Dienst Uitvoering Onderwijs
Ministerie van Onderwijs, Cultuur en Wetenschap

CONCEPT/~~VERTROUWELIJK~~/~~DEFINITIEF~~
Expertgroep OAuth

memo

OAuth2.0 protocol en mandaten

Directie
DFS-ICT-SWH

Afdeling
Facet

Contactpersoon
René Bosscher
Software Architect

Datum
29-10-2025

Bijlagen
nvt

Binnen het onderwijsveld is al enige tijd een wens ontstaan om gegevensuitwisseling tussen systemen te moderniseren. Om hier een invulling aan te geven, is deze memo geschreven. Deze memo kan gebruikt worden als input binnen de werkgroepen die hierbij betrokken zijn om samen tot een gedeelde 'best practice' te komen.

Om een goed afgewogen voorstel te kunnen maken, worden onderstaande onderwerpen in deze memo uitgewerkt:

1 Doel

Het doel van deze memo is om een voorstel uit te werken voor een nieuw generiek profiel ('best practice') t.b.v. gegevensuitwisseling tussen systemen onder het mandaat van een onderwijsinstelling.

De achtergrond waaruit de wens is ontstaan, wordt uit dit document weggelaten. Daar zit een business en een technisch belang achter. Dat is wat mij betreft niet waar deze memo voor bedoeld is.

Er zijn al initiatiefnemers geweest die een invulling hebben gegeven aan deze wens. De vraag daarbij is in hoeverre de al gemaakte stappen passen bij één gedeelde 'best practice' binnen het onderwijsveld. Een vervolgvraag is in hoeverre één gedeelde 'best practice' past voor alle partijen.

Per implementatie kunnen eventueel aanvullende details toegevoegd worden. Die aanvullende details zijn geen onderdeel van deze memo en dienen tussen de betrokken partijen voor die implementatie afgestemd te worden.

2 Gebruikte terminologie

In de communicatie tot op heden rond dit onderwerp merk ik dat men soms het zelfde bedoelt, maar anders benoemt. Ook merk ik dat men soms met een term verschillende dingen bedoelt. Door hier vooraf een paar definities te geven, hoop ik dat te voorkomen. Dat wil niet zeggen dat dit de juiste termen of uitleg daarvan zijn, maar dit is wel hoe ik de termen in dit document gebruik.

2.1 Mandaat

Deze term wordt al enige tijd binnen het onderwijsveld gebruikt. Een paar formele toelichtingen hierover:

- "In het Nederlands betekent "mandaat" een machtiging om namens een ander te handelen, vaak in een formele of officiële context";
- "Een mandaat kan een bevoegdheid zijn die een bestuursorgaan aan een ander orgaan of persoon verleent om namens hem te handelen";
- "Bij een mandaat blijft de oorspronkelijke bevoegdheidhouder (de mandaatgever) verantwoordelijk voor de uitoefening van de bevoegdheid, ook al wordt deze gemandateerd";
- "Mandaat verschilt van delegatie, waarbij de bevoegdheid wordt overgedragen en de delegerende partij de verantwoordelijkheid verliest".

Deze toelichtingen zijn letterlijk wat ik versta onder de term. Dat is voor mij de reden om deze term ook bewust in deze context te blijven gebruiken in de communicatie en documentatie.

2.2 LAS/SIS/TPS

Er wordt vaak gebruik gemaakt van de termen LAS, SIS en/of TPS. Deze termen verwijzen sterk naar een leverancier van een administratie-/planningsysteem. Ondanks dat dit wel hele relevante termen zijn, probeer ik in dit document zo veel mogelijk te schrijven over 'systemen' om het generiek te houden.
NB. De afkortingen staan voor:

- LAS: Leerling Administratie Systeem, term voor het PO/VO
- SIS: Student Informatie Systeem, term voor het MBO
- TPS: Toets Planning Systeem, term voor het MBO

2.3 Toetsaanbieder

Een platform waarmee toetsen worden aangeboden, gepland en afgenomen. Ondanks dat dit wel een hele relevante term is, probeer ik in dit document zo veel mogelijk te schrijven over 'systemen' om het generiek te houden.

2.4 Onderwijsinstelling

Verwijzend naar een erkenning van een organisatie voor het verzorgen van onderwijs binnen de context van een bevoegd gezag.

3 Uitgangspunten

Deze memo schrijf ik vanuit het scenario waarbij gegevensuitwisseling plaatsvindt van systeem-A (client) naar systeem-B (server) onder het mandaat van een onderwijsinstelling.

3.1 Actoren en systemen

In dit document worden de volgende actoren en systemen erkend:

- Onderwijsinstelling, een rechtspersoon welke verwerkingsverantwoordelijke en bevoegdheidhouder is;
- Medewerker onderwijsinstelling, een natuurlijk persoon met de bevoegdheid om namens de onderwijsinstelling gegevens aan te leveren;
- Systeem-A, een client die die een verzoek naar een server stuurt;
- Medewerker systeem-A, een natuurlijk persoon met de bevoegdheid om namens de organisatie achter systeem-A gegevens aan te leveren;
- Systeem-B, een server die een verzoek van een client ontvangt;
- Medewerker systeem-B, een natuurlijk persoon met de bevoegdheid om namens de organisatie achter systeem-B gegevens aan te leveren.

NB. De gegevensuitwisseling wordt in dit document in één richting uitgewerkt, maar een systeem kan de rol van systeem-A en -B tegelijk invullen voor een gegevensuitwisseling in twee richtingen.

3.2 Mandaatregister

We zouden nog een actor kunnen erkennen, namelijk een centraal mandaatregister. Maar in de communicatie t.b.v. de expertgroep OAuth is aangegeven de voorkeur te hebben voor een decentraal mandaatregister. Daarbij ligt de keuze waarop de mandaten beheerd worden ook bij de systeemeigenaar van systeem-B. Daarom neem ik verder in dit document het uitgangspunt aan dat systeem-B ook de actor voor het mandaatregister is. Die actor zal dus niet separaat uitgewerkt dan wel erkend worden, maar de businessfunctie voor een mandaatregister wel. Hoe dat mandaatregister geïmplementeerd wordt, is een detail dat buiten de scope van dit document valt.

3.3 Inhoudelijke gegevens

Middels het nieuwe profiel dat we aan het uitwerken zijn, willen ketenpartners inhoudelijke gegevens uitwisselen. Voor de implementatie dient een specificatie van het koppelvlak uitgewerkt te zijn. Die inhoudelijke specificatie (o.a. modelleren van de request/response berichten) valt buiten de scope van dit document. De logistiek, identificatie, authenticatie en autorisatie zijn wel in scope van dit document en komen in volgende hoofdstukken en paragraven terug.

3.4 RFC-8693 en RFC-6750

In dit document worden de RFC's 8693 (token delegation) en 6750 (bearer token) van de Internet Engineering Task Force (IETF) als uitgangspunt voorgesteld voor de toepassing van OAuth2.0 en dan met name betreffende token delegation, ook bekend als een 'On-Behalf-Of' flow.

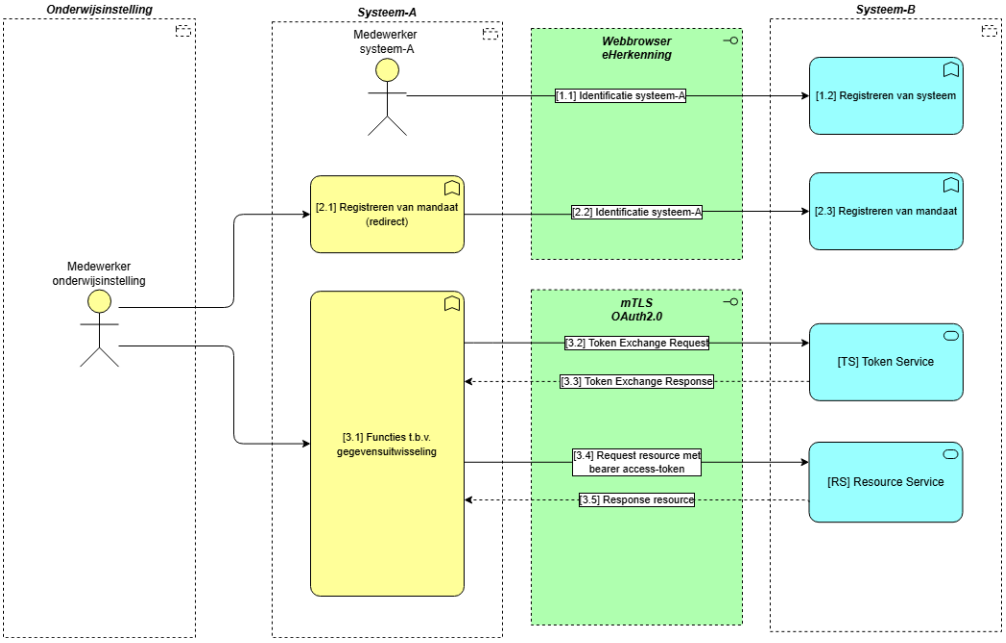
NB. Meer informatie en details over de RFC's via onderstaande links:

- https://www.rfc-editor.org/rfc/rfc8693.html#name-delegation-token-exchange-e
- https://www.rfc-editor.org/rfc/rfc6750.html#section-2

4 Scenario gegevensuitwisseling

Het scenario waarvoor we een nieuw profiel uitwerken wordt hieronder gevisualiseerd en toegelicht.

4.1 Schema



4.2 Toelichtingen

De toelichting op de processen worden onderverdeeld in twee categorieën.

Als eerste de eenmalige registratie-acties. Deze acties worden benoemd om het scenario volledig te maken, maar de details kunnen door de implementerende leveranciers zelf (anders) uitgewerkt worden.

Als tweede het OAuth2.0 koppelvlak t.b.v. de daadwerkelijke uitwisseling tussen twee systemen.

4.2.1 Eenmalige registraties

Om systeem-B een mandaatvalidatie te kunnen laten uitvoeren moet het aanroepende systeem-A bekend zijn bij systeem-B en moet er een mandaat geregistreerd zijn. Het registreren van die gegevens kan geïmplementeerd worden via een portaal van systeem-B welke beschikbaar wordt gemaakt via een webbrowser. Bij aanmelding via eHerkenning is bij systeem-B direct duidelijk namens welke organisatie de medewerker de registratie doet.

Dit resulteert in onderstaande processtappen.

4.2.1.1 Registreren identificatie systeem-A

Paragrafen 4.2.2 en 4.2.2.1 beschrijven het opzetten van een beveiligde bevinding en het uitgeven van een access token. Om die stappen mogelijk te maken, moet Systeem-A eerst identificerende informatie, zoals een client certificaat, een public key en een client_id registreren bij Systeem-B

Voor de registratie van systeem-A zou een medewerker van systeem-A (na aanmelding via eHerkenning) het client-certificaat kunnen uploaden bij systeem-B en laten koppelen aan een client_id (een OIN of een UUID). Dit wordt in 4.1 gevisualiseerd met "*[1.1] Identificatie systeem-A*".
Voor gegevensuitwisseling met de overheid zal het client-certificaat een PKIo certificaat moeten zijn met o.a. een OIN in de subject van het betreffende client-certificaat. Voor een

gegevensuitwisseling tussen twee commerciële systemen kan hiervoor een regulier PKI certificaat gebruikt worden.

4.2.1.2 Registreren mandaat systeem-A

In 4.2.2.1 wordt het uitgeven van een access-token beschreven. Daarbij moet m.b.v. een mandaat gevalideerd worden of een access-token daadwerkelijk uitgegeven mag worden.

Om te kunnen valideren of een access-token uitgegeven mag worden namens een onderwijsinstelling, dient het vereiste mandaat bekend te zijn bij systeem-B. Voor de registratie van het mandaat zou een medewerker onderwijsinstelling bij systeem-B (na aanmelding via eHerkenning) een mandaat kunnen registreren. Daarbij is middels eHerkenning-validatie de onderwijsinstelling bekend en kan de medewerker een keuze uit beschikbare systemen en de betreffende OAuth scopes maken om te mandateren.

Nog mooier is als de medewerker onderwijsinstelling via systeem-A doorgestuurd wordt naar systeem-B waarbij systeem-A het eerder geregistreerde client_id (zie 4.2.1.1), een return-url en optioneel een mandaat-uuid meestuurt. Daarmee hoeft de medewerker onderwijsinstelling het mandaat alleen nog maar te bevestigen. Het mandaat-uuid dient als referentie. Dit wordt in 4.1 gevisualiseerd met "*[2.2] Identificatie systeem-A*".

4.2.2 OAuth2.0

Het uitwisselen van gegevens dient middels OAuth2.0 plaats te vinden. Voor een uitwisseling met de overheid wordt een mTLS verbinding met PKIo certificaten vereist. Twee commerciële partijen kunnen daarvan afwijken, maar de uitwisseling dient ook daar via een beveiligde verbinding (bijvoorbeeld met TLS) plaats te vinden.

Over die (m)TLS verbinding worden gegevens uitgewisseld. Het modelleren van die berichten valt buiten scope van dit document.

Het protocol waarmee de berichten worden uitgewisseld dient via OAuth2.0 te gebeuren waarbij uit de claims van de tokens een onderscheid gemaakt kan worden tussen de gemandateerde partij (systeem-A) en de bevoegdheidhouder (onderwijsinstelling). Daarvoor kan 'Token delegation' uit RFC-8693 gebruikt worden.

Dit resulteert in onderstaande processtappen.

4.2.2.1 Uitwisselen tokens

In RFC-8693 wordt 'Delegation Token Exchange' toegelicht met voorbeelden. Hieronder een samenvatting van de stappen met een vertaling naar ons scenario.

Samenstelling JSON Web Token
In onderstaande teksten wordt geschreven over het gebruik van JSON Web Tokens (JWT's). Voor de leesbaarheid nog een korte toelichting over de samenstelling van een JWT.

Een JWT bestaat uit drie delen, namelijk een 'header', 'payload' en een 'signature'. Elk deel is middels Base64 gecodeerd.

Header
De header wordt gevuld met bijvoorbeeld het gebruikte algoritme van de signature. Dat kan er uit zien als:

- {"alg":"RS512"}

Payload
De payload bevat de content waar het met name om gaat. In onderstaande paragrafen wordt geschreven over een JWT claims set. De claims set staat in de payload van de betreffende JWT.

Signature
De signature van een JWT is een versleutelde hash. De versleuteling wordt gedaan met een private key, een sleutel die alleen bij de verzender bekend is. Een ontvangende server kan de signature ontsleutelen met de (reeds bekende) public key van de verzender. Het resultaat is de hash die is berekend door de verzender. De ontsleutelde hash kan vergeleken worden met een nieuw berekende hash. De hash waardes moeten dan overeenkomen. Hiermee wordt de integriteit van de payload vastgesteld (de hashes moeten hetzelfde zijn) en wie het ondertekend heeft (de signature kan alleen versleuteld zijn met de private key van de verzender). Dit betekent nog niet dat de server ook zeker weet dat dit vanuit systeem-A is verstuurd, daarvoor is de mTLS nodig.

Request
Het aanvragen van een access-token staat in 4.1 gevisualiseerd met "*[3.2] Token Exchange Request*".

A.2.1. Token Exchange Request (zie link voor voorbeelden)
Systeem-A vraagt een access-token aan bij de token-service van systeem-B middels een 'token exchange request'. Dit is een POST op het token-endpoint van systeem-B met daarin de onderstaande request-parameters (voorbeeld):

request-parameter	Waarde ('unescaped')
grant_type	urn:ietf:params:oauth:grant-type:token-exchange
audience	urn:example:cooperation-context
subject_token	*JWT-encoded, zie 'A.2.2. Subject Token Claims'*
subject_token_type	urn:ietf:params:oauth:token-type:jwt
actor_token	*JWT-encoded, zie 'A.2.3. Actor Token Claims'*
actor_token_type	urn:ietf:params:oauth:token-type:jwt

Met name 'subject_token' en 'actor_token' zijn hierin interessant om nader te beschrijven. Dit zijn request-parameters met een JWT claims set. De 'subject_token' en 'actor_token' worden hieronder separaat beschreven.

A.2.2. Subject Token Claims (zie link voor voorbeelden)
De 'subject_token' betreft een JWT claims set. De JWT bevat onder andere claims betreffende het subject. De belangrijksten zijn:

- De 'sub' (subject) claim betreft de partij namens wie (on behalf of whom) een access token wordt aangevraagd. Dat is de onderwijsinstelling, bij voorkeur geïdentificeerd met een OIN.
- De 'scope' (toegangsbereik) claim betreft een indicatie voor welke data-onderdelen toegang gevraagd wordt. N.B.: dit kunnen meerdere zijn, gescheiden met een spatie.

NB. *Het gebruik van een subject token welke door systeem-A wordt meegestuurd (en dus ondertekend) is wel wat discutabel te noemen. Het subject token zou eigenlijk door de subject zelf aangeleverd/ondertekend moeten zijn.*

A.2.3. Actor Token Claims (zie link voor voorbeelden)
De 'actor_token' betreft een JWT claims set. De JWT bevat onder andere claims betreffende de actor. De belangrijkste is:

- De 'sub' (subject) claim betreft de partij die de access-token daadwerkelijk gaat gebruiken. Dat is systeem-A, geïdentificeerd met een client_id welke in 4.2.1.1 is beschreven.

Response
Het retourneren van een access-token staat in 4.1 gevisualiseerd met "*[3.3] Token Exchange Response*".

A.1.3. Token Exchange Response (zie link voor voorbeelden)
In RFC-8693 worden meerdere voorbeelden gegeven voor een Token Exchange Response. In A.2.4. Token Exchange Response staat zo'n voorbeeld uitgewerkt, maar daarbij wordt een token_type "N_A" gebruikt. Wij willen graag een rechtstreeks te gebruiken access-token hebben, en daarvoor moet token_type "Bearer" worden gebruikt. Daarom is A.1.3. Token Exchange Response een beter voorbeeld welke ik voorstel te gebruiken.

Systeem-B stuurt als antwoord op een Token Exchange Request, een Token Exchange Response retour naar systeem-A met daarin de onderstaande attributen (voorbeeld):

Attribuut	Waarde ('unescaped')
access_token	JWT-encoded, zie 'A.2.5. Issued Token Claims'
issued_token_type	urn:ietf:params:oauth:token-type:access_token
token_type	Bearer
expires_in	Het aantal seconden dat het access-token geldig is, default 3600s

Met name 'access_token' is hierin interessant om nader te beschrijven. Dit is een veld in het response met een JWT claims set. De 'access_token' wordt hieronder separaat beschreven als 'A.2.5. Issued Token Claims'.

A.2.5. Issued Token Claims (zie link voor voorbeelden)
De 'access_token' betreft een JWT claims set. De JWT bevat claims betreffende de (toegekende) toegang. De belangrijksten zijn:
- De 'act' (actor) claim betreft de partij die de access-token daadwerkelijk gaat gebruiken. Dat is systeem-A, geïdentificeerd met een client_id welke in 4.2.1.1 is beschreven. Dit moet overeenkomen met de subject in de actor-token.
- De 'sub' (subject) claim betreft de partij namens wie (on behalf of whom) de token is uitgegeven. Dat is de onderwijsinstelling, bij voorkeur geïdentificeerd met een OIN.
- De 'scope' (toegangsbereik) claim betreft een indicatie voor welke data-onderdelen toegang toegekend is met dit access-token. NB. Dit kunnen meerderen zijn, gescheiden met een spatie.

4.2.2.2 Uitwisselen resources

Het in 4.2.2.1 beschreven access-token zou als bearer token gebruikt moeten worden voor het uitwisselen van resources. In '7.1. Access Token Types' van RFC-6749 wordt een mooi voorbeeld gegeven (als verwijzing naar RFC-6750) over het gebruik van een bearer token.

Hieronder een samenvatting van de stappen met een vertaling naar ons scenario.

Request
Het request van een resource staat in 4.1 gevisualiseerd met "*[3.4] Request resource met bearer access-token*". Het request wordt gedaan door een HTTP request te sturen naar een endpoint van een resource-service met o.a. de header 'Authorization'. De waarde van die header is "Bearer <access-token>".

De resource-service valideert o.a. het onderstaande bij ontvangst van het request:
- Of de geldigheid ('expires_in') van het access-token nog niet is verlopen;
- Of de ondertekening van het access-token geldig is (zie ook de uitleg over signature in 4.2.2.1);
- Of resource onderdeel is van 'scope' claim in access-token;
- Of gevraagde/geleverde gegevens behoren tot de 'sub' claim (onderwijsinstelling) in access-token.

Response
Als alle validatie stappen van het request positief zijn, dan wordt het request verwerkt en een response geretourneerd. Het retourneren van het response wordt in 4.1 gevisualiseerd met "*[3.5] Response resource*".