

Issuer *

The URL of the IDP issuer.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

issuer: https://issuer.example.com/ # Required

Client Auth Method

Client Auth Method

Client Secret Basic

Default ▲

Client Secret Post

Client Secret Basic

None

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

clientAuthMethod: client_secret_basic # If no `clientAuthMethod` defined, will default to `client_secret_basic`

clientAuthMethod: client_secret_post

clientAuthMethod: none

Client ID

Client ID *

The client identifier assigned by the IDP.

Config Example

```
spec:

components:

  synapse:

    config:


      delegatedAuth:

        oidc:

          clientId: example_client_id
```

Client Secret

Secrets / Synapse / **OIDC Client Secret** ▾

Client Secret 

The client secret assigned by the IDP.

Config Example

- deployment.yml

```
spec:
```

components:

synapse:

config:

delegatedAuth:

oidc:

clientSecretSecretKey: oidcClientSecret

- secrets.yml

apiVersion: v1

kind: Secret

metadata:

name: synapse

namespace: element-onprem

data:

oidcClientSecret: U2VjdXJIT0lEQ0NsaWVudFNlY3JldA==

Allow Existing Users

☐ Allow Existing Users

Whether to allow existing users or not. This option does nothing if Matrix Authentication Service is deployed.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

Scopes

Scopes

A list of scopes requested during the authorization process.



Standard scopes include openid, profile, email.

openid

Default



Standard scopes include openid, profile, email.

profile

Default



Standard scopes include openid, profile, email.

email

Default

Add more Scopes

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

scopes:

- openid

- profile

- email

User Mapping Provider

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

userMappingProvider:

Subject Template

Subject Template



The claim used to identify the subject of the ID token.

Subject Template *

|



The claim used to identify the subject of the ID token.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

userMappingProvider:

subjectTemplate: '{{ user.subject }}'

Localpart Template

Localpart Template



The template used to generate the local part of the user's Matrix ID.

Localpart Template *

{{ user.preferred_username }}

Edited

The template used to generate the local part of the user's Matrix ID.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

userMappingProvider:

localpartTemplate: '{{ user.preferred_username }}'

If using legacy auth, you should use jinja python to format your template; if using MAS, you should use jinja rust formatting instead. For example, to get the a valid localpart from an email, you would use `{{ user.preferred_username.split('@')[0] }}` if using Legacy Auth, or `{{ (user.preferred_username | split('@'))[0] }}` if using MAS.

Display Name Template

Display Name Template



The template used to generate the user's display name in Matrix.

Display Name Template *

`{{ user.name }}`

Edited

The template used to generate the user's display name in Matrix.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

userMappingProvider:

displayNameTemplate: '{{ user.name }}'

Email Template

Email Template



The template used to generate the user's email address in Matrix.

Email Template *

{{ user.email }}

Edited

The template used to generate the user's email address in Matrix.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

userMappingProvider:

emailTemplate: '{{ user.email }}'

Endpoints Discovery

Auto Discovery

Endpoints Discovery

Use OpenID Connect Discovery to discover the server configuration. The endpoint `<openid issuer>/well-known/openid-configuration` needs to be reachable for automatic discovery to work.

☐ Manual ☒ Auto

☐ Skip Verification

Set to 'true' to skip metadata verification. Use this if you are connecting to a provider that is not OpenID Connect compliant. Defaults to false. Avoid this in production.

Default

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

- clientId: synapsekieranml

clientSecretSecretKey: oidcClientSecret

endpointsDiscovery:

skipVerification: false

idpId: 01JDS2WKNYTQS21GFAKM9AKD9R

idpName: Keycloak

issuer: https://keycloak.ems-support.element.dev/realms/matrix

scopes:

- openid

- profile

- email

userMappingProvider:

displayNameTemplate: '{{ user.name }}'

emailTemplate: '{{ user.email }}'

Skip Verification

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

- clientId: synapsekieranml

clientSecretSecretKey: oidcClientSecret

endpointsDiscovery:

skipVerification: false

idpId: 01JDS2WKNYTS21GFAKM9AKD9R

idpName: Keycloak

issuer: https://keycloak.ems-support.element.dev/realms/matrix

scopes:

- openid

- profile

- email

userMappingProvider:

displayNameTemplate: '{{ user.name }}'

emailTemplate: '{{ user.email }}'

Backchannel Logout Enabled

The Matrix Authentication Service does not support configuring Backchannel Logout. You can only configure Backchannel logout if you have enabled [Legacy Auth](#) from the [Host Section](#).

☐ Backchannel Logout Enabled

An optional flag to enable/disable backchannel logout support.

Config Example

spec:

components:

synapse:

config:

delegatedAuth:

oidc:

- clientId: synapsekieranml

clientSecretSecretKey: oidcClientSecret

endpointsDiscovery:

skipVerification: false

idpId: 01JDS2WKNYTQS21GFAKM9AKD9R

idpName: Keycloak

issuer: https://keycloak.ems-support.element.dev/realms/matrix

scopes:

- openid

- profile

- email

userMappingProvider:

displayNameTemplate: '{{ user.name }}'

emailTemplate: '{{ user.email }}'

SAML

The Matrix Authentication Service does not support SAML and it is recommended to switch to OIDC. You can only enable SAML authentication if you have enabled [Legacy Auth](#) from the [Host Section](#).

LDAP

Local Accounts

Revision #4

Created 4 June 2025 09:18:09 by Kieran Mitchell Lane

Updated 4 June 2025 09:53:06 by Kieran Mitchell Lane