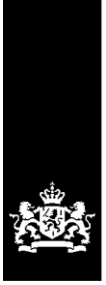


Logius
Ministerie van Binnenlandse Zaken en Koninkrijksrelaties

Koppelvlakspecificatie CGI - DigiD

Versie 2.5

Datum 31 augustus 2016

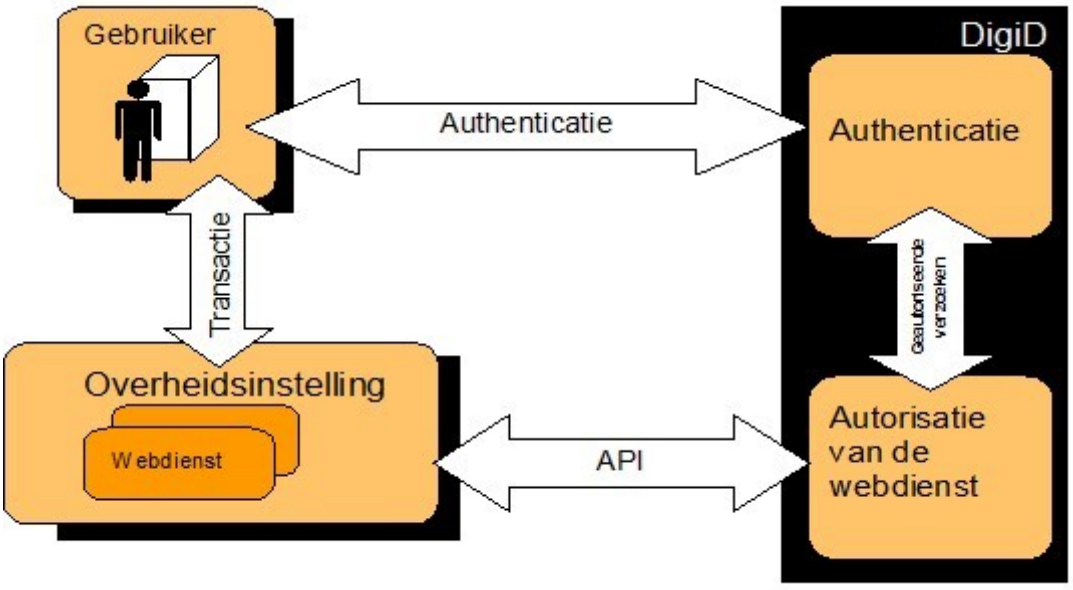
Colofon

Projectnaam	DigiD
Versienummer	2.5
Organisatie	Logius Postbus 96810 2509 JE Den Haag T 0900 555 4555 (10 ct p/m) servicecentrum@logius.nl

Inhoud

1 Inleiding

DigiD is een gemeenschappelijk authenticatiesysteem dat overheidsinstellingen in staat stelt om de identiteit te verifiëren van klanten die gebruikmaken van hun elektronische diensten. Webdiensten moeten hiervoor het schema van authenticatie van DigiD implementeren
dat bestaat uit een verzameling van API-aanroepen gecombineerd met het doorsturen (redirecten) van de gebruiker naar DigiD.



figuur 1: Positionering van de interface

DigiD heeft twee interfaces. De eerste interface is een interface voor webdiensten die met behulp van API-aanroepen beschikbaar is. Het doel van deze interface is het initiëren van de authenticatie van een gebruiker. De tweede interface is de gebruikersinterface waarmee interactie met de gebruiker plaatsvindt. De totale communicatieketen over beide interfaces is met behulp van redirects (via de browser van de gebruiker) gekoppeld.

Meer informatie over het totale concept van DigiD is te vinden in "Handreiking DigiD" en verdere relevante documentatie die aan webdiensten ter beschikking is gesteld. Voor de meest recente informatie en documentatie kunt u terecht op www.logius.nl/digid.

Voor de verklaring van afkortingen en begrippen verwijzen wij u naar www.logius.nl/begrippenlijst

2 Principes van DigiD Authenticatie

2.1 Technisch Overzicht

DigiD handelt API-verzoeken af als een webservice en is aan te roepen via CGI (Common Gateway Interface) parameters

Webdiensten kunnen met DigiD communiceren met behulp van de in dit document beschreven mechanismen en bijbehorende API. Webdiensten roepen DigiD als webservice aan en krijgen een antwoord terug. Wanneer DigiD zelf moet communiceren met Webdiensten gebeurt dit door de gebruiker te redirecten naar de webdienst. DigiD plaatst dan parameters achter de URL van de webdienst. Met behulp van deze parameters kunnen Webdiensten de authenticatie van een gebruiker voltooien door de authenticatie met DigiD te verifiëren. DigiD geeft dan de identiteit van de gebruiker vrij. Webdiensten dienen derhalve rekening te houden met door DigiD gereserveerde parameters waarop zij moeten reageren.

2.2 Principes en Vervolg in de Webdienst

Voor een Webdienst is een gebruiker geauthenticeerd wanneer de Webdienst op een betrouwbare manier het resultaat van authenticatie en derhalve de identiteit van de gebruiker van DigiD heeft ontvangen. DigiD heeft dan zijn werk gedaan.

Na een succesvolle authenticatie gaat de webdienst een sessie aan met de gebruiker waarin de identiteit van de gebruiker voor de volledige duur van deze sessie vaststaat. Sessies worden doorgaans bijgehouden met behulp van een uniek, moeilijk te raden sessie-identificatiecode (sessie-id). Dit sessie-id wordt aangemaakt door de webdienst en moet bij elk verzoek van de browser van de gebruiker worden meegegeven aan de Webdienst. Dit kan bijvoorbeeld door het op te slaan in een non-persistent cookie[1]. Het sessie-id is binnen de Webdienst gekoppeld aan de identiteit van de gebruiker, plus eventuele aanvullende informatie over de huidige sessie.

Webdiensten zijn zelf verantwoordelijk voor het aangaan en beheren van sessies met geauthenticeerde gebruikers. Deze bijlage gaat hier niet verder op in, daar dit buiten de verantwoordelijkheid van DigiD valt.

2.3 Procedure van authenticatie

De keten van communicatie tussen webdiensten en DigiD tijdens een volledige authenticatieslag bestaat uit een combinatie van API-aanroepen en redirects van de browser van de gebruiker. De samenhang binnen deze keten wordt gewaarborgd door voor elke authenticatieslag een authenticatiesessie aan te maken[2]. Elke keer dat een webdienst een authenticatie van een gebruiker initieert maakt DigiD een unieke authenticatiesessie aan die tijdens de gehele authenticatieslag geldig is.

---

[1] *Dit zijn cookies die niet op disk bij de gebruiker maar in het geheugen van de browser worden opgeslagen. Wordt de browser of browser instantie gesloten, dan is het cookie vernietigd. De gebruiker moet echter wel toestaan om cookies te gebruiken. Hiervoor dient hij/zij zijn browser als zodanig in te stellen.*

[2] *Deze sessie staat los van de sessie die na een succesvolle authenticatie aangemaakt dient te worden aangemaakt door de Webdienst zelf.*

Webdiensten moeten deze authenticatiesessie zelf ook bijhouden en verifiëren om misbruik van de dienst te voorkomen.

1. Een authenticatieslag wordt geïnitieerd door de webdienst. Dit gebeurt met behulp van een API-aanroep naar DigiD.
2. Vervolgens zal DigiD reageren met onder andere de identifier van bovengenoemde authenticatiesessie en een URL waar de gebruiker naar toe gestuurd dient te worden.
3. Zodra de gebruiker naar deze URL geleid is, begint de eigenlijke authenticatie bij DigiD.
4. Wanneer de authenticatie succesvol verlopen is, wordt de gebruiker terug naar de Webdienst geredirect. In die redirect naar de Webdienst worden vercijferde authenticatiegegevens als parameters doorgegeven in de URL van de webdienst. De webdienst kan deze herkennen aan de door DigiD gereserveerde parameternamen.
5. De authenticatiegegevens worden vervolgens geverifieerd bij DigiD. Dit gebeurt door de gegevens (rechtstreeks en zonder aanpassing) ter verificatie aan te bieden bij DigiD met behulp van een API-aanroep.
6. Als de gegevens kloppen, antwoordt DigiD met onder andere het resultaat van de authenticatie en identificerende gegevens over de gebruiker.
7. Zodra dit antwoord is ontvangen, is de identiteit van de gebruiker bij de Webdienst bekend en is de authenticatieslag afgerond. Vanaf dat moment is de webdienst verantwoordelijk hoe er met de identiteit en overige gegevens om wordt gegaan.

3 DigiD authenticatie

DigiD is als webservice op het internet bereikbaar onder SSL. Zoals in het voorgaande beschreven is zijn er twee momenten waar webdiensten de DigiD webservice moet aanroepen: (1) bij initiatie van authenticatie en (2) bij verificatie van een authenticatie.

3.1 Conventies

Elke API-aanroep naar DigiD bestaat uit een enkel bericht met daarin een reeks parameters. DigiD ondersteunt daarbij CGI parameters.

3.1.1 Parameters

Webdiensten dienen het volgende t.a.v. parameters in API aanroepen in acht te nemen:

- Parameternamen zijn hoofdlettergevoelig.
- Volgorde van de parameters is irrelevant (zowel in aanroep als antwoord)
- DigiD geeft na elk verzoek een resultaatcode terug. De lijst van resultaatcodes is gegeven in Hoofdstuk 5. Webdiensten dienen bij het verwerken van het antwoord van DigiD altijd deze resultaatcode te controleren.

3.2 CGI

De CGI van DigiD wordt aangeroepen met behulp van de HTTPS/GET[3] methode waarbij de parameters in CGI-syntax aan de URL van DigiD worden meegegeven.

Parameters aan DigiD hebben de volgende syntax:

```
key1=value1[&key2=value2[...&keyN=valueN]]
```

Parameters worden gescheiden door het '&'-teken en het hele verzoek bestaat uit één regel zonder spaties of andere niet-valide URL-karakters, afgesloten met een *newline* (carriage return gevolgd door linefeed). In programmeertalen zoals Java of C/C++ bijvoorbeeld zou dan een "\r\n" achter het verzoek toegevoegd moeten worden.

Het antwoord van DigiD is in dezelfde syntax en bestaat uit één regel met de antwoord parameters gescheiden door het '&'-teken en afgesloten met *newline*.
De Webdienst moet de parameters hieruit zelf destilleren.

[3] *De HTTPS/POST methode wordt bij gebruik van CGI interface niet ondersteund door DigiD*

3.3 Initiatie van Authenticatie

Om een authenticatieslag te initiëren moet een Webdienst een `request=authenticate` verzoek sturen. De volledige parameterlijst is:

Key	Type	Verplicht	Value	Omschrijving
request	String	Ja	authenticate	Initiatie van authenticatie.
a-select-server	String	Ja	[a-select-server-ID]	Naam van de DigiD Burger Server.
app_id	Int	Ja	[Applicatie ID]	Uniek id van de Webdienst applicatie uitgegeven door DigiD Beheer.
shared_secret	String	Ja	[Authenticatiecode]	Geeft de door DigiD uitgegeven authenticatiecode aan.
app_url	String	Ja	[URL van de Webdienst]	Volledige URL van de webdienst zelf. Na afloop van de authenticatie wordt de gebruiker hier naar geredirect. Webdiensten kunnen eigen parameters in deze URL hebben. **De inhoud van deze parameter moet URL encoded zijn.** **De URL dient in hetzelfde domein te zijn als de pagina voor het inloggen (gebruiker start en eindigt op hetzelfde domein)**

Het antwoord van DigiD bestaat uit:

Key	Type	Value	Omschrijving
rid	String	[Request ID]	Een unieke identifier van de authenticatiesessie
as_url	String	[URL van DigiD]	URL van DigiD. De webdienst moet de browser naar deze volledige URL doorsturen. De webdienst moet aan deze URL de rid toevoegen.
a-select-server	String	[a-select-server-ID]	Naam van de DigiD Burger Server.
result_code	String	[resultaatcode]	Resultaat van de aanroep. '0000' indien succesvol.

3.3.1 CGI voorbeeld

Let op: onderstaande waarden dienen slechts als voorbeeld.

De Webdienst heeft de URL:	https://diensten.hengelo.nl/secureportal
Het Applicatie id is:	hengelo_digid_portal
De Authenticatie code is:	123456-kd2s-s3kg-72kf-k2f3-mk2e-aoe3
DigiD heeft als URL:	https://was.digid.nl/was/server
DigiD heeft als ID:	digidas1

Om authenticatie te initiëren zal de Webdienst DigiD aanroepen zoals hieronder gegeven:

```
https://was.digid.nl/was/server?request=authenticate&app_url=
https%3A%2F%2Fdiensten%2Ehengelo%2Enl%2Fsecureportal&app_id=he
ngelo_digid_portal&shared_secret=123456-kd2s-s3kg-72kf-k2f3-
mk2e-aoe3&a-select-server=digidas1
```

DigiD zal vervolgens in één regel het antwoord terugsturen. Bijvoorbeeld:

```
rid=A77C582B33C03912&as_url=https://as.digid.nl/aselectserver/
server?request=login1&a-select-
server=digidas1&result_code=0000
```

De Webdienst zal de browser van de gebruiker dus moeten redirecten naar:

```
https://as.digid.nl/aselectserver/server?request=login1&rid=A7
7C582B33C03912&a-select-server=digidas1
```

Opmerkingen

- Let op de concatenatie van de parameters rid en a-select-server aan de verkregen URL van DigiD voordat de browser wordt geredirect naar DigiD.
- Webdiensten dienen de verkregen rid parameter te onthouden als onderdeel van hun eigen sessie management tijdens de authenticatieslag.

3.4 Verificatie van Authenticatie

Verificatie van authenticatie moet gebeuren wanneer de browser bij de Webdienst komt met in de URL een boodschap afkomstig van DigiD. Deze boodschap bevat vercijferde informatie die de Webdienst onveranderd naar DigiD moet sturen ter verificatie. Hiervoor stuurt de Webdienst een request=verify_credentials verzoek. Als de verificatie succesvol is, zal DigiD informatie terugsturen over de geauthenticeerde gebruiker.

DigiD stuurt drie parameters naar Webdiensten die zij moeten herkennen:

- aselect_credentials
- rid
- a-select-server

Webdiensten die eigen parameters in hun URL gebruiken dienen rekening te houden met bovenstaande gereserveerde namen.
Webdiensten dienen bij het ontvangen van dit verzoek van DigiD controles te doen op de rid en a-select-server parameters. Deze moeten immers voorkomen in de lijst van lopende sessies met DigiD.

Key	Type	Verplicht	Value	Omschrijving
request	String	Ja	verify_credentials	Verificatie van authenticatie
a-select-server	String	Ja	[a-select-server-ID]	Naam van de DigiD Burger Server.
aselect_ credentials	String	Ja	[vercijferde informatie]	De vercijferde informatie die de Webdienst onveranderd door moet sturen naar DigiD.
shared_ secret	String	Ja	[Authenticatiecode]	Geeft de door DigiD uitgegeven authenticatiecode aan.
rid	String	Ja	[Request ID]	De rid die gebruikt is tijdens initiatie van authenticatie. Deze parameter moet voorkomen in een door de DigiD eerder geïnitieerde en nog lopende authenticatie sessie met DigiD.

Het antwoord van DigiD bestaat uit:

Key	Type	Value	Omschrijving
uid	String	[Gebruikers ID]	De identiteit van de geauthenticeerde gebruiker.
app_id	Int	[Applicatie ID]	Uniek ID van de Webdienst applicatie uitgegeven door DigiD Beheer.
organization	String	[DigiD organisatie]	Naam van DigiD organisatie.
betrouwbaarheidsniveau		[betrouwbaarheidsniveau]	Het betrouwbaarheidsniveau van de authenticatie.
rid	String	[Request ID]	De rid die gebruikt is tijdens initiatie van authenticatie
a-select-server	String	[DigiD ID]	Server-id van DigiD.
Result_code	String	[resultaatcode]	Het resultaat van de aanroep. '0000' indien succesvol.

Opmerkingen

- DigiD gebruikt betrouwbaarheidsniveaus. Op dit moment zijn de volgende betrouwbaarheidsniveaus gedefinieerd:
    - 10: (basis): De gebruiker heeft online een gebruikersnaam met wachtwoord aangevraagd en geactiveerd, en heeft zich geauthenticeerd met zijn wachtwoord.
    - 20: (midden): De gebruiker heeft online een gebruikersnaam met wachtwoord aangevraagd en geactiveerd, en heeft daarnaast een mobiel telefoonnummer als extra factor ontsloten (sms-authenticatie). De gebruiker heeft zich geauthenticeerd met zijn permanente wachtwoord en een tijdsgebonden sms-transactiecode.
    - 25: (substantieel): De gebruiker heeft online een authenticatie middel aangevraagd. Bij de uitgifte van dit middel is het identiteitsdocument van de gebruiker eenmalig gecontroleerd. De gebruiker heeft zich geauthenticeerd met een two-factor authenticatie.
    - 30: (hoog): De gebruiker heeft online een koppeling gelegd tussen zijn DigiD en een identiteitsdocument. Dit document bevat een persoonlijk certificaat dat kan worden uitgelezen. De gebruiker heeft zich geauthenticeerd met het gekoppelde identiteitsdocument.

Belangrijk: De webdienst dient zelf te controleren of er een authenticatie op het juiste betrouwbaarheidsniveau heeft plaatsgevonden. Met andere woorden: er vindt een match plaats tussen het door DigiD teruggekoppelde betrouwbaarheidsniveau (A) en het door de webdienst gewenste minimale betrouwbaarheidsniveau (B). Als (A) < (B) wordt de authenticatie niet geaccepteerd; als (A) ≥ (B) wordt de authenticatie geaccepteerd.

Let op: In de verdere toekomst zullen ook andere middelen op betrouwbaarheidsniveau Midden, Substantieel en Hoog worden ontsloten. De gebruikte parameterid's voor de parameter zekerheid zullen analoog hieraan oplopen. Om aanpassingen op dat moment te voorkomen, adviseren we u het volgende:

- Configureer uw webdienst zodanig dat deze alle betrouwbaarheidsniveaus accepteert, die gelijk of hoger zijn dan degene die minimaal voor uw webdienst zijn genoodzaakt. Voorbeeld: als uw webdienst minimaal authenticatie vereist op niveau basis (dit zou 10 zijn), zorg er dan voor dat alle authenticaties worden geaccepteerd die minimaal 10 of hoger zijn (bijvoorbeeld ≥ 10).

Meer informatie over de betrouwbaarheidsniveaus kunt u vinden in de handreiking.

3.4.1 CGI voorbeeld

Let op: onderstaande waarden dienen slechts als voorbeeld.

De Webdienst heeft de URL:	https://diensten.hengelo.nl/secureportal
Het Applicatie id is:	hengelo_digid_portal
De Authenticatie code is:	123456-kd2s-s3kg-72kf-k2f3-mk2e-aoe3
DigiD heeft als URL:	https://was.digid.nl/was/server
DigiD heeft als ID:	DigiD1

De Webdienst krijgt een verzoek van de browser met een boodschap van DigiD. De Webdienst is bijvoorbeeld als volgt aangeroepen:

```
https://diensten.hengelo.nl/secureportal?aselect_credentials=X
&rid= A77C582B33C03912&a-select-server=digidas1
```

De Webdienst controleert of de rid voorkomt in de lijst van lopende sessies. Om de authenticatie te verifiëren zal de Webdienst het volgende HTTPS/GET verzoek naar DigiD sturen.

```
https://was.digid.nl/was/server?request=verify_credentials&ase
lect_credentials=X&rid=A77C582B33C03912&shared_secret=123456-
kd2s-s3kg-72kf-k2f3-mk2e-aoe3&a-select-server=digidas1
```

DigiD zal vervolgens in één regel het antwoord terugsturen. Bijvoorbeeld:

```
rid=A77C582B33C03912&uid=190382582&app_id=hengelo_digid_portal
&betrouwbaarheidsniveau=10&organization=DigiD&a-select-
server=digidas1&result_code=0000
```

4 Praktische Aspecten

In deze sectie worden praktische aspecten belicht voor de implementatie van de koppeling met DigiD.

4.1 Protocol

Voor deze interface wordt HTTP met SSL (HTTPS) gebruikt.

4.2 Secure Socket Layer (SSL)

DigiD gebruikt Secure Socket Layer (SSL) voor de communicatie met de burger. DigiD gebruikt daarnaast ook SSL voor zijn authenticatiedienst (de Webservice) voor Webdiensten. SSL biedt een uitstekende mogelijkheid om een veilige verbinding op te zetten, waarbij de Webdienst er zeker van kan zijn dat zij met DigiD communiceert door het server-certificaat van DigiD te verifiëren. Webdiensten dienen echter wel de nodige maatregelen te nemen waarbij het certificaat van DigiD, zoals die in productie wordt gebruikt, wordt vertrouwd.

4.3 Sessie Management

Er zijn feitelijk twee relevante sessies met betrekking tot DigiD, (1) tijdens het authenticatie proces waarbij DigiD een unieke, tijdelijke identifier gebruikt voor een authenticatiesessie en (2) na authenticatie, wanneer de Webdienst de identiteit van de geauthenticeerde gebruiker vast moet houden.

De webdienst moest daarnaast voldoen aan de volgende eisen:

- Wanneer gebruik wordt gemaakt van sessie-ID's, mogen deze na authenticatie niet worden gewijzigd.
- Sessies mogen door de web-applicatie niet worden doorgegeven aan derden.
- Sessies mogen niet blijven 'hangen' en moeten door de web-applicatie actief worden afgesloten.
- De web-applicatie moet om kunnen gaan met geannuleerde sessies vanuit DigiD.

4.4 Tijdens Authenticatie

DigiD identificeert een authenticatie sessie met een unieke identifier, de rid . Bij initiatie van authenticatie krijgt de Webdienst deze terug in het antwoord van DigiD. Bij verificatie van authenticatie moet de Webdienst dezelfde rid meegeven als die tijdens de initiatie van authenticatie verkregen is. DigiD neemt zelf de nodige maatregelen om ervoor te zorgen dat niet bestaande of niet meer geldige lopende authenticatie sessies worden misbruikt door aanvallers[4]. Webdiensten moeten misbruik ook herkennen door te controleren of de rid die ze krijgen in een antwoord van DigiD geldig is. Dit moet worden gecontroleerd worden tegen een lijst van lopende authenticatie sessies met DigiD.

In de praktijk kan het voorkomen dat gebruikers niet terugkeren bij de Webdienst omdat ze de authenticatie niet voltooien. Het is aan te raden

[4] *Onder misbruik wordt verstaan een replay attack waarbij een voormalig antwoord van DigiD wordt verstuurd door een aanvaller.*

dat Webdiensten verlopen sessies met DigiD verwijderen uit hun lijst van lopende sessies als onderdeel van een efficiënt sessie management.

4.5 Na Authenticatie

Het belangrijkste doel van authenticatie is niet de authenticatie zelf maar de sessie die daarna moet worden opgebouwd, gebaseerd op dat moment, met die gebruiker, met die browser van dat IP-adres, etc.

De API van DigiD is betrekkelijk eenvoudig en dit resulteert dan ook in een eenvoudige situatie op het moment na authenticatie. De Webdienst krijgt van DigiD een identificerend gegeven van de gebruiker en zal dit moeten gebruiken om een sessie met die gebruiker op te zetten en te behouden.

Meestal kiezen Webdienstbouwers om een geauthenticeerde sessie te implementeren in de vorm van tijdelijke sessies geïdentificeerd door een uniek en niet te raden reeks van tekens (sessie-id). Dit sessie-id wordt dan opgeslagen in de browser als non-persistent cookie en komt bij elk verzoek van de browser naar de Webdienst mee. De Webdienst moet dan controleren of dit inderdaad een geldige sessie is. Hierbij is het de verantwoordelijkheid van de Webdienst bouwer om een sessie mechanisme te bouwen danwel te vertrouwen op software componenten zoals de gebruikte Webserver of andere services.

5 DigiD Resultaatcodes

DigiD geeft in elk antwoord tijdens initiatie en verificatie van authenticatie een resultaat in de parameter result_code terug.

In onderstaande tabel zijn de mogelijke waarden met hun beschrijving opgenomen.

result_code	**Omschrijving**
0000	Geen fout, de verwerking van het request was succesvol.
0001	DigiD is tijdelijk buiten dienst.
0003	DigiD is niet in staat om het verzoek te verwerken. DigiD is tijdelijk buiten dienst.
0004	De verificatie van authenticatie is mislukt omdat DigiD credentials in de DigiD boodschap ongeldig zijn. Dit is meestal een foutief gevormde verificatie verzoek. Controleer de syntax van het verify_credentials verzoek.
0007	De verificatie van authenticatie is mislukt omdat de DigiD credentials in DigiD boodschap ongeldig zijn. Dit kan duiden op een foutief gevormde verificatie verzoek. Controleer de syntax van het verify_credentials verzoek.Deze fout kan ook duiden op een aanval. Als deze fout optreedt en u bent er zeker van dat de syntax van verify_credentials correct is dan wordt aangeraden maatregelen te nemen tegen de gebruiker die niet bestaande DigiD credentials heeft meegegeven (b.v. replay attack).
0030	Ongeldig verzoek. Het verzoek is niet bekend of bevat een syntax fout. Controleer de meegegeven parameters.
0032	De app_url parameter bevat een ongeldige waarde. Webdienst URL's moeten volledig en URL- encoded zijn.
0033	De parameter a-select-server klopt niet. Controleer of u de juiste a-select-server parameter hebt meegegeven voor de juiste URL van DigiD. Raadpleeg de aansluitgegevens die u van DigiD Beheer hebt gekregen.
0040	De gebruiker heeft het authenticatie proces geannuleerd.
0050	DigiD is bezig. DigiD heeft teveel huidige authenticatie sessies af te handelen. Aangeraden wordt om enkele seconden te wachten en het verzoek opnieuw te sturen en dit desnoods een paar keer te herhalen met eventueel een langere periode in acht te nemen.
0070	Ongeldige sessie; de sessie bestaat niet bij DigiD.
0080	De webdienst is gedeactiveerd aan de zijde van DigiD. Indien dit niet correct is dient u contact op te nemen met het servicecentrum Logius.
0099	Webdienst is niet geautoriseerd. De meegestuurde authenticatie code (parameter shared_secret) komt niet overeen met het afgesproken authenticatie code bij DigiD voor uw Webdienst. Deze code wordt ook teruggestuurd indien de app_id niet bekend is of niet aan uw Webdienst is gekoppeld.

Afnemers zijn verplicht zorg te dragen voor een adequate afhandeling van eventuele foutcodes. (Zie bijlage Checklist Testen.)