Edukoppeling

MDX Secure API OAuth profiel
voor
M2M gegevensuitwisseling binnen het onderwijs

Edustandaard

Datum: januari 2023

Versie: 0.1

Status: concept

edustandaard

Inhoudsopgave

1. Historie

Versie	Auteur	Datum	Opmerking
0.1	Werkgroep Edukoppeling	januari 2023	Initiële versie gebaseerd op uitgangspunten in het discussiestuk (versie 0.3).

edustandaard

2. Inleiding

2.1. Aanleiding

De aanleiding voor de introductie van Edukoppeling in het onderwijsdomein is een steeds groter wordende stroom van geautomatiseerde (machine-machine) processen in het onderwijs. Dit wordt veroorzaakt door vernieuwingen in het onderwijs zelf, in wetgeving, in de beschikbare techniek en de wens om het aantal (technische) koppelvlakafspraken binnen de perken te houden. In toenemende mate lopen de processen over organisaties heen, tussen onderwijsorganisaties (zowel op bestuursniveau van de onderwijsaanbieders, de “scholen”) onderling, tussen onderwijsorganisaties en overheidsorganisaties en tussen onderwijsorganisaties en bedrijven. En vaak, als er iets nieuws komt, wordt er dan pas nagedacht over de benodigde infrastructuur. Als men niet oppast worden er evenveel infrastructurele oplossingen gerealiseerd als er geautomatiseerde processen zijn. Met Edukoppeling verandert dat. Edukoppeling is een meervoudig inzetbare infrastructuur waarvan de ontwikkeling en het beheer gemeenschappelijk wordt aangepakt.

In het onderwijs is het normaal geworden dat onderwijsinstellingen veel van hun processen laten ondersteunen door zogeheten SaaS-diensten (diensten ‘in the cloud’). Dit geldt voor onderwijskundige processen als ook voor hun administratieve processen. Het Edukoppeling Mandated Data eXchange (MDX) protocol en verwante profielen houden met deze ontwikkeling rekening. De diensten van leveranciers waar een onderwijsorganisatie gebruik van maakt beheren gegevens (administraties) en wisselen vaak namens de onderwijsorganisatie gegevens uit met ketenpartijen. De Edukoppeling Mandated Data eXchange houden expliciet rekening met het uitwisselen van gegevens tussen verwerkers namens een eindorganisatie.

Dit document beschrijft de Edukoppeling MDX Secure API OAuth profiel (verder aangeduid als OAuth-profiel) en is onderdeel van de Edukoppeling Architectuur. Het beschrijft op welke punten het afwijkt van het Edukoppeling MDX Secure API REST profiel[1]. Het kan als een extensie op het MDX Secure API REST profiel gezien worden. Bij het MDX Secure API OAuth-profiel (deze specificatie) wordt een autorisatietoken bij de uitwisseling toegepast. Het is gebaseerd op het OAuth client credentials profiel. Het bouwt dus voort op het MDX Secure API REST-profiel en de betreffende eisen zijn dus ook van toepassing op dit OAuth-profiel, zoals de toepassing van het Mandated Data eXchange (MDX) protocol.

2.2. Doel en doelgroep

Het doel dat met dit profiel nagestreefd wordt is het op een generieke manier kunnen uitwisselen van gegevens binnen de onderwijssector. Het model ondersteunt het scenario waarbij een Eindorganisatie zijn systeem zelf beheert in de eigen ICT-infrastructuur, als het scenario waarbij de Eindorganisatie deze als (SaaS-)dienst van een verwerker (leverancier) afneemt.

Dit document is bedoeld voor ICT-specialisten die betrokken zijn bij het ontwerpen en ontwikkelen van systeem-naar-systeem (M2M) koppelingen. Het gaat hier om werknemers (ontwikkelaars, architecten, projectmanagers, informatiemanagers etc.) werkzaam bij onderwijsgerelateerde organisaties, zowel in de publieke als private sector. Edukoppeling is voor een groot deel compliant aan de overheidsstandaard Digikoppeling. De Edukoppeling-documentatie dient derhalve naast de Digikoppeling-documentatie gebruikt te worden.

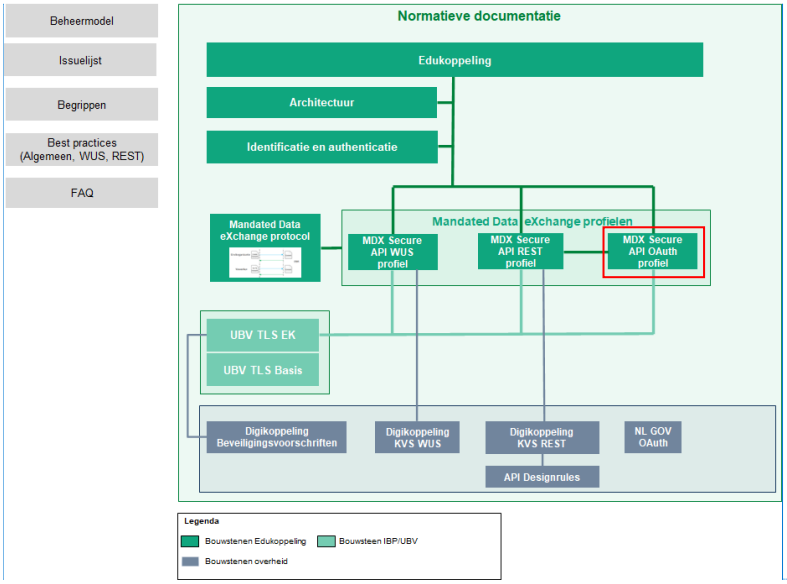
De lezer van dit document willen wij vragen om zaken die ontbreken of onduidelijk zijn te melden bij de beheerorganisatie Edustandaard[2].

[1] Het Edukoppeling REST profiel is gebaseerd op het Digikoppeling REST-profiel in beheer bij Logius https://www.logius.nl/diensten/digikoppeling/documentatie

[2] https://www.edustandaard.nl/standaarden/afspraken/afspraak/edukoppeling/. Reageren kan via info@edustandaard.nl.

2.3. Positionering binnen Edukoppeling Architectuur

Het Edukoppeling OAuth-profiel is onderdeel van de Edukoppeling Architectuur. Dit OAuth-profiel is gebaseerd op het Edukoppeling MDX Secure API REST-profiel. Het beschrijft met name op welke punten er van het MDX Secure API REST-profiel afweken wordt. In het volgende hoofdstuk wordt dit inhoudelijk beschreven.



Figuur 1 - Positionering van OAuth-profiel binnen de Edukoppeling Architectuur

Met opmerkingen [ER1]: Deze versie sluit volledig aan op het RESt profiel (oa.a querystring voor routeringskenmerk en API design rules voor API (protected resources)

2.4. Functioneel toepassingsgebied

Het functionele toepassingsgebied van het OAuth-profiel betreft M2M-gegevensuitwisseling via een point-to-point verbinding voor uitwisseling van vertrouwelijke gegevens via een gesloten API[3]. Er worden bevragingen (pull) en meldingen (push) op basis van een request-response uitwisselingspatroon ondersteund. De client is in deze context geen browser, maar een systeem (applicatie). De systemen worden beheerd door verwerkers en voeren de uitwisseling uit op basis van een mandaat van een eindorganisatie. Identificatie en authenticatie wordt in de point-to-point verbinding gerealiseerd met behulp van mTLS.

De gegevens kunnen op basis van de afspraken binnen dit profiel gerouteerd worden van verwerker naar eindorganisatie. Het profiel kan ook worden toegepast indien de eindorganisatie ook zelf de rol van verwerker heeft. Als er sprake is van een transparante intermediair, of er is noodzaak voor onweerlegbaarheid dan wordt het Edukoppeling MDX Secure API WUS-profiel toegepast. Bij gegevensuitwisseling op basis van XML heeft het de voorkeur om het MDX Secure API WUS profiel toe te passen.

[3] Het Kennisplatform heeft o.a. een API Strategie ontwikkeld waarin verschillende soorten API's worden onderkend (https://geonovum.github.io/KP-APIs/API-strategie-algemeen/). Open API's: voor ontsluiten van diensten zonder toegangsbeperking bijv. open data. Gesloten API's: voor ontsluiten van diensten met toegangsbeperking bijv. persoonsgegevens en vertrouwelijke gegevens of diensten voor specifieke partijen (access-restricted and purpose-limited API's)

2.5. Notatiewijze voorschriften

Voor elk voorschrift wordt aangegeven in welke mate hier invulling aan moet worden gegeven. Hiermee kunnen we duidelijk aangeven wat de grenzen van dit profiel zijn ten opzichte van de mogelijke externe bron(nen) waar het voorschrift eventueel van wordt overgenomen. We gebruiken hiervoor de notatiewijze van RFC2119[4]. Deze gebruikt de volgende termen: "MUST" (“MOET”), "MUST NOT"(“MOET NIET”), "REQUIRED" (“VEREIST”), "SHALL" (“ZAL”), "SHALL NOT" (“ZAL NIET”), "SHOULD" (“ZOU”), "SHOULD NOT" (“ZOU NIET”), "RECOMMENDED" (“AANBEVOLEN”), "NOT RECOMMENDED" (“NIET AANBEVOLEN”), "MAY" (“MAG”), and "OPTIONAL" (“OPTIONEEL”).

[4] https://tools.ietf.org/html/rfc2119

3. Edukoppeling OAuth-profiel

Het Edukoppeling OAuth-profiel sluit aan op de generieke voorschriften van het MDX Secure API REST profiel (zie Figuur 1). Het MDX Secure API REST profiel is op zijn beurt weer afgeleid van het Digikoppeling REST API profiel[5]. Voor dit OAuth-profiel dient men dus ook kennis te nemen van het Edukoppeling MDX Secure API REST profiel en het Digikoppeling REST API profiel.

3.1. Generieke voorschriften vanuit MDX Secure API REST profiel

1. Het OAuth-profiel houdt expliciet rekening met gebruik van een openbaar netwerk (Internet).
2. Het OAuth-profiel vereist transportbeveiliging conform het UBV TLS Edukoppeling profiel.
3. Het OAuth-profiel stelt eisen aan identificatie van organisaties.
4. Het OAuth-profiel kan worden toegepast voor zowel bevragingen als meldingen
5. Het OAuth-profiel wordt gebruikt in combinatie met het Onderwijs serviceregister[6] (autorisatie deelname ketensamenwerking op basis van een mandatering).
6. Het OAuth-profiel sluit aan op een aantal eisen aan de foutafhandeling.

Zie voor verdere details rond deze generieke voorschriften het MDX Secure API REST profiel.

3.2. Specifieke voorschriften vanuit MDX Secure API REST profiel

Voor het OAuth-profiel worden alle specifieke REST voorschriften overgenomen:

1. geen eisen rond berichtbeveiliging;
2. aansluiten op overheidsbrede afspraken.

Zie voor verdere details rond deze specifieke voorschriften het MDX Secure API REST profiel.

3.3. Specifieke voorschriften OAuth-profiel

3.3.1. Context

Edukoppeling standaardiseert de uitwisseling van vertrouwelijke gegevens binnen het onderwijs op basis van een aantal MDX Secure API-profielen. Het gaat hierbij om scenario's waarbij een verwerker dit doet namens een eindorganisatie als onderdeel van een ketensamenwerking. De MDX Secure API-profielen bieden de volgende gestandaardiseerde functies:

1. autorisatie op basis van een mandaat voor een ketensamenwerking (zie MDX Secure API protocol);
2. routeren op basis van een routeringskenmerk (zie MDX Secure API REST en WUS-profiel);
3. identificatie, authenticatie en vertrouwelijkheid op basis van mTLS/PKIo en OIN (zie UBV TLS Edukoppeling profiel);
4. identificatie, authenticatie, integriteit, vertrouwelijkheid en onweerlegbaarheid op gegevensniveau op basis van ondertekening en versleuteling (WUS be-S en WUS be-SE).

[5] Het Digikoppeling REST API Profiel maakt gebruik van de Kennisplatform REST-API Design Rules (zie https://publicatie.centrumvoorstandaarden.nl/api/adr/)

[6] https://www.kennisnet.nl/onderwijs-service-register/

Het OAuth-profiel voegt aan deze lijst een extra autorisatie in de applicatielaag toe:

5. autorisatie voor een confidential client applicatie o.b.v. een Access Token (AT).

De autorisatie voor een client-applicatie wordt gerealiseerd op basis van het OAuth client credentials grant type. Hierbij krijgt een client toegang tot een API (protected resource) als deze over een geldig Access Token (AT) beschikt. Het profiel stelt dat de client credentials in principe voldoende zijn voor toegang tot de API. Het gaat ervan uit dat er geen sprake is van een resource owner. Dit Edukoppeling OAuth-profiel gaat er echter vanuit dat er impliciet wel sprake is van een resource owner die toestemming moet hebben gegeven voor de uitwisseling. Toestemming moet op twee lagen worden gegeven. Ten eerste op het niveau van een ketensamenwerking welke geregistreerd wordt in het OSR. De tweede is op het niveau van de API en een specifieke dataset die deze kan ontsluiten welke wordt geregistreerd in een OAuth authorization server. Dit Oauth-profiel richt zich met name op het tweede niveau. Het eerste niveau wordt gerealiseerd door de specificaties in het Edukoppeling MDX Secure API REST-profiel.

Een AT wordt door een authorization server uitgegeven die de API (protected resource) beveiligt. De resource server zorgt ervoor dat de client alleen met een geldige AT toegang heeft tot de API. Verschillende partijen zullen API's ontsluiten dus er zullen meerdere lokale authorization servers en resource servers zijn. Verder nemen we aan dat één authorization server meerdere API's beveiligt. Het AT is een bearer token en geeft een client toegang tot de gespecificeerde API.

Met opmerkingen [ER2]: Helaas biedt RFC8705 geen ondersteuning voor subject.serial van het certificaat en is een "proof-of-possession" token (nog) niet mogelijk.

Met opmerkingen [ER3]: In het token request is de resources parameter verplicht. Deze moet gelijk zijn aan de "aud" claim in het AT

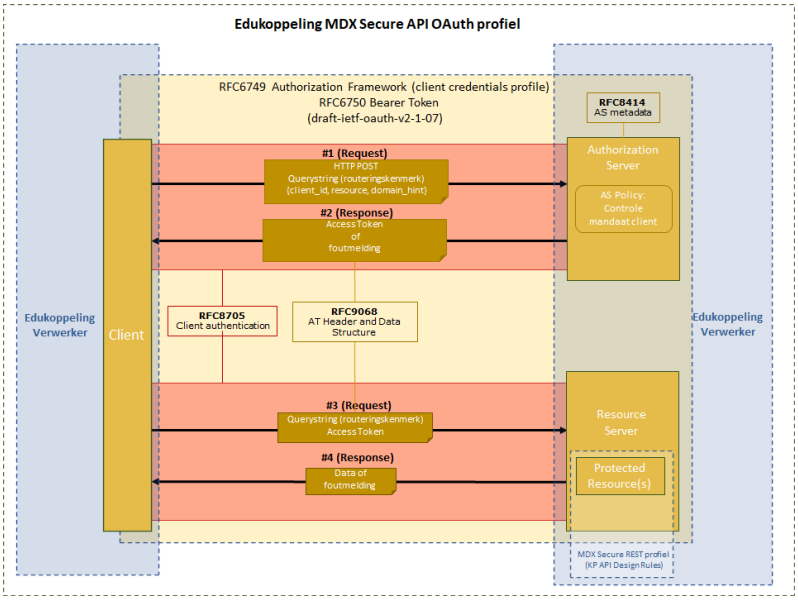
We gaan er verder vanuit dat één partij de resource server beheert en de betreffende authorization server die deze beveiligt, i.e. de verwerker heeft de OAuth authorization server en resource server rol. Deze verwerker kan geïdentificeerd worden op basis van een OIN. De client is de andere verwerker die via mTLS en OIN geauthentiseerd en geïdentificeerd wordt.

In Figuur 2 worden de twee interacties die we binnen het OAuth-profiel onderkennen schematisch weergegeven. Hierbij geldt het volgende:

heeft verwijderd: Figuur 2

heeft opmaak toegepast: Nederlands (standaard)

1. de interacties zijn gebaseerd op RFC6749 client credentials grant type;
2. de protected resource is ontwikkelt (API design) conform het REST profiel;
3. het client request #1 bevat een querystring met het routeringskenmerk conform het REST-profiel en een bearer token conform dit OAuth-profiel;
4. de authorization server valideert het request (#1) en geeft respons #2;
5. de respons #2 bevat een access token conform dit OAuth-profiel bij een valide request;
6. de respons #2 bevat een foutmelding indien request #1 niet succesvol gevalideerd kan worden;
7. het client request #3 bevat een querystring met het routeringskenmerk conform het REST-profiel en het access token uit respons #2;
8. de respons #4 van de resource server bevat de data (protected resource) of een foutmelding.



Figuur 2 - Edukoppeling MDX Secure API OAuth profiel (RFC's)

Een mandaat betreft de autorisatie voor deelname aan een bepaalde ketensamenwerking. Beheerders van API's (bronhouders) zullen in de authorization server policy mogelijk aanvullende randvoorwaarden opnemen voordat aan een bepaalde client een AT uitgegeven kan worden. Zij maken hierin eigen keuzes. De aanvullende randvoorwaarden in de authorization server policy betreft bijvoorbeeld de eis dat de bronhouder wil kunnen vaststellen dat de "resource owner" een client toegang heeft gegeven tot een specifieke dataset die door een API ontsloten wordt. Hiervoor moet er een meer fijnmazige autorisatie worden geregistreerd. Zaken die hierbij een rol spelen zijn bijvoorbeeld:

1. definitie van de dataset die de API ontsluit;
2. registratie van autorisaties voor een bepaalde client door een "resource owner";
3. het kunnen identificeren van de client applicatie.

De dataset wordt bepaald door de set attributen (zoals voornaam en achternaam) en een lijst betrokkenen die wordt bepaald door een bepaalde organisatorische eenheid. Deze entiteit wordt geïdentificeerd met een door de beheerder bepaalde identifier. Deze entiteit heeft waarschijnlijk een relatie met opleidingseenheden zoals die door RIO gedefinieerd worden en een Edukoppeling eindorganisatie. Of dit past of niet en waarom niet is onderdeel van de verdere ontwikkeling van dit profiel. In deze versie wordt hier een aparte parameter (domain_hint) voor onderkend om een tot een eenduidige inrichting van het OAuth profiel te komen. Zo kan bij request (#1) naar het token endpoint de AS bepalen tot welke dataset de client via de API toegang heeft.

Binnen het OAuth profiel wordt de client bij de AS geregistreerd en daarbij krijgt de client een eigen identiteit die intern naar een OIN herleid kan worden. Deze meer fijnmazige identificatie is nodig omdat het OIN (incl. suffix[7]) niet voldoende opties biedt om de mogelijk vele clients (applicaties) te identificeren. Het maakt het profiel ook meer generiek en kan ook gebruikt worden buiten de MDX context.

[7] Een alternatief zou de suffix van het OIN kunnen zijn, maar naast het beperkte bereik maakt de ontkoppeling van het OIN (en PKIo) het gebruik van het profiel ook bruikbaar in andere contexten (anders dan MDX).

De “resource owner” in deze context betreft een beheerder bij een onderwijsinstelling die verantwoordelijk is voor het delen van gegevens van betrokkene die die de API ontsluit. Deze is dus bevoegd om namens de onderwijsinstelling derde partijen toegang te geven tot deze datasets. Hoewel we hier dus een resource owner kunnen onderkennen gebruiken we toch een OAuth client credentials profiel omdat het niet om een real time proces gaat, de beheerder zal tijdens de sessie niet de interactie initiëren. Het gaat vaak om een batch proces (nachtelijk synchroniseren van data) waarbij de verwerkers de uitvoerders zijn. Hoe een resource owner de autorisatie voor een client registreert valt buiten de scope van Edukoppeling.

3.3.2. Normatieve afspraken

Met opmerkingen [ER4]: TODO: Verder uitwerken

- De API (protected resource) MOET in een OAS specificatie zijn gedefinieerd
  - Minimaal versie 3.1
- De API (protected resource) MOET in de OAS specificatie aangegeven dat dit OAuth-profiel van toepassing is.
  - Bevat een verwijzing naar de AS die het AT uitgeeft.
- De authorization server meta data MOET worden gespecifieerd volgens RFC8414[8]
  - De grant_types_supported MOET de waarde “client_credentials” bevatten.
  - De token_endpoint_auth_methods_supported MOET de waarde "tls_client_auth" bevatten.
- De client die toegang wil tot de beveiligde API MOET geregistreerd zijn bij de Authorization Server.
  - De client (applicatie) MOET bij registratie een fijnmazige identifier (client_id) toegewezen krijgen.De client_id parameter MOET herleidbaar zijn naar de verwerker (OIN) die de beheerder (verwerker) van de client identificeert.
- De authorization server valideert request #1. Hierbij geldt dat:
  - een mTLS verbinding mogelijk is conform het UBV TLS Edukoppeling profiel;
    - authenticatie van de client op basis de certificaat hiërarchie;
    - identificatie (verwerker) op basis van OIN;
  - de querystring een routeringskenmerk MOET bevatten conform het REST-profiel;
  - de client (verwerker OIN) een mandaat MOET hebben voor de betreffende ketensamenwerking conform het MDX Secure API protocol;
  - de body een JSON object MOET bevatten met de volgende parameters:
    - grant_type parameter met de waarde “client_credentials”;
    - client_id parameter;
    - resource parameter;
      - de “aud” claim in het AT moet gelijk zijn aan deze waarde;
    - domain_hint parameter;
  - het JSON object MAG een scope parameter bevatten.
- Op basis van de gegevens in het JSON object MOET de authorization server verifiëren of er een fijnmazige autorisatie voor de client bestaat.
- Als de autorisatie service request #1 succesvol heeft gevalideerd, MOET deze een HTTP 200 OK status code respons (#2) geven conform RFC6749 sectie 5.1.
  - In de body zijn de volgende parameters opgenomen:
    - een AT (access_token);
    - de levensduur (expires_in maximaal 3600 seconden, één uur);
    - de scope(s).
  - De respons MOET NIET een refresh token bevatten
    - RFC6748:”A refresh token SHOULD NOT be included.”

Met opmerkingen [ER5]: Vanaf deze OAS versie kan mTLS gespecificeerd worden

Met opmerkingen [ER6]: Het betreft use cases waarbij verwerkers een (nachtelijke batch) interactie uitvoeren. Doordat de verwerkers deze uitwisseling initiëren en niet een beheerder bij de eindorganisatie wordt het OAuth client credentials profiel gebruikt en niet het code grant profiel.

Met opmerkingen [ER7]: Door toepassing van mTLS client authenticatie hoeven we geen RFC7523 toe te passen en hoeft het geen JWT te zijn

De client in request #1 wordt geauthenticeerd op basis van een mTLS client certificaat;

Met opmerkingen [ER8]: Of form parameters

Met opmerkingen [ER9]: Om de meer fijnmazige privacy maatregel te ondersteunen op het niveau van een dataset MOET in request #1 een domain_hint parameter opgenomen zijn. Deze parameter MOET worden gevuld met een identifier die herleid kan worden naar een set attributen (zoals voornaam en achternaam) en een lijst betrokkenen voor een bepaalde organisatorische eenheid.

[8] https://www.rfc-editor.org/rfc/rfc8414.html

    - Het access token MOET ondertekend zijn volgens een algoritme uit het UBV TLS Edukoppeling profiel.
- Als de autorisatie service request #1 niet succesvol heeft gevalideerd, MOET deze een HTTP 400 Bad Request status code respons (#2) geven.
    - In de body is een foutmelding opgenomen conform sectie 5.2 in RFC6749.
    - De Authorization Server MOET een invalid_client foutmelding geven conform RFC6749 als het client requests geen certificaat aanbiedt, of deze niet voldoet aan de PKIo hiërarchie.
- De client MOET de AT niet verwerken maar enkel gebruiken voor toegang tot de protected resource. Dit MOET plaatsvinden binnen de levensduur van het AT.
- ….

Met opmerkingen [ER10]: Nog niet compleet

4. Bijlage A: Overwegingen

We moeten nog een eerste versie vaststellen, maar we onderkennen nu al een aantal punten die we nu nog niet meenemen, maar wel op de roadmap zetten voor een volgende versie. Het betreft het volgende:

1. OAuth versie 2.1
2. Token binding o.b.v. RFC8705
3. Routeringskenmerk o.b.v. een JWT
4. Scope naming notation convention
5. …

4.1. OAuth versie 2.1

Er is een nieuwe OAuth versie in ontwikkeling https://datatracker.ietf.org/doc/html/draft-ietf-oauth-v2-1-07

This specification replaces and obsoletes the OAuth 2.0 Authorization Framework described in RFC 6749 and the Bearer Token Usage in RFC 6750.

Impact moet bepaald worden.

4.2. Token binding o.b.v RFC8705

RFC8705 ondersteunt niet subject.serial (PKIo vereist OIN in cert subject.serial)
In conceptversie nu volgende opgenomen:
token_endpoint_auth_methods_supported": ["tls_client_auth", "tls_client_auth_subject_dn"]
Beschikbare alternatieven zijn bijv. tls_client_auth_san_dns, tls_client_auth_san_uri, tls_client_auth_san_uri_ip

We gebruiken wel de metadata van RFC8705 om aan te geven dat een AS mTLS vereist.

4.3. Routeringskenmerk o.b.v. een JWT

Bij het REST profiel hebben we destijds nog niet voor een JWT gekozen gezien de extra implementatie effort die dit introduceert. We hebben toen ervoor gekozen om het routeringskenmerk op te nemen in query string met het uitgangspunt dat het point-2-pooint verbindingen betreft (geen intermediairs) en de respons bevat impliciet een routeringskenmerk voor dezelfde eindorganisatie.

Voor dit OAuth-profiel hebben we eerder besloten om zowel in het request als de respons een routeringskenmerk in een JWT te definiëren. In deze versie van het OAuth-profiel wordt echter op dit punt afgeweken om beter op het huidige REST profiel te kunnen aansluiten. We gaan dus ook in dit OAuth-profiel uit van een point-2-point uitwisseling waarbij slechts een routeringskenmerk in het request volstaat. Hiermee hebben we een meer vereenvoudigde eerste versie van het OAuth-profiel.

Het opnemen van het routeringskenmerk in een JWT in request en respons waarbij de respons een andere eindorganisatie kan bevatten dan het request kunnen we in een toekomstige versie gaan ondersteunen (eventueel ook in het REST-profiel).

4.4. Scope Naming notation convention

Als de client niet geautoiriseerd is voor de scope in request #1, MOET er dan een foutmelding gegeven worden of laten we dit over aan de implementatie (AS)? Alternatief kan bijvoorbeeld zijn om een token met een passende scope uit te geven.

5. Bijlage B: Uitgangspunten

5.1. Uitgangspunten discussiestuk versie 0.3

1. SaaS-Context en SaaS profielen zijn begrippen die we in de communicatie kunnen blijven gebruiken, maar we veranderen de naam van de profielen in 'Secure API-profielen'. We onderkennen de volgende Secure API profielen[9]:
    a. Secure API WUS(be, be-S & be-SE)-profiel
    b. Secure API REST-profiel
    c. Secure API OAuth-profiel
2. In de Edukoppeling documentatie en die van OSR hebben we het over een mandatering. Digikoppeling heeft het in een vergelijkbare procesafspraak (bevoegdheid intermediair/SAAS partij door 'machtigen') over een machtiging. In Edukoppeling blijven we het begrip mandatering hanteren.
3. De procesafspraken rond het OSR[10] worden onderdeel van de normatieve voorschriften voor Edukoppeling. De procesafspraken worden beschreven in een Secure API Protocol (zie **Fout! Verwijzingsbron niet gevonden.**).
    a. Bij een Secure API profiel[11] mandateert een eindorganisatie een verwerker om als onderdeel van een bepaalde ketensamenwerking vertrouwelijke gegevens te laten verwerken[12]. De gemandateerde verwerkers controleren vooraf aan de uitwisseling hun eigen en elkaars mandaat. Bij alle Secure API profielen wordt altijd een mandaat toegepast.
    b. De interacties voor registratie en verificatie van het mandaat en bijbehorende kaders zijn onderdeel van het Edukoppeling Secure API Protocol.
    c. Er wordt een generiek interactiepatroon gehanteerd. Dit betekent dat ook een Agentschap of onderwijsinstelling die zowel de rol van eindorganisatie als verwerker heeft voor zichzelf een mandaat registreert om binnen een bepaalde ketensamenwerking vertrouwelijke gegevens uit te wisselen. Dit dus om partijen binnen de ketensamenwerking in staat te stellen een generiek interactiepatroon rond verificatie in te richten die vooraf aan de uitwisseling wordt uitgevoerd.
    d. Het OSR is naast een mandatenregister ook een service register. Vanuit die functie is het wenselijk (geen onderdeel van de normatieve afspraak) om ook AS metadata op te nemen in het OSR. Voor de Resource Server / Protected Resources ligt het al voor de hand dat dergelijk informatie in het OSR beschikbaar is[13]. Mede omdat OSR ook uitgaat van een ketensamenwerking (afspraak) gaan we er vanuit dat partijen ook via bilaterale kanalen over de benodigde informatie kunnen beschikken.
    e. Het beheer van het Secure API protocol ligt in principe bij Kennisnet, maar Edustandaard en specifiek de Edukoppeling werkgroep is hierbij een belangrijke stakeholder.

heeft verwijderd: Figuur 1

Met opmerkingen [BD11]: Check of de definitie in het ROSA Begrippenmodel overeenkomt met die uit de AVG: https://rosa.wikixl.nl/index.php/F9420848-2723-409d-9a76-c21f1543a450 (de bron is hiervoor NORA Begrippenkader)

Met opmerkingen [KR12]: Als dit altijd 1:1 zou zijn dan kunnen we dat ook als regel toepassen zonder OSR mandaat (je bent altijd gemandateerd om namens jezelf te verwerken).. Maar zodra de gemandateerde verwerker een ander ' niveau' is dan de eindorganisatie (bv ihgv subOIN's of iets dergelijks) dan is het handiger om dit idd in de OSR op te nemen.

[9] De WUS be-S & be-SE profielen bieden extra functionaliteit (integriteit en integriteit icm vertrouwelijkheid op data niveau) doordat de data in transport ondertekend of ondertekend en versleuteld kan worden. Het OAuth profiel biedt extra beveiliging doordat de autorisatie technisch geborgd is. Bij de uitwerking van de architectuur kunnen wellicht nog verschillende beveiligingsniveaus aan de profielen toekennen.
[10] Nader onderzoek moet nog uitwijzen of alle scenario's door OSR ondersteund kunnen (gaan) worden.
[11] Met Secure API duiden we profielen aan waarbij ook de machtiging aan een verwerker een rol speelt. Als we in de Architectuur meer profielen gaan ondersteunen, bijvoorbeeld API key, dan is het wellicht beter om per profiel met een beveiligingsniveau aanduiding te gaan werken.
[12] Verwerken is hier vergelijkbaar als in AVG gedefinieerd.
[13] Er vanuit gaande dat dit geen publieke info is, maar alleen toegankelijk voor partijen die als verwerker actief zijn binnen een bepaalde ketensamenwerking.

4. De Secure API profielen worden toegepast bij de uitwisseling van vertrouwelijke gegevens. Dit kunnen persoonsgegevens, maar ook bedrijfskritische gegevens zijn.
    a. Het gaat om vertrouwelijke gegevens en het juridisch kader (AVG) wordt dus niet expliciet binnen het Secure API Protocol opgenomen. De Edukoppeling Secure API Profielen voldoen wel aan algemeen geldende beveiligings- en privacy maatregelen die bij het verwerken van persoonsgegevens verwacht kunnen worden. Het Edustandaard Certificeringsschema geeft aan wanneer deze toegepast dienen te worden.
    b. We definiëren eigen Edukoppeling begrippen (eindorganisatie en verwerker) en sluiten dus niet direct aan op het juridisch (AVG) kader. Wel is het handig als we voor het begrip 'verwerker' nog een ander begrip te gaan gebruiken om verwarring te voorkomen. En moeten wellicht ook de definitie aanscherpen.
5. Het functionele toepassingsgebied van Secure API profielen blijft ongewijzigd.
6. We gebruiken (voorlopig[14]) de internationale open standaard OAuth (RFC6749[15] en RFC6750) als vertrekpunt voor de ontwikkeling van het OAuth profiel (dus niet het iGOV-NL OAuth). Onze use case past het beste bij het OAuth client credentials en dit OAuth profiel zal als basis dienen.
7. Het Secure API OAuth profiel gaat ervan uit dat er meerdere lokale Authorization Servers en Resource Servers[16] zijn die door verschillende partijen worden beheerd.
8. In de context van het Secure API OAuth profiel wordt een meer specifieke rolaanduiding wenselijk. In de context van het OAuth profiel kunnen we namelijk spreken van een verwerker die een OAuth AS en RS beheert en een verwerker als client die de protected resource wil verwerken. De relatie tussen bestaande Edukoppeling rollen en nieuwe OAuth rollen wordt weergegeven in **Fout! Verwijzingsbron niet gevonden.**
9. De lokale Authorization Servers moeten vooraf aan de uitgifte van een Access Token hebben geverifieerd of de client gemandateerd is.
    a. Hoe de AS de OSR-verificatie van client vertaalt[17] naar het wel of niet uitgeven van een access token valt buiten de Edukoppeling afspraak.
10. ~~Het Secure API OAuth profiel (wijzigingen en/of aanvullingen op de internationale open standaard) wordt in het Engels geformuleerd.~~
11. ~~Als wijzigingen overeenkomen met die van het iGOV-NL OAuth profiel dan wordt dit expliciet aangegeven met "<iGOV-NL>"[18]. We doen dit voor hele tekstblokken.~~
12. Het Secure API OAuth profiel gaat uit van het OAuth client credentials profiel[19]. Het gaat om confidential clients die een Access Token krijgen op basis van hun identiteit[20]. De client moet

Met opmerkingen [KR13]: Een andere begrip kan ook verwarring stichten. Ik vind ' verwerker' juist een heel mooi generiek begrip waarvan iedereen weet wat ermee bedoeld wordt

Met opmerkingen [KR14]: Dat betekent ook een registratie van de client in alle auth servers waar je een koppeling mee wilt leggen?. Een OSR als ketenauthserver, zou dat niet mooier zijn?

heeft verwijderd: Figuur 3

Met opmerkingen [ER15]: We documenteren alles in het Nederlands. Mogelijk wel teksten overnemen uit RFC's

Met opmerkingen [ER16]: We specificeren op basis van RFC's. EK heeft een ander vertrekpunt (M2M/CC) dan iGOV NL

[14] Het Kennisplatform ontwikkelt nog OAuth profielen. Mochten we op een later moment kunnen aansluiten dan nemen we dat in overweging.
[15] https://datatracker.ietf.org/doc/html/rfc6749 (ook OAuth wordt doorontwikkeld: https://datatracker.ietf.org/doc/draft-ietf-oauth-v2-1/).
[16] We gaan bij deze versie uit van een lokale AS en RS. Er kan op termijn besloten dat er een centrale AS komt (wellicht OSR) voor de beveiliging van API's bij meerdere Resource Servers van verschillende partijen. Dit zou dan wel meer een OSR implementatietraject zijn en een afsprakenstelsel zijn wat wellicht in het Secure API protocol beschreven kan worden. De interfaces van OSR beschrijven dan in zijn geheel het OAuth profiel.
[17] Het OSR registreert een mandaat voor een bepaalde ketensamenwerking. De OAuth AS beveiligd een bepaalde resource. Er is geen 1-op-1 relatie te leggen vanuit de standaard.
[18] Gezien het iGOV-NL profiel gebaseerd is op het iGOV profiel (https://logius-standaarden.github.io/OAuth-NL-profiel/#bib-igov.oauth2) en hier ook teksten van overneemt kan het zijn dat binnen een <iGOV-NL> tag alleen iGOV teksten staan. Voor de leesbaarheid worden er niet geneste (iGOV/iGOV-NL) tags toegepast. Voor de iGOV-NL referenties wordt de volgende in ontwikkeling zijnde draftversie gebruikt: https://logius-standaarden.github.io/OAuth-NL-profiel/
[19] De clients binnen het Edukoppeling Secure API OAuth profiel betreffen alleen confidential clients in de vorm van een web application Zie toelichting bijlage A.
[20] Dus niet op basis van toestemming door een gebruiker (Resource Owner).

zich bij de AS registreren zodat identificatie mogelijk is om de verificatie van het mandaat bij het OSR uit te voeren en te kunnen bepalen of een Access Token geleverd mag worden.

13. ~~Client identificatie op basis van OIN.~~
14. Verwerker met de AS/RS combi zijn zelf verantwoordelijk voor het registreren van een client. ~~Hiervoor moet het OIN[21] gebruikt te worden~~.
15. Een client wordt geauthentiseerd op basis van het UBV TLS Edukoppeling profiel (mTLS). In OAuth (RFC6749) worden een aantal methoden voor client authenticatie[22] onderkend. Het Secure API OAuth vereist client authenticatie op basis van mTLS en PKIo certificaten.
16. ~~Access Tokens worden beveiligd op basis van RFC8705[23]. Met het toepassen van mTLS kunnen we ook de Mutual-TLS Client Authentication and Certificate-Bound Access Tokens (RFC8705) standaard toepassen. We sluiten hiermee ook aan bij nieuwe ontwikkelingen rond OAuth[24]. Een Access Token is in principe een bearer token, het kan worden gebruikt door iedereen die in het bezit is van het token. Met RFC 8705 wordt een bewijs van bezit (proof-of-possession) aan het token gebonden. Het bezit is de asymmetrische sleutel van het mTLS client certificaat. De koppeling van de sleutel aan het token wordt ook doorgegeven aan de beveiligde API (RS). De client kan het bezit aantonen doordat deze beschikt over de private sleutel en dit ook kan aantonen.~~
17. We adviseren het toepassen van beveiliging best practices, OAuth 2.0 threat model and security considerations [RFC6819]. Ook wordt aanbevolen om kennis te nemen van de OAuth 2.0 security best current practices [OAUTH-SBP] en JSON Web Tokens [JSONWT-BP].

Met opmerkingen [ER17]: Client_id wordt door AS uitgegeven. Beheerdwer van clent is verwerker en wordt geidentificeerd obv OIN

Met opmerkingen [ER18]: Helaas niet mogelijk, subject.serial wordt niet ondersteund

[21] Het OIN wordt nu binnen Edukoppeling en het OSR gebruikt. Het kan later blijken dat bij een AS een fijnmazige identificatie van een client nodig is. Zolang het een lokale (decentrale) AS betreft kunnen AS beheerders zou een client die onderdeel is van een bepaalde verwerkersorganisatie (OIN) aanvullend geïdentificeerd kunnen worden op basis van bijvoorbeeld een GUID.
[22] https://datatracker.ietf.org/doc/html/rfc6749#section-2.3
[23] https://datatracker.ietf.org/doc/rfc8705/
[24] Oauth 2.1 (https://datatracker.ietf.org/doc/draft-ietf-oauth-v2-1/)"*It is RECOMMENDED to use asymmetric (public-key based) methods for client authentication such as mTLS [RFC8705] or a JWT [RFC7523].*"

6. Bijlage C: Bronnen

6.1. Normatief

[RFC6749]
The OAuth 2.0 Authorization Framework. D. Hardt, Ed.. IETF. October 2012. Proposed Standard. URL: https://datatracker.ietf.org/doc/html/rfc6749

[RFC6750]
The OAuth 2.0 Authorization Framework: Bearer Token Usage. M. Jones; D. Hardt. IETF. October 2012. Proposed Standard. URL: *https://www.rfc-editor.org/rfc/rfc6750*

[RFC9068]
JSON Web Token (JWT) Profile for OAuth 2.0 Access Tokens, V. Bertocci, 2021-10-21 , Proposed Standard URL *https://datatracker.ietf.org/doc/rfc9068/*
*In-market use has shown that many commercial OAuth 2.0 implementations elected to issue access tokens using a format that can be parsed and validated by resource servers directly, without further authorization server involvement. The approach is particularly common in topologies where the authorization server and resource server are not co-located, are not run by the same entity, or are otherwise separated by some boundary. At the time of writing, many commercial implementations leverage the JSON Web Tokens (JWT) **[RFC7519]** format.*

This specification aims to provide a standardized and interoperable profile as an alternative to the proprietary JWT access token layouts going forward. Besides defining a common set of mandatory and optional claims, the profile provides clear indications on how authorization request parameters determine the content of the issued JWT access token, how an authorization server can publish metadata relevant to the JWT access tokens it issues, and how a resource server should validate incoming JWT access tokens.

*Please note: although both this document and [**RFC7523**] use JSON Web Tokens in the context of the OAuth2 framework, the two specifications differ in both intent and mechanics. Whereas [**RFC7523**] defines how a JWT Bearer Token can be used to request an access token, this documents describes how to encode access tokens in JWT format.*

Rationale: We stellen voor deze draft op te nemen. De toepasbaarheid moet wel vooraf aan vaststelling bij implementaties getoetst worden.

[RFC8414]
OAuth 2.0 Authorization Server Metadata. M. Jones; N. Sakimura; J. Bradley. IETF. June 2018. Proposed Standard. URL: https://www.rfc-editor.org/rfc/rfc8414

[RFC7518]
JSON Web Algorithms (JWA), **RS256**

[RFC6819]
OAuth 2.0 Threat Model and Security Considerations. T. Lodderstedt, Ed.; M. McGloin; P. Hunt. IETF. January 2013. Informational. URL: *https://datatracker.ietf.org/doc/html/rfc6819*

Met opmerkingen [ER19]: Sinds het profiel is opgesteld, heeft IETF de RFC 9068 uitgebracht. RFC9068 beschrijft een standaard JWT formaat voor access tokens. Deze RFC meenemen in het profiel kan overwogen worden voor interoperabiliteit.
Concreet wordt met name sectie 3.2.1 van het profiel geraakt. Om dit in lijn met sectie 2.2 van RFC9068 te brengen, zullen de claims 'iat' en 'azp' als verplicht moeten worden toegevoegd. Verder moet de claim 'sub' ook verplicht worden gesteld, dit is lijn met issue #7.
Daarnaast is de claim 'client_id' toegevoegd, die zou overeenkomen met 'azp'. Hoe hiermee om te gaan zal nader uitgewerkt moeten worden.
Daarnaast is er ook een wijziging in de JWT header. De JWT header hoort 'typ' (content-type) 'at+jwt' te krijgen, conform RFC9068 sectie 2.1. Ook hiervoor zal nader uitgewerkt moeten worden hoe hiermee om te gaan.
Bovenstaande is geen volledige analyse, er kan dus nog verdere impact zijn.

[RFC7519]
JSON Web Token (JWT). *M. Jones; J. Bradley; N. Sakimura. IETF. May 2015. Proposed Standard. URL: https://www.rfc-editor.org/rfc/rfc7519*

[RFC7523]
JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants. M. Jones; B. Campbell; C. Mortimore. IETF. May 2015. Proposed Standard. URL: *https://www.rfc-editor.org/rfc/rfc7523*

[RFC7662]
OAuth 2.0 Token Introspection. J. Richer, Ed.. IETF. October 2015. Proposed Standard. URL: https://www.rfc-editor.org/rfc/rfc7662

[RFC7800]
Proof-of-Possession Key Semantics for JSON Web Tokens (JWTs). M. Jones; J. Bradley; H. Tschofenig. IETF. April 2016. Proposed Standard. URL: https://www.rfc-editor.org/rfc/rfc7800