



Delegation high level design

09-02-2026

Hoofdstuk 1
Context

Hoofdstuk 2
Requirements

Hoofdstuk 3
High level Design

Hoofdstuk 4
Vervolgstappen

Hoofdstuk 1

Context

Overzicht FSC proces

Eenmalig, tot expiration date



Publiceren in directory

- Partijen
- Services

Eenmalig, tot expiration date



Contracten tekenen

- Aanbieder
- Afnemer (eventueel via delegatee)

Zo lang contract & certificaat geldig is



Informatie uitwisselen

- Afnemer (delegatee)
- Aanbieder

Waarom delegatie oplossing

- ▶ Doel: Gegevens die worden uitgewisseld met een intermediair, worden alleen uitgewisseld met toestemming van zowel de aanbieder als de afnemer van de service.
- ▶ Risico: informatie uitwisselen met een intermediair die hiervoor geen geldige toestemming heeft
- ▶ Oplossing: contract door alle partijen laten ondertekenen en bij uitgeven token valideren

Ondertekenen | FSC delegatie

1. Gebaseerd op certificaten en gestructureerd document: **FSC Contract**
2. Alle partijen hebben FSC manager nodig om contract te ondertekenen
3. Cryptografische basis
4. Validatie van contract (aanvraag token) is gebaseerd op certificaat dat wordt gebruikt om het moment van token aanvraag
 1. Nieuwe handtekening nodig op het moment dat een accept signature is gezet met een certificaat dat verloopt voordat het contract verloopt

Situaties

1 Delegatee (consumer)

Organisatie A biedt API-1 aan als provider. Organisatie C wil deze gebruiken met behulp van Organisatie D.

Directory

Partij	Service
Organisatie A	API-1
Organisatie C	
Organisatie D	

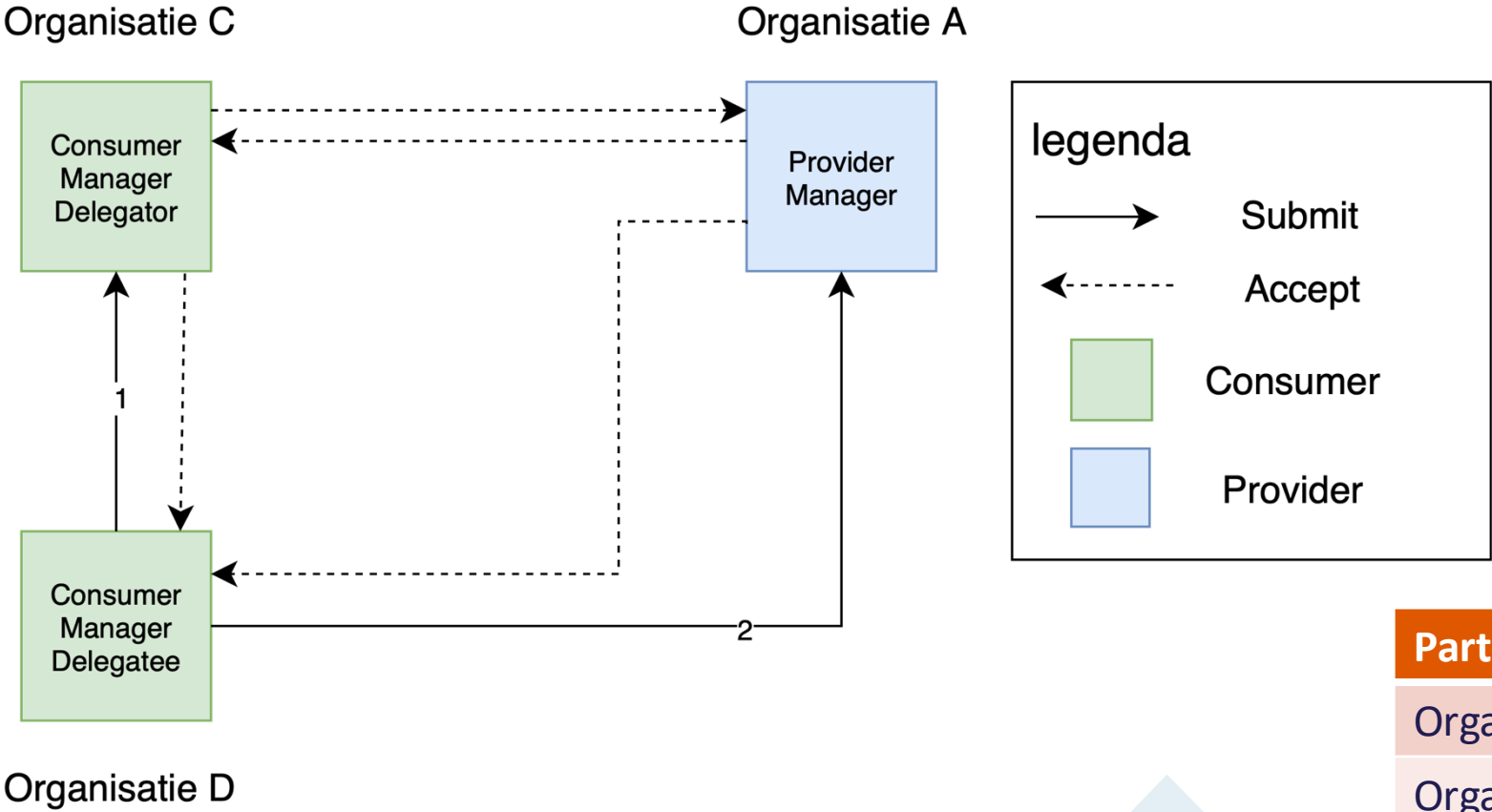
2 Delegatee

Organisatie A biedt met behulp van Organisatie B API-1 aan. Organisatie C wil deze gebruiken m.b.v. Organisatie D.

Directory

Partij	Service	Aangeboden door
Organisatie B	API-1	Organisatie A
Organisatie C		
Organisatie D		

Huidige FSC oplossing

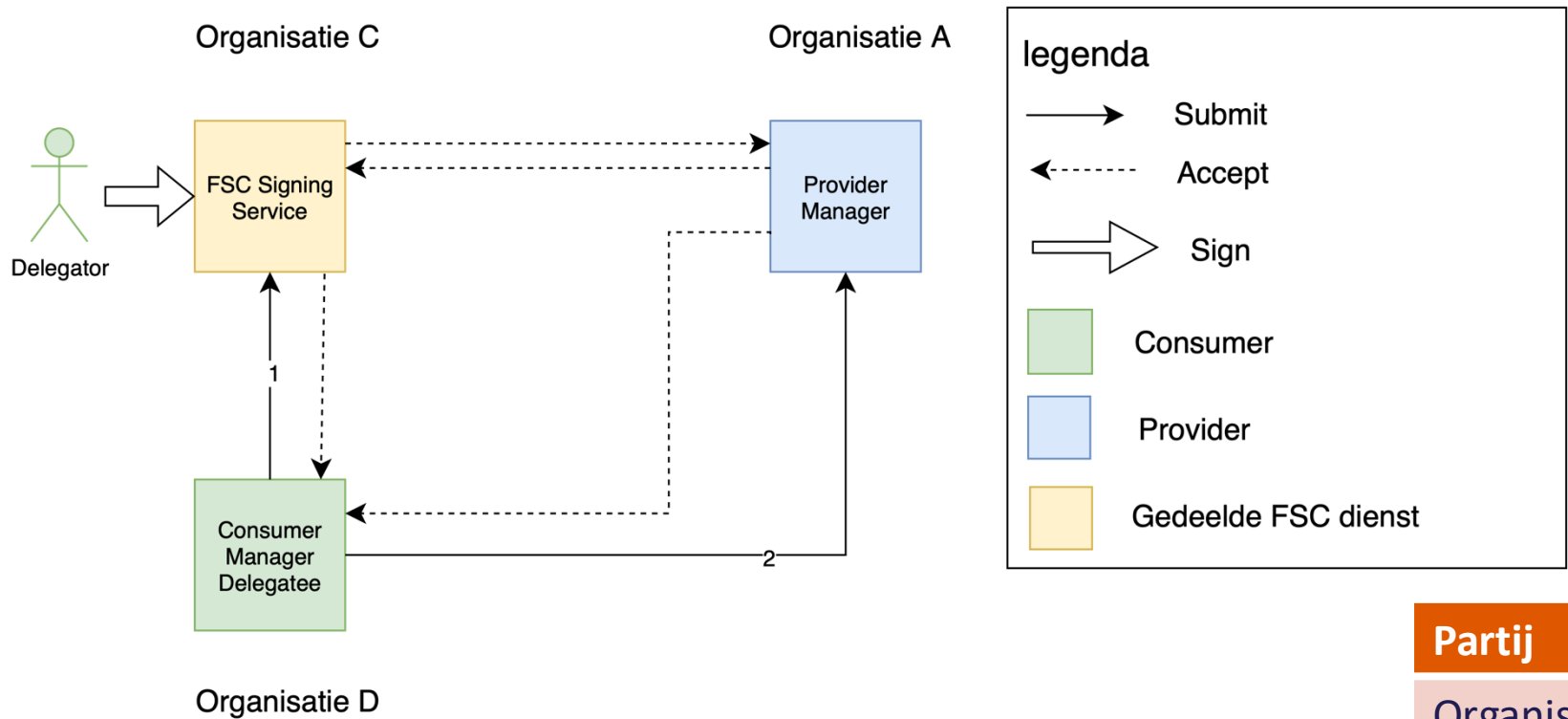


Partij	Service
Organisatie A	API-1
Organisatie C	
Organisatie D	

Welk probleem veroorzaakt het

- ▶ Partijen hebben zelf geen kennis en ervaring
 - ▷ Om de FSC manager te installeren en beheren
 - ▷ Geen certificatenbeheer (denk bijvoorbeeld aan scholen), of geen Digikoppeling certificaten (denk bijvoorbeeld aan notarissen)
- ▶ Vertraging omdat het vernieuwen van certificaten en beheren van de FSC componenten bij partijen wordt gelegd die hiervoor geen kennis en menskracht hebben
- ▶ Technische beperkingen bij het uitwisselen over netwerken heen.

Oplossing: de FSC Signing Service



Partij	Service
Organisatie A	API-1
Organisatie C	
Organisatie D	

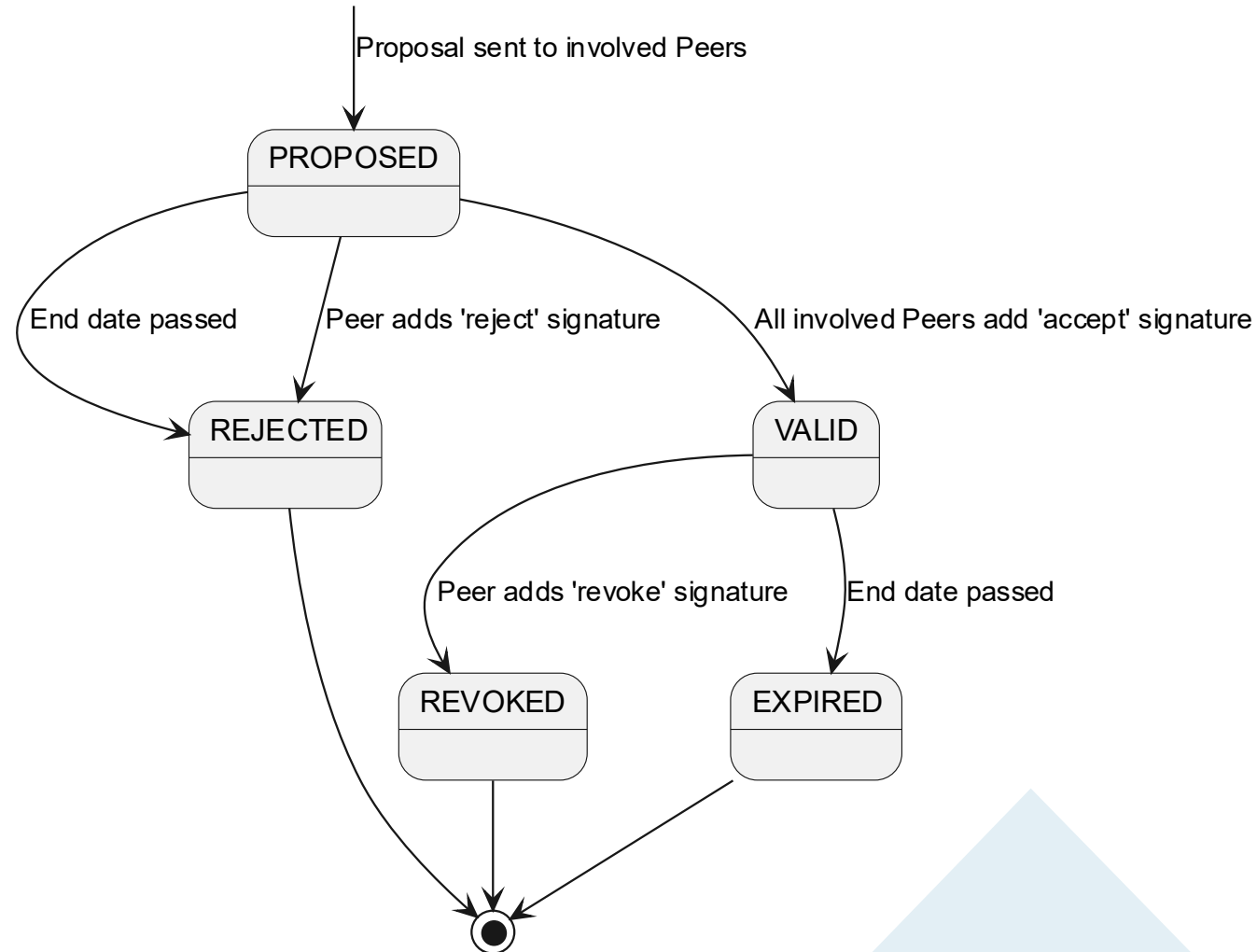
Requirements

Eisen aan de oplossing

- ▶ Intrekken, accepteren en weigeren van contracten moet mogelijk zijn via een user interface
- ▶ Multi-tenant oplossing die gehost kan worden, waarbij inzichtelijk is dat er gebruik is gemaakt van de signing service (audit)
- ▶ Signing service kan bewijzen dat er was ingelogd bij de trusted third party voordat de handtekening werd geplaatst.
- ▶ Partijen op het contract kunnen dit bewijs opvragen
- ▶ Login via een trusted third party, die kan garandeert dat de persoon een medewerker is van de organisatie namens wie getekenend wordt
- ▶ Meerdere trusted third parties moeten geconfigureerd worden
 - ▷ eHerkenning / eIDAS
 - ▷ ... ?
- ▶ Het identificerende nummer moet configureerbaar zijn (ook organisaties zonder OIN, denk aan scholen of notariskantoren).

High level design

Contract statuses conform FSC (huidig)



Overzicht FSC proces

Eenmalig, tot expiration date



Publiceren in directory

- Partijen
- Services

Eenmalig, tot expiration date



Contracten tekenen

- Aanbieder
- Afnemer (eventueel via delegatee)

Zo lang contract & certificaat geldig is



Informatie uitwisselen

- Afnemer (delegatee)
- Aanbieder

Proces publiceren in de directory (optioneel)

1. Provider kondigt zichzelf aan in de directory
2. Provider publiceert API in de directory (optioneel)
3. Signing service kondigt zichzelf aan in de directory
4. Leveranciers kondigen zichzelf aan in de directory

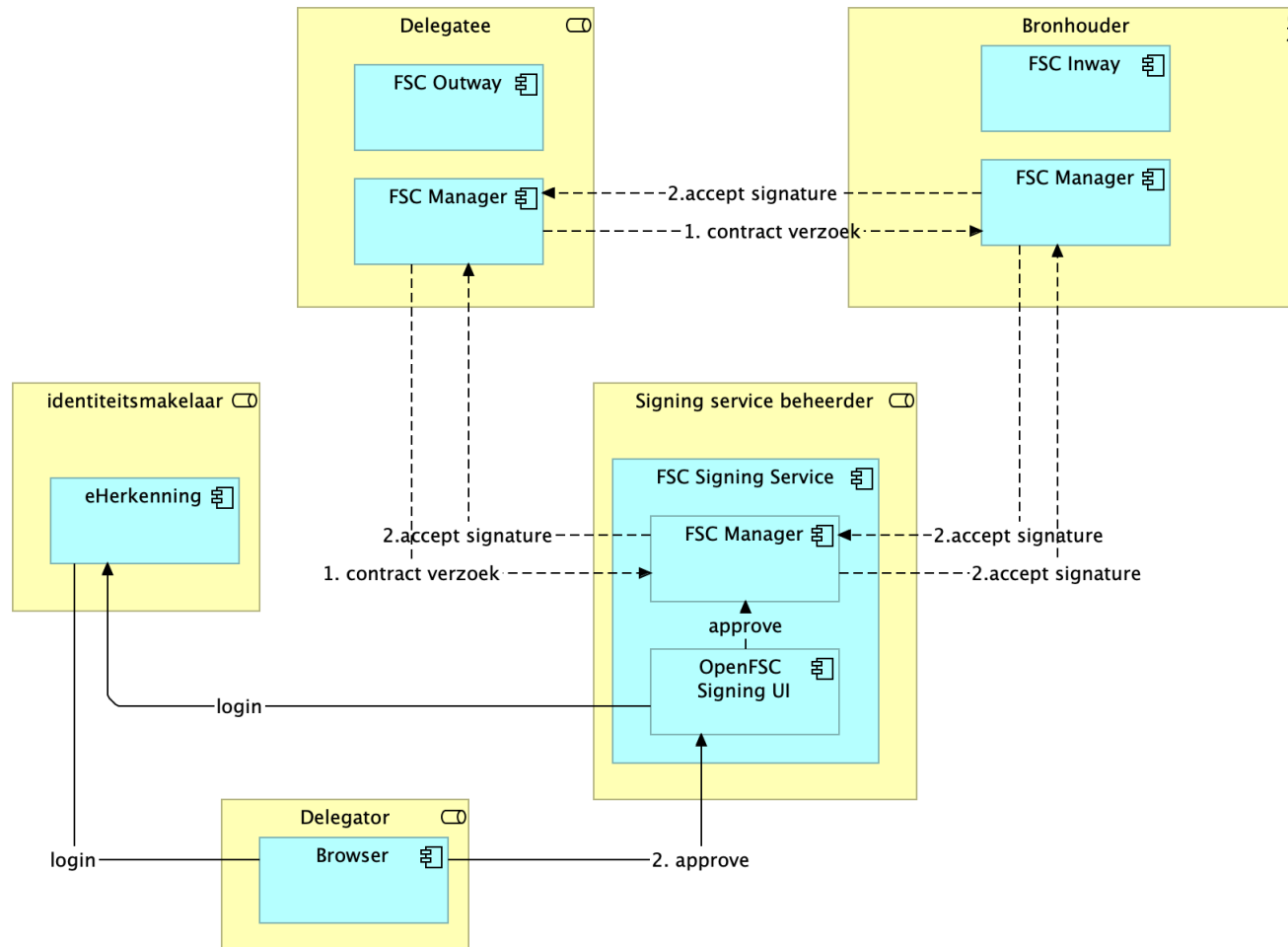
Contracten ondertekenen

1. Leverancier X stuurt een contract verzoek voor de API naar provider en de Signing Service
2. Bronhouder ondertekent contract
3. Medewerker van organisatieA logt in bij de Signing service via eHerkenning
4. Medewerker van organisatie A keurt het contract goed
5. Sessie en approval wordt gelogd
6. Signing Service stuurt approval handtekening met daarin informatie over de delegator naar provider en leverancier

Informatie uitwisselen

1. Leverancier X vraagt informatie token op namens organisatie A bij FSC Manager Provider
2. FSC Manager Bronhouder controleert contract
3. FSC Manager Bronhouder geeft token uit

Contract ondertekenen – goedkeuring contract

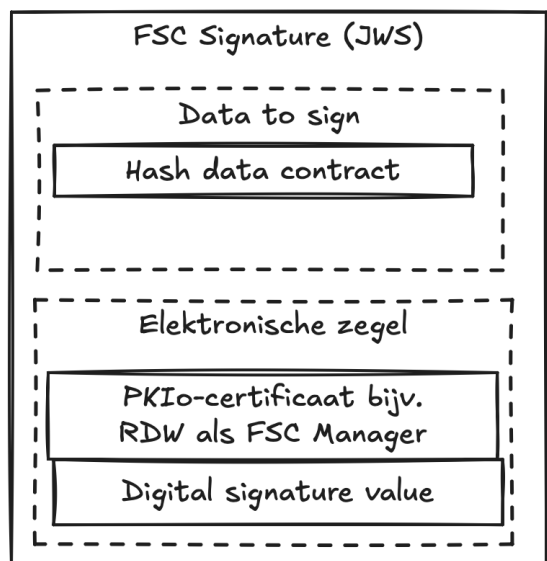


Vragen.

1. Kunnen we een eIDAS/eHerkenning signature gebruiken of tekent de manager van de signing service en breiden we het delegator object uit?

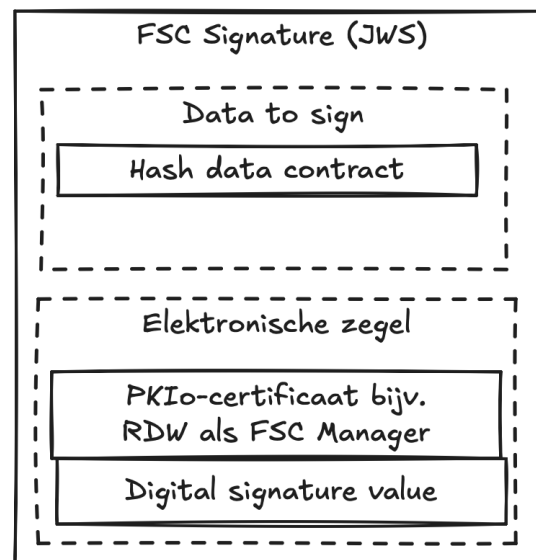
Signature

via eigen FSC Manager, bijv. RDW



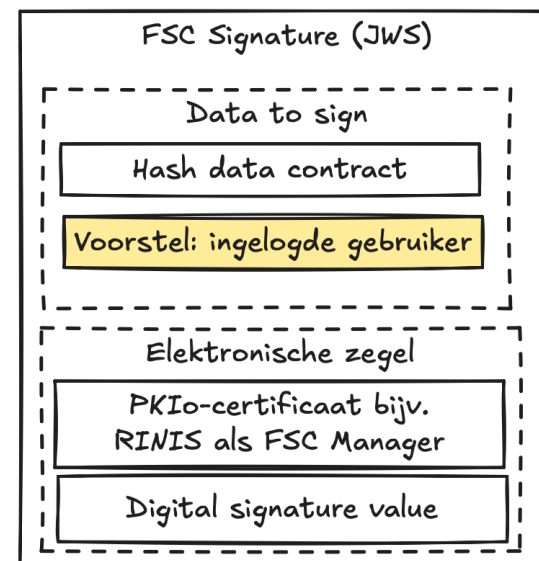
Provider

via eigen FSC Manager, bijv. Leverancier



Delegatee

via multi-tenant FSC Manager,
gemeente X (signing service)



Delegator

advies Sander:
pas FSC aan: gebruik
JAdES-standaard voor
elektronische zegels

Veranderingen huidige implementatie

Open FSC

1. Signing service UI (uitgekleedde controller) realiseren
2. Multi-tenant FSC manager
3. Aanpassing delegator object in signature
4. Koppeling met eHerkenning en andere inlog middelen
5. Opvraagbare audit trail (API) voor sessie plus approval

FSC standaard

1. Aanpassen van de core standaard (nog te analyseren)
2. Aanpassen van de transaction log extension (nog te analyseren)
3. Extensie toevoegen (? Nog te analyseren)
4. Optioneel: gap analyse jADES standaard

Vervolgstappen

Aanpak

High level design

OpenFSC

FSC standaard

Implementatie
& beheer

Vervolgstappen (1/2)

1. Validatie van het high level design
 1. Validatie projectgroep rijbewijsregister BOA's ✓
 2. Validatie Uniforme bronontsluiting (Cleverbase) ✓
 3. Validatie FDS ✓
 4. Validatie FSC subwerkgroep
2. Open FSC
 1. *User interface maken voor FSC Signing service (multi-tenant versie van controller)*
 2. Configuratie toevoegen om te koppelen met authenticatiemiddelen
 3. Release maken

Vervolgstappen (2/2)

3. *FSC standaard*
 1. *Impact analyse op FSC standaard*
 2. Extensie definiëren of bestaande standaard aanpassen
 3. Tekst opstellen
 4. Bespreken in FSC werkgroep van het TO
 5. Bespreken in TO, publieke consultatie, programmeringstafel en programmeringsraad GDI (3 maanden)
4. Implementatie van de Signing service
 1. Installatie van de Signing Service
 2. Koppelen met eHerkenning
 3. Onboarding van de partijen via het programma
 4. In beheername

Vragen/opmerkingen

- ▶ Login via een trusted third party, die kan garanderen dat de persoon een medewerker bevoegd is om namens de organisatie te tekenen. FSC wilt niet weten dat de medewerker werkzaam is bij de organisatie maar dat de medewerker tekenbevoegd is.
- ▶ Duidelijk formuleren waarom we meerdere trusted third parties willen toelaten. Dit is omdat FSC wil dat de standaard niet vast zit aan een bepaalde trusted third party. Een signing service is vrij om een trusted third party te kiezen (E.g. E-Herkenning, eIDAS).
- ▶ De volgorde van tekenen maakt voor FSC niet uit. De volgorde van tekenen in de presentatie is een suggestie, geen verplichting.
- ▶ Wil FSC ondersteuning bieden voor meerdere lagen van delegatie? Voorbeeld: Een gemeente machtigt een leverancier om een bron te bevragen en de leverancier schakelt sub-aannemer om de bevraging te doen. Werkgroep sessie organiseren. Vooraf usecases verzamelen. Ivo Lodewijks heeft een usecase.
- ▶ In het plaatje van de signature (Die samen met Sander is gemaakt) tekst aanpassen "ingelogde gebruiker" -> "bewijs tekenbevoegdheid"
- ▶ De signing service tekent de contracten met een PKIO certificaat. Wat gebeurt er met de handtekeningen als het certificaat van de signing service vervangen wordt? Wordt alles opnieuw getekend? Of moeten alle delegators weer opnieuw inloggen?
- ▶ Wat gebeurt er met een handtekening als de tekenbevoegdheid wordt ingetrokken? Niks. Volgens de werkgroep heeft dit geen gevolgen voor de handtekening. Juridisch gezien is het zo dat de handtekening nog steeds geldig is.