



Datadelen via DjustConnect DATA-AANBIEDER

Om veilig datadelen, mét toestemming van het landbouwbedrijf, op DjustConnect mogelijk te maken, zijn er enkele stappen en randvoorwaarden om data via een API te kunnen aanleveren:

- Enkel HTTP GET requests worden ondersteund
- Een server SSL certificaat, bvb. van Letsencrypt, is vereist
- Het inkomende SSL client certificaat van DjustConnect valideren of OAuth2 authenticatie voorzien
- API-documentatie onder de vorm van een Open API Spec (Swagger file)
- Als een landbouwbedrijf toestemming moet verlenen:
 - o Een extra endpoint aanmaken: *farm-id*
 - o Zich ervan verzekeren dat elke endpoint als request parameter of in de response body een landbouwbedrijf identificatie bevat

Contents

1	API ontwikkeling	2
1.1	Farm-id endpoint.....	3
1.2	OpenAPI Spec.....	4
2	Configuratie in DjustConnect	4
2.1	Partner Details.....	4
2.2	API Configureren.....	5
2.3	Toestemming verlenen.....	6
3	Partner API (optioneel)	6
4	Push API (Events)	6
4.1	Pushmelding Configureren.....	7
4.2	Pushmelding uitsturen.....	8



1 API ontwikkeling

Enkel HTTP GET requests worden op dit moment ondersteund door DjustConnect. Om er zich van te verzekeren dat enkel DjustConnect toegang krijgt tot de data van de aangeboden API zijn er twee authenticatie methodes voorzien: enerzijds mutual SSL waarbij enkel calls gemaakt met een bepaald SSL certificaat toegestaan worden of anderzijds via oauth2 authenticatie. Optioneel kunnen ook nog extra HTTP header parameters verplicht worden gemaakt. Deze moeten dan wel medegedeeld worden aan de datagebruikers opdat de gebruikers deze headers kunnen toevoegen.

Om het inkomend DjustConnect SSL certificaat te valideren in een Asp.NET Core C# applicatie draaiende in IIS moeten volgende stappen uitgevoerd worden:

1. Web applicatie toevoegen aan IIS, domein configureren, SSL server certificaat toevoegen en https binding configureren
2. In het IIS *Features* paneel, ga naar *SSL Settings*, *Require SSL* aanzetten en *Client certificates* op *Accept* zetten
3. In C# (zie onderstaande code)
 - a. Implementeer de *IAuthorization* interface als een *RequireCertificateAttribute*
 - b. Gebruik *[RequireClientCertificate]* als attribute voor elke controller klasse

```
public class RequireClientCertificateAttribute : Attribute, IAuthorizationFilter
{
    private const string DJUSTCONNECTCERT_THUMB =
"CEB6DA224A27FC0376464633261452050723F24C";
    public void OnAuthorization(AuthorizationFilterContext context)
    {
        X509Certificate2 cert =
context.HttpContext.Connection.ClientCertificate;
        if (cert == null)
        {
            context.Result = new UnauthorizedResult();
        }
        else if (!cert.Verify() || !IsValid(cert))
        {
            context.Result = new StatusCodeResult(403);
        }
    }

    private bool IsValid(X509Certificate2 cert)
    {
        return string.Equals(cert.Thumbprint, DJUSTCONNECTCERT_THUMB,
StringComparison.InvariantCultureIgnoreCase);
    }
}
```

1.1 Farm-id endpoint

Om te weten te komen welke landbouwbedrijven hun toestemming moeten geven voor het gebruik van de aangeleverde data, is er een aparte endpoint nodig die DjustConnect op regelmatige basis kan aanroepen. We raden aan om deze endpoint uit te sluiten van de Open API Spec van de aangeleverde API.

Het contract waaraan de *farm-id* endpoint moet voldoen is het volgende:

- POST request met als body een JSON-object met volgende properties:
 - o *ResourceUrl*: volledige url van de endpoint in kwestie
 - o *FarmIds*: ids van de landbouwbedrijven waarvoor DjustConnect wenst te weten of de endpoint hiervoor data ter beschikking heeft. Deze is leeg wanneer *All* gelijk is aan True
 - o *All*: geeft aan of één van de datagebruikers data voor alle beschikbare landbouwbedrijven wenst op te halen. Wanneer dit True is dan wordt als resultaat de volledige lijst van beschikbare landbouwbedrijf nummers verwacht als antwoord.
- Response bestaande uit een JSON-lijst van landbouwbedrijfnummers

Voorbeeld code voor een *farm-id* endpoint:

```
[ApiExplorerSettings(IgnoreApi = true)]
[RequireClientCertificate]
[Route("api/[controller]")]
[ApiController]
public class FarmIdController : ControllerBase
{
    IDatabaseContext _context;

    public FarmIdController(IDatabaseContext context)
    {
        _context = context;
    }
    [HttpPost]
    [ProducesResponseType(typeof(IEnumerable<string>)), (int)HttpStatusCode.OK]
    public IActionResult Post([FromBody]FarmIdRequest req)
    {
        var resultset = new HashSet<string>(_context.Farms.Select(f => f.Kbo));
        if (req.All)
        {
            return new JsonResult(resultset);
        }
        else
        {
            return new JsonResult(resultset.Intersect(req.FarmIds));
        }
    }
}
public class FarmIdRequest
{
    public string ResourceUrl { get; set; }
    public IEnumerable<string> FarmIds { get; set; }
    public bool All { get; set; }
}
```

1.2 OpenAPI Spec

Om de datagebruikers te kunnen voorzien van voldoende informatie over de aangeleverde API wordt er een OpenAPI Spec bestand, ook wel gekend als een Swagger file, verwacht. In het geval van .NET kan deze gegenereerd worden door het gebruik van NSwag. Voor het gebruik hiervan verwijzen we graag naar de [online documentatie](#). De informatie aanwezig in de OpenAPI Spec wordt weergegeven in het developer portaal (<https://portal.djustconnect.be>).

2 Configuratie in DjustConnect

2.1 Partner Details

Hier vindt u de gegevens van de geselecteerde partner.

Name
Test Provider

Purpose
Provider for testing

Application Url

Primaire sleutel
.....  

Secundaire sleutel
.....  

Certificaat 

Thumbprint huidig certificaat: null

Opslaan

Aan te leveren informatie voor het tabblad *Partner details*, deze informatie wordt geconfigureerd door een DjustConnect beheerder:

- Naam van de dataleverancier
- Purpose: Bijkomende informatie over de dataleverancier
- Url van de dataleverancier

Indien de optioneel te gebruiken partner API van DjustConnect ook aangesproken wordt, dan kan op het tabblad *Partner details* de *Primaire sleutel* die bij elke API call meegegeven moet worden als een HTTP header met sleutel: *DjustConnect-Subscription-Key* terug gevonden worden en het SSL certificaat (.cer bestand) geconfigureerd worden.

2.2 API Configureren

Om een API te configureren moet er eerst algemene informatie ingegeven worden:

- API Naam
- API Url: basis url van de API bv. <https://wa-sample-api.azurewebsites.net/>
- OpenAPI Spec URL: URL naar de OpenAPI JSON Specificatie bv. <https://wa-sample-api.azurewebsites.net/swagger/v1/swagger.json>
- Url naar de *farm-id* endpoint: bv. <https://wa-sample-api.azurewebsites.net/FarmId>

Nieuwe API

API Naam *

De naam van uw API

API URL *

De basis-URL van uw API

Open API Spec URL

De locatie van de Open API-specificatie waarin uw API wordt uitgelegd

Landbouwbedrijf ID API URL

URL van de landbouwbedrijf ID API endpoint die de Datahub zal gebruiken om de aanwezigheid van bepaalde landbouwbedrijf IDs na te gaan

Annuleren Nieuwe API aanmaken

Vervolgens moet er gekozen worden tussen SSL Certificaat Authenticatie en OAuth2 authenticatie. Bij OAuth2 Authenticatie moeten er extra parameters geconfigureerd worden. Bij SSL Certificaat Authenticatie moet zich men ervan verzekeren dat men het DjustConnect certificaat valideert.

Naast het suggereren van enkele tags waaronder de API terug te vinden zal zijn in het Developer portaal van DjustConnect (<https://developer.djustconnect.be/>), is de laatste en misschien wel belangrijkste stap het configureren van de verschillende endpoints van de API.

Voor elke van de databronnen moet er aangegeven worden:

- Welke toestemming er nodig is voor het gebruik van de databron:
 - o Geen toestemming nodig
 - o Enkel van de dataleverancier
 - o Zowel van de dataleverancier als van het landbouwbedrijf
- Het type landbouwidentificator dat gebruikt wordt (KBO, Leveraarsnummer, ...)
- Locatie van de landbouwidentificator:
 - o Request parameter: als de ID zich in de request url bevindt dan geef je de naam van de parameter zoals deze gekend is in de OpenAPI Spec
 - o JSON-pad: als de ID's zich in de body van de JSON-response bevinden dan moeten er twee parameters ingevuld worden:
 - JSON-pad naar het object: Als de response een lijst van JSON-objecten is, dan is het JSON-pad: \$
 - JSON-pad naar Landbouwbedrijf-ID: Als het response object in de lijst bv. een *kbo* property heeft dan is het JSON-pad: *\$.kbo*
- Beschrijving van de databron

Indien er JSON-paden gedefinieerd moesten worden dan kunnen deze uitgetest worden op <https://jsonpath.com/>.

Na indiening voor goedkeuring zal een DjustConnect beheerder de API publiceren zodat de datagebruikers toestemming kunnen beginnen vragen om de verschillende databronnen binnen de API te beginnen gebruiken.

2.3 Toestemming verlenen

Wanneer datagebruikers de gepubliceerde databronnen waarvoor toestemming van de dataleveranciers nodig is, wensen te gebruiken dan komen deze aanvragen terecht in het [Toegangsbeheer](#).

Geef per databron aan of u de data wilt delen. De data worden enkel gedeeld met de aanvrager voor het doel van de aanvraag

Aanvrager	Doel	Databron	Aangevraagd op	Indatatum	Alle landbouwbedrijven	Status	Review datum	Acties
Test Consumer	Consumer for testing	/api/FarmData/complex - GET	03-08-2020		Nee	Goedgekeurd	03-08-2020	✓ ✗
Test Consumer	Consumer for testing	/api/FarmData/simple - GET	03-08-2020		Nee	Goedgekeurd	03-08-2020	✓ ✗
Test Consumer	Consumer for testing	/api/OAuth/farmId - GET	03-08-2020		Nee	Goedgekeurd	03-08-2020	✓ ✗
Test Consumer	Consumer for testing	Returns a number	03-08-2020		Nee	Goedgekeurd	03-08-2020	✓ ✗

In de actie kolom kan al dan niet toestemming verleend worden.

3 Partner API (optioneel)

Alle acties en overzichten in de DjustConnect Portal kunnen ook uitgevoerd worden met behulp van de [Partner API](#).

Voor dataleveranciers zijn de relevante endpoints:

- *ProviderAPI*: aanmaken, configureren en publiceren van een nieuwe API
- *ResourceAccessRequest*: Goedkeuren/afkeuren van data aanvragen

4 Push API (Events)

Indien gewenst kunnen er ook pushevents aangemaakt worden door dataleveranciers in het DjustConnect portaal. Dit is vooral nuttig wanneer datagebruikers in realtime op de hoogte willen gebracht worden van nieuwe data of veranderingen in de huidige data. We denken hierbij aan bv. nieuwe laboresultaten. Hierbij kan gekozen worden om de resultaten in één tijd met het pushbericht mee te geven of om het ID van het nieuwe of aangepaste datapunt door te sturen waarbij dan de data, indien gewenst, door de datagebruiker via een web API opgehaald kan worden. DjustConnect gebruikt onderliggend Azure Event Grid, dus de tooling en documentatie hiervoor, aangeboden door Microsoft, kan ook steeds geraadpleegd worden.

4.1 Pushmelding Configureren

De eerste stap om een pushmelding te configureren is het configureren van een pushmelding endpoint door op "Pushmelding endpoint configureren" te klikken. Hierbij worden 3 zaken aangemaakt: een endpoint url en 2 keys.

Pushmelding endpoint

Om pushmeldingen aan te maken moet je eerst een push endpoint definiëren

Pushmelding endpoint configureren

Pushmelding types

Nieuw type pushmelding toevoegen

Naam	Id	Status	Acties
No data to display			
Totaal: 0			

Na klikken op "Pushmelding endpoint configureren" krijgt men dan zoiets:

Pushmelding endpoint

Endpoint url

<https://provider-7aff8418-15e7-5709-ad96-0cde0e97ea57.northeurope-1.eventgrid.azure.net/api/events>

Key1

XkOA1AWZik2g3ZKqCKWqAbdb7jOxPLYZzguE_INVALID_KEY-

Key2

aoR35qmDzUeWrvoltHbjBVPXguiycEHPghdN_INVALID_KEY-

Naast het aanmaken van een pushmelding endpoint moeten er ook pushmelding types geconfigureerd worden. Dit komt conceptueel overeen met de configuratie van een API en is dus ook gelijkaardig. De in te vullen velden zijn:

1. Naam van het eventtype
2. Een beschrijving van het eventtype
3. Documentatie url of document. Hiervoor is geen standaard documentatie formaat maar de bedoeling is om het verstuurd JSON-formaat te verduidelijken op een webpagina of in een document.
4. Aangeven of er al dan niet toestemming verleend moet worden alvorens een pushmelding naar de datagebruiker verstuurd mag worden. Indien ook het landbouw/bedrijf zijn toestemming moet verlenen, dan is er ook hier een *farm-id* endpoint vereist. Deze kan net als elke andere web API gebruikmaken van SSL of OAuth autorisatie en eventueel extra vereiste headers.

Na de configuratie kan deze gevalideerd worden en ingediend worden ter goedkeuring.

Hier vindt u alle details van de geselecteerde pushmelding.

Naam	Id
new-lab-result	1a491352-e452-448f-fbbd-o8d8c29€

Voeg een beschrijving toe

Nieuw labo resultaat is a ...

Documentatie URL

Documentatie bestand

Provider An... Type landbou... Landbouwbedrijf ID API URL

Beslag nu... URL van de landbouwbedrijf ID API endpoint die de Datahub zal

Certificaat Authorisatie OAuth Authorisatie

Headerconfiguratie toevoegen

Terug **Verwijderen** **Opslaan** Valideren **Indienen voor goedkeuring**

4.2 Pushmelding uitsturen

Om een pushmelding uit te sturen moet een JSON-bericht verstuurd worden met de juiste metadata. Dit kan door gebruik te maken van de door Microsoft Azure aangeleverde .NET bibliotheek: Microsoft.Azure.EventGrid die beschikbaar is op Nuget:

```
// Topic te kopiëren bij Mijn pushmeldingen -> Pushmelding endpoint
var topic = "https://provider-90d2a26d-YOUR-ENDPOINT-URL-HERE.northeurope-1.eventgrid.azure.net/api/events";
// Key te kopiëren bij Mijn pushmeldingen -> Pushmelding endpoint
var topicCredentials = new TopicCredentials("tLZY9Nvn2=YOUR-EVENT-ENDPOINT-KEY-HERE");
// Id te kopiëren bij Mijn pushmeldingen -> Pushmelding types
var eventType = "1a491352-e452-YOUR-EVENT-TYPE-ID-HERE";
var farmNumber = "0262172489";
var data = "Voorbeeld event, kan ook een object zijn";
var events = new List<EventGridEvent>
{
    new EventGridEvent
    {
        Id = Guid.NewGuid().ToString(),
        EventTime = DateTime.Now,
        EventType = eventType,
        Subject = farmNumber,
        Data = data,
        DataVersion = "1.0"
    }
};

using (var client = new EventGridClient(topicCredentials))
    await client.PublishEventsAsync(new Uri(topic).Host, events);
```


Of je kan je bericht zelf samenstellen en versturen als JSON. De request heeft dan de volgende parameters en eigenschappen:

- Het moet een POST request zijn naar de url die vermeld staat op het DjustConnect portaal;
- Er moet een extra header *aeg-sas-key* geconfigureerd worden met als waarde de vermelde key in het portaal;
- Het root element van de body is een array met één of meerdere pushmeldingen;
- Een pushmelding bestaat uit:
 - o id: unieke id voor elk event
 - o eventType: id van het eventtype, terug te vinden in het portaal
 - o subject: identificatienummer van het landbouwbedrijf waarover deze pushmelding gaat
 - o eventTime: tijdstip van het event
 - o data: JSON-waarde of object met de eigenlijke data
 - o dataVersion: versie van het gebruikte dataformaat

Meer lezen over Azure Event Grid:

- <https://docs.microsoft.com/en-us/azure/event-grid/>
- <https://docs.microsoft.com/en-us/dotnet/api/overview/azure/eventgrid>
- <https://social.technet.microsoft.com/wiki/contents/articles/53692.azure-eventgrid-submitting-from-postman-to-custom-topic.aspx>