Onboarding nieuwe dataleverancier MDS REST API

Last updated by | Kuil E van der (Erik) | Jun 12, 2024 at 12:25 PM GMT+2

Contents

Introductie

Tellingen kunnen vanuit een dataleverancier automatisch naar MDS worden verstuurd middels een REST API. Hierop kunnen tellingen middels HTTP POST worden afgeleverd. De REST API ontsluit endpoints per type telling en is beveiligd: zonder de juiste authenticatie (is de leverancier bekend in het systeem?) en autorisatie (wat mag de leverancier doen?) zullen aangeboden tellingen niet worden geaccepteerd. Dit document beschrijft hoe leveranciers toegang kunnen krijgen tot de endpoints om tellingen aan te bieden aan MDS.

Authenticatie

Voor authenticatie hanteert MDS een combinatie van IP-whitelisting en RSA-gesigneerde JSON Web Tokens (JWT's) - ook wel JSON Web Signatues (JWS) genoemd.

- IP-whitelisting biedt strakke controle op welke bronsystemen data mogen leveren aan MDS. Ander verkeer wordt op firewall-niveau geblokkeerd.
- JWT authenticatie laat zich goed automatiseren voor bulk-datalevering.
- RSA-gesigneerde JWT's zijn - vergeleken met statische wachtwoorden en geheime sleutels - robuuster tegen onderschepping.

RSA is een asymmetrische cryptografie-methode, waarbij er twee sleutels worden gegenereerd door de dataleverancier. De geheime sleutel wordt gebruikt voor signeren en hoeft nooit gedeeld te worden met ICT InTraffic. De publieke sleutel kan vrijuit worden gedeeld en kan alleen worden gebruikt om JWT’s te

valideren. Dit verkleint de kans op uitlekken van de geheime sleutel aanzienlijk en biedt extra bescherming tegen impersonatie van de dataleverancier.

Alleen bij hoge uitzondering en met goede onderbouwing kan één van de authenticatie-elementen worden versoepeld (symmetrische cryptografie, JWT óf whitelisting) – nooit meerdere elementen. Deze uitzondering kan alleen plaatsvinden indien de opdrachtgever, de dataleverancier en ICT InTraffic akkoord zijn en de risico's accepteren.

Autorisatie

ICT InTraffic configureert voor elke dataleverancier expliciet welke authenticatiemethode afgesproken is, welke endpoints toegankelijk zijn en vanaf welke IP-adres(sen) data aangeboden mag worden.

Proces

Het consistent en veilig onboarden van een nieuwe dataleverancier voor de MDS REST API is gewaarborgd door het doorlopen van de stappen in onderstaand proces.

Afspraken tussen dataleverancier en ICT InTraffic

Tussen de datalerancier en ICT InTraffic dienen afspraken gemaakt te worden over de volgende zaken:

- Welke type(n) data gaat de leverancier leveren en naar welke endpoint(s)?
- Vanaf welk(e) IP-adres(sen) gaat de leverancier leveren?
- Welke authenticatiemethode wordt gebruikt?
- Welke cryptografie wordt gebruikt voor signeren en valideren van JWT's?
- Wat is de maximale geldigheidsperiode van JWT's?
- Welke processen worden ingeregeld om te voorkomen dat datalevering onderbroken wordt bij wijzigingen in IP-adressen, sleutels en/of andere problemen in de keten?

Uitwisselen gegevens tussen dataleverancier en ICT InTraffic

De dataleverancier en ICT InTraffic wisselen op basis van de afspraken gemaakt in de vorige stap de benodigde gegevens uit om JWT payload (dataleverancier) en client-configuratie (ICT InTraffic) in te regelen.

ICT InTraffic

1. Levert skelet voor verwachte JWT structuur (header, payload, signature).
2. Levert unieke identifier aan voor de leverancier.
3. Levert endpoint URI voor elk overeengekomen type datalevering aan leverancier.
4. Levert unieke identifier voor elk keypair dat leverancier heeft aangeleverd.

Dataleverancier

1. Levert IP-adres(sen) waarvandaan data geleverd gaat worden.
2. Genereert RSA-keypair en levert de public key aan ICT InTraffic.

Implementatie door dataleverancier

De dataleverancier implementeert JWT met de structuur en attributen zoals uiteengezet in Vereisten Signed JWT hieronder. De dataleverancier genereert voor elke datalevering een nieuwe JWT en voegt die als HTTP Authorization header toe aan de HTTP POST aanroep:

```
Authorization: Bearer <Base64 encoded JWT>
```

Testen op acceptatieomgeving

Na implementatie kan er getest worden op de MDS acceptatieomgeving. Hiervoor dienen de volgende stappen doorlopen te worden.

1. ICT InTraffic configureert toegang voor dataleverancier op de MDS acceptatieomgeving en informeert de dataleverancier.
2. Dataleverancier verstuurt dataleveringen met gesigneerde JWT's (als HTTP Authorization Header) naar het overeengekomen endpoint.
3. ICT InTraffic controleert correcte ontvangst en validatie van de JWT.
4. ICT InTraffic en dataleverancier bespreken de resultaten.

Toegang tot productieomgeving

Pas na volledig succesvolle afronding van tests op de MDS acceptatieomgeving zal de dataleverancier toegang krijgen tot de overeengekomen endpoints op de MDS productieomgeving. Hiervoor dienen de volgende stappen doorlopen te worden.

1. ICT InTraffic configureert toegang voor dataleverancier op de MDS productiomgeving en informeert de dataleverancier.
2. Dataleverancier kan vanaf dit moment data versturen naar de MDS productieomgeving. Naar gelang de situatie wordt dit moment overeengekomen en gemonitord door de dataleverancier en ICT InTraffic samen.

Vereisten Signed JWT/JWS

Opbouw

De JWT die in de header van elke datalevering aanwezig is, dient een aantal verplichte attributen te bevatten om door alle validatiestappen heen te komen. De leverancier vult, genereert en signeert de JWT zelf. De inhoud van de attributen wordt gebruikt om de bijpassende client-configuratie in MDS op te zoeken en vervolgens te verifiëren dat de datalevering aan alle gestelde eisen voldoet. De verplichte attributen (inclusief voorbeeld-invulling) in elke JWT zijn hieronder weergegeven.

Header

```
{
  "alg": "RS512",
  "typ": "JWT",
  "kid": "BAAE="
}
```

Attribuut	Opmerking
`alg`	Gebruikt algoritme voor signeren JWT.
`typ`	Dient altijd gelijk te zijn aan 'JWT'.
`kid`	Unieke identifier van exact 5 tekens voor de (public) key die MDS moet gebruiken voor validatie van de signature. Deze is standaard gelijk aan de laatste vijf tekens van de (public) key.

Payload

```
{
  "sub": "test-leverancier",
  "aud": "/endpoint1 ",
  "iss": "1.2.3.4",
  "iat": 1692871094,
  "exp": 1712318296
}
```

Attribuut	Opmerking
`sub`	Unieke identifier voor de dataleverancier.
`aud`	Endpoint waar de levering voor bedoeld is.
`iss`	IP-adres waarvandaan de data verstuurd is.
`iat`	Epoch timestamp (millis) waarop JWT gegenereerd is.
`exp`	Epoch timestamp (millis) waarop JWT verlopen is.

Voorbeeld JWT

Last updated on 23 Nov 2023

De bovenstaande JWT-header en payload worden volgens standaard JWT aanpak gesigneerd. In het geval van het voorbeeld hierboven wordt de signature gegenereerd middels:

```
RSASHA512(base64Encode(header) + "." + base64Encode(payload), privKey)
```

De resulterende (base64-encoded) JWT String is als volgt:

```
eyJhbGciOiJSUzUxMiIsInR5cCI6IkpXVCIsImtpZCI6IkJBQUU9In0.eyJzdWIiOiJ0ZXN0LWxldmVyYW5jaWVyIiwiYXVkIjoiL2VuZHBvaW50MSAiLCJpc3MiOiIxLjIuMy40IiwiaWF0IjoxNjkyODcxMDk0LCJleHAiOjE3MTIzMTgyOTZ9.3XIbd3TzooooaViEfjaX-SbxoQWZ-3mVyIMa65DEVHnD5ZJim-Qi7bsBLzHmZMC9iAbnih17aq5I30uCGQ9KuBfdSy1bd8_WYa-rSYBuPOEuQP_0ouunNuB5p-j2CEitZpSnkJO9mIa7uz8TzXzFfkB8xAGE-ImEAnvxqXweyYrdtDLy1AiE-hu6FRE64AYg8_ZG-Iy64sZGDC0kqVsFZwyZt0wcmhNaaT0BtfUheoJIgt_gfr_xS9be9Nb87kH-AFbSvgSvJF5ZRvnv2Dx8FIn9Klmtngl7C4q8qSDVYjcC0gMRXSEKnbSWA3YYf1p4V2avX3GyMxPzbsSakDg_fTQJR-ks0r3AMQGQciNttHLfAMNm4gIvuKtrvrzeVjjP1h-L850nUTfZ8WQswwBLF6b7m37TbvXrKivXJu5udvs-wRhlUusLCs_A0Ts5fDT3JayZ3ihFsHQKcju7bkkONDDLZAbwAj7UDAn2cKQMjj7IMKc3dgEbGZIHUDzxnl9C
```

Debuggen van JWT's kan met de debugger op https://jwt.io/ . Als hier bovenstaande JWT String wordt geplakt, zullen de header en payload uit de voorgaande sectie zichtbaar zijn. Om de signature te valideren, kan de volgende public key worden gebruikt:

```
MIIBojANBgkqhkiG9w0BAQEFAAOCAY8AMIIBigKCAYEA6P9eEpPoTz36txbNgRyOtPxK/Qsgix+U5+IXVfGPQTcD9lSyiPmMnPUmQFRMTOoB
sOPpG8j5MT97gmCe9noUR1smtwj1IK+8rab8IeCi3G7kb8lN75bt9YOQJhTf/RA23JfEbszNR4p/Xm4rrTegrH5jLlnKVqarGr92psgqvDkF
qaCdm65vmc1qKWPIyNAL8x1rWyDZxuMpAgV26RnV6nrLqUhQuYAmq+U9z2XoySxMAUUzH/5g5g2byG+qSOzf+pX6JWzgxxDxXicCyK0cesnE
StyWhTD2n5fqqaTx+4xNqJ8Kk9vVuG0j3bEwfiZpyl/yEqEjftXf/uyHV8HE/AnHtMAFdQdTQhWcSrpbEmLUo7qT8LdCrStpK6OzqhkjdAjT
9CBoXij7fTGfZlk/ng43pSvKVYKEE35vv5/PsOp39WaXgPwFVp2MeiM8KeN/IifmCSAfEIAq/9vYiYGp1gMniLHBrhKcx9b/GD2qD5bgiu0J
dJCsXIs81Y/V+zuHAgMBAAE=
```