TOP-KT-023 - Identity Provisioning

Versiegeschiedenis...

Versie	Datum	Status	Wijzigingen
1.0.2	15 jan 2026	definitief	Multiple IdP Support als subpagina toegevoegd aan topic 23. & IAM Eisen uitgebreid met Multiple IdP eisen.
1.0.1.	11 Feb 2025	definitief	Onder de combinatie van de configuratie van de Identity Provisioning is aan het gebruikerstype de RelatedPerson toegevoegd. Verder is de tekst aangepast waardoor de identifier voor mapping ook een RelatedPerson.identifier.system mag zijn. Een typefout is gerepareerd van SAMRT on FHIR naar SMART on FHIR. Ook is de Naaste opgenomen bij de paragraaf over IdP per User type en de kapotte link is gewijzigd naar AB.017.
1.0.0	21 Jun 2023	definitief	

Beschrijving

De koppeltaal launch betreft de transitie van een portal applicatie naar een module van een user agent, dat laatste is meestal webbrowser. Binnen koppeltaal zijn de applicatie geauthenticeerd en geautoriseerd door middel van SMART on FHIR backend services. De SMART on FHIR app launch kent een authenticatiestap waar de user agent geauthenticeerd moet worden. In koppeltaal 1.x werd er in deze stap niets ondernomen. In Koppeltaal 2.0 wordt in deze stap de user agent wel geauthenticeerd. Dit gebeurt door middel van een Identity Provider (IdP) die binnen het domein beschikbaar is. Koppeltaal 2.0 ondersteunt daarnaast het gebruik van meerdere IdP's binnen één domein (Multiple IdP Support). De keuze voor de juiste IdP wordt bepaald op basis van de `idp_hint` in het HTI-token, conform de Koppeltaal 2.0 Multiple IdP-specificatie.

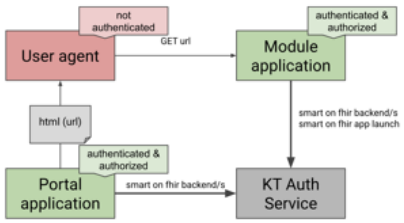
Overwegingen

Autorisatie en SSO

De Identity Provisioning is onderdeel van de koppeltaal launch. In de launch is het noodzakelijk de user agent (webbrowser) te identificeren in het launch proces. Zie ook TOP-KT-007 - Koppeltaal Launch. Het uitgangspunt is dat de gebruiker zich in het

domein (nogmaals) moet identificeren om van de module die gestart wordt gebruik te maken. Om dit te doen dient in het domein een Identity Provider beschikbaar te zijn. Hoewel Koppeltaal niet bepaald welke Identity Provider dit moet zijn en hoe deze exact moet werken, geeft Koppeltaal 2.0 wel aan dát er een Identity Provider moet zijn. Door de Identity Provisioning ter hoogte van de autorisatie service te beleggen wordt deze taak voor zowel de portalapplicatieleverancier als de moduleapplicatie leverancier geabstraheerd. Indien de IdP de SSO methode goed implementeert blijven de cookies op de user agent actief en hoeft de gebruiker niet nogmaals in te loggen.

"Het diagram hieronder licht het probleem toe. In een launchescenario zijn alle systemen geauthenticeerd en geautoriseerd, met uitzondering van de user agent (webbrowser). Om deze te authenticeren is de autorisatiestap noodzakelijk. Zie voor meer uitleg AB.017 - Authenticatie van de gebruiker tijdens de launch .



Omdat Koppeltaal 2.0 meerdere IdP's binnen één domein ondersteunt, bevat het HTI-token een `idp_hint`. Deze hint geeft aan welke IdP gebruikt moet worden voor de authenticatie van de gebruiker. De autorisatieservice valideert of deze IdP is toegestaan binnen het domein en gebruikt vervolgens de bijbehorende configuratie. In " TOP-KT-023 a-Multiple IdP Support voor Patient, Practitioner & RelatedPerson | Oplossingsbeschrijving " wordt het gebruik van Multiple IdP Support verder toegelicht.

Aansluiting bij standaarden en raamwerken

Om te voldoen aan de eIDAS-verordening, de Wet Digitale Overheid (WDO) en de toekomstige normen voor toegang tot online diensten, zoals de applicaties binnen een Koppeltaal-domein, onderzoeken we nu al hoe we in de toekomst kunnen aansluiten op IdP-diensten met erkende middelen op eIDAS-niveau 'hoog'. In een dergelijk toekomstig scenario moet het mogelijk zijn om via Federated Identity Management (FIM) als Koppeltaal IdP te vertrouwen op een erkende IdP. Om dit toekomstige scenario mogelijk te maken, hebben we ervoor gekozen om de IdP te baseren op Open Id Connect (OIDC)

Afhankelijk van de inhoud

De vraag of de identificatie door middel van een IdP noodzakelijk is, is afhankelijk van de vraag of de module persoonsgegevens en/of medische gegevens verwerkt. Verwerken de modules van een applicatie-instantie geen gegevens over de gebruiker (bijvoorbeeld door alleen media en tekst aan te bieden), dan verwerkt deze geen persoonsgegevens of medische gegevens. In dergelijk geval is er geen noodzaak voor een extra authenticatie van de user agent. Niet in alle gevallen is er dus een noodzaak voor een IdP. De aanbieder van de module moet communiceren met de domeinbeheerder of de module persoonsgegevens en/of medische gegevens verwerkt. Koppeltaal doet wél de observatie dat er een noodzaak is binnen alle domeinen zoals deze op dit moment bekend zijn.

IdP en SSO standaarden

Koppeltaal kiest enkel voor de OpenID Connect standaard (OIDC) om met de Identity Provider te communiceren. Hoewel Security Assertion Markup Language(SAML) tevens een gangbare standaard is, gaat de voorkeur primair uit naar OIDC. Dit om de implementatie eenvoudig te houden.

IdP per user type

In het architectuurbesluit AB.017 - Authenticatie van de gebruiker tijdens de launch is opgenomen dat de configuratie van de IdentityProvider op de combinatie domein, applicatie-instantie en user type (Patient, Practioner en RelatedPerson) plaatsvindt. Het moet dus mogelijk zijn om per user type een andere IdP in een domein toe te wijzen. Met de introductie van Multiple IdP Support in Koppeltaal 2.0 kan de autorisatieservice bovendien op basis van de `idp_hint` in het HTI-token bepalen welke IdP gebruikt moet worden. Dit maakt het mogelijk om niet alleen per user type, maar ook per launch-context of per applicatie-instantie dynamisch de juiste IdP te selecteren. Dit is essentieel in domeinen waar medewerkers, cliënten en naasten verschillende IdP's gebruiken. In TOP-KT-023a-Multiple IdP Support voor Patient, Practitioner & RelatedPerson | Oplossingsbeschrijving wordt verder uitgewerkt hoe Multiple IdP Support wordt toegepast in de Koppeltaal Standaard.

Dit is noodzakelijk omdat het vaak voorkomt dat medewerkers in een domein een andere IdP gebruiken dan de cliënten en zijn/haar Naasten.

AuditEvent logging

Zoals alle interacties van de gebruiker met koppeltaal 2.0 moet ook de authenticatiepoging gelogd worden als AuditEvent, ongeacht van de uitkomst van de poging. De authorisatie service moet een AuditEvent aanmaken van het type 110114, User Authentication. De outcome is de plaats om aan te geven of het succesvol is gebleken of niet. Zie ook TOP-KT-011 - Logging en tracing voor het specifieke AuditEvent.

Toepassing, restricties en eisen

Toepassing IdP binnen een domein

Binnen een domein moet op basis van de modules die worden ingezet bepaald worden of een IdP noodzakelijk is. Of een IpP vereist is, kan ingesteld worden in de combinatie domein - applicatieinstantie - gebruikerstype. Is er sprake van een heel klein domein met modules die geen persoonsgegevens of medische gegevens verwerken, is er geen IdP noodzakelijk. De WDO is het raamwerk waarnaar Koppeltaal verwijst om te bepalen of dit van toepassing is. Het geldt tevens voor het middel wat de IdP implementeert. Ook hiervoor verwijzen we naar de WDO om te bepalen of het authenticatiemiddel voldoende is.

Configuratie in het domein

De configuratie van de Identity Provider (IdP) vindt plaats in het domeinbeheer, de authorization service maakt gebruik van deze configuratie om tijdens de SMART on FHIR app launch procedure de user agent op de juiste manier te authenticeren. De configuratie van de IdP gebeurt op de combinatie domein - applicatieinstantie - gebruikerstype.

Op de combinatie van de volgende drie kenmerken wordt de configuratie van de Identity Provisioning vastgesteld:

1. Het domein
2. Het gebruikerstype (Patient/Practitioner/RelatedPerson)
3. De applicatie-instantie

Afhankelijk van de implementatie kan er gewerkt worden met defaults, zo is het logisch per domein - gebruikerstype een default in te kunnen stellen, en deze dan per applicatie optioneel te kunnen configureren. Dit om de complexiteit van de configuratie terug te brengen.

Op dit niveau wordt vastgesteld:

- Of een IdP vereist is, zie ook de overweging afhankelijk van de inhoud.

- Zo ja,
  - welk type (vooralsnog enkel OIDC), indien er meerdere typen in de toekomst worden gebruikt kan dit veld meervoudig worden ;
  - welk veld het identifier attribuut bevat, hiermee wordt geconfigureerd welke attribuut van de user-assertion wordt gebruikt;
  - voor welk type FHIR resource deze IdP is (Patient/Practitioner/RelatedPerson),
  - op welke identifier van de FHIR resource de identiteit gemapped is (Patient.identifier.system/Practitioner.identifier.system/RelatedPerson.identifier.system);
  - de kenmerken van de Identity Provider (URL, public key/JWKS URL etc.).

Het is de verantwoordelijkheid van het domeinbeheer deze configuratie mogelijk te maken, de authrosiatieservice maakt van deze configuratie gebruik om de SMART on FHIR app launch uit te voeren.

Integratie in het domein

De IdP speelt een rol in de autoriseren van de gebruiker in de SMART on FHIR app launch, daar wordt in de authenticatie (/authenticate) stap de beslissing tot authenticatie met het IdP genomen. Daarnaast staat het natuurlijk verschillende applicaties in het domein te adviseren gebruik te maken van dezelfde Identity Provisioning, aangezien dit het Single Sign On (SSO) effect in het domein sterker maakt. Het volgende onderdeel zet deze stappen uiteen.

Uitvoer van de Identity Provisioning stappen

De stappen van de OIDC flow worden in de specificaties in detail besproken, samenvattend komen ze op de volgende neer:

- De autorisatie service ontvangt een authenticatieverzoek in de SMART on FHIR app launch flow met als launch parameter het HTI token, pakt het token uit en zoekt de configuratie op.
- De autorisatie service service bereidt een authenticatieverzoek voor met daarin de gewenste verzoekparameters op basis van de configuratie en de referentie van de gebruiker in het HTI token.
- De autorisatie service stuurt het verzoek naar de IdP door middel van een redirect (302).
- IdP verifieert de user agent en de gebruiker.
- IdP verkrijgt toestemming/autorisatie van de gebruiker.
- IdP stuurt de user agent terug naar de autorisatie service met een code door middel van een redirect (302).
- De autorisatie service vraagt een antwoord met behulp van de code op het token endpoint van de IdP.
- De autorisatie service ontvangt een antwoord met een ID-token.
- De autorisatie service valideert het ID-token en haalt de juiste attribuut van de gebruiker op, op basis van de configuratie.
- De autorisatie service haalt de juiste FHIR resource op, en valideert of de gebruiker bestaat, actief is en overeenkomt met de referentie van de gebruiker in het HTI token.
- De autorisatie service maakt een AuditEvent aan van het type 110114 en sub_type 110122. Zie ook TOP-KT-011 - Logging en tracing voor het specifieke AuditEvent.
- De autorisatie service vervolgt de SMART on FHIR app launch flow met als geïdentificeerde gebruiker de FHIR resource uit de vorige stap.

Gebruikte standaarden

OpenID Connect OIDC

Wet Digitale Overheid (WDO)

Voorbeelden