Trust Anchor

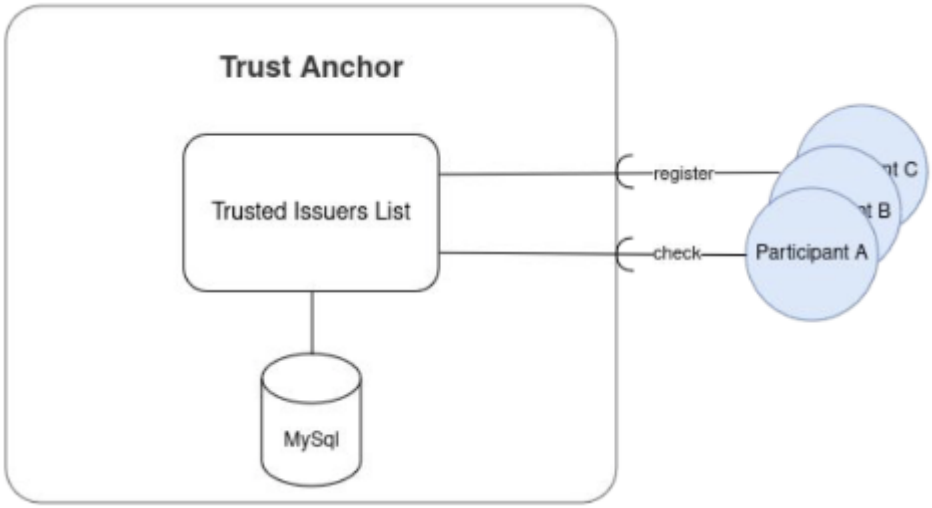
Table of Contents

Introduction

The Trust Anchor includes **Trusted Issuers List (TIL)** to provide 2 APIs:

- EBSI Trusted Issuers Registry (TIR) API
- TIL API for registration

MySql acts as a storage backend for the participants.



Configuration values.yaml

MySql

```
# -- configuration for the mysql to be deployed as part of the trust-
anchor, see https://github.com/bitnami/charts/tree/main/bitnami/mysql
for all options
mysql:
  # -- should it be enabled? set to false if one outside the chart is
used.
  enabled: true

  # -- configure persistence, depending on the storageClasses available
in your cluster
  primary:
    persistence:
      # use one of the classes provided by your cluster
      storageClass: local-path

  ## defaults

  # -- allows to set a fixed name for the services
  fullnameOverride: trust-anchor-mysql
  # -- configure authentication to mysql
  auth:
    # -- name of the secret to take the passowrds from
    existingSecret: mysql-database-secret
  # -- scripts to be executed on db startup
  initdbScripts:
    create.sql: |
      CREATE DATABASE tirdb;
```

Trusted Issuers List

```
# -- configuration for the trusted-issuers-list to be deployed as part
of the trust-anchor, see https://github.com/FIWARE/helm-
charts/tree/main/charts/trusted-issuers-list for all options
trusted-issuers-list:
  # -- should it be enabled? set to false if one outside the chart is
used.
  enabled: true

  # -- configuration to make the list "publicly" accessible through our
ingress controller
  ingress:
    tir:
      enabled: true
      hosts:
        - host: tir.127.0.0.1.nip.io
    til:
      enabled: true
      hosts:
        - host: til.127.0.0.1.nip.io
```

```
  ## defaults

  # -- allows to set a fixed name for the services
  fullnameOverride: tir
  # -- connection to the database
  database:
    # -- should persistence be used?
    persistence: true
    # -- name of the db user
    username: root
    # -- configuration for the existing secret to get the passwords from
    existingSecret:
      enabled: true
      name: mysql-database-secret
      key: mysql-root-password
    # -- host of the database
    host:  trust-anchor-mysql
    # -- name of the schema inside the db
    name: tirdb
```

Use TIL API to register a participant to the Trust Anchor

```
curl -X POST http://til.127.0.0.1.nip.io:8080/issuer \
    --header 'Content-Type: application/json' \
    --data '{
      "did": "did:key:<PARTICIPANT-KEY>",
      "credentials": []
    }'
```

Use TIR API to fetch the global trusted issuers list

```
curl -X GET http://tir.127.0.0.1.nip.io:8080/v4/issuers
```

Deployment of the Trust Anchor

1. Render the template

```
helm repo add data-space-connector https://fiware.github.io/data-space-
connector/

cd trust-anchor

helm template data-space-connector/trust-anchor --version 2.0.0 -f
```

```
values.yaml --name-template=trust-anchor --namespace=trust-anchor --
output-dir rendered
```

2. Create 'trust-anchor' namespace

```
kubectl create namespace trust-anchor
```

3. Install the Trust Anchor

```
helm install trust-anchor-dsc data-space-connector/trust-anchor --
version 2.0.0 -f values.yaml --namespace=trust-anchor

watch kubectl get pods -n trust-anchor
```

4. Apply rendered templates

If helm template was executed, this can be used to apply all files in the folder

```
kubectl apply -R  -f rendered/

watch kubectl get pods -n trust-anchor
```

Uninstall

```
helm uninstall trust-anchor-dsc -n trust-anchor
```

Return to index