

Toptien van Security-scans bren

Nu de meeste bedrijven hun netwerkbeveiliging op orde hebben, is databasebeveiliging niet graag op straat ziet liggen. Het is dus zaak om ook hier de beveiliging hoog

T-dienstverlener Motiv voerde in de afgelopen jaren tal van security-scans uit op databases van klanten. Daaruit kwam naar voren dat er op databasegebied nog veel moet gebeuren. Zo kon Motiv in 90 procent van de uitgevoerde scans de database binnenkomen doordat er standaardaccounts (meegeleverd door de databaseleverancier) met standaardwachtwoorden gebruikt werden. Welke kwetsbaarheden komen het meest voor bij databases? Hoe voorkomt een bedrijf problemen op dit gebied? In dit artikel vindt u op deze vragen een antwoord. Het accent ligt hierbij op de Oracle-database, maar dezelfde problematiek is ook terug te vinden bij bijvoorbeeld de SQL*Server van Microsoft.

Om een toptien van Oracle-beveiligingskwetsbaarheden te kunnen samenstellen, moeten we eerst bepalen waarom de ene kwetsbaarheid boven de andere staat. We zien regelmatig toptienlijsten op allerlei gebied. Zelden wordt duidelijk gemaakt in wat volgorde deze beveiligingskwetsbaarheden gerangschikt zijn. Voor dit artikel gaat het om een toptien van *meest voorkomende* beveiligingskwetsbaarheden, zoals we die in de afgelopen jaren hebben gevonden bij security-scans.

Toptien

1. Standaard accounts met standaard wachtwoorden

Oracle wordt standaard uitgeleverd met een aantal standaard accounts zoals DBSNMP. De aanwezigheid van

standaard accounts maakt het mogelijk om de hashes van wachtwoorden eenvoudig uit te lezen en offline via een Brute Force Attack de wachtwoorden van deze standaardaccounts te kraken. Als deze standaardaccounts gekraakt zijn, heeft een kwaadwillende gebruiker voldoende privileges om het werkproces binnen een organisatie te verstoren.

2. Geen limiet op foute inlogpogingen

Als er geen limiet aan het aantal foute login-pogingen wordt gesteld, is het mogelijk om via een Brute Force Attack toegang tot de database te krijgen. Door het aanpassen van een parameter van de database is het eenvoudig mogelijk om deze beveiligingszwakheid te neutraliseren.

3. Zwakke wachtwoorden

Als er zwakke wachtwoorden aanwezig zijn binnen de database, lukt het vaak om binnen een half uur wachtwoorden te achterhalen en in te loggen op de database. Meestal hebben deze accounts dan voldoende rechten om gegevens uit de database te verzamelen. Niet alleen zwakke wachtwoorden zijn een probleem. Binnen veel bedrijven maakt men ook vaak gebruik van elkaars inloggegevens. Hierdoor is het moeilijker te achterhalen wie wanneer wat gedaan heeft.

4. Listener slecht beschermd

De listener is een onderdeel van Oracle, dat de connectie tussen gebruikers en de database regelt. Zonder listener is

het als gebruiker niet mogelijk om een connectie met de database te maken en daardoor zijn dan ook de bedrijfsgegevens onbereikbaar. Oracle biedt mogelijkheden om de listener te beschermen, maar de meeste organisaties schenken hier weinig aandacht aan.

5. Verkeerde file-permissies

Indien de file-permissies op de database files verkeerd staan, kan een database nog zo goed beveiligd zijn, maar is het met name op een Unix-systeem toch mogelijk om de database in zijn geheel weg te gooien. Als de database eenmaal uit de lucht is, komt het primaire proces van de organisatie stil te liggen, met alle negatieve gevolgen van dien.

6. Niet geïnstalleerde security patches

Oracle brengt elke drie maanden een securitypatch uit. Deze worden bij veel bedrijven om verschillende redenen niet onmiddellijk geïnstalleerd. Een veel gehoord argument is dat men 24*7 online moet zijn en er dus weinig of geen tijd is om de patch in een testomgeving te testen. Daarnaast is er het nadeel dat Oracle aangeeft welke problemen opgelost zijn met de security patch. Dit geeft hackers ook meteen inzicht wat zij kunnen gebruiken om een database te hacken in het geval dat de laatste security patches niet geïnstalleerd zijn.

7. Gebruikers met teveel rechten op databaseobjecten

Bij het ontwikkelen van applicaties

database-beveiligingskwetsbaarheden gen risico's in kaart

ging een logische volgende stap. Binnen databases is een schat aan gegevens te vinden die men op de agenda te houden. Maar in de dagelijkse praktijk blijkt dit toch lastig. GERARD UITERWAAL

wordt over het algemeen niet goed gekeken hoe om te gaan met de rechten van eindgebruikers op database-objecten. Dit leidt er vaak toe dat (eind)gebruikers te veel rechten hebben op de objecten. Het gevolg is dat zij gegevens kunnen benaderen die niet nodig zijn voor hun functie. Zij kunnen daardoor misbruik maken van de overbodige privileges. Zeker

9. Wachtwoorden in clear text opgeslagen in database of programmatuur

In veel gevallen is een database zo ingericht dat bij koppelingen naar andere databases de wachtwoorden zonder encryptie opgeslagen worden. Hierdoor zijn deze wachtwoorden uit een tabel in de database op te vragen. Een ander veel voorkomend probleem is dat veel

gehouden worden met een technisch en een organisatorisch aspect.

Technische oplossingen zijn meestal overzichtelijk. De consequenties zijn goed in te schatten en de implementatietijd is meestal goed te voorspellen. Bij bedreigingen waarvan de oplossing organisatorische aspecten heeft, moeten beslissingen veelal door meer mensen genomen worden. Vaak heeft

Als deze standaardaccounts gekraakt zijn, heeft een kwaadwillende gebruiker voldoende privileges om het werkproces te verstoren

wanneer gebruikers een tool zoals discoverer of SQL*Plus ter beschikking hebben, waarmee ze selecties op de databases kunnen maken, kunnen gebruikers toegang krijgen tot vertrouwelijke informatie die niet voor hun ogen bestemd is.

8. Geen of gebrekkige auditing

Het automatisch vastleggen van transacties die plaatsvinden op vertrouwelijke data geeft de organisatie een beter beeld van wie er wanneer gebruikt maakt van databaseobjecten. Onderzoek kan dan uitwijzen of gebruikers terecht toegang hebben tot bepaalde objecten. Met een goed ingestelde auditing en bijbehorende alarmering kan de organisatie afwijkend databasegebruik vaststellen en de nodige maatregelen treffen. Het is uiteraard niet in alle gevallen nodig om alles te auditen, maar een aantal basisaudits is toch zeer gewenst. Zeker met het oog op eventueel forensisch onderzoek.

applicaties gebruik maken van batch-programmatuur waar in de code een gebruikersnaam en wachtwoord in clear text is opgenomen. Deze programmatuur wordt in een batch opgestart en maakt dan connectie met de database.

10. Objecten die ten onrechte in de system tablespace staan

De system tablespace is een gedeelte van de database waar gegevens opgeslagen worden die Oracle gebruikt om zijn database draaiende te houden. In dit gedeelte worden bijvoorbeeld de gebruikersnamen met wachtwoorden opgeslagen. Indien tabellen die niet thuishoren in de system tablespace daar wel geplaatst worden, is het mogelijk dat dit gedeelte van de database volloopt. De database kan dan stoppen met werken.

Moelijkheidsgraad

Om vast te stellen hoe moeilijk het is om een zwakte aan te pakken en hoeveel tijd dat kost, moet rekening

de implementatie ook invloed op de manier van werken. Een voorbeeld hiervan is dat het bij veel bedrijven gebruikelijk is dat ontwikkelaars toegang hebben op de productiedatabase met gevoelige data. Volgens diverse regelgevingen is dat niet toegestaan. Wanneer het toch noodzakelijk is dat een ontwikkelaar op de productiedatabase toegang krijgt om een probleem op te lossen dan moet er een procedure doorlopen worden om de ontwikkelaar tijdelijk toegang te geven tot de productiegegevens. Zo'n procedure wordt vaak als lastig en tijdrovend beschouwd maar is voor de beveiliging van gevoelige data noodzakelijk.

Hoe gevaarlijk zijn beveiligingsbedreigingen eigenlijk? In de beveiligingswereld wordt dit klassiek berekend als *de kans dat een zwakte misbruikt wordt vermenigvuldigd met de kosten hiervan*. Omdat de kosten van een bedreiging afhangen van de organisatie, is ook het gevaar van een bedreiging niet objectief vast te stellen. ▶



Voor een gestructureerde aanpak van de beveiliging van de database moet de organisatie een aantal stappen doorlopen.

Plan van aanpak

Het is duidelijk dat ook databases op een goede manier beveiligd moeten worden. Daarvoor is een gestructureerde aanpak nodig waarbij de organisatie een aantal stappen doorloopt. Dit is en blijft overigens een iteratief proces. Iedere keer worden immers nieuwe beveiligingszwakheden ontdekt, maar ook interne regelgeving kan veranderen.

gingsplan en de gevonden problemen omgezet in regels en toegepast op de database.

4. Bewaking risico's en beveiligingsregels.

Door de geïmplementeerde regels te bewaken, zorgt de organisatie ervoor dat er op elk moment inzicht is in de status van de databasebeveiliging.

5. Vastleggen DBMS-gebruik In het beveiligingsplan is ook opgenomen welke acties geaudit moeten worden. Denk

Technische invulling

In het hiervoor beschreven stappenplan staat de organisatorische kant van de zaak centraal. Het is daarnaast ook nodig om een structuur op te zetten voor de implementatie van de technische maatregelen. Omdat er continu nieuwe (Oracle-)beveiligingszwakheden ontdekt worden, is het niet mogelijk om een of meer projecten te definiëren waarbinnen alle zwakheden opgelost worden. Om de werkzaamheden overzichtelijk te houden, is het aan te raden om een aantal trajecten te definiëren. De trajecten kunnen worden ingedeeld aan de hand van de tijd die een medewerker nodig heeft op het beveiligingslek te dichten:

- ▶ Het oplossen van beveiligingszwakheden die minder dan een halve dag werk kosten.
 - ▶ Het oplossen van beveiligingszwakheden die een dag werk kosten.
 - ▶ Het oplossen van beveiligingszwakheden die meer dan een dag werk kosten.
 - ▶ Het beoordelen van nieuwe beveiligingslekken en het aan de hand van die beoordeling plaatsen in één van de bovenstaande trajecten.
- Tevens wordt hierbij de prioriteit bepaald van de nieuwe beveiligingslekken ten opzichte van de in dit traject nog openstaande beveiligingslekken. Deze trajecten worden uitgevoerd door teams bestaande uit security

Security officers houden zich voornamelijk bezig met de organisatorische aspecten en de dba's met de technische aspecten

1. **DBMS-beveiligingsplan** De eerste stap is het vaststellen van de beveiligings-eisen, die de organisatie opneemt in een databasebeveiligingsplan. De basis van dat plan is in de regel het informatiebeveiligingsplan zoals dat binnen de organisatie aanwezig is (of hoort te zijn).

2. **Controle database** Door regelmatig scans uit te voeren, is controle mogelijk op de aanwezigheid van eventuele lekken of kwetsbaarheden in de database.

3. **Implementatie van het beveiligingsplan.** In deze stap worden het DBMS-beveili-

gelingen aan voorschriften en regelgevingen. Daarnaast kan deze vastlegging later gebruikt worden voor onderzoek naar het gebruik van de databasegegevens.

6. **Review databasebeveiliging** Een regelmatige review van waarschuwingen en de systematische vastlegging van het databasegebruik leiden tot het verder beveiligen van de database-omgeving. Daarbij kunnen items uit de reviewfase opgenomen worden in het beveiligingsplan, waarna ze geïmplementeerd worden in de database en de organisatie.

officers en dba's (senior en junior). Daarbij houden de security officers zich voornamelijk bezig met de organisatorische aspecten en de dba's met de technische aspecten.

Nog geen halve dag

Nadat vastgesteld is of de oplossing beter is dan de kwaal, kan de organisatie onmiddellijk aan de slag. Het is niet nodig prioriteiten te stellen. Voorbeelden van dit soort beveiligingslekken zijn:

- ▶ Beveiliging van de listener
- ▶ Verwijderen van default wachtwoord

den en standaard accounts

- ▶ Goed instellen van file permissies
- In zijn algemeenheid zijn dit puur technische wijzigingen, die geen impact hebben op het normale gebruik van de database. Natuurlijk moet er wel goed geïnventariseerd worden of er geen gebruik gemaakt wordt van de default accounts.

Een tip voor de technici: bij het wijzigen van het wachtwoord van dbsnmp is het handig om ook create script van dbsnmp in de directory rdbms/admin aan te passen. Anders zal een en ander bij de eerstvolgende patch worden teruggezet. Bij het installeren van een nieuwe versie van Oracle moet het script natuurlijk weer worden aangepast.

Een dag werk

Bij beveiligingszwakheden waarvan de oplossingstijd in de orde van grootte van een werkdag ligt, is een echte planning niet nodig. Het volstaat om deze op de *to do*-lijst van een (groep van) medewerkers te plaatsen. Het tijdstip waarop deze beveiligingszwakheden daadwerkelijk worden aangepakt, is afhankelijk van de actuele workload van de beheerders. Het vaststellen van prioriteiten is hier wel van belang. Voorbeelden van dit soort beveiligingszwakheden zijn:

- ▶ Voorbereiden en op kleine schaal invoeren van auditing en monitoring
- ▶ Stroomlijning van uitgegeven privileges
- ▶ Installatie van patches

Indien er nog geen auditing is ingevoerd, kan men het beste beginnen met het auditen van foutieve inlogpogingen. De hoeveelheid data die verzameld wordt, is overzichtelijk en het belang ervan is groot. Zorg ervoor dat privileges niet direct, maar via rollen worden toegelast.

Meer dan een dag

Voor deze beveiligingszwakheden waarvan de oplossing meer dan een dag werk kost, is de projectvorm de beste keus. Het belangrijkste verschil met de vorige categorie is, dat er op deze manier een eindtijd kan worden gepland en dus bewaking mogelijk is.

Een voorbeeld van dit soort beveiligingszwakheden is:

- ▶ Configuratie van auditing en monitoring
- ▶ Installatie uitgebreide patches

Bij het configureren van auditing zal eerst een inventarisatie gemaakt moeten worden van wat men wil auditen. Daarna zal gekeken moeten worden

In de system tablespace worden gegevens opgeslagen die Oracle gebruikt om zijn database draaiende te houden

wat de technische consequenties zijn. En als laatste en belangrijkste punt: de uitkomsten van de auditing moeten worden gemonitord.

Beoordeling

Binnen dit traject worden nieuwe security threats beoordeeld en in één van de bovenstaande categorieën ingedeeld. Tevens wordt er gekeken of de voorgestelde oplossing organisatorische consequenties heeft en hoeveel kennis er nodig is om de oplossing te implementeren. Beveiligingszwakheden die tussendoor kunnen worden opgelost, krijgen een prioriteit en voor de langetermijnoplossingen wordt een planning gemaakt. Voor alle categorieën wordt tevens bepaald wie (welke groep) het probleem krijgt toegewezen.

Verantwoording

Om een toptien van Oracle-beveiligingskwetsbaarheden te kunnen samenstellen, moet eerst bepaald worden waarom een bepaalde beveiligingskwetsbaarheid boven een andere komt te staan. Er worden regelmatig toptien-overzichten gepubliceerd op verschillende gebieden. Zelden wordt duidelijk gemaakt in welke volgorde deze beveiligingskwetsbaarheden gerangschikt zijn. Voor dit artikel werd gekozen voor de 10 meest voorkomende beveiligingskwetsbaarheden zoals Motiv die gevonden heeft in security-scans voor klanten.

Er zijn uiteraard ook andere mogelijkheden om beveiligingszwakheden te rangschikken. Denk hierbij aan:

- ▶ Hoeveel kost het een organisatie als een security threat daadwerkelijk misbruikt wordt?
- ▶ Hoeveel resources kost het om een security threat dicht te timmeren?
- ▶ Hoe gemakkelijk is het misbruik te maken van een security threat?
- ▶ Hoe moeilijk is het om een security threat te neutraliseren?

- ▶ Hoe lang duurt het om een security threat dicht te timmeren?

Tot slot

In dit artikel kwam vooral de technische kant van beveiligingszwakheden aan bod. Het is uiteraard voor iedere organisatie van belang om een totaal-aanpak voor informatiebeveiliging te creëren, inclusief een organisatiebreed informatiebeveiligingsplan. Wanneer dat niet gebeurt, blijft beveiliging een puur technische zaak en zal het draagvlak voor beveiliging en beveiligingsmaatregelen nooit optimaal zijn.

Toptien Oracle-beveiligingskwetsbaarheden

- ▶ **Standaard accounts met standaard wachtwoorden**
- ▶ **Geen limiet op gefaalde inlogpogingen**
- ▶ **Zwakke wachtwoorden**
- ▶ **Listener slecht beschermd**
- ▶ **Verkeerde file permissies**
- ▶ **Niet geïnstalleerde security patches**
- ▶ **Gebruikers met teveel rechten op databaseobjecten**
- ▶ **Geen of gebrekkige auditing**
- ▶ **Wachtwoorden in clear text opgeslagen in database of programmatuur**
- ▶ **Objecten die ten onrechte in de system tablespace staan**