Installation Guide

Prerequisites

- Kubernetes cluster set up with a namespace dedicated for the application
- Three configured DNS records with active SSL certificates and keys
  - main (ui) domain
  - admin domain
  - identity provider (IDP) domain
- Kubernetes node pool with taints "`project=geoss:NoExecute`"
  Minimal requirements for the nodes are: `4 vCPU, 8Gb RAM`
  Minimal number of nodes in the node pool is `6`
- Allowed access from Kubernetes cluster to docker repository containing application images
- Helm installed on the server used to deploy the application
- Persistent volumes and storage class provided in the cluster accordingly to the following list
- Elasticsearch operator Helm chart installed

Required persistent volumes

PV names are examples provided for DEPLOY_ENV = “prod”. You should edit the names accordingly when using other value.

PV name	Minimal size	Comments
geoss-prod-db-data-pv	10 GB	Requires high I/O throughput (should utilize raw disk access solution instead of file storage like NFS, S3 etc.)
geoss-prod-contents-repository-storage-pv	10 GB	Should allow manual access (e.g. by FTP)
geoss-prod-kibana-storage-pv	1 GB	
geoss-prod-matomo-storage-pv	1 GB	

Required storage class

Create a storage class named `elasticsearch-storage-class` (high I/O throughput required)

Example:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name:  elasticsearch-storage-class
provisioner: disk.csi.azure.com
volumeBindingMode: WaitForFirstConsumer
allowVolumeExpansion: true
reclaimPolicy: "Retain"
parameters:
  skuName: "Premium_LRS"
```

Elasticsearch operator installation

```
helm repo add elastic https://helm.elastic.co
helm repo update
helm install elastic-operator elastic/eck-operator -n elastic-system --create-namespace
```

Deployment process

Eversis CI/CD pipeline will build and upload images into external images repository available for clients. We will also prepare Helm charts files available for downloads. Users will be able to download Helm charts, edit variables and deploy application on their servers.

Installation guide

1. Copy the Helm charts to the machine with access to Kubernetes cluster

2. Configure ingresses section accordingly to your hosting solution

Ingesses configuration is located in geoss-nginx/values.yaml.template

(Default configuration is designed for AKS cluster connected with Application Load Balancer in Azure Cloud)

3. Set application variables

Create .env file by copying .env.template and filling variables values

VARIABLE NAME	Description
DOCKER_REPOSITORY_ADDRESS	Address of Docker repository containing application images
K8S_NAMESPACE	Kubernetes namespace where the application should be deployed
DEPLOY_ENV	Purpose of the environment. (“dev”, “uat” or “prod”)
DOCKER_IMAGE_TAG	Tag of a specific image release in the container registry

UI_DOMAIN_NAME	Public domain name of the portal
IDP_DOMAIN_NAME	Public domain name of the Keycloak service
ADMIN_DOMAIN_NAME	Public domain name of the admin portal
CSP_DOMAINS	A comma-separated list of domains that should be added to `Content-Security-Policy` header All public domains of the application (UI, IDP, ADMIN) should be present on this list
INGRESS_ALLOWED_CIDR	IP block (in CIDR notation) of the network that is allowed to connect to the application
INGRESS_EXTERNAL_IP	Public IP of the ingress. Should have the same value as the DNS record of the domains
BASIC_AUTH_ENABLED	Should be set to yes if access to the application needs to be restricted
BASIC_AUTH_LOGIN	Login for the basic authentication
BASIC_AUTH_PASSWORD	Password for the basic authentication
BASIC_AUTH_WHITELIST	Comma separated list of IP addresses and IP blocks in CIDR notation that are not required to provide the basic auth password
MAINTENANCE_ON	Should be set to yes if maintenance mode is required
MAINTENANCE_WHITELIST	A comma-separated list of IP addresses and IP blocks in CIDR notation that are permitted to access the site during maintenance mode
MARIADB_ROOT_PASSWORD	Password of the root MariaDB user
DATABASE_USERNAME_APP	Username of the database user
DATABASE_PASSWORD_APP	Password of the database user
ELS_ELASTIC_PASSWORD	Password of default elastic user
ELS_KIBANA_PASSWORD	Password of kibana_system user
ELS_GEOSS_PASSWORD	Password of geoss admin user
KEYCLOAK_ADMIN_USERNAME	Username of the Keycloak admin user
KEYCLOAK_ADMIN_PASSWORD	Password of the Keycloak admin user
MAIL_HOST	Hostname of the mail server
MAIL_PORT	Port number of the mail server
MAIL_USERNAME	Username of the mail account
MAIL_PASSWORD	Password of the mail account
DATASOURCE_AMERIGEOSS_CKAN_BASE_URL	
DATASOURCE_ZENODO_BASE_URL	
WORKER_DAB_GEODAB_BASE_URL	
WORKER_DAB_VLAB_BASE_URL	
WORKER_DAB_VLAB_API_TOKEN	
WORKER_SDG_DEFAULT_LOGO	
WORKER_SDG_UN_BASE_URL	
WORKER_WIKIDATA_API_URL	

WORKER_WIKIDATA_CATEGORIES_SPARQL_URL	
WORKER_WIKIDATA_CATEGORIES_SPARQL_DEFAULT_GRAPH_URI	
WORKER_THESAURUS_ESA_BASE_URI	
WORKER_THESAURUS_ESA_TOP_CONCEPTS_URIS	
WORKER_THESAURUS_EOSTERM_BASE_URI	
WORKER_THESAURUS_EARTH_BASE_URI	
NEXT_AUTH_SECRET	Should be generated using command `openssl rand -base64 32`
SERVICES_PROVIDERS	Link to services providers (optional)
DATABASE_USERNAME_MATOMO	Matomo database user (optional)
DATABASE_PASSWORD_MATOMO	Matomo database password (optional)
MATOMO_USERNAME	Matomo user name (optional)
MATOMO_PASSWORD	Matomo user password (optional)
MATOMO_DATABASE_NAME	Matomo database name (optional)
MATOMO_TOKEN	Matomo authorization token for fetching statistics (optional)

4. Provide SSL files(certificates and keys)

Main domain SSL

Provide:
`./ui.crt` - certificate in following format (PEM):

```
-----BEGIN CERTIFICATE-----
MIICyDCCAbCgAwIBAgIUIb8q5kLJx... (certificate for your domain)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 1, if
applicable)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 2, if
applicable)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 3, if
applicable)
-----END CERTIFICATE-----
```

`./ui.key` - certificate key in following format (PEM - unencrypted):

```
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFA... (private key)
-----END PRIVATE KEY-----
```

Admin domain SSL

Provide:
`./admin.crt` - certificate in following format (PEM):

```
-----BEGIN CERTIFICATE-----
MIICyDCCAbCgAwIBAgIUIb8q5kLJx... (certificate for your domain)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 1, if
applicable)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 2, if
applicable)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 3, if
applicable)
-----END CERTIFICATE-----
```

`./admin.key` - certificate key in following format (PEM - unencrypted):

```
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFA... (private key)
-----END PRIVATE KEY-----
```

IDP domain SSL

Provide:
`./idp.crt` - certificate in following format (PEM):

```
-----BEGIN CERTIFICATE-----
MIICyDCCAbCgAwIBAgIUIb8q5kLJx... (certificate for your domain)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 1, if applicable)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 2, if applicable)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE0DCCA7igAwIBAgIUIb8q5kLJx... (intermediate certificate 3, if applicable)
-----END CERTIFICATE-----
```

`./idp.key` - certificate key in following format (PEM - unencrypted):

```
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFA... (private key)
-----END PRIVATE KEY-----
```

5. Run installation script

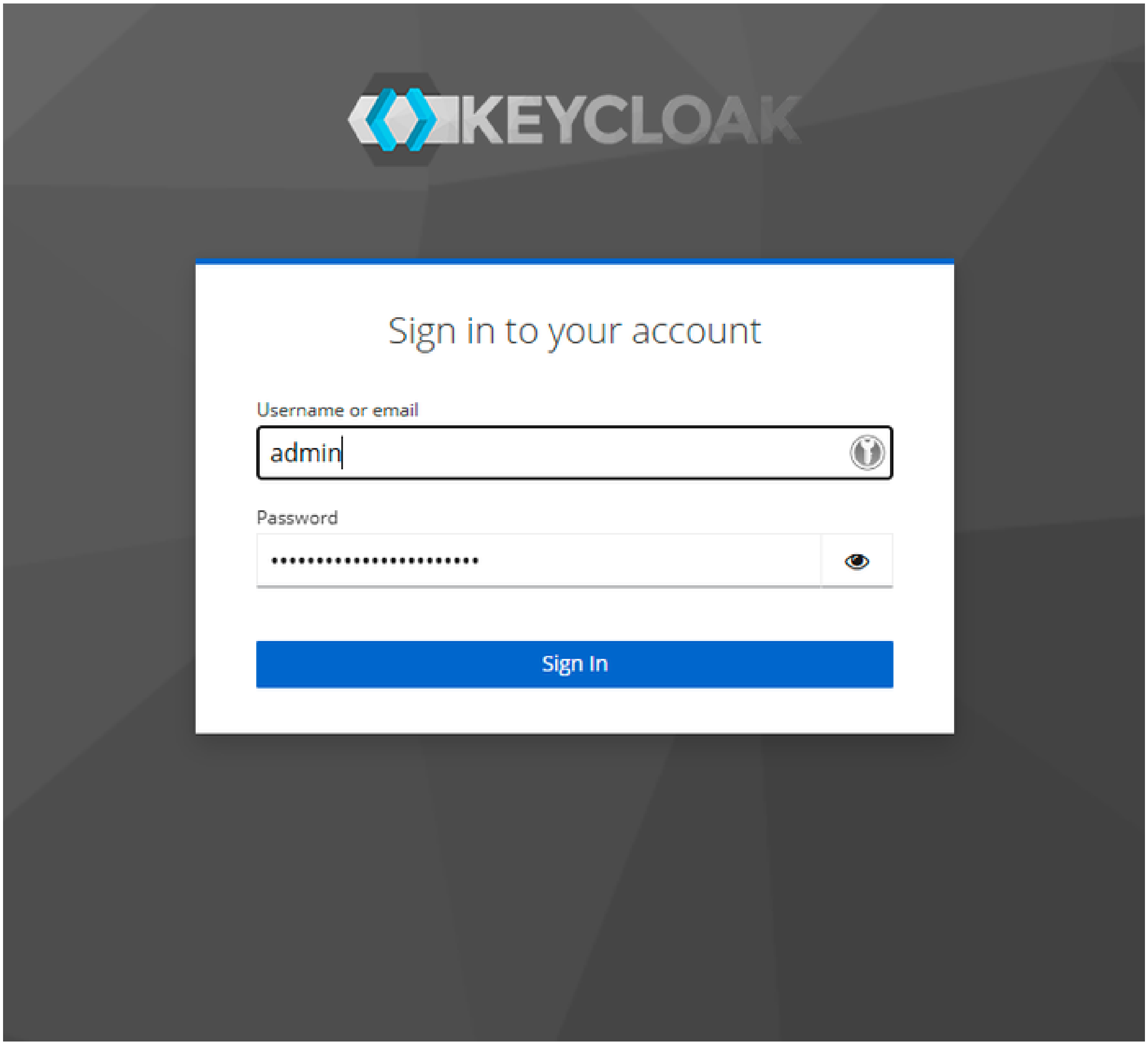
`chmod +x ./install.sh` - make the script executable

`./install.sh` - execute the script

6. Regenerate client secrets in keycloak

In previous step application has been started using default keycloak secret which isn’t secure.
This secret has to be regenerated by taking following actions:

1. Go to the IDP domain address, you have set in the variables. (IDP_DOMAIN_NAME)
2. Fill the username and password. (KEYCLOAK_ADMIN_USERNAME and KEYCLOAK_ADMIN_PASSWORD)

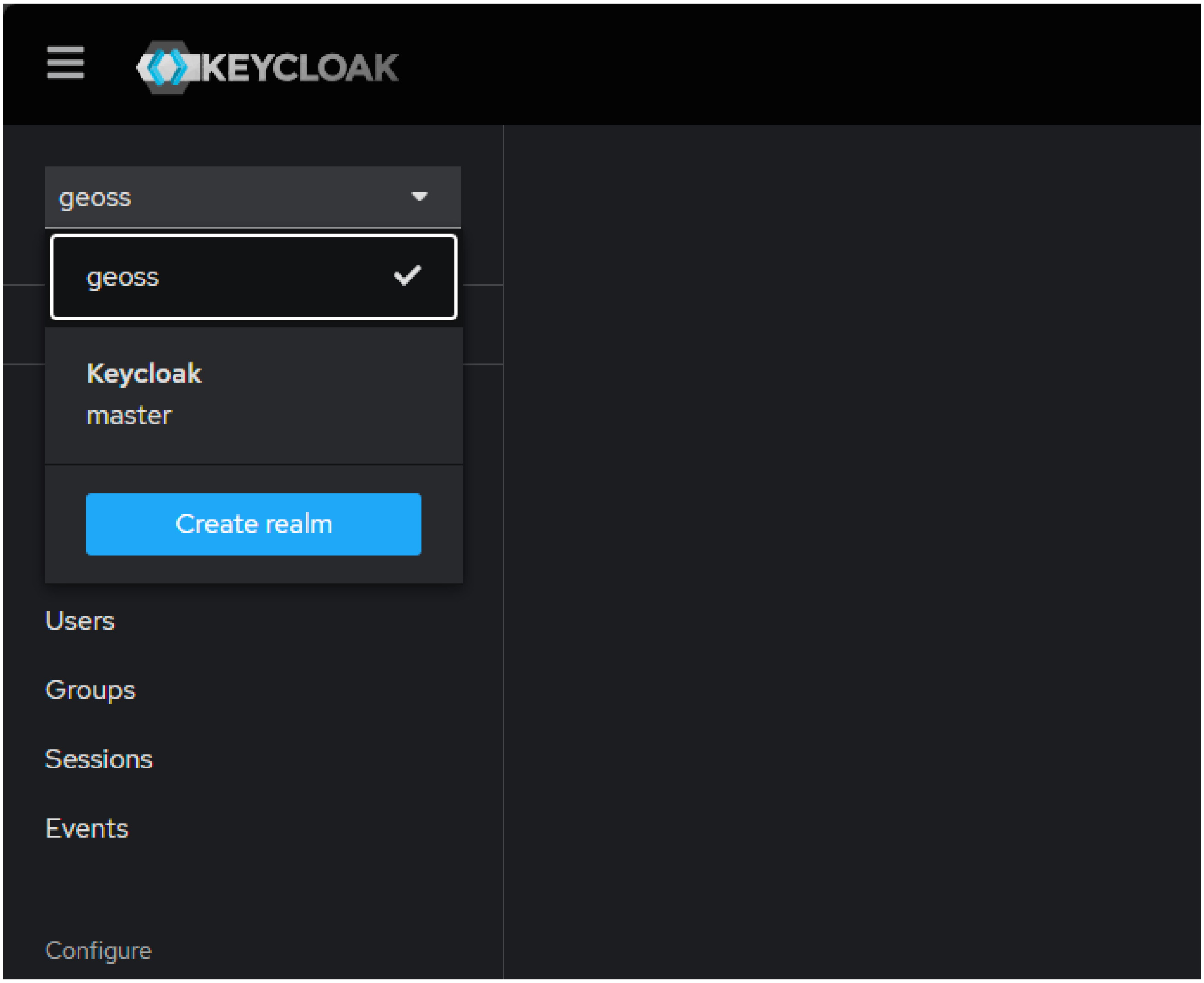


Sign in to your account

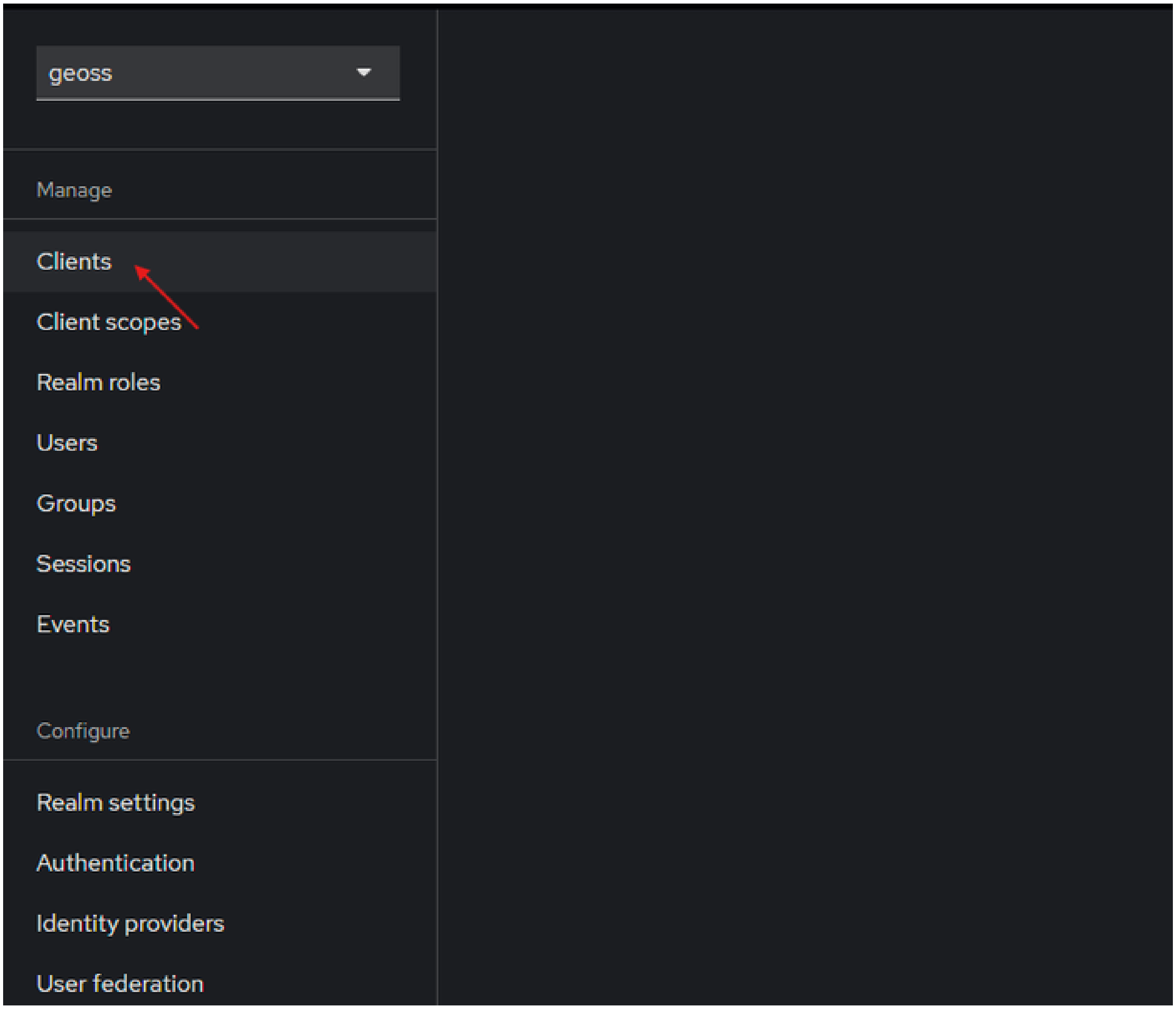
Username or email
admin

Password
••••••••••••••••••••••••

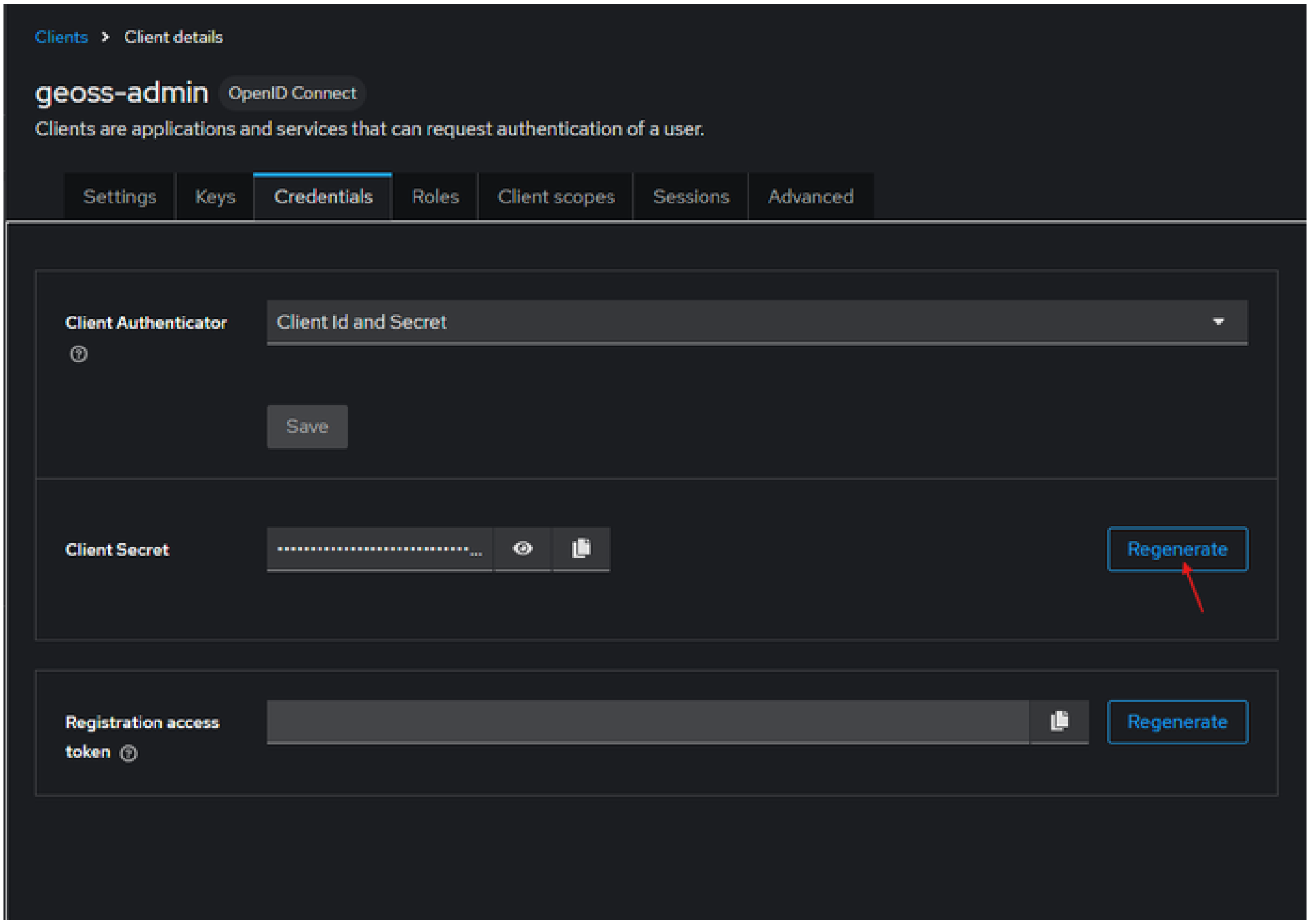
3. In top left corner choose geoss realm.



4. Open `Manage Clients` section



5. Open `geoss-admin` , go to `Credentials` section and click `Regenerate`



Clients > Client details

geoss-admin OpenID Connect

Clients are applications and services that can request authentication of a user.

Client Authenticator: Client Id and Secret

Client Secret: ·································...

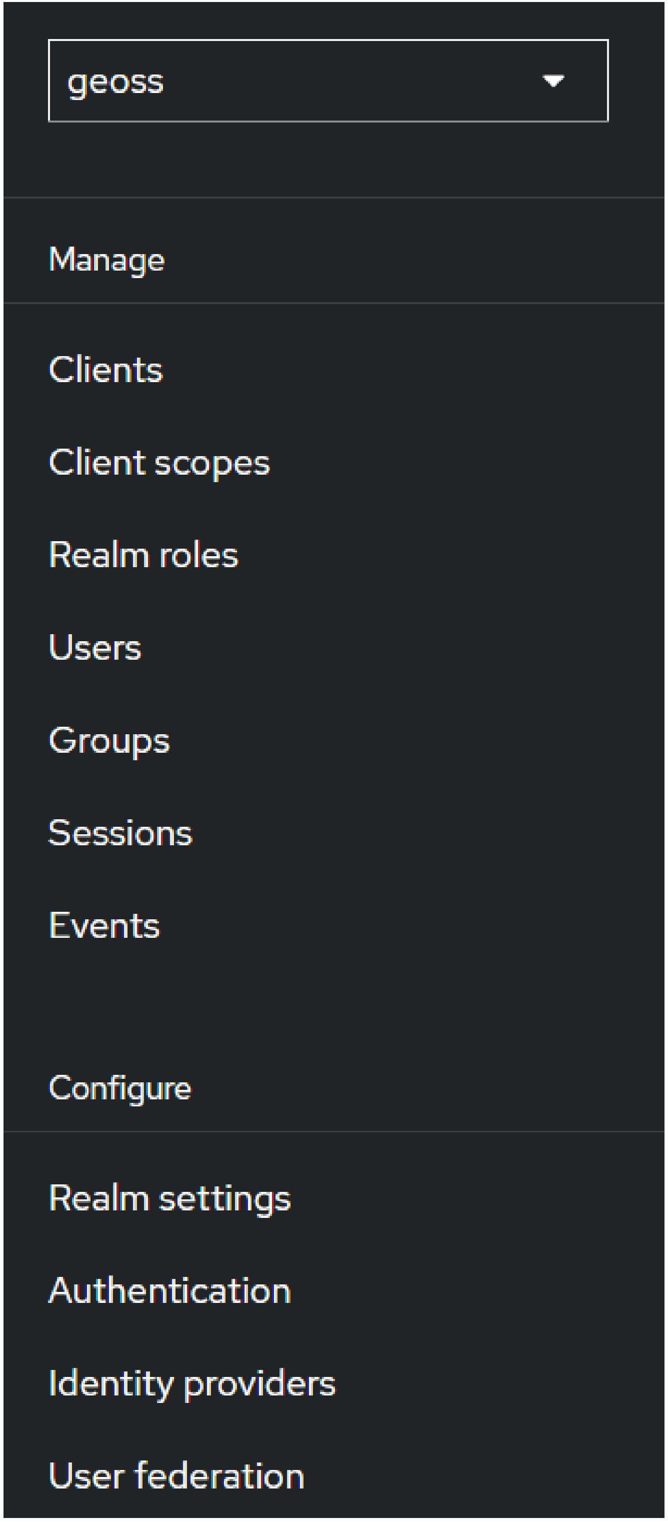
Registration access token

6. Copy the new `Client Secret` and paste it into `KEYCLOAK_CLIENT_SECRET_ADMIN` variable in .env file.
7. Repeat steps 5-6 for all clients starting with `geoss-` and replace .env file variables according to following list:

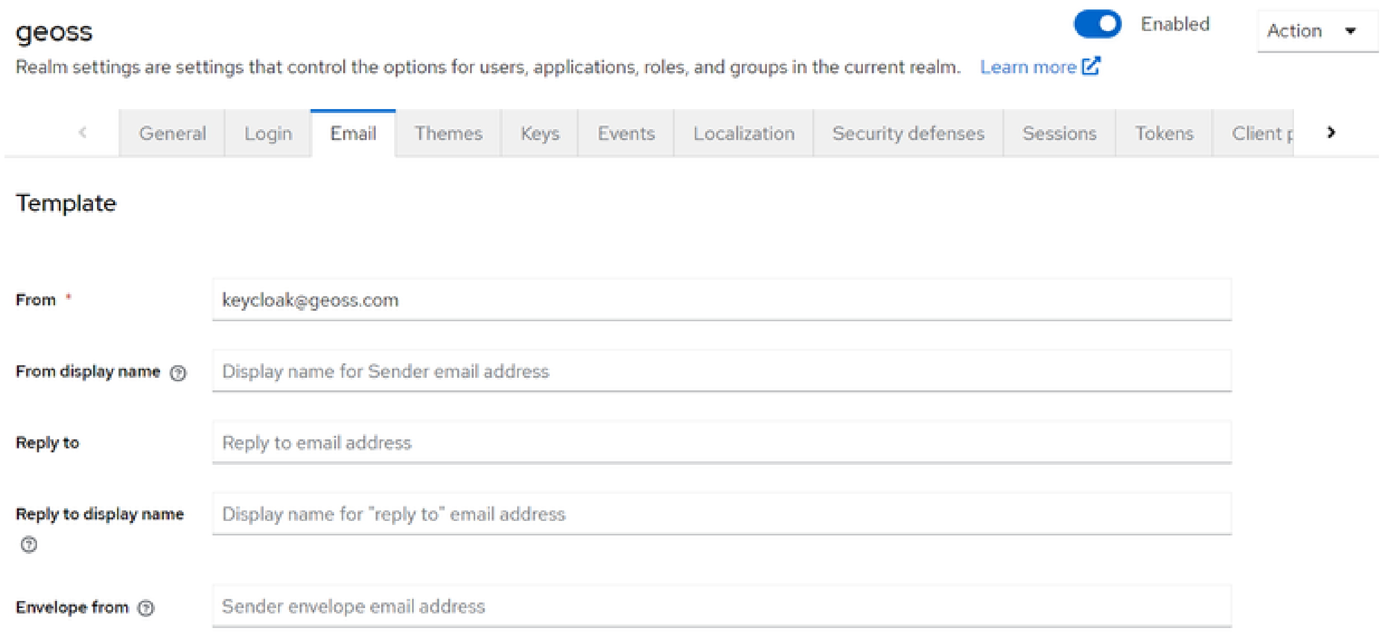
VARIABLE NAME	COMPONENT_NAME	Description
KEYCLOAK_CLIENT_SECRET_ADMIN	geoss-admin	Keycloak client secret for user geoss-admin
KEYCLOAK_CLIENT_SECRET_CURATED	geoss-curated	Keycloak client secret for user geoss-curated
KEYCLOAK_CLIENT_SECRET_MATOMO	geoss-matomo	Keycloak client secret for user geoss-matomo
KEYCLOAK_CLIENT_SECRET_PERSONALDATA	geoss-personaldata	Keycloak client secret for user geoss-personaldata
KEYCLOAK_CLIENT_SECRET_PROXY	geoss-proxy	Keycloak client secret for user geoss-proxy
KEYCLOAK_CLIENT_SECRET_SEARCH	geoss-search	Keycloak client secret for user geoss-search
KEYCLOAK_CLIENT_SECRET_SETTINGS	geoss-settings	Keycloak client secret for user geoss-settings
KEYCLOAK_CLIENT_SECRET_WORKER_GEODAB	geoss-worker-geodab-worker	Keycloak client secret for user geoss-worker-geodab-worker
KEYCLOAK_CLIENT_SECRET_WORKER_SDG	geoss-worker-sdg-worker	Keycloak client secret for user geoss-worker-sdg-worker
KEYCLOAK_CLIENT_SECRET_WORKER_THESAURUS	geoss-worker-thesaurus-worker	Keycloak client secret for user geoss-worker-thesaurus-worker
KEYCLOAK_CLIENT_SECRET_WORKER_WIKIPEDIA	geoss-worker-wikipedia-worker	Keycloak client secret for user geoss-worker-wikipedia-worker

7. Update SMTP configuration in keycloak

In geoss realm open Configure -> Realm Settings



Next go to Email tab



geoss

Realm settings are settings that control the options for users, applications, roles, and groups in the current realm. Learn more

Template

From * keycloak@geoss.com

From display name: Display name for Sender email address

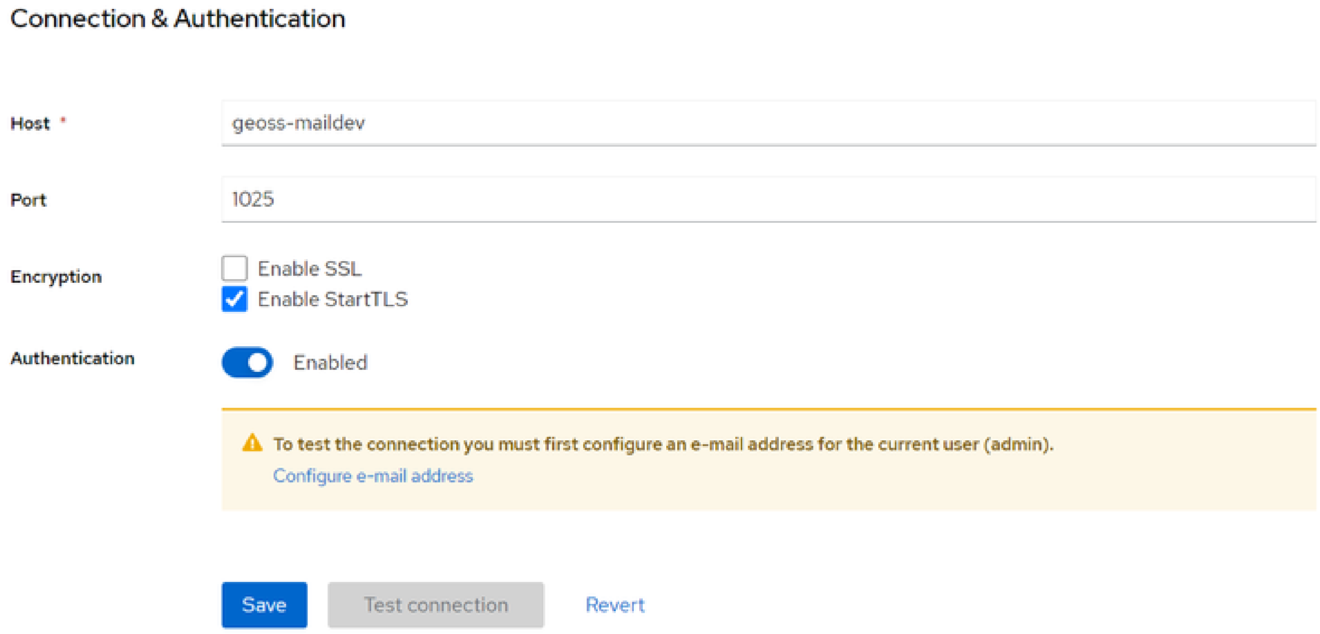
Reply to: Reply to email address

Reply to display name: Display name for "reply to" email address

Envelope from: Sender envelope email address

Scroll down to Update Connection & Authentication

Update Connection & Authentication



Connection & Authentication

Host * geoss-maildev

Port 1025

Encryption □ Enable SSL ☑ Enable StartTLS

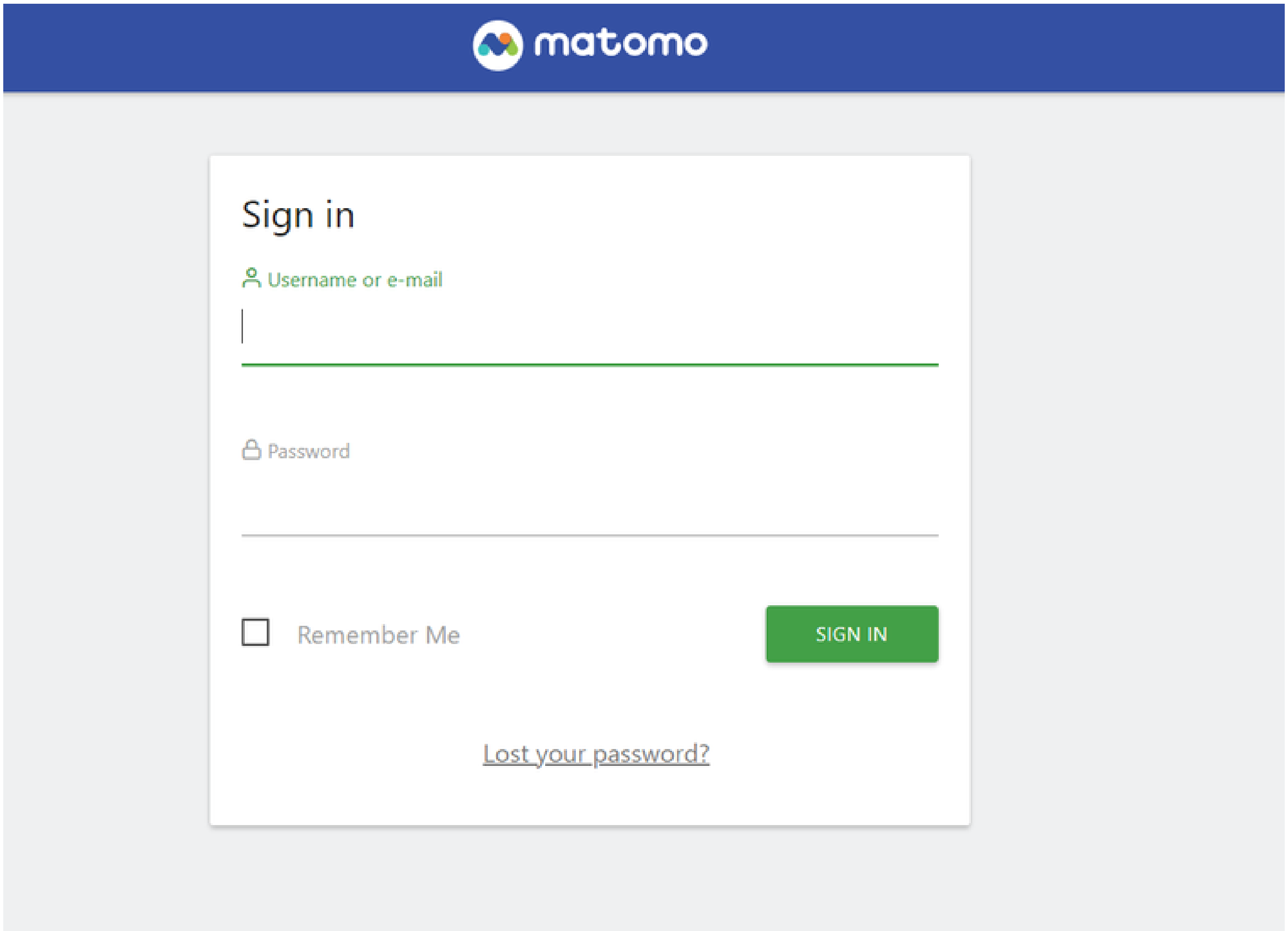
Authentication: Enabled

To test the connection you must first configure an e-mail address for the current user (admin).

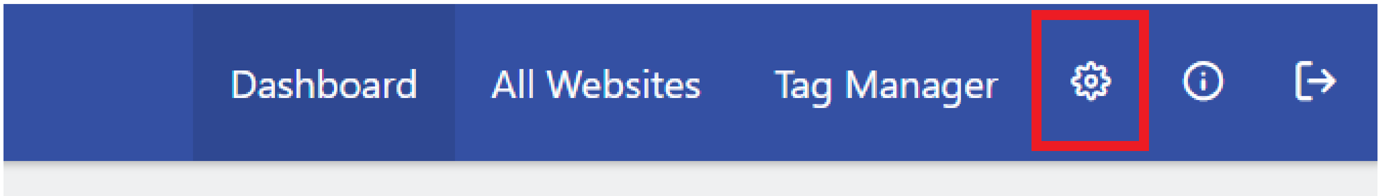
Provide host, port and authentication to your SMTP server. Next press Save button.

8. Create matomo token

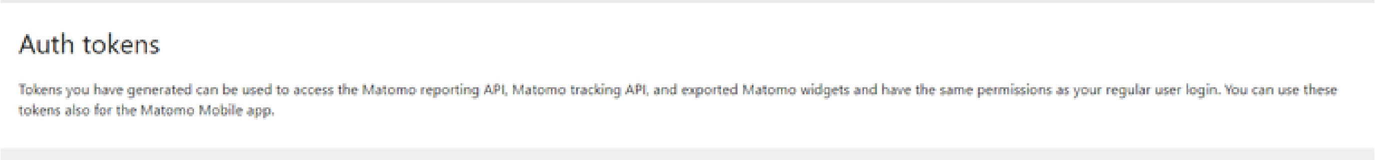
1. Go to the admin domain address, you have set in the variables. (ADMIN_DOMAIN_NAME)
2. Open matomo application on admin domain https://[ADMIN_DOMAIN_NAME]/matomo/
3. Fill the username and password. (MATOMO_USERNAME and MATOMO_PASSWORD)



In top right corner choose Administration.



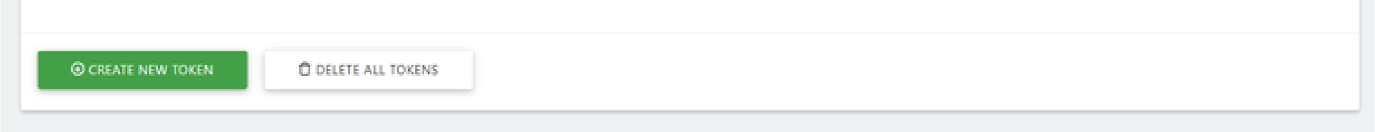
Go to Personal Security and scroll down to Auth tokens



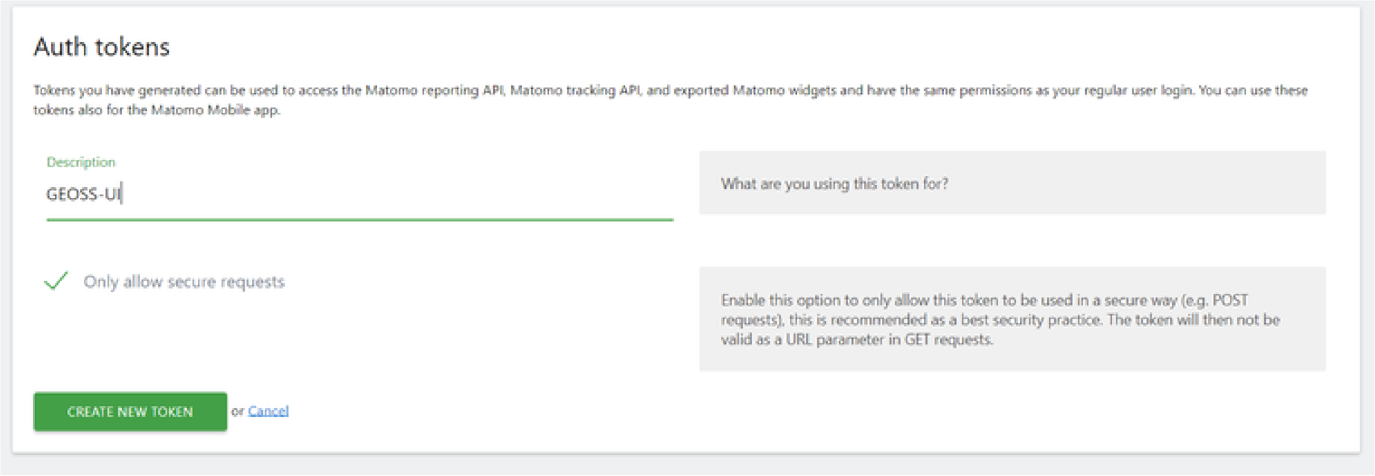
Auth tokens

Tokens you have generated can be used to access the Matomo reporting API, Matomo tracking API, and exported Matomo widgets and have the same permissions as your regular user login. You can use these tokens also for the Matomo Mobile app.

Press button CREATE NEW TOKEN



in the description field enter GEOSS-UI and press button CREATE NEW TOKEN



Auth tokens

Tokens you have generated can be used to access the Matomo reporting API, Matomo tracking API, and exported Matomo widgets and have the same permissions as your regular user login. You can use these tokens also for the Matomo Mobile app.

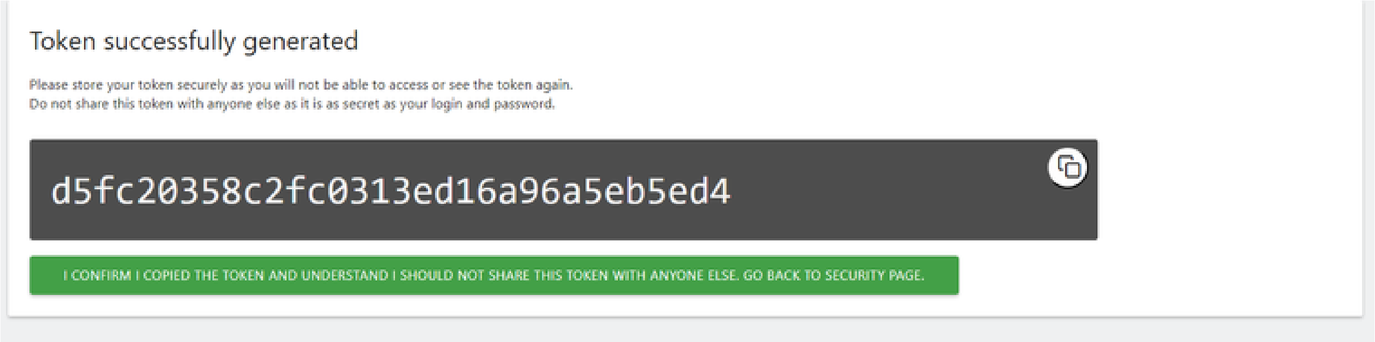
Description: GEOSS-UI

What are you using this token for?

✓ Only allow secure requests

Enable this option to only allow this token to be used in a secure way (e.g. POST requests), this is recommended as a best security practice. The token will then not be valid as a URL parameter in GET requests.

Copy the new `TOKEN` and paste it into `MATOMO_TOKEN` variable in .env file.



Token successfully generated

Please store your token securely as you will not be able to access or see the token again.
Do not share this token with anyone else as it is as secret as your login and password.

```
d5fc20358c2fc0313ed16a96a5eb5ed4
```

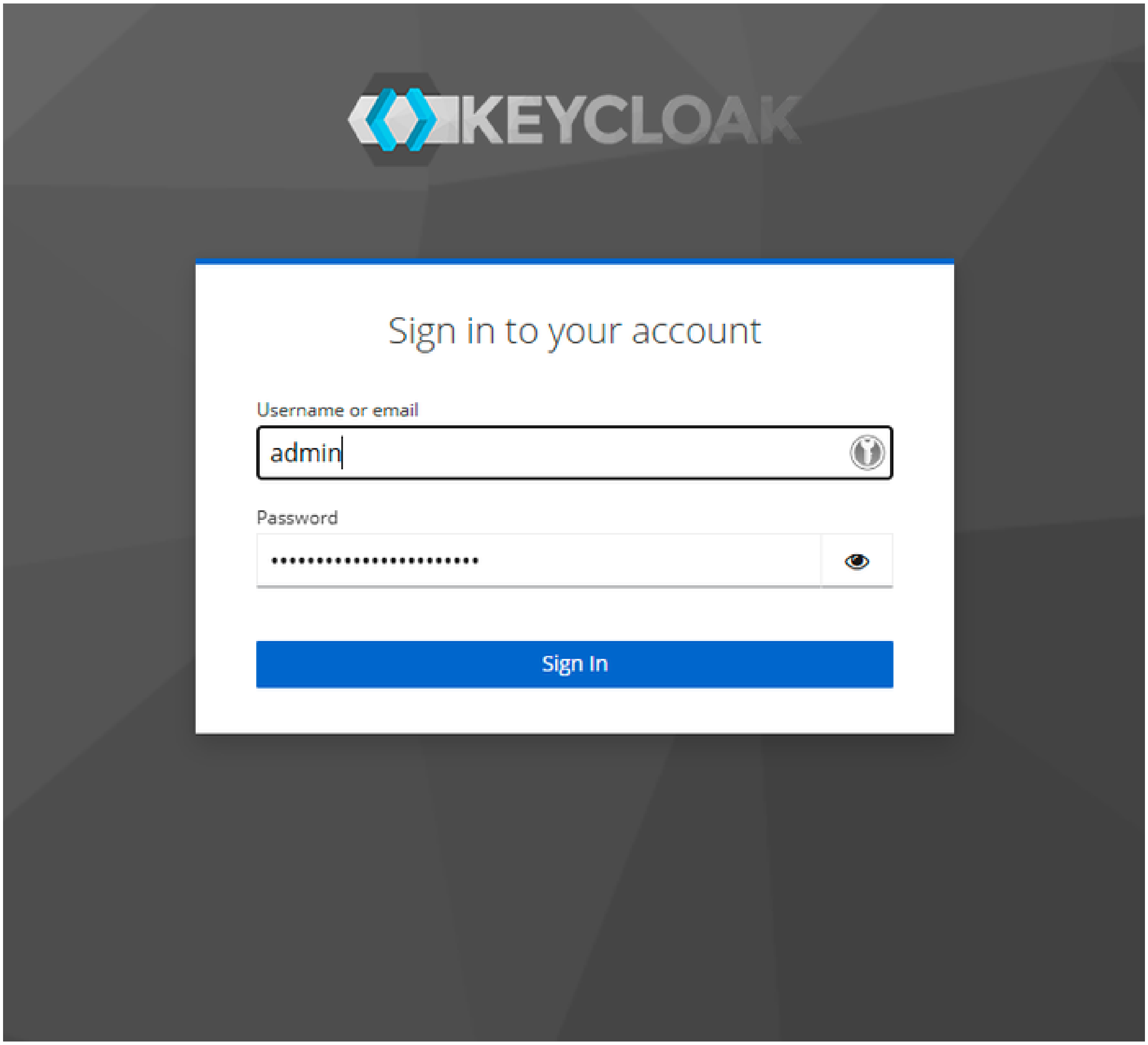
I CONFIRM I COPIED THE TOKEN AND UNDERSTAND I SHOULD NOT SHARE THIS TOKEN WITH ANYONE ELSE. GO BACK TO SECURITY PAGE.

Next press confirm button.

9. Run installation script again to reload Keycloak secrets

10. Create Administration account in keycloak

1. Go to the IDP domain address, you have set in the variables. (IDP_DOMAIN_NAME)
2. Fill the username and password. (KEYCLOAK_ADMIN_USERNAME and KEYCLOAK_ADMIN_PASSWORD)

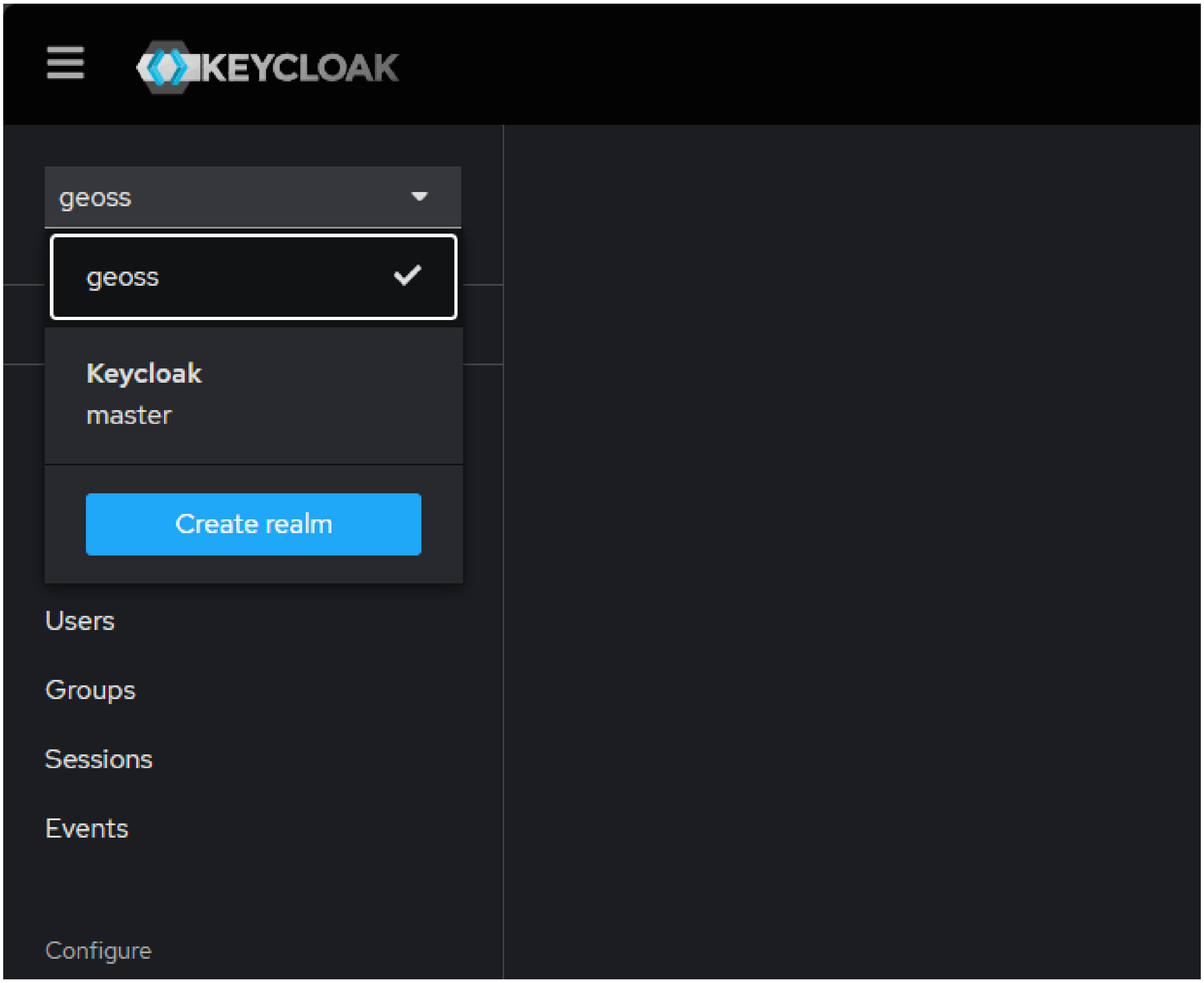


Sign in to your account

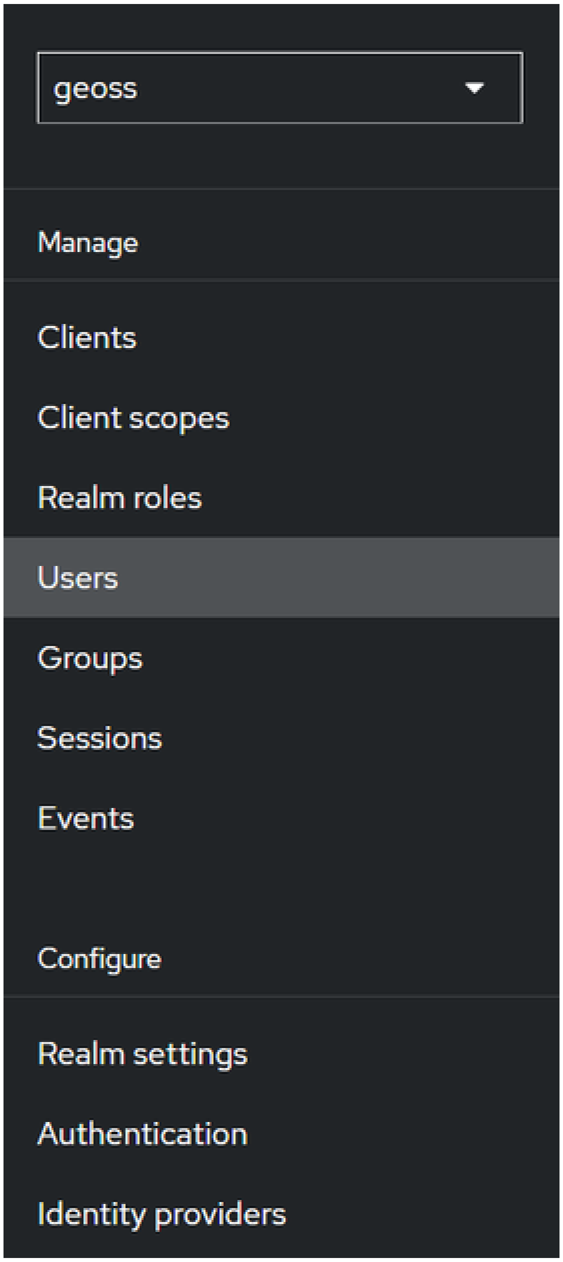
Username or email: admin

Password: ••••••••••••••••••••••••

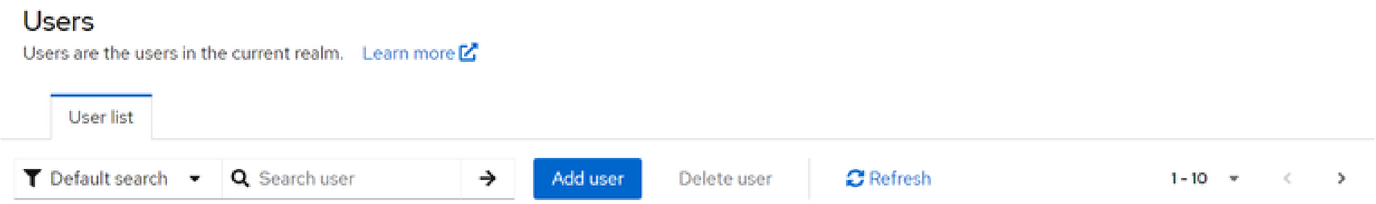
3. In top left corner choose geoss realm.


KEYCLOAK
geoss
geoss ✓
Keycloak
master
Create realm
Users
Groups
Sessions
Events
Configure

4. Go to Users

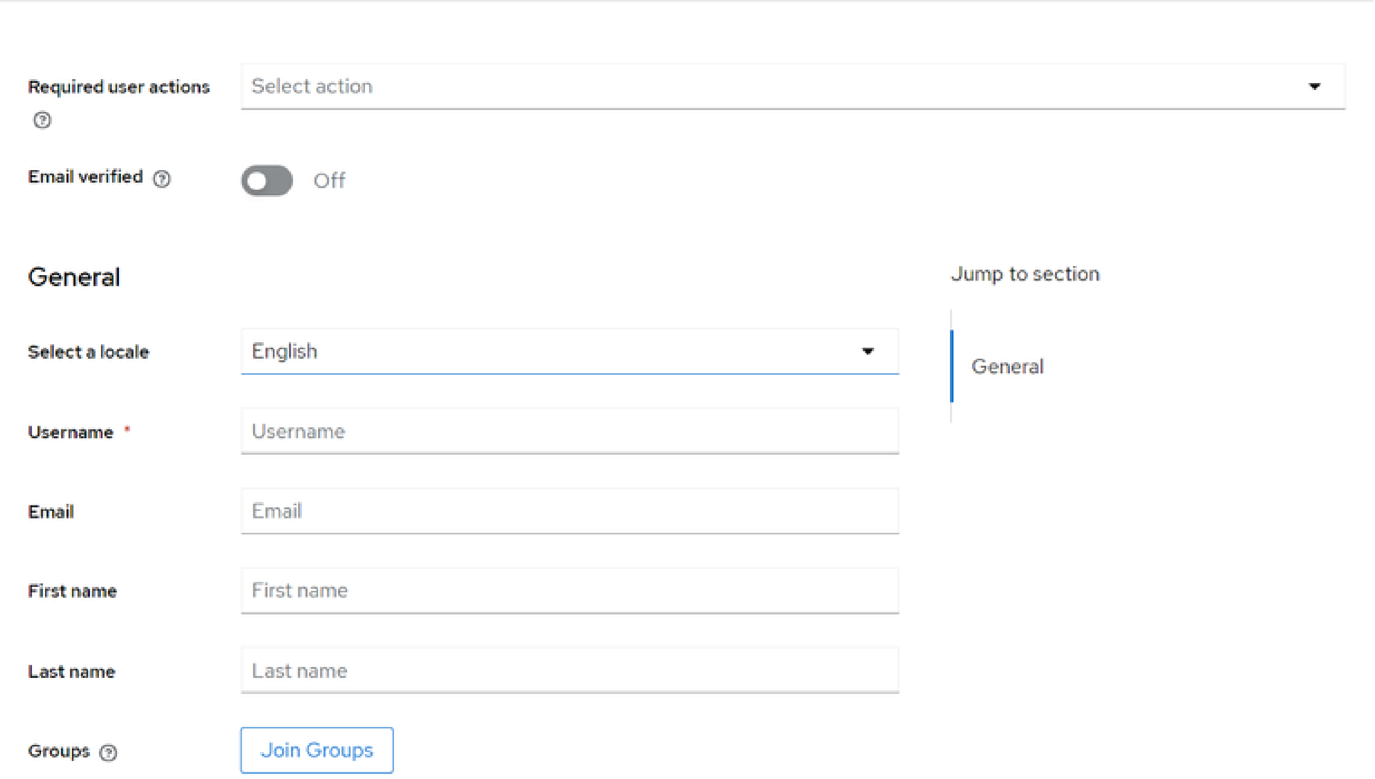


1. Add new user



Users

Users are the users in the current realm. Learn more

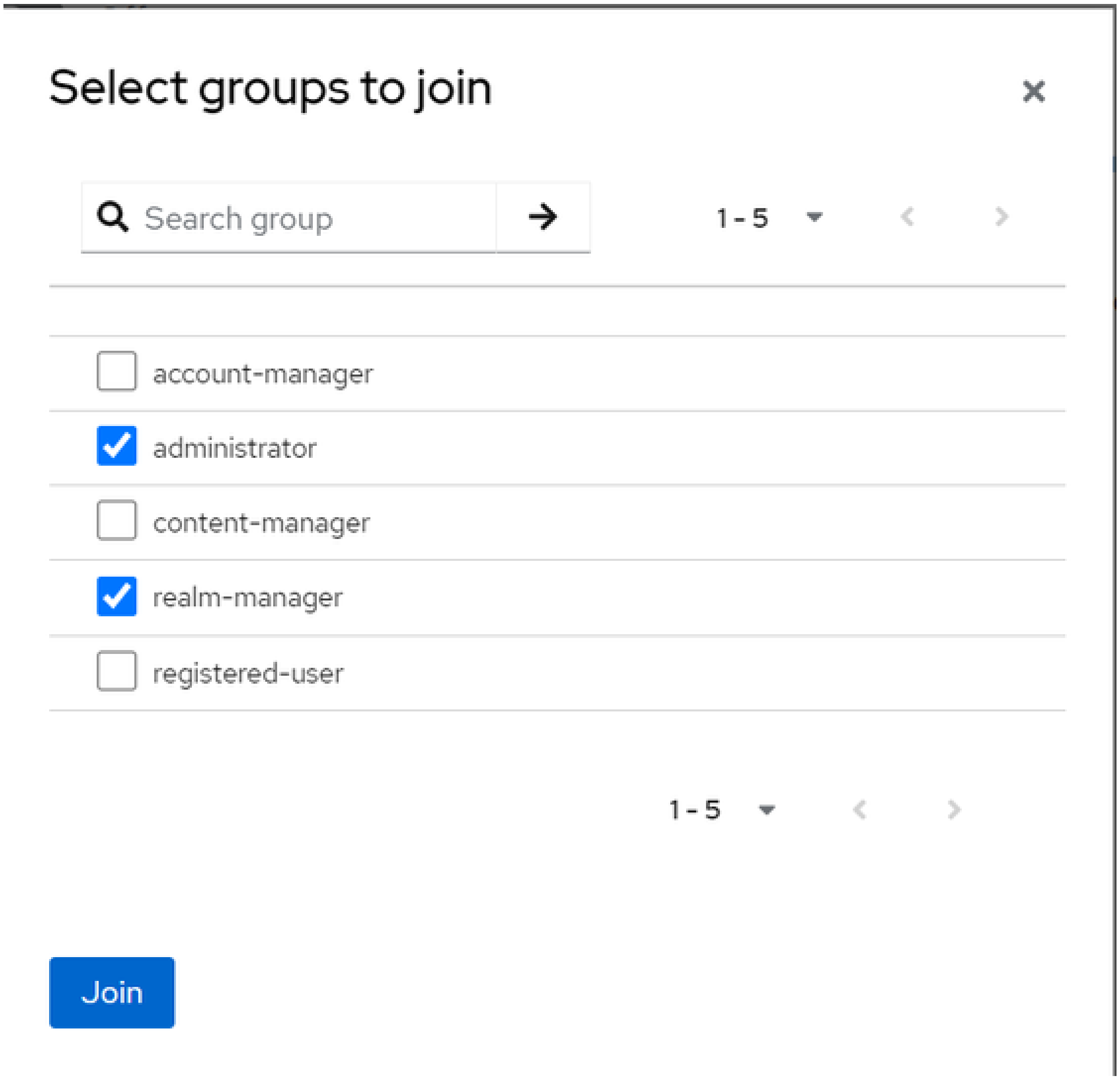


Required user actions	Select action
Email verified	Off

General

Select a locale	English
Username *	Username
Email	Email
First name	First name
Last name	Last name
Groups	Join Groups

1. Join Groups administrator and realm-manager



Select groups to join

☐ account-manager

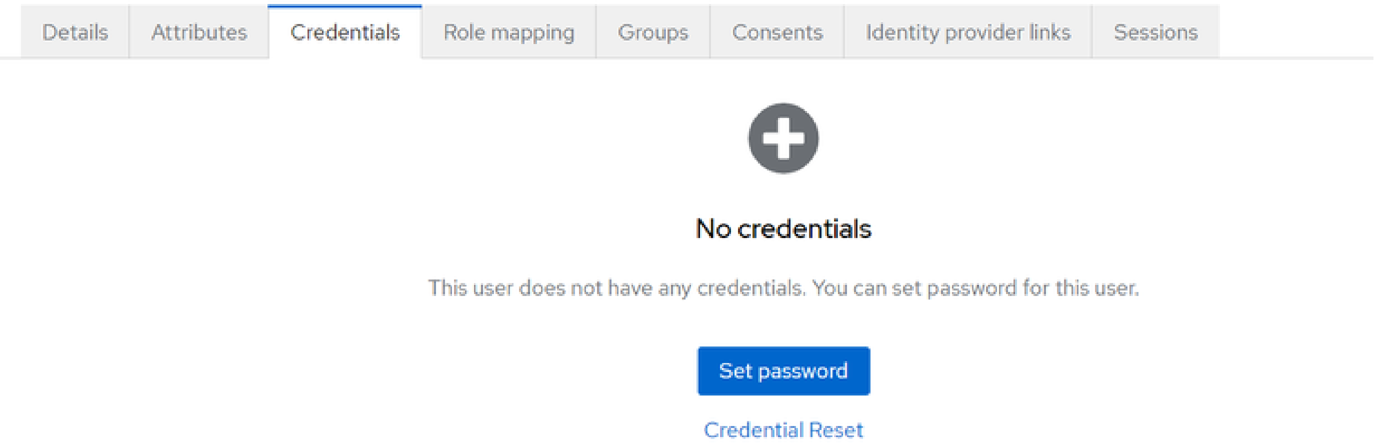
☑ administrator

☐ content-manager

☑ realm-manager

☐ registered-user

1. Set password in credentials tab



No credentials

This user does not have any credentials. You can set password for this user.

Optional components

Matomo and geoss-ui statistics page

To correctly set-up geoss-ui statistics page matomo component must be configured.

1. Set-up environment variables for geoss-matomo service
2. Run installation script to deploy geoss-matomo service
3. Open matomo website (https://<UI_DOMAIN_NAME>/matomo) and log-in into admin account
4. Generate new matomo API token and copy it’s value
5. Connect to applications server
6. Paste matomo token into MATOMO_TOKEN environment variable
7. Run installation script to reload variables

External DAB services providers source configuration

To correctly set it up:

1. Set SERVICES_PROVIDERS environment variable (example: "http://yp.geodab.eu/yp-publisher/services/yp/providers") - **link must be valid otherwise the container won’t start**
2. Run installation script to reload variables

Additional notes

Firewall and monitoring

This application setup does not provide any kind of firewall or application monitoring. Such solutions have to be provided separately basing on the hosting architecture.