

# Database Lees- en Schrijfrechten

Dit is een overzicht van de technische randvoorwaarde van een lees- en schrijfaccount op de database.

## Doel/gebruik van de generieke databaseaccounts

Er is voor iedere OpenWave omgeving (zowel acceptatie als productie) een generiek leesaccount (wavereader001) en generiek schrijfaccount (wavewriter001) aanwezig op de database. De rechten voor deze wavereader001 en wavewriter001 zijn voor alle omgevingen gelijk en worden door REM onderhouden en bijgewerkt bij iedere grote OpenWave release (update van de database). Het leesaccount en/of schrijfaccount wordt pas geactiveerd nadat de daarvoor benodigde contracten akkoord zijn. Voor het schrijfaccount gelden nog extra voorwaarden. Zie kopje *Schrijfaccount (wavewriter001)*.

**N.B. Het daadwerkelijk gebruiken van deze accounts is dus verbonden aan voorwaarden!**

Indien men nog geen toegang heeft tot gebruik van lees- dan wel schrijfaccount dient eerst contact op genomen te worden met support voor aanvraag van het gebruik van het gewenste account. Op deze pagina wordt verder alleen informatie verstrekt over de mogelijkheden van de accounts: dit zal eerst voor het leesaccount beschreven worden en vervolgens voor het schrijfaccount.

## Leesaccount (wavereader001)

Het leesaccount (de zogenaamde wavereader001) wordt alleen geactiveerd voor de klant na contractovereenkomst en zal daarna altijd toegankelijk zijn. Met het leesaccount kan de database bevroegd worden om gegevens op te halen via zogenaamde **select statements** (het is niet mogelijk om gegevens in de database te wijzigen, data aan te maken of data te verwijderen. Voor deze acties is het schrijfaccount benodigd). Met het leesaccount kan men via tools als Pgadmin een connectie maken met de database en vervolgens select statements maken/de database bevroegen. Het is ook mogelijk om zogenaamde BI tools toegang te geven tot het leesaccount. Zo kunnen deze tools de benodigde rapportages maken met de gegevens uit de OpenWave database.

### Waarschuwing

In principe mag men (bijna) alle databasevelden opvragen met de select statements maar er gelden uitzonderingen. Denk aan wachtwoorden die in OpenWave staan opgeslagen: velden waarin deze ongecrypt liggen opgeslagen mogen niet benaderd worden door het leesaccount.

Welke select statements niet mogen worden uitgevoerd zijn gelijk voor zowel het lees- als schrijfaccount en staan beschreven bij kopje *Select statements* op deze pagina.

## Schrijfaccount (wavewriter001)

Het schrijfaccount (de zogenaamde wavewriter001) wordt alleen geactiveerd voor de klant na contractovereenkomst en zal altijd tijdelijk toegankelijk zijn. Dit betekent dat alleen in overleg met support en ICT van OpenWave op afgesproken dag/tijdstip de klant van het schrijfaccount gebruik kan maken. Alle schrijfacties worden gelogd en zijn intern door REM te bekijken. Op deze pagina wordt per type database actie beschreven wat het schrijfaccount wel, en wat het account niet mag uitvoeren. Dit is de situatie in de huidige versie. Het kan zijn dat er in de toekomst tabellen of tabelvelden zijn waarop restricties gelden die nu nog niet bekend zijn.

### Select statements

In principe mag op alle tabellen en views van OpenWave data gelezen worden via select statements. Echter geldt hierop de uitzondering dat men geen ongecrypte wachtwoorden mag inzien. Voor de tabellen die zulke informatie bevatten (voor views komt dit nu nog niet voor) geldt dat ze wel te bevragen zijn met een select statement maar niet met een select \*. Men zal dan de select moeten specificeren door ieder veld van interesse in de select op te nemen m.u.v. de niet toegestane velden.

### Concrete voorbeelden welke selects NIET mogen

```
select * from tbmedewerkers;
select * from tb33gemeente;
select * from tbcompartiment;
select * from tbgeowms;
select * from tbinitialisatie;
select * from tbexportcontainer;
```

Indien men voor deze tabellen toch de data wil zien kan men of select \* op de views doen, of dus select op aparte velden van de tabel doen. Uitzondering: tbinitialisatie is niet te bevragen. In deze tabel staan alle configuratie instellingen en wachtwoorden en is niet toegankelijk voor het schrijfaccount.

### Voorbeelden hoe dan toch de data uit deze tabellen te bekijken

```
select * from vwfrmmedewerkers;
select * from vwfrmlokaties;
select * from vwfrmgeowms;
select * from vwfrminitialisatie;
select * from vwfrmexportcontainer;
select dvcode from tbmedewerkers;
select dvomschrijving from tb33gemeente;
select dvomschrijving from tbcompartiment;
```

```
select dvlaag naam from tbgeowms;  
select dvomschrijving from tbexportcontainer;
```

## Update statements

Ook hier geldt de uitzondering dat men geen velden mag updaten die wachtwoorden bevatten. Men mag geen updates uitvoeren op tabel tbinitialisatie en tabel tbaudit. Alle overige velden van alle tabellen mogen geüpdate worden. Mits de nieuwe waarde aan eventuele constraints voldoet. Met constraints wordt bedoeld bijvoorbeeld een null waarde proberen te zetten voor een veld wat niet null mag zijn. Dit zal niet mogelijk zijn: als er in pgadmin gewerkt wordt geeft het programma een melding met de specifieke constraint die het uitvoeren van update statement verhindert.

## Insert statements

Insert statements kunnen uitgevoerd worden op alle tabellen (m.u.v. tbinitialisatie en tbaudit). Het kan zijn dat men tegen constraints aanloopt als het insert statement niet alle verplichte velden meegeeft. Men dient dan het insert statement aan te passen en de verplichte velden mee te geven.

Indien men een insert doet waardoor een mogelijke gap in dnkeys ontstaat, is het zaak dat men na de insert de nextvalue goed zet voor de tabel. Dit om (eventueel GROTE) problemen in toekomst te voorkomen. Deze gaps ontstaan als men in de insert de waarde van primary keyvelden zelf bepaald in plaats van door de database te laten genereren.

Het is raadzaam om de database zijn werk te laten doen maar mocht het toch nodig zijn eigen primary keys mee te geven dan is het noodzakelijk dat men daarna de nextvalue goed zet met het setval command (hieronder als voorbeeld voor tabel tbomgvergunning geschreven):

```
select setval('tbomgvergunning_dnkey_seq'::regclass,(select  
coalesce(max(dnkey)+ 1,1) from tbomgvergunning),false);
```

## Delete statements

Ook voor delete statements geldt dat ze kunnen worden uitgevoerd op alle tabellen (m.u.v. tbinitialisatie en tbaudit). Ook hier kan men tegen constraints aanlopen. Bijvoorbeeld als het te verwijderen record 1 of meer velden heeft die verwijzen naar een record in een andere tabel EN waarvoor geldt dat deze geen delete cascade heeft. In dat geval dient men eerst de verwijzingen in de andere tabellen te updaten (of ook te verwijderen) alvorens het delete statement uitgevoerd kan worden.

## Create/ alter en drop statements

De door REM uitgeleverde tabellen en views kunnen NOOIT per ongeluk gedropt worden met het schrijfaccount. Het schrijfaccount is daarvoor niet bevoegd en er zal een foutmelding verschijnen. Idem dito kan men geen alter table statements uitvoeren voor de door REM uitgeleverde tabellen.

Het is wel mogelijk, MITS zo afgesproken in separaat contract (toekomstplannen), om eigen tabellen en views te creëren.

## Overige statements/rechten

Het schrijfaccount heeft geen rechten om een eigen nieuwe rol aan te maken of om zichzelf meer rechten toe te kennen. Wat wel mag is het bekijken en zetten van sequences: dit om bijvoorbeeld de nieuwe primary key value van een tabel goed te zetten. De sequences kunnen niet verwijderd worden en men kan geen nieuwe sequences aanmaken. Het is ook niet mogelijk om constraints/triggers aan te passen van de door REM uitgeleverde tabellen. Verder is gewaarborgd dat na de verstrijken van de toegezegde periode men niet meer kan inloggen met het schrijfaccount. Er dient dan bij REM een nieuwe login te worden aangevraagd.

**Mocht het zo zijn dat men scripts wilt draaien waarvoor het schrijfaccount niet bevoegd is,** dan dient voor deze acties een aparte aanvraag te worden gedaan via support. In deze gevallen wordt vaak op afspraak een moment gepland dat de scripts of door Rem zelf uitgevoerd zullen worden op de database, of dat men tijdelijk toegang krijgt tot een zogenaamde superuser.

From:  
<https://doc.open-wave.nl/> - Documentatie

Permanent link:  
[https://doc.open-wave.nl/doku.php/openwave/1.31/techniek/database\\_lees\\_schrijfrechten?rev=1732597677](https://doc.open-wave.nl/doku.php/openwave/1.31/techniek/database_lees_schrijfrechten?rev=1732597677)

Last update: 2024/11/26 06:07

