

# Welkom

API security

**We beginnen om 14.50 uur**

# OpenID Connect – hoe was het ook al weer? -

- OpenID connect is een modern protocol uit 2014
- OAuth2 (een autorisatieprotocol) met een identity laag er bovenop
- Gebaseerd op moderne standaarden als JSON en REST
- Ook geschikt voor authenticatieflows in mobiele apps
- In SURFconext alleen nog voor SPs, niet voor IDP's

# OpenID Connect & OAuth2 – Rollen

- **OP:** OpenID Connect Provider: Server die zorgt voor de afhandeling van authenticatie en het uitdelen van “tokens”
- **RP:** Relying Party: De applicatie die een gebruiker wil authenticeren (mobiel, native of web)
- **Resource Server:** Een API met gegevens. Kan door de RP worden bevraagd *namens* de gebruiker
- **Resource Owner:** De eigenaar van de gegevens achter de resource server.

# SURFconext OpenID Connect

- **Nieuwe implementatie** in 2019 (in gebruik genomen in 2020, migratie net afgerond)
- **Alle features van SURFconext** zoals autorisatie, statistieken, attribuut aggregatie en groepen via SURFconext teams, alleen geen eduGAIN
- Veel **OIDC features** worden ondersteund (PKCE, implicit, hybrid en code flow, 'claims requested')
- **API security:** Ook resource servers kunnen nu worden gebruikt
- **Consent op scopes:** Logo en begeleidende tekst van de API kunnen worden getoond
- Selfservice opvoeren van Relying Parties en Resource Servers via het **SP Dashboard**
- **Playground applicatie:** <https://oidc-playground.test.surfconext.nl>

# SURFconext API security

- **API** wordt ook wel **Resource Server** genoemd
- OIDC kan gebruikt worden om een API te beveiligen
- API krijgt *dezelfde claims* en *identiteiten* als de Relying Party
- De API behoort tot dezelfde service als de Relying Party: Ze vallen onder hetzelfde contract

# Waarom SURFconext OIDC voor je API?

- OIDC (of OAuth2) is *de* defacto standaard voor authenticatie bij mobiele apps
- Invloed op de ontwikkelingen
- Alle voordelen van SURFconext (een koppeling, centrale features zoals autorisatie en statistieken)
- Veel geavanceerde features zoals bijvoorbeeld PKCE
- Ingebouwde autorisatie: Introspectie is alleen mogelijk als een Resource Server gekoppeld is aan de Relying Party waar het token aan is uitgedeeld.
- Mogelijkheid tot sterke authenticatie met SURFsecureID
- Geen afhankelijkheid van een commerciële cloud provider
- Integratie met andere SURF initiatieven, zoals studentmobiliteit en de OOAPI



# API security 2.0

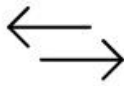
- Ondersteuning van **scopes** en **consent op scopes**
- Ondersteuning voor lang levende tokens (met **refresh\_tokens**)
- Een **interface** voor gebruikers om deze tokens in te kunnen trekken

# Scopes en consent op scopes

- Een scope geeft een access\_token een *doel* (bijvoorbeeld “cijferinformatie” of “roosterinfo”)
- SURFconext laat een scherm zien waarop de gebruiker wordt geïnformeerd over deze scope (“de applicatie Studentenapp wil graag cijferinformatie ophalen”)
- Inhoud van de tekst wordt door de RP bepaald



Do you give permission to share your information?



## Grant Playground Client permission

You have given your consent to the disclosure of certain personal information earlier. The service you are currently using however also needs to exchange **additional** data. You must give your consent for this separately.



### Group memberships

All your group memberships

Allow

No, I don't agree

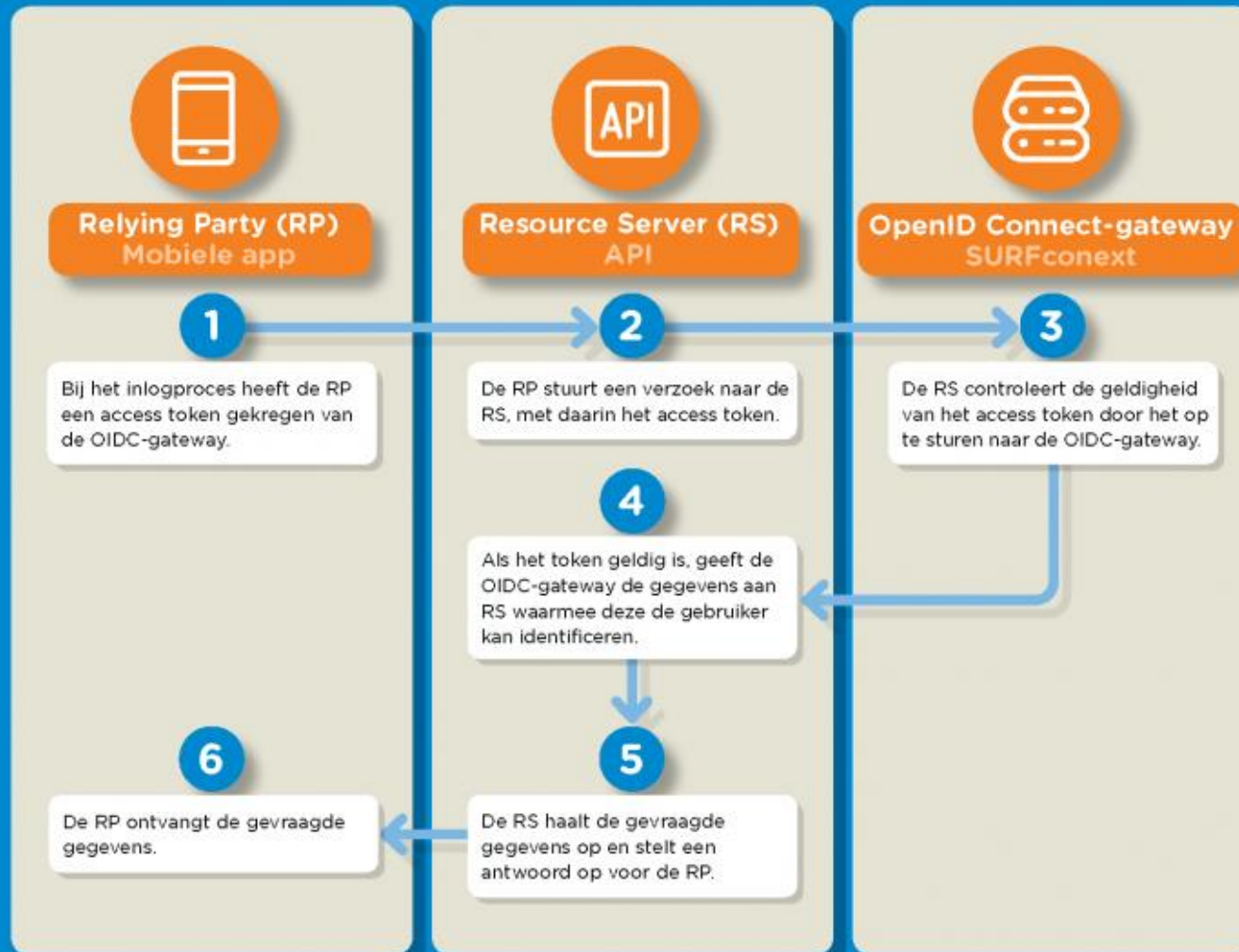
NL | EN

# Hoe werkt het API security technisch?

- De **RP** (mobiele app of webclient) krijgt nadat de gebruiker is ingelogd, een `access_token`. Deze heeft een beperkte geldigheidsduur
- Bij het aanvragen van dat token *kan* een RP vragen om een **scope**, bijvoorbeeld “mijnapi.univanharderwijk.nl/cijfers”
- Indien ingesteld, toont SURFconext een *consent* scherm met het logo en naam van de API, en de beschrijving van de scope (bijvoorbeeld: Je cijferinformatie kan worden opgehaald)
- De **RP** doet nu een request naar de **API** en gebruikt het `access_token` om te authentifieren
- De **API** doet een *introspect* call naar SURFconext
- SURFconext vertelt de API of het token geldig is, en geeft identifiers en attributen terug van de gebruiker
- De **API** weet nu wie de gebruiker is en kan de juiste gegevens aan de RP leveren



# Gegevens opvragen bij een API die met OIDC is beveiligd



# Een instelling aan het woord

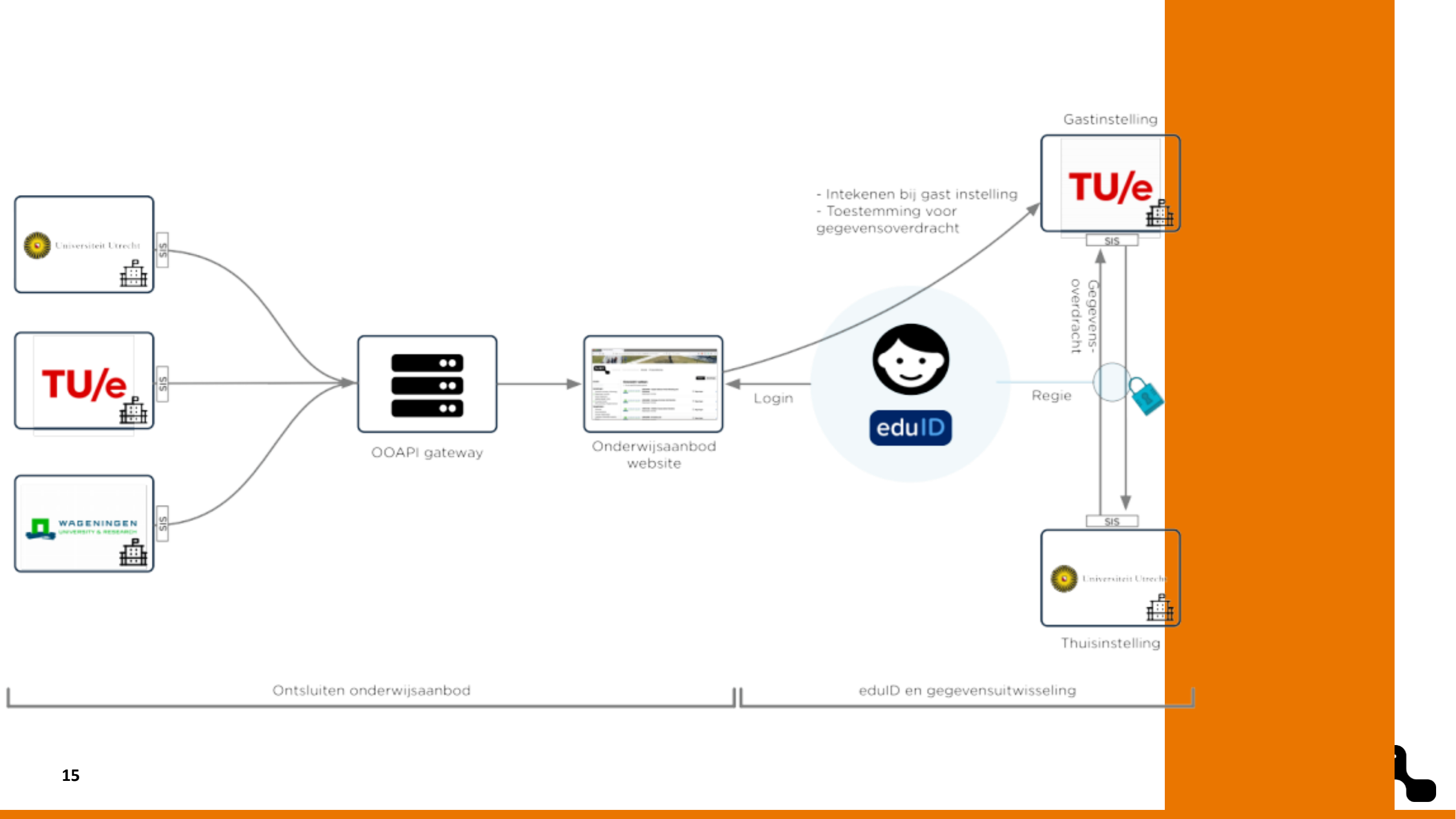
**Gerben Meuleman**

Project digitale Campus



## Use case 2: Studentmobiliteit

- Studenten kunnen makkelijk inschrijven bij andere instellingen
- De “gastinstelling” verkrijgt een access\_token, de “thuisinstelling” heeft een API waar gegevens van de student opgehaald kunnen worden
- SURFconext API security levert de tokens en informatie



# Use case 3: Studentenapps

- Een aantal instellingen, mede gedreven door studentmobiliteit, zijn studentenapps aan het bouwen
- Informatie van deze apps komen uit APIs
- Veel instellingen nog zoekende: API gateway van \$leverancier met bijbehorend Identity management systeem?
- API security van SURFconext kan hierbij helpen.

# En verder?

- Wil je aan de slag met APIs en SURFconext? Neem even contact op!
- Zelf spelen? <https://oidc-playground.test.surfconext.nl>



**OOK AANSLUITEN?  
NEEM CONTACT OP!**



**Bart Geesink**



**E-mail: [bart.geesink@surf.nl](mailto:bart.geesink@surf.nl)**



**[www.surf.nl](http://www.surf.nl)**

**Driving innovation together**