Edukoppeling

Discussiestuk OAuth/Secure API-profiel

Edustandaard

Datum: oktober 2022

edustandaard

Inhoudsopgave

1. Documenthistorie

Versie	Auteur	Datum	Opmerking
0.1	E. Reinhoud	12 september 2022	Initiële versie
0.2	E. Reinhoud	12 oktober 2022	Verwerking opmerkingen en besluiten 21 sept 2022

2. Inleiding

2.1. Aanleiding

Edukoppeling is een belangrijke bouwsteen in de ketenreferentiearchitectuur ROSA[1]. Het functionele toepassingsgebied van de Edukoppeling betreft de M2M uitwisseling van vertrouwelijke gegevens via een beveiligde verbinding.

Voor hun koppelingen in de ECK-keten heeft Topicus een OAuth client credentials profiel in gebruik genomen dat ook als onderdeel is van het nieuwe SEM Ecosysteem. Dit profiel wordt toegepast bij uitwisselingen waarbij Topicus als bronhouder van persoonsgegevens die namens een onderwijsinstelling deze gegevens verstrekt aan een derde partij. In de ECK-keten zijn meerdere partijen die als bronhouder persoonsgegevens kunnen delen met derde partijen en dus is de verwachting dat meerdere partijen een dergelijk OAuth-profiel willen gaan gebruiken. Het risico hierbij is dat er bij de inrichting verschillende keuzes worden gemaakt. Er wordt nu aan de Edukoppeling werkgroep gevraagd om tot een gestandaardiseerd OAuth-profiel te komen die breed inzetbaar is binnen het onderwijs[2].

We pakken de ontwikkeling van het Edukoppeling OAuth-profiel op zonder vooraf al een duidelijke voorkeur te hebben over de nog te maken keuzes. Deze keuzes zijn in principe aan de werkgroep, maar we hebben wel te maken met de kaders van de ROSA. Daarnaast zijn er verschillende ontwikkelingen die het definiëren van een OAuth standaard complex maken. Met dit document hopen we de complexiteit te beperken door overzicht te bieden. In dit document willen we de aandachtspunten kaderen en de besluiten herleidbaar maken. Op basis van deze besluiten zal het uiteindelijke OAuth-profiel opgesteld worden.

2.2. Doel en status van dit document

Dit is versie 0.2 van dit document. Het zal geen onderdeel vormen van de uiteindelijke Edukoppeling afspraak, maar dient alleen voor transparantie van het proces door het inzichtelijk maken van relevante informatie en het traceerbaar maken van de verschillende te maken keuzes.

In deze versie zijn de besproken voorstellen vertaald naar besluiten in de vorm van uitgangspunten die voor het OAuth profiel gaan gelden. Voordat een profiel opgesteld kan worden moet er consensus zijn rond de uitgangspunten. Om welke uitgangspunten het uiteindelijk gaat is onderdeel van de discussie.

[1] https://www.wikixl.nl/wiki/rosa/index.php/Hoofdpagina

[2] Partijen buiten het onderwijs kunnen dit ook toepassen. De interoperabiliteit binnen het onderwijs heeft echter prioriteit.

3. Samenvatting bespreking 21 september 2022

Op 22 september is de 0.1 versie van dit discussiestuk besproken. Hieronder wordt per laag hiervan verslag gegeven.

3.1. Algemeen

Begrippen

We heroverwegen een aantal bestaande begrippen. Het huidige functioneel toepassingsgebied van de Edukoppeling SaaS-profielen wordt gehandhaafd, maar we gaan de naam veranderen. We hebben het niet meer over SaaS-profielen omdat we het profiel niet alleen gebruiken in de context van een SaaS-dienst[3] en uitwisseling van persoonsgegevens. Er wordt voorgesteld om de naam 'Secure API-profielen' te gaan gebruiken[4]. We hebben in de huidige situatie de volgende Secure API-profielen[5]:

- Secure API WUS profiel
- Secure API REST profiel
- Secure API OAuth profiel

Het gaat niet alleen om de uitwisseling van persoonsgegevens, maar om uitwisseling van vertrouwelijke gegevens (dus bijvoorbeeld ook bedrijfskritische gegevens). Er kan sprake zijn van persoonsgegevens, maar we willen dit niet expliciet onderdeel maken van het profiel omdat het dus ook andere vertrouwelijke gegevens kan betreffen.

In de huidige Edukoppeling documentatie en die van OSR wordt wel van een mandatering gesproken. Er wordt voorgesteld het meer algemenere begrip machtiging te gebruiken. We sluiten hiermee ook aan op een recente Digikoppeling handreiking waar een vergelijkbaar concept wordt beschreven (Bevoegdheid intermediair/SAAS partij door 'machtigen'[6]).

Opbouw OAuth profiel binnen het Edukoppeling Informatie-uitwisselingsmodel

De Secure API-profielen bevatten conform de huidige situatie voorschriften voor:

- registratie en verificatie van de machtiging[7] door eindorganisatie aan verwerker voor verstrekking van vertrouwelijke gegevens (proceslaag);
- routeren o.b.v. een routeringskenmerk van een verwerker naar een bepaalde eindorganisatie (applicatielaag);

Met opmerkingen [KR1]: Ik vind het een mooie naam echter met SaaS gaven we een didelijk onderscheid aan met Digikoppeling. De ontkoppeling van gegevensverwerker en formele eindpartij is de basis van Edukoppeling. Dat zijn we nu wel een beetje kwijt in de naamgeving

Met opmerkingen [ER2R1]: Dit was mijn conclusie op basis van vorig overleg. Het gaat niet altijd om een SaaSdienst en ook niet altijd om persoonsgegevens. Nieuwe naam is meer nautraal. Voorzie op architectuurniveau nog wel dat een extra aanduiding van beveiligingsniveau wenselijk is. Er kan bijvoorbeeld ook een Secure API profiel komen op basis van TLS en API key. Maar is meer discussie rond architectuur document

Met opmerkingen [ER3R1]: In omschrijving wel SaaS blijven benoemen, zoals ook nu al het geval is in AR. Meer generiek gaat het ook om use case waarbij een partij een andere opdracht moet kunnen geven voor uitwisseling van vertrouwelijke gegevens. Aanduiding niveaus om profielen te kunnen onderkennen

Met opmerkingen [GR4]: Het verrast mij dat OAuth wordt opgesomd naast WUS en REST. Dat zijn naar mijn idee niet dezelfde grootheden.

Met opmerkingen [ER5R4]: Klopt, idee is om in stappen eea vast te leggen. Eerste stap is hoe registratie en verificatie van machtiging wordt ingericht. Keuze is nu om dit generiek te houden met bestaande WUS/REST en OSR,
Volgende stap is inderdaad kijken of nadere aanduiding nodig is. Ik hint hier al naar door bijvoorbeeld met een level aanduiding te gaan werken

heeft verwijderd: kan

Met opmerkingen [EB6]: In welke zin is dit begrip algemener? In www.encyclo.nl zijn het synoniemen. Ik kan geen bron vinden die dit echt onderscheid. Als er geen onderscheid is stel ik voor in de begrippenlijst mandaat als synoniem op te nemen.

Met opmerkingen [ER7R6]: geen probleem om het als synoniem op te nemen als we uberhaubt besluiten op DK aan te sluiten

Met opmerkingen [ER8R6]: Wij houden Mandatering, wel relatie met DK machtiging benoemen

[3] De SaaS-context kan wel helpen als sprekend voorbeeld dat een verwerker gegevens deelt in opdracht van een eindorganisatie.
[4] Deze naam wordt dus ook verder in dit document gebruikt. Ook de naam van het document is aangepast.
[5] In de architectuur zullen we wellicht meerdere profielen gaan onderkennen. Bijvoorbeeld profielen waarbij geen machtiging van eindorganisatie van toepassing is.
[6] https://github.com/Logius-standaarden/Digikoppeling-Handreiking-Adressering-en-Routering/blob/main/documenten/Analyse_knelpunten_Routering_Intermediairs.md
[7] Dit is dus een toestemming in de context van Edukoppeling (OSR) en staat los van enig juridisch kader.

- beveiligd (P2P[8]) transport (infrastructuurlaag);

Het nieuwe OAuth profiel heeft extra functionaliteit in de applicatielaag[9]. Gegevens kunnen niet worden verstrekt zonder een geldig Access Token in het request.

Het voorstel om kolom 'D' binnen het Informatie-uitwisselingsmodel te kiezen wordt niet overgenomen. In het overzicht waren vanwege leesbaarheid niet alle mogelijke varianten opgenomen. Binnen de verschillende lagen zijn nog meer keuzes (op detailniveau) die als variant onderkend zouden kunnen worden. De meeste discussie gaat over de proceslaag (blauw).

3.2. Proceslaag

De meeste discussie gaat over de proceslaag en betreft met name de machtiging door eindorganisatie en relevantie van het juridisch kader. Hieronder wordt per voorstel verslag gedaan van de discussie.

P1:De proceslaag wordt onderdeel van de normatieve voorschriften voor Edukoppeling Secure API profielen.

Er is consensus om de proceslaag onderdeel te laten zijn van de normatieve Edukoppeling afspraken.

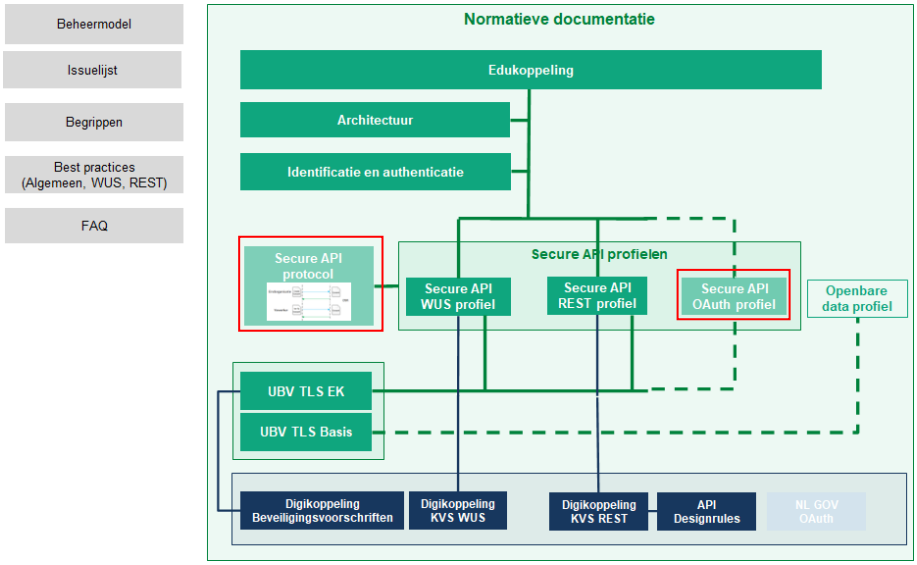
P2: **De kaders voor de proceslaag worden in een Secure API protocol beschreven.**

Er is ook consensus om de proceslaag op te nemen in een apart document. Omdat het uitwisseling van vertrouwelijke gegevens betreft en niet alleen persoonsgegevens, noemen we dit normatieve document 'Secure API protocol' in plaats van 'SaaS privacy-protocol' (zie Figuur 1). We sluiten dan ook niet direct aan op de Edustandaard Architectuuraanpak toegang waar het met name maatregelen betreft de uitwisseling van persoonsgegevens. Echter is deze discussie daar ook nog niet helemaal beslecht en wordt mogelijk later ook een bredere scope gehanteerd. Aansluiting kunnen we dus wellicht in een later stadium heroverwegen.

heeft verwijderd: Figuur 1

heeft opmaak toegepast: Lettertype: 11 pt

[8] Het WUS profiel bevat ook voorschriften om berichten te ondertekenen en versleutelen.
[9] Binnen de architectuur is het wellicht wenselijk om met verschillende niveaus voor Secure API profielen te gaan werken. Bijvoorbeeld Level3 (mTLS + toestemming Eindorganisatie en OAuth), Level2 (mTLS + toestemming Eindorganisatie) en Level1 (TLS + API key)



Figuur 1 – Positionering van Secure API protocol en OAuth profiel binnen Edukoppeling

P3: **Onderdeel van het Secure API protocol is het juridisch kader.**

Zoals eerder aangegeven is besloten dat we de profielen toepassen bij vertrouwelijke gegevens en niet alleen de uitwisseling van persoonsgegevens betreft. Binnen de profielen maken we ook geen expliciet onderscheid of het persoonsgegevens of andere vertrouwelijke gegevens betreft. Hiermee kunnen we ook de AVG niet als bovenliggend kader gebruiken. Het juridisch kader van de AVG is dus niet leidend binnen deze Edukoppeling-profielen.

P4: **Onderdeel van het Secure API protocol zijn de interacties.**

Er was al consensus om de proceslaag onderdeel te laten zijn van de normatieve Edukoppeling afspraken. Er is nu ook consensus om hierin de interacties rond de registratie en verificatie van de machtiging die een eindorganisatie een verwerker geeft op te nemen.

De meeste discussie gaat er over of de registratie en verificatie via een centraal of decentraal (lokaal) register ondersteund moet worden. Binnen het voorstel werd uitgegaan van een decentraal registratie conform SEM consent API. Hiermee zou een directe binding tussen registratie en beveiliging van API met AS bereikt kunnen worden. De decentrale variant kent echter ook een aantal nadelen, zoals mogelijke toename van administratieve last voor onderwijsinstellingen. Daarnaast hebben we met het OSR al een centraal register. Er wordt dan ook besloten dat de registratie en verificatie centraal wordt ingericht. Met het Edukoppeling Secure API protocol sluiten we dus aan op het OSR voor registratie en verificatie van machtigingen. De blauwe laag is dus gelijk aan al bestaande praktijk. Het is alleen zaak dat nog in een normatief document op te nemen. We hadden al eerder besloten dit te doen voor het bestaande WUS en REST profiel.

Met opmerkingen [KR9]: Is het niet handig om aan te geven dat als het om persoonsgeevens gaat dat dan alsnog de AVG wel leidend is? Anders krijgen we daar straks weer gedoe mee

Met opmerkingen [ER10R9]: Op basis van vorige discussie begreep ik dat we zo generiek mogelijk willen zijn binnen het profiel. Het gaat om vertrouwelijke gegevens en generieke maatregelen. Met het onderkennen van persoonsgegevens zou er weer een subprofiel ontstaan.

Met opmerkingen [BD11R9]: Dat klopt, het profiel moet voor een breed scale aan vertrouwelijke gegevens toepasbaar zijn. Dat kunnen persoonsgegevens zijn, maar ook bedrijfskritische gegevens. Maar als de eisen uit het AVG voor persoonsgegevens ook gelden voor de bedrijfskritische gegevens dan voldoen we daarmee toch impliciet aan de eisen die vanuit het AVG worden gesteld, daar waar dat relevant is? Ik zie niet zo gauw in dat er een subprofiel dan weer nodig is. Het AVG is dan wel niet formeel leidend maar we voldoen er wel aan.

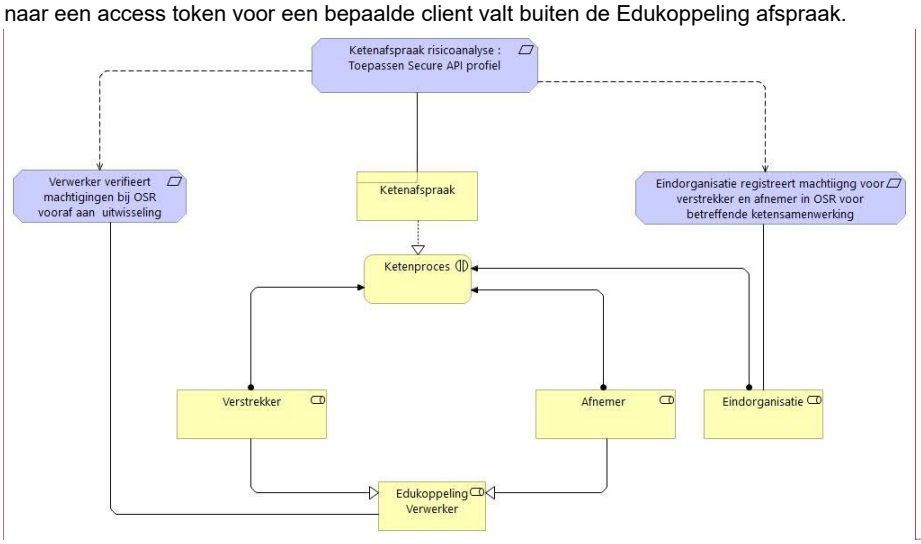
Met opmerkingen [GR12R9]: Je mag het AVG kader ook toepassen voor niet persoonsgegevens. Daar kan niemand tegen zijn.

Met opmerkingen [ER13R9]: Gaat om vertrouwelijke gegevens, dit kunnen persoonsgegevens zijn. We onderkennen hier echter geen specifieke maatregelen voor. De maatregelen moeten voldoende zijn voor alle onderliggende scenario's

In Figuur 2 zijn op basis van Archimate de proceslaag rollen van het Secure API profiel opgenomen. Bij de verschillende rollen is aangegeven welke verantwoordelijkheden hierbij horen rond de registratie en verificatie van de machtiging voor het delen van gegevens. De Eindorganisatie heeft de verantwoordelijkheid om voor verwerkers[10] binnen een bepaalde ketensamenwerking een machtiging te registreren. De verwerkers moeten elkaars machtiging en die van hunzelf verifiëren voorafgaand aan de uitwisseling van vertrouwde gegevens. Bij de verwerkers wordt er een onderscheid gemaakt in een gegevensverstrekker en een gegevensafnemer om aan te geven dat de verstrekker de afnemer in de context van OAuth registreert als client. In de context van het OAuth profiel heeft de verstrekker (die OAuth AS en RS beheert) de verantwoordelijkheid om bij het OSR te verifiëren of de geregistreerde client onderdeel is van de beoogde ketensamenwerking voordat een access token geleverd wordt. Het Access Token geeft toegang tot een API, de verificatie bij OSR geeft aan of de client (afnemer) gemachtigd is om namens eindorganisatie deel te nemen aan de ketensamenwerking, Hoe de verstrekker de OSR-verificatie van afnemer vertaalt naar een access token voor een bepaalde client valt buiten de Edukoppeling afspraak.

Met opmerkingen [BD14]: Gezien de voetnoot, zullen we het begrip dan ook van een andere definitie moeten voorzien voor de context van Edukoppeling.

Met opmerkingen [ER15R14]: lijkt mij een goed idee, maakt het ook expliciet waar we het wel en niet over hebben



Met opmerkingen [GR16]: De paarse blokjes zijn archimate requirements. Ik stel voor om die te omschrijven met een user storie: als.. wil ik... zodat ik ...

Met opmerkingen [ER17R16]: Kan, is onderdeel van het Secure API protocol wat we gaan opstellen. Met de OSR interacties

Figuur 2 - Secure API Rollen (gelden voor OAuth, REST en WUS)

Doordat we nu niet meer alleen normatief de logistieke laag (bericht en transport) voorschrijven, maar ook de kaders voor het proces (rollen en interacties rond registratie machtiging) kunnen we stellen dat we een afsprakenstelsel voor OSR opnemen. Het beheer hiervan ligt bij Kennisnet, maar ook de Edukoppeling werkgroep is hierbij een belangrijke stakeholder.

Er is verder aangegeven dat het wenselijk is om een generiek interactiepatroon te houden ongeacht de actoren die uitwisselen. Ook een Agentschap of onderwijsinstelling die de rol van eindorganisatie als verwerker heeft, registreert voor zichzelf een machtiging om binnen

Met opmerkingen [GR18]: Verder graag definiëren wat de andere blokjes in de figuur 2 betekenen. Impliciet lijkt dit te gaan over een lees-API, maar API's kun je ook gebruiken om te schrijven.

Met opmerkingen [ER19R18]: Klopt, belangrijkste stap leek mee om gegevensuitwisselingen te bespreken, ook irt push/pull. Volgende stap is inderdaad crud functies onderkennen. Zie dat als onderdeel van applicatielaag, bijv obv scopes

[10] Verwerker in de context van Edukoppeling is niet hetzelfde als de verwerker in de context van de AVG.

een bepaalde vertrouwelijke gegevens uit te wisselen. Ook deze registratie wordt vooraf aan de uitwisseling door de partijen geverifieerd.

P5: **Het functionele toepassingsgebied van het Secure API OAuth profiel betreft een RESTful M2M gegevensuitwisseling via een beveiligde point-to-point verbinding waarbij tenminste één partij dit in opdracht doet van een verwerkingsverantwoordelijke.**

Er wordt besloten om het functionele toepassingsgebied niet te wijzigen. Het gaat om de uitwisseling van vertrouwelijke gegevens en niet alleen persoonsgegevens. De rollen vanuit het juridisch kader (AVG) zijn dus niet per definitie van toepassing.

3.3. Applicatielaag

A1: **Het NL GOV OAuth profiel vormt de basis voor de applicatielaag.**

Dit voorstel wordt niet overgenomen. Er wordt besloten om voorlopig de open internationale standaard als basis te nemen (RFC6749). Men mist in het NL GOV OAuth profiel de onderbouwing bij de verschillende afslagen.

Omdat de eindorganisatie de machtiging voor de verwerker niet decentraal registreert kiezen we ook impliciet voor het OAuth client credentials profiel[11]. Zoals aangegeven is de beheerder van de AS en API (RS) er verantwoordelijk voor dat een AS pas een Access Token kan uitgeven aan een client als in het OSR een machtiging bestaat voor het gebruik van de API in de context van de betreffende ketensamenwerking.

We weten dat het Kennisplatform API's[12] een client credentials profiel gaat ontwikkelen en we blijven dit volgen. Bij de ontwikkeling van ons OAuth profiel kunnen we hier in ieder geval deels gebruik van maken. Verder wordt ook de OAuth[13] standaard doorontwikkeld. Daar waar wenselijk maken we een keuze die op deze nieuwe versie aansluit. Met name dit soort keuzes zal wel vooraf aan vaststelling validatie van implementeerbaarheid in de keten vereisen.

A2: **We gebruiken het OAuth client credentials profiel als basis voor de interacties in de applicatlelaag.**

Dit voorstel wordt overgenomen (zie ook A1).

A3: **Een OAuth Authorization Server wordt door meerdere partijen geïmplementeerd.**

Dit voorstel wordt overgenomen (zie ook P4). Het profiel gaat uit van een decentrale (lokale) Authorization Server. Dit betekent dat zowel de registratie als de uitgifte van het Access Token door Authorization Servers (AS) wordt uitgevoerd die in hetzelfde (netwerk/beveiligings)domein staan als de Resource Server b(RS).

---

[11] Niet het OAuth code grant profiel

[12] https://www.geonovum.nl/themas/kennisplatform-apis

[13] draft-ietf-oauth-v2-1-06 - The OAuth 2.1 Authorization Framework

3.4. Infrastructuurlaag

Dit voorstel wordt overgenomen. In de infrastructuurlaag wordt het UBV TLS Edukoppeling profiel toegepast (mTLS/PKIo). Deze keuze maakt ook de toepassing van RFC8705 in de applicatielaag mogelijk.

4. Uitgangspunten

Op basis van de samenvatting van het overleg van september zijn hieronder uitgangspunten geformuleerd. Deze worden tijdens het volgende overleg besproken ter vaststelling. De vastgestelde uitgangspunten vormen het kader voor het op te stellen Edukoppeling Secure API OAuth profiel.

1. SaaS-Context en SaaS profielen zijn eigenlijk begrippen die niet goed passen. We veranderen de naam van de SaaS profielen naar ‘Secure API-profielen’. We onderkennen de volgende Secure API profielen:
    a. Secure API WUS-profiel
    b. Secure API REST-profiel
    c. Secure API OAuth-profiel
2. Daar waar het begrip mandateren gebruikt wordt gaan we het begrip ‘machtigen’ gebruiken. We sluiten hiermee ook aan op Digikoppeling.
3. De proceslaag wordt onderdeel van de normatieve voorschriften voor Edukoppeling Secure API profielen. Deze wordt beschreven in een Secure API Protocol (zie Figuur 1).
4. Bij een Secure API profiel[14] machtigt de eindorganisatie een verwerker om als onderdeel van een bepaalde ketensamenwerking vertrouwelijke gegevens te delen met een derde partij. Verwerkers controleren vooraf aan de uitwisseling elkaars machtiging. Bij alle Secure API profielen wordt altijd een dergelijke machtiging toegepast.
5. De interacties voor registratie en verificatie van de machtiging en bijbehorende kaders zijn onderdeel van het Secure API Protocol.
6. Het Secure API Protocol is gebaseerd op het OSR (beide verwerkers moeten in een keten elkaars machtiging kunnen verifiëren[15]).
7. De Secure API profielen worden toegepast bij de uitwisseling van vertrouwelijke gegevens. Dit kunnen persoonsgegevens maar ook bedrijfskritische gegevens zijn.
8. Het gaat om vertrouwelijke gegevens[16] en het juridisch kader wordt dus niet binnen het Secure API Protocol opgenomen. We definiëren eigen rollen (eindorganisatie en verwerker) die niet direct aan het juridisch kader gekoppeld zijn.
9. Met het ontkoppelen van het juridisch kader is het ook niet meer relevant of een organisatie een agentschap is of niet. Alle partijen worden als verwerker beschouwd die in opdracht van de eindorganisatie uitwisselen. Net als voor een agentschap geldt dit ook voor een onderwijsinstelling die eindorganisatie en verwerker is. Beide

heeft opmaak toegepast: Lettertype: 11 pt

heeft verwijderd: Figuur 1

Met opmerkingen [EB20]: Tekstvoorstel: een andere door de eindorganisatie of anderzijds gemachtigde verwerker binnen dezelfde ketensamenwerking. (anderzijds onder de aanname dat verwerkers als DUO niet door de eindorganisatie gemachtigd hoeven te worden)

Met opmerkingen [ER21R20]: tekstvoorstel prima, maar we hanteren een generieke benadering en dus is het niet relevant of het duo/agentschap is of niet. Die zou zichzelf machtigen. er wordt vooraf een machtiging geverifieerd

Met opmerkingen [EB22]: Tekstvoorstel: Verwerkers controleren hun eigen en elkaars machtiging.

Met opmerkingen [ER23R22]: prima

[14] 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.

[15] Nader onderzoek moet nog uitwijzen of alle scenario’s door OSR ondersteund kunnen (gaan) worden.

[16] Dus niet per definitie om persoonsgegevens

hebben zichzelf gemachtigd als verwerker[17]. Rationale hierbij is dat we een uniform interactiepatroon willen hebben.

10. Het functionele toepassingsgebied van Secure API profielen blijft ongewijzigd.
11. We gebruiken (voorlopig[18]) de internationale open standaard OAuth als vertrekpunt voor de ontwikkeling van het OAuth[19] profiel (dus niet NL GOV OAuth). Onze use case past het beste bij het OAuth client credentials en dit profiel zal als basis dienen.
12. Voor het Secure API OAuth profiel besluiten we (voorlopig) dat er meerder lokale Authorization servers zijn die door verschillende partijen beheerd worden. De AS en RS (en de beveiligde API's) staan zo in hetzelfde beveiligingsdomein.
13. De beheerder van een AS/RS moet ervoor zorgen dat de AS slechts een Access Token aan een client levert na verificatie bij het OSR dat de betreffende eindorganisatie de client gemachtigd heeft voor de betreffende uitwisseling.
14. Hoe een beheerder van de AS/RS de verificatie bij het OSR en de autorisatie door de AS door de uitgifte van een access token aan een bepaalde client levert, valt buiten de scope van de Edukoppeling afspraak.
15. Anders dan bij het huidige Secure API REST profiel wordt de routering bij het Secure API OAuth profiel opgenomen in een JWT token.

Met opmerkingen [EB24]: Is de implicatie dan dat een school ook het agentschap moet machtigen als verwerker als "derde partij" als genoemd bij punt 4 bij vertrouwelijke gegevens?

Met opmerkingen [ER25R24]: Ja, we hebben afgesproken dat er een generieke aanpak is, altijd machtiging registreren en verifiëren

Met opmerkingen [EB26]: Tekstvoorstel ipv besluiten we: "nemen we waar". Ik overzie namelijk niet de consequenties, technisch en/of voor de diverse beheerders.

Met opmerkingen [ER27R26]: Nee, uitgangspunten worden vastgesteld. We gaan uit van meerder lokale ASen. Tenzij we dus iets centraal doen maar dan is het meer een OSR implementatie

Met opmerkingen [KR28]: Vanuit de verschillende SaaS partijen kan ik me dit wel voorstellen. Maar worden deze AS-sen dan in de OSR opgenomen? Hoe weet ik anders waar ik ze kan bereiken? Kan de OSR ook niet voor bepaalde ketens als AS fungeren? Zeg maar een duale inrichting, soms decentraal, soms centraal? Dat zou ons veel gedoe schelen bij DUO denk ik

Met opmerkingen [ER29R28]: Begreep uit discussie dat we de machtiging (mandatering) centraal willen houden. De extra beveiliging met OAuth voor een API gebeurt met de lokale AS. En centrale AS kan, als OSR dat zou willen gaan ondersteunen. De RS en AS staan dan niet meer in hetzelfde beveiligingsdomein en vraagt dus wel specifieke kaders die anders zijn dan lokale

Met opmerkingen [EB30R28]: Goede vraag. De voordelen/nadelen van centraal/decentraal plus de technische en organisatorische implicaties dienen we daartoe mijns inziens in kaart te brengen. Ik zou graag donderdag bespreken hoe we dat kunnen insteken.

Met opmerkingen [ER31R28]: Vind dat prima, maar is wat mij betreft een basis voor uitwerking voor het EK profiel die we al eerder besloten hebben. Kortom, als het allemaal centraal via OSR gaat is het meer een implenentatietraject van OSR en niet een standaard die we Q1 2023 willen vaststellen

Met opmerkingen [EB32]: In OSR kan men "real time" de machtiging intrekken. Hoe zorgen we er dan voor dat uitgedeelde Oauth tokens dan niet leiden tot datalekken? Ook dit kan technisch/organisatorisch op verschillende manieren maar ik zou wel een regel willen toevoegen hierover.

Met opmerkingen [ER33R32]: Dit profiel is nog in ontwikkeling. Dit deze details zijn nog niet besproken. Lijkt mij logisch dat er een relatie bestaat tussen de twee. Als je bij OSR nu elk moment je machtiging kan intrekken lijkt mij dat ook niet werkbaar. Wellicht een dag geldig of langer maar is wat mij betreft later discussie als we hoog over consensus hebben

Met opmerkingen [KR34]: Deze snap ik niet. Moet denk ik verder uitgewerkt worden

Met opmerkingen [ER35R34]: Volgens mij is het niet wenselijk om in het OAuth profiel met een querystring voor routering te werken. De gedachte is dus opname in JWT, maar wellicht is deze al impliciet bekend en niet meer nodig. moet idd nog nader besproken worden wat nodig is en hoe we het inrichten

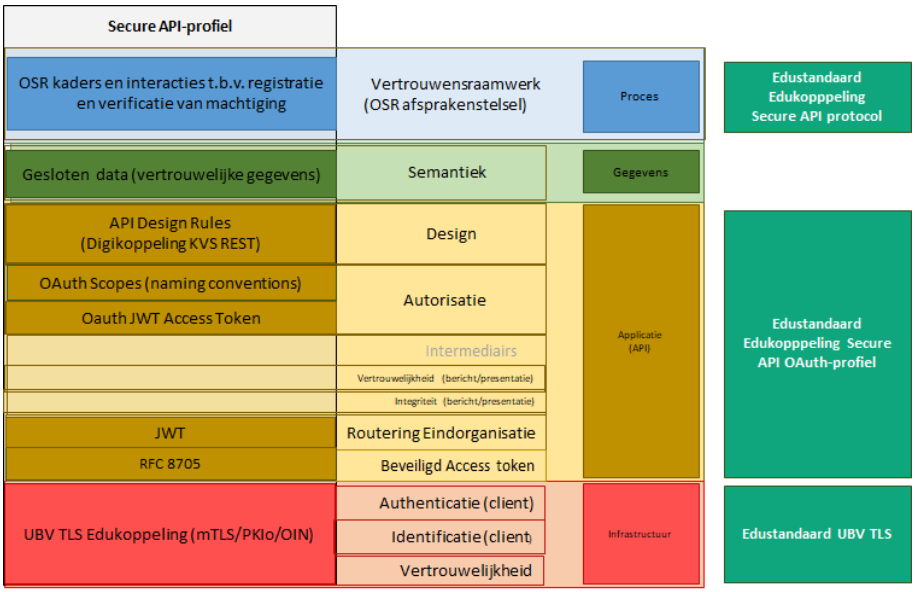
[17] Nader onderzoek moet nog uitwijzen of alle scenario's door OSR ondersteund kunnen (gaan) worden.
[18] Het Kennisplatform ontwikkelt nog OAuth profielen. Mochten we op een later moment kunnen aansluiten dan nemen we dat in overweging.
[19] https://datatracker.ietf.org/doc/html/rfc6749 (ook OAuth wordt doorontwikkeld: https://datatracker.ietf.org/doc/draft-ietf-oauth-v2-1/).

5. Nieuwe overwegingen

In dit hoofdstuk zijn nieuwe overwegingen opgenomen die we gebruiken om in een volgende versie tot uitgangspunten te komen. De overwegingen zijn ingedeeld naar het Edukoppeling informatie-uitwisselingsmodel.

Voorstel rond opbouw binnen het Edukoppeling Informatie-uitwisselingsmodel

In Figuur 3 wordt de nieuwe voorgestelde opbouw van het OAuth profiel weergegeven.



Figuur 3 – Oauth profiel binnen het Edukoppeling Informatie-uitwisselingsmodel

In Figuur 4 wordt schematisch weergegeven wat er in het Edukoppeling Secure API OAuth profiel en wat in het Edukoppeling Secure API protocol wordt opgenomen. Wat hierin op detailniveau wordt opgenomen is onderdeel van de lopende discussie.

Vooraf aan de uitwisseling moeten de verstrekker en afnemer elkaars machtiging verifiëren. Hoe verstrekkers de bevoegdheid voor een Access Token voor een bepaalde afnemer koppelen aan de verificatie van de machtiging is buiten scope van het profiel. Partijen richten dit op hun eigen manier. IT denk ik

Figuur 4 – Edukoppeling Secure API OAuth (client credentials profile) en Edukoppeling Secure API protocol

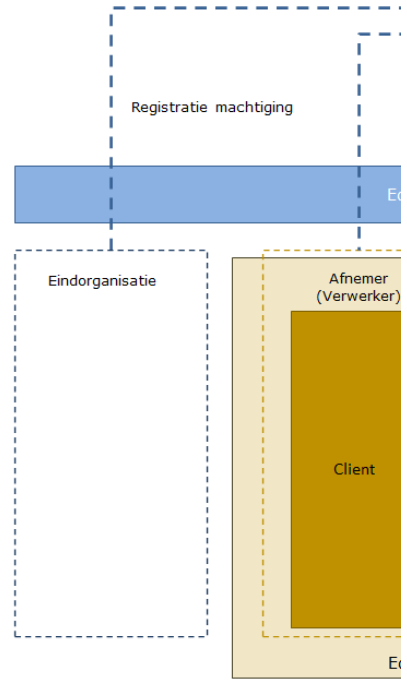
heeft verwijderd: Figuur 4

Met opmerkingen [KR36]: met de introductie van oauth introduceren we een soort asymmetrie. Ik bedoel daarmee het volgende: als een client een gegeven wil ophalen van een RS, moet de RS verifieren of dat mag (middels het uitgegeven token door de AS). Klaar. Echter als ik als client iets op wil sturen is het ook belangrijk dat ik als client RS wel geautoriseerd is om dit gegeven te mogen ontvangen als verwerken namens de formele eindpartij. Dit gaat vervolgens niet met oauth. Een post/put kent dus andere richtlijnen dan een GET. of zie ik dat verkeerd?

Met opmerkingen [ER37R36]: Klopt wat mij betreft. Vorige keer dit niet besproken, maar wat mij betreft ondersteund OAuth alleen een pull. We houden de OAuth rollen zuiver in het EK profiel. Heeft dus mogelijk ook impact op interactiepatronen. En zoals vorige keer besproken betekend dit mogelijk in een keten dat beide partijen een AS hebben en dat zij om en om de client rol hebben

heeft verwijderd: in.

heeft verwijderd: ¶



Stappen binnen het OAuth client credentials profiel:

1. Verkrijgen Access Token via OAuth 2 Access Token endpoint (A/B);

2. Ophalen Beveiligde resource met valide Access Token in request (C/D).

5.1. Applicatielaag

Voorstel A4: De wijzigingen die betrekking hebben op het OAuth profiel (de internationale open standaard) worden in het Engels geformuleerd.
Dit verduidelijkt de aanvulling die we binnen Edukoppeling willen opnemen en sluit ook aan hoe hier bij het Kennisplatform API's en Logius mee om wordt gegaan.

Voorstel A5: Als wijzigingen overeenkomen met die van het NL GOV OAuth profiel dan wordt dit expliciet aangegeven.
Een aantal wijzigingen zullen overeenkomen met die van het NL GOV OAuth profiel. Waar hier sprake van is wordt dit aangegeven met "<NLGOV>[20]". We doen dit voor hele tekstblokken. Daar waar we op dit niveau afwijken worden oude teksten doorgestreept en aanvullingen zijn een vet letter type. We houden zo overzicht waar we wel aansluiten.

Voorstel A6: Bij het client credentials profiel gaan we uit van confidential clients die een Access Token krijgen op basis van hun identiteit[21]
De client hoeft zich alleen te registreren bij AS zodat identificatie mogelijk is. Echter, onderdeel van het Edukoppeling Secure API profiel is verificatie van de machtiging vooraf aan afgifte van een Access Token. Binnen de Edukoppeling afspraak is dit geborgd in het Secure API protocol (onderdeel van de proceslaag).

De clients binnen het Edukoppeling OAuth client credentials grant profiel betreffen alleen confidential clients in de vorm van een web application[22].

Voorstel A7: Client identificatie op basis van OIN

---

[20] Gezien het NL GOV 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 <NLGOV> tag alleen iGOV teksten staan. Voor de leesbaarheid worden er niet geneste (iGOV/NLGOV) tags toegepast. Voor de NLGOV referenties wordt de volgende versie gebruikt: https://logius-standaarden.github.io/OAuth-NL-profiel/
[21] Dus niet op basis van toestemming door een gebruiker (Resource Owner).
[22] Zie clients profile bijlage A.

De verwerkers in de rol van verstrekker (in opdracht van eindorganisatie) zijn zelf verantwoordelijk voor het registreren van een client. Hiervoor dient in principe het OIN gebruikt te worden. Het OIN wordt nu binnen Edukoppeling en het OSR gebruikt.

Het kan zijn dat een fijnmazige identificatie van clients nodig is. Zolang het een lokale (decentrale) AS betreft kunnen AS beheerders er voor kiezen een client als onderdeel van een bepaalde verwerkersorganisatie (OIN) te identificeren op basis van bijvoorbeeld een GUID.

Voorstel A8: Client authenticatie op basis van mTLS
In OAuth (RFC6749) worden een aantal methoden voor client authenticatie[23] onderkend. Binnen het Secure API profielen wordt hiervoor mTLS toegepast. Zie keuze voor mTLS in infrastructuurlaag

Voorstel A9: Toepassing van Mutual-TLS Client Authentication and Certificate-Bound Access Tokens (RFC8705)
We stellen voor om RFC8705[24] toe te passen om het Access Token te beveiligen. We sluiten hiermee aan bij nieuwe ontwikkelingen rond OAuth[25].

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.

In NL GOV OAuth wordt er iets anders gekeken naar het koppelen van een bewijs van bezit aan een access token:
<NLGOV>Proof of possession can be implemented using various methods. An example of such an implementation is using TLS with mutual authentication, where the client is using a PKIoverheid certificate. The authorized party (azp) can then be verified with the client certificate to match the authorized party. As an alternative, the authorization server can include a cnf parameter in the JWT by the authorization server, see [rfc7800[26]]. The key referenced in cnf can be validated using a form of client authentication, e.g. using an private_key_jwt. </NLGOV>

---

[23] https://datatracker.ietf.org/doc/html/rfc6749#section-2.3
[24] https://datatracker.ietf.org/doc/rfc8705/
[25] 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]."*

[26] RFC 7800: Proof-of-Possession Key Semantics for JSON Web Tokens (JWTs) (rfc-editor.org)

Voorstel A10: Beveiliging best practices
De partijen die een OAuth rol implementeren worden gewezen op de 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] and similarly for the use of JSON Web Tokens [JSONWT-BP].

<NLGOV>
All servers *MUST* conform to applicable recommendations found in the Security Considerations sections of [rfc6749] and those found in the "OAuth Threat Model Document" [rfc6819].
</NLGOV>

6. Bijlage A: OAuth 2.0

Binnen de OAuth-standaard worden 4 rollen onderkend, de User (Resource Owner / RO), de Resource Server (RS), de Client en de Authorization Server (AS). OAuth introduceert een expliciete autorisatielaag tussen de RO en Client. Het gaat hierbij om zogenaamde gedelegeerde autorisatie. De RO bepaalt welke van de gegevens aan de client beschikbaar worden gesteld. Deze gegevens worden via beschermde API's ontsloten die zijn gehost op een bepaalde RS. De RO registreert hiervoor bevoegdheden bij de AS en de client kan via een door de AS geleverd toegangstoken toegang krijgen tot de gegevens van de API die binnen de betreffende bevoegdheden vallen. De RO vertrouwt de client dus voldoende dat deze over betreffende gegevens komt te beschikken.

6.1. Client typen

OAuth definieert twee typen clients. De indeling is gebaseerd op de mate waarin een betrouwbare authenticatie door de autorisatieserver mogelijk is. Het betreft hierbij in hoeverre ze in staat zijn hun credentials vertrouwelijkheid te bewaren. De OAuth-specificatie onderkent op basis hiervan Confidential clients en Public clients.

- Confidential
  Clients capable of maintaining the confidentiality of their credentials (e.g., client implemented on a secure server with restricted access to the client credentials), or capable of secure client authentication using other means.

- Public
  Clients incapable of maintaining the confidentiality of their credentials (e.g., clients executing on the device used by the resource owner, such as an installed native application or a web browser-based application), and incapable of secure client authentication via any other means.

De OAuth-specificatie onderkent verder verschillende client profielen die meer context geven bij een bepaald client type. De specificatie onderkent een web application, user-agent en een native application client profile.

- web application
  "*A web application is a confidential client running on a web server. Resource owners access the client via an HTML user interface rendered in a user-agent on the device used by the resource owner. The client credentials as well as any access token issued to the client are stored on the web server and are not exposed to or accessible by the resource owner.*"

- user-agent-based application
  "*A user-agent-based application is a public client in which the client code is downloaded from a web server and executes within a user-agent (e.g., web browser) on the device used by the resource owner. Protocol data and credentials are easily accessible (and often visible) to the resource owner. Since such applications reside within the user-agent, they can make seamless use of the user-agent capabilities when requesting authorization.*"

- native application
  "*A native application is a public client installed and executed on the device used by the resource owner. Protocol data and credentials are accessible to the resource owner. It is assumed that any client authentication credentials included in the application can be extracted. On the other hand, dynamically issued credentials such as access tokens or refresh tokens can receive an acceptable level of protection. At a minimum, these credentials are protected from hostile servers with which the application may interact. On some platforms, these credentials might be protected from other applications residing on the same device*."

6.2. OAuth 2.0 - Client credentials grant profile[27]

Het OAuth 2.0 profiel is het client credentials grant profiel is een profiel waarbij geen RO betrokken is. De client verkrijgt alleen een toegangstoken voor een resource van de Autorization Server via zijn eigen client credentials, in tegenstelling tot het aanvragen van toegang namens een RO. In de context van OAuth zou gesteld kunnen worden dat de client zelf de RO is: er is geen delegatie.

Binnen de context van de Edukoppeling Secure API profielen worden bevoegdheden impliciet uitgedrukt door het uitgeven van een Access Token. De bepaling van de bevoegdheden vallen buiten de scope van de OAuth standaard. De interacties binnen en buiten het OAuth profiel wordt schematisch weergegeven in Figuur 4[28]

De clients binnen het Edukoppeling OAuth client credentials grant profiel betreffen alleen confidential clients (web application profile clients).

6.3. RFC8705

De RFC8705 standaard bevat aanvullende beveiligingsmaatregelen voor de OAuth standard bij de toepassing van mTLS. mTLS ondersteund de client authenticatie[29]. De RFC beschrijft het hoe het mTLS client certificaat van het request gebonden kan worden aan het Access Token. Dit zorgt er voor dat de beveiligde API alleen gebruikt kan worden door een client met een aan het certificaat gebonden Access Token[30]. De client moet beschikken over de private sleutel van dat certificaat.

---

[27] Zie voor meer details https://datatracker.ietf.org/doc/html/rfc6749

[28] In Figuur 4 wordt ook de verificatie van de machtiging bij het OSR weergegeven. Dit is onderdeel van de Secure API proceslaag (geen onderdeel van de OAuth applicatielaag). Een access token wordt pas geleverd als er is vastgesteld dat voor de client een valide machtiging bestaat. Figuur 4

[29] A in stap 1 van Figuur 4

[30] C in stap 2 van Figuur 4

heeft verwijderd: Figuur 4

heeft verwijderd: Figuur 4

heeft verwijderd: Figuur 4

7. Bijlage B: Normatieve referenties en best practices

7.1. Normatief

[RFC6749]

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

[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*

[RFC8705]

OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens Campbell, B., Bradley, J., Sakimura, N., and T. Lodderstedt, " url:https://www.rfc-editor.org/rfc/rfc8705 Proposed Standard (February 2020). URL: RFC 8705 - OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens (ietf.org)

7.2. Best Practices

[OAUTH-SBP]

OAuth 2.0 Security Best Current Practice (draft-ietf-oauth-security-topics-11). IETF. December 28, 2019.
Best Current Practice. URL: https://datatracker.ietf.org/doc/draft-ietf-oauth-security-topics/

[JSONWT-BP]

JSON Web Token Best Current Practices (draft-ietf-oauth-jwt-bcp-04). IETF. November 08, 2019.
Best Current Practice. URL: https://datatracker.ietf.org/doc/draft-ietf-oauth-jwt-bcp/