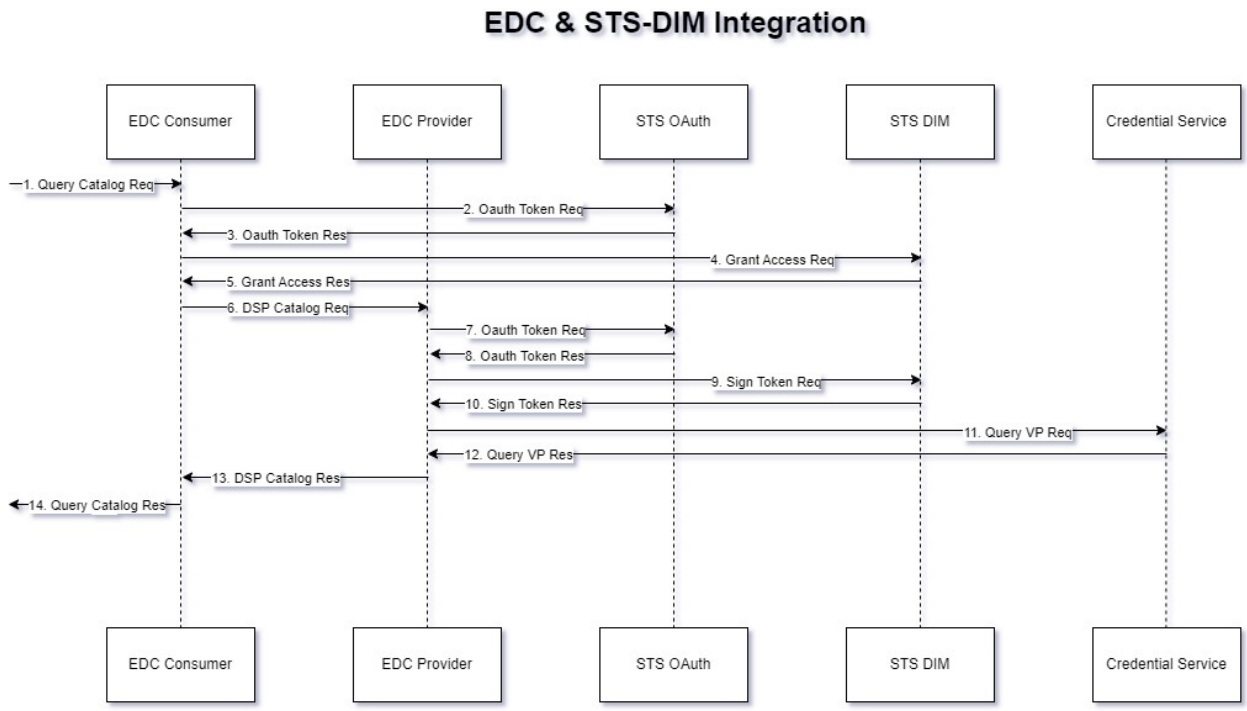
Subject: RE: MIW Open Question
Date: Tuesday, 16. April 2024 at 08:23:02 Central European Summer Time
From: PATEL, PRATAPI HEMANT (external - Project)
To: Boris Rizov
CC: NICUTA, CIPRIAN (external - Project), EUNA, ISLAM (external - Project), Danciu, Alexandru
Attachments: image001.png, image002.jpg

Hi @Boris Rizov,

Thanks for your response.

We further explored options to integrate MIW with EDC. Below are the details.

1. eclipse-tractusx/tractusx-edc provides an implementation of Secure Token Service (STS) which looks like below.



2. Upstream Project eclipse-edc/Connector provides an implementation of Secure Token Service (STS), identity-trust-sts-remote-client which replaces STS Oauth + STS DIM (in above diagram) with a single endpoint. Alex suggested that this is the extension using which we can integrate MIW's '/token' API.
3. We explored latest snapshot release of eclipse-tractusx/tractusx-edc with version **0.7.0-rc1**, which has been shipped with STS DIM implementation. So we rebuilt the EDC with identity-trust-sts-remote-client instead of DIM STS and tried integrating with MIW but it failed. MIW's `/token` API expects a JSON request body, but EDC sends an application/x-www-form-urlencoded request body.
4. eclipse-edc/Connector has an embedded STS-API implementation for testing purposes which can be taken as a reference in terms of request / response structure identity-trust-sts-api. This extension has a `/token` endpoint and it is compatible with the token request EDC sends when identity-trust-sts-remote-client extension is used.

At this point, MIW doesn't look compatible with either STS implementation.

Please let me know your thoughts about this. We can setup a meeting as well to talk about this.

Regards,
Hemant

From: Boris Rizov <B.Rizov@cluetec.de>
Sent: Friday, April 5, 2024 6:24 PM
To: PATEL, PRATAPI HEMANT (external - Project) <pratapi.hemant.patel@sap.com>
Cc: Danciu, Alexandru <alexandru.danciu@sap.com>
Subject: Re: MIW Open Question

Hi,

Do you have any documentation for `/token` API which was implemented in MIW. Was the requirement documented?
All the requirements come directly from the iatp specification repository. There is, in my opinion, no additional information needed on our behalf. Please be more specific, what you mean by documentation, so I can provide a more in-depth answer.

Does MIW still requires Keycloak or it can function without Keycloak?
The MIW doesn't strictly require Keycloak, rather an OAuth | OpenID provider somewhere. This provider issues the tokens which can be used to access the MIW. We have included a Keycloak instance for local development and testing, BUT we do not provide the production environment identity provider. As long as you setup an identity provider, you'll be able to use the MIW normally, as it will use normal OAuth flows to acquire cryptographic material and verify incoming requests and the attached tokens.

Does '/token' API generates an OAuth token?
No, strictly speaking it is not an OAuth token. It was the intention of the iatp specification to make the `access_token`, contained within the `self-issued token`, compatible with the OAuth | OpenID4VC specifications.
The token endpoint can be used in two ways: 1) provide a set of scopes and client authorization to receive an `access_token` wrapped by a self-issued token. The `access_token` is OAuth | OpenID4VC compatible. 2) Provide an `access_token` and get it wrapped in a self-issued token. In both cases you'll use the `access_token` to get authorization with the issuing party.

Hope this helps, best regards
Boris

From: PATEL, PRATAPI HEMANT (external - Project) <pratapi.hemant.patel@sap.com>
Date: Friday, 5. April 2024 at 11:57
To: Boris Rizov <B.Rizov@cluetec.de>
Cc: Danciu, Alexandru <alexandru.danciu@sap.com>
Subject: MIW Open Question

Hi Boris,

We are currently investigation IATP and STS implementation in EDC.
We found all existing MIW configs were replaced via IATP configs in TractusX-EDC helm charts.

```
            ############################
            ## IATP / STS / DIM CONFIG ##
            ############################
            - name: "EDC_IAM_STS_OAUTH_TOKEN_URL"
              value: {{ .Values.iatp.sts.oauth.token_url | required ".Values.iatp.oauth.token_url is required" |
quote}}
            - name: "EDC_IAM_STS_OAUTH_CLIENT_ID"
              value: {{ .Values.iatp.sts.oauth.client.id | required ".Values.iatp.sts.oauth.client.id is required"
| quote}}
            - name: "EDC_IAM_STS_OAUTH_CLIENT_SECRET_ALIAS"
              value: {{ .Values.iatp.sts.oauth.client.secret_alias | required
".Values.iatp.sts.oauth.client.secret_alias is required" | quote}}
            - name: "EDC_IAM_STS_DIM_URL"
              value: {{ .Values.iatp.sts.dim.url | required ".Values.iatp.sts.dim.url is required" | quote}}
```

We have some questions which we are trying to understand.

- Do you have any documentation for `/token` API which was implemented in MIW. Was the requirement documented.
- Does MIW still requires KeyCloak or it can function without KeyCloak.
- Does '/token' API generates an OAuth token? As per new IATP configuration, EDC generates a new token at the url defined in **EDC_IAM_STS_OAUTH_TOKEN_URL** with given client id and secret, and using the OAuth token it sends request to url defined **in EDC_IAM_STS_DIM_URL**.
  We are trying to explore what services / endpoints these two URLs point to.

Regards,
Hemant