# Oracle en Security (5)

## *Oracle Security Alert #68*

***Sinds begin dit jaar is Oracle er toe overgegaan om geen security patches meer uit te leveren voor individuele security lekken. Voortaan zullen met een vooraf bekende regelmaat security patches worden uitgebracht, waarin alle tot dan toe opgeloste security lekken gebundeld worden aangeleverd. Oracle voert als reden voor dit besluit aan, dat het onderhoud op deze manier beter te plannen zou zijn. Microsoft werkt al langer op deze manier.***

Oracle gaat hier mijns inziens voorbij aan een belangrijk verschil tussen Oracle- en Microsoft-sites. Bij een Microsoft-site is de uptime van systemen meestal veel korter dan de tijd die er zit tussen het uitkomen van twee security patches. Bij vele Oracle-sites is op zijn minst de gewenste uptime veel langer. In het laatste geval is de door Oracle genoemde voordeel met betrekking tot planning alleen maar een planning waarmee de doelstelling van een langere uptime nooit gehaald zal worden. Dat deze afweging (security versus uptime) daadwerkelijk velen beroerd blijkt onder meer uit de vele discussies hierover op internet (bijvoorbeeld vragen als 'Hoe kan ik zien of deze patch voor mijn site noodzakelijk is').

### Enquête

Een mooie inventarisatie hoe DBA's omgaan met de laatste security patch komt van een Belgische DBA-site (www.dba-village.com). In een online enquête werd DBA's gevraagd, welke actie ze hadden ondernomen naar aanleiding van Security Alert 68 (zie tabel 1)

***Security awareness moet het doorgaans afleggen tegen beschikbaarheidseisen***

**"What action do/did you take for the recent Security Alert 68?"**

| Antwoord | Percentage |
|---|---|
| I don't know security alert 68 | 33% |
| Not urgent, one day I'll apply that patch | 30% |
| Serious, I applied the patch immediately | 20% |
| No risk in my environment, no need to patch | 17% |

*Tabel 1. Uitslag van een online poll op de website www.dba-village.com*

Een aantal opmerkingen hierover:

- Toen ik de site bezocht hadden een kleine 500 DBA's gereageerd;
- Voor een derde van hen is security blijkbaar nog geen issue;
- Bijna een vijfde concludeerde dat de patch voor hun omgeving geen risico oplevert. De beschrijving van security patch #68 is dermate algemeen en er worden zoveel issues in behandeld, dat het merkwaardig is, dat sommige DBA's tot deze conclusie zijn gekomen;
- De mogelijkheid "Ik zou hem graag willen aanbrengen, maar er is nog geen patch aangeboden voor de versie die ik heb draaien" is een reële mogelijkheid. Voor versie 9.2.0.4 was er eind september nog geen patch. Toch is voor velen de upgrade naar 9.2.0.5 geen optie door een serieuze bug.

Het lijkt er dus op, dat de security awareness toeneemt, maar dat die awareness het doorgaans moet afleggen tegen andere eisen als beschikbaarheid.

### Buiten de applicatie om

Security patches (zoals die ten behoeve van Oracle security alert #68) zijn met name bedoeld om ervoor te zorgen dat gebruikers die toegang hebben verkregen tot het interne netwerk, maar geen toegang hebben tot de Oracle databases, verhinderd worden kwaad aan te richten. In dit deel wil ik beschrijven hoe voorkomen kan worden dat gebruikers die wel legaal toegang hebben tot de Oracle databases, al dan niet

gewild verkeerde dingen doen. Hiervoor zijn verschillende opties beschikbaar, waaronder PRODUCT_USER_PROFILE.

Al in 1996 heeft Oracle de eerste poging gedaan om er gestructureerd voor te kunnen zorgen dat gebruikers niet op een andere manier dan via de applicatie in de database kunnen komen. Er is hiervoor een tabel gemaakt (PRODUCT_USER_PROFILE) gemaakt waarin opgeslagen kan worden welke gebruiker, welke commando's in welke tool mag gebruiken. Voor SQL*Plus werkt dit prima. Alleen hebben diverse third party tool-leveranciers zoals met name TOAD niet meegewerkt. Indien eindgebruikers niet zelf software op het net kunnen zetten en SQL*Plus het enige tool is waar ze bij kunnen, is er veel mee mogelijk en wel per gebruiker (zie tabel 2).

## V$SESSION

In de v$views is niet alleen informatie beschikbaar over de sga, maar ook over de processen die draaien. In V$SESSION is bijvoorbeeld te vinden welke executable er gebruikt wordt. Dit kan gebruikt worden om "bepaalde executables alleen in bepaalde gevallen" te laten gebruiken. In het onderstaande voorbeeld afkomstig van metalink, wordt hier door middel van een database trigger gebruik gemaakt om af te dwingen dat alleen DBA's gebruik mogen maken van TOAD. De onderstaande trigger moet als SYS worden aangemaakt:

```
create or replace trigger on_logon after logon on database
declare
   t_isdba    varchar2(10);
   t_program v$session.program.type;
begin
   execute immediate 'select program
                      from   v$session
                      where  audsid =
sys_context(''userenv'', ''SESSIONID'')'
                      into   t_program;
   execute immediate 'select
sys_context(''userenv'',''ISDBA'')
                      from   dual'
                      into   t_isdba;
   if  upper(t_program) = 'TOAD.EXE'
   and t_isdba          = 'FALSE'
   then
     raise_application_error(-20001,'Toad is alleen voor dba''s',true);
   end if;
end;
/
```

Dit leidt ertoe dat de volgende foutmeldingen als een niet-DBA probeert aan te loggen met TOAD:

ORA-00604: error occurred at recursive SQL level 1
ORA-20001: TOAD is alleen voor DBA's
ORA-06512: at line 11

Een paar opmerkingen hierbij: onder een DBA wordt hier verstaan een gebruiker met de rol OSDBA. De rol DBA is niet genoeg. De trigger gaat alleen af voor een executable met de naam TOAD.EXE. Het zou beter zijn de kolom MODULE te gebruiken. Die wordt namelijk in de executable gezet en kan niet aangepast worden, maar deze kolom wordt pas gevuld nadat de ON_LOGON trigger is afgegaan. Let erop dat als een logon trigger niet compileert, dit grote gevolgen heeft. (Bijna) niemand kan meer aanloggen. Dit geldt nog meer voor een startup trigger van de database. In het uiterste geval is het mogelijk om de database op te starten zonder dat er database triggers afgaan door het zetten van een initialisatie parameter (_SYSTEM_TRIG_ENABLED=FALSE).

## Virtual Private Database

In Oracle10g bestaat er wel een mogelijkheid om de informatie van MODULE te gebruiken, omdat deze is opgenomen in de userenv. Dit kan met de Virtual Private Database (VPD) gecontroleerd worden. Er kan per object gechecked worden of een bepaalde policy functie van toepassing is. Als policy functie gebruiken we dan:

```
create or replace function geen_toad (schema in varchar2
                                      ,object in varchar2)
return varchar2
as
begin
  return
'upper(substr(sys_context(''userenv'',''module''),1,4))<>''
TOAD''';
end;
/
```

Voor objecten die we tegen het gebruik in TOAD willen beschermen kunnen we deze policy functie als volgt gebruiken:

```
begin
  dbms_rls.add_policy
     (OBJECT_SCHEMA    => 'EIGENAAR'
     ,OBJECT_NAME      => 'OBJECT_NAAM'
     ,POLICY_NAME      => 'POLICY_NAAM'
     ,FUNCTION_SCHEMA => 'SYS'
     ,POLICY_FUNCTION => 'GEEN_TOAD'
     ,STATEMENT_TYPES => 'SELECT,INSERT,DELETE,UPDATE'
     ,UPDATE_CHECK     => TRUE
     ,ENABLE            => TRUE
     ,STATIC_POLICY    => FALSE);
end;
/
```

Indien het 'noodzakelijk' is om iemand toch toe te staan om met TOAD dit object te benaderen, is dit mogelijk door hem het systeem privilege EXEMPT ACCESS POLICY te geven.

| Verboden actie vanuit SQL*Plus | Verbied het gebruik van de commando's |
|---|---|
| In zijn geheel niet gebruiken | CONNECT |
| OS commando's uitvoeren [1] | HOST |
| DML commando's | INSERT,DELETE,UPDATE,COPY |
| DDL commando's | CREATE, DROP,TRUNCATE |
| Procedures uitvoeren | EXECUTE |
| DScripts draaien | RUN |

*1] Dit werkt niet voor PL/SQL procedures, die zelf inherent iets op het OS doen*

*Tabel 2. De tabel PRODUCT_USER_PROFILE voor SQL*Plus*

## Binnen de applicatie

Bovenstaande oplossingen voor het probleem van ongeautoriseerde toegang buiten de applicatie om zijn eigenlijk lapmiddelen. Eigenlijk is een applicatie niet compleet als er niet voor wordt gezorgd, dat de data structureel worden beveiligd tegen oneigenlijk of onjuist gebruik. Dat betekent niet, dat deze lapmiddelen niet nodig zijn. Vele applicaties krijgen we tenslotte zoals ze zijn en niet zoals we ze willen hebben. In deze twee laatste paragraven schets ik een aantal methodes, waarmee het binnen de applicatie geregeld kan worden. Een ruwe, maar goed werkbare methode is:

- Wijs gebruikers binnen de applicatie rollen toe die beschermd zijn met een wachtwoord;
- Laat de applicatie dit wachtwoord bepalen en niet de gebruiker;
- Zorg ervoor dat deze wachtwoord bepaling niet statisch is.

Netter is het om de business logica geheel binnen de database te implementeren. Indien deze goed is uitgezocht kunnen wijzigingen beperkt blijven tot correcte wijzigingen, uitgevoerd door gebruikers die dit mogen door:

- Constraints
- Triggers
- Procedures

Het gebruik van "constraints" en "triggers" is hierbij evident. Database procedures kunnen hiervoor op de volgende manier gebruikt worden:

- Sta nooit wijzigingen toe op individuele database objecten;
- Implementeer wijzigingen door database procedures die een complete business transactie afhandelen.

**Gerard Uiterwaal**
is Oracle- en security expert en werkzaam bij Motiv IT Masters. Indien u in een van de volgende uitgaven van Optimize graag een specifiek onderdeel belicht zou willen zien dan kunt u dit aangeven via e-mail: gerard.uiterwaal@motiv.nl. De auteur streeft ernaar zoveel mogelijk vragen te beantwoorden.


## Geodan en Jentro winnen Oracle Award voor innovatie

Oracle heeft de eerste Oracle PartnerNetwork Innovation Award voor de EMEA-regio toegekend aan het Nederlandse Geodan Mobile Solutions. Geodan kreeg de eerste prijs gewonnen voor Geodan Movida, een geïntegreerde oplossing voor mobiele locatiediensten. De tweede prijs ging naar het Duitse Jentro Technologies voor activepilot, een op Java gebaseerde off-board navigatieservice via de mobiele telefoon. De awards zijn in het leven geroepen om innovatie bij partners te stimuleren en te belonen. Ze brengen oplossingen, diensten of initiatieven onder de aandacht waarin technologie van Oracle is toegepast om meerwaarde voor de klant te creëren en nieuwe afzetmogelijkheden te ontwikkelen. Met Geodan Movida kunnen informatiegegevens over locaties vanuit verschillende technologieën worden geïntegreerd. Wi-Fi, RFID en mobiele technologieën voorzien applicaties die draaien op Oracle 10g van gestandaardiseerde, real-time lokalisatiegegevens. Een ziekenhuis kan bijvoorbeeld zijn ambulances lokaliseren met behulp van GPS of mobiele technologieën. Met RFID kan het materiaal worden gemerkt en kan bijvoorbeeld het aantal diefstallen worden teruggedrongen, en met Wi-Fi kan belangrijke medische apparatuur worden getraceerd en zo onmiddellijk worden gelokaliseerd. Geodan Movida brengt al deze informatie samen, standaardiseert het en maakt het in real-time beschikbaar voor de database en applicaties. Geodan Mobile Solutions is een Oracle Certified Partner.