Oracle-Datenbank-Security-Monitoring

Carsten Mützlitz, ORACLE Deutschland B.V. & Co. KG

Wann haben Sie die Sicherheit Ihrer Oracle-Datenbank das letzte Mal überprüft? Kennen Sie die Bedrohungen und die davon abgeleiteten Risiken? Vielleicht sagen Sie jetzt: „Meine Risiken sind unter Kontrolle. Ich kenne den notwendigen Schutzbedarf und habe diesen in meiner Datenbank bedacht." Für alle anderen fasst dieser Artikel ein paar Fakten darüber zusammen, was passiert, wenn Sie kein Security-Monitoring aktiviert haben. Es geht um das Wissen von Bedrohungen und Risiken und die damit verbundene und notwendige Transparenz und Kontrolle. Die Lösung dahinter beschreibe ich als Security-Monitoring.

Natürlich bietet Oracle verschiedene Möglichkeiten für ein Security-Monitoring out-of-the-box an. Hierunter fallen folgende Lösungen:

- *Oracle Database Audit*
  Standard-Funktionen in der Datenbank, die Aktivitäten innerhalb der Datenbank protokollieren. In der neuen Datenbank-Version 12*c* heißt dieses Audit „Unified Auditing".
- *Oracle Database Vault*
  Lösung, um zusätzliche Sicherheitszonen in der Datenbank einzuziehen, die Regeln in Form von Policies erzwingen und den Gebrauch und Missbrauch dieser Policies automatisch protokollieren.
- *Oracle Audit Vault & Database Firewall*
  Ein zentraler Protokoll-Server, der alle Aktivitäten, die in der Datenbank stattfinden, revisionssicher protokolliert und gegebenenfalls auch blockieren kann, ebenso wie Kommandos, bevor sie die Datenbank erreichen.
- *Oracle Compliance Dashboard*
  Eine Funktionalität innerhalb des Enterprise Manager Cloud Control 12*c*, die man mit dem „Lifecycle Management Pack" erwirbt. Damit können entsprechende Security-Policies überwacht und auf Einhaltung überprüft werden. In der Regel sind das Konfigurations-Einstellungen.

Dieser Artikel zeigt, was passiert, wenn man überhaupt kein Security-Monitoring betreibt. Dafür wurde eine kleine Auswahl sogenannter „DB Security Top Issues" zusammengestellt, die entstehen, wenn der Blick auf die Security fehlt. Der Autor hat in der Vergangenheit viele sogenannte „Datenbank-Security-Reviews" durchgeführt, also ein manuelles Security-Monitoring für eine ausgewählte Datenbank. Diese zeigen den Sicherheitszustand einer Datenbank auf und betrachten hierbei folgende Bereiche:

- Konfiguration der Datenbank entsprechend des Schutzbedarfs
- Monitoring- und Audit-Einstellungen innerhalb und außerhalb der Datenbank
- Einstellungen und Konzept-Implementierungen hinsichtlich Verfügbarkeit
- Die eingestellte Zugriffskontrolle, auch für Daten, die die Datenbank verlassen
- Maßnahmen für die Compliance-Einhaltung sowie Nachhaltigkeit

Aus vielen Untersuchungen nachfolgend eine Auswahl der „DB Security Top Issues".

Top Issue 1: Kaum Wissen und Verantwortungen definiert

Die Erfahrung aus vielen manuellen Security-Monitorings zeigt, dass selten das Wissen zu dem notwendigen Schutzbedarf vorhanden ist. Es werden weder Daten sicherheitstechnisch klassifiziert noch sind die teilweise zwingenden und unternehmensabhängigen Anforderungen auch aus den Gesetzen wie Bundesdatenschutzgesetz, Sozialgesetzbuch, SOX, PCI DSS etc. im Detail bekannt. Somit besteht selten ein wirkliches Wissen über den notwendigen Schutzbedarf bei den Personen, die für den Schutz sorgen müssen.

Maßnahme: Es sind organisatorische Maßnahmen zu treffen, die dafür sorgen, dass der Sicherheitszustand einer Datenbank in der Verantwortung einer definierten Business-Rolle liegt. Der Schutzbedarf für die klassifizierten Daten muss bekannt sein, Maßnahmen müssen umgesetzt werden.

Top Issue 2: Selten eigene Mindestsicherheit implementiert

Eine unternehmensweite Mindest-Sicherheit aller Datenbanken ist sehr selten implementiert, obwohl Oracle hier Vorgaben und Anregungen definiert, wie im „Oracle Database Security Guide 11*g*, Chapter 10, Keeping your database secure" beschrieben. Wenn von Mindestsicherheit die Rede ist, dann sind damit insbesondere die Maßnahmen gemeint, die jede Datenbank aktiviert haben muss. Man könnte auch sagen, dass ein IT-Grundschutz entsprechend des Bundesamts für Sicherheit in der Informationstechnik (BSI) einheitlich umgesetzt ist. Natürlich bietet eine Oracle-Datenbank viele Funktionen, die bei Aktivierung eine Sicherheit erhöhen können. Diese gilt es einzuschalten.

Maßnahme: Das Kapitel 10 im Database Security Guide (für die Datenbank 12*c* steht das Kapitel im Anhang) anwenden und darauf eigene Anpassungen vornehmen, sodass ein einheitlicher Unternehmensstandard für die Mindestsicherheit besteht. Es existiert außerdem in der My Oracle Support Knowledge Base ein Artikel: „10

Basic Steps to Make your DB secure from Attacks (ID 1545816.1)“. Dieser Unternehmensstandard kann dann mit entsprechenden Tools auf Einhaltung geprüft werden.

Top Issue 3: Schwache Zugriffskontrolle und Zwecktrennung

In den Bereichen „Zugriffskontrolle“ und „Zwecktrennung“ gibt es verschiedene Konzepte, die teilweise sehr veraltet oder zu komplex sind. Andere ergeben durchaus Sinn, werden jedoch durch andere Konzepte ausgehebelt. Im Bereich der Zugriffskonzepte sind die größten Fettnäpfchen:

- Nutzung von Standard-Rollen wie „CONNECT“ oder „RESOURCE“ für Nicht-Admin-User. Die „RESOURCE“-Rolle vergibt Rechte für die Anlage von Objekten und „UNLIMITED TABLESPACE QUOTA“ (wurde ab Version 12*c* aufgehoben). Wenn man Standard-Rollen an Datenbank-Benutzer vergibt, muss man die Berechtigungen in den Rollen kennen und entscheiden, ob diese wirklich notwendig sind. Oft wird die „RESOURCE“-Rolle vergeben, obwohl diese Benutzer gar keine eigenen Objekte besitzen. Wofür benötigt der Datenbank-Benutzer dann die „RESOURCE“-Rolle?
- Nutzung keiner Rollen, sondern ausschließliche und explizite Vergabe von Privilegien an den Benutzer. Das erhöht die Komplexität in der Zugriffskontrolle, da für weitere Benutzer viele „GRANTS“ vergeben werden müssen anstelle eines Rollen-„GRANT“.
- Sehr komplexe Zugriffskonzepte mit vielen Tausend „GRANTS“, doppelten Berechtigungen und redundanten Privilegien. Diese komplexen Zugriffskonzepte sind schwer zu kontrollieren. Eine Vereinfachung ist sofort sichtbar, wenn man das Berechtigungskonzept genauer betrachtet.
- Aushebelung der Zugriffskontrolle durch Vergabe von Privilegien an „PUBLIC“ sowie Erstellung von „PUBLIC Database“-Links auf die gleiche Datenbank mit Zugriff auf die Anwendung.
- Zu mächtige „GRANTS“ an Anwendungs-Owner mit „ANY“-Privilegien, die auch gleichzeitig als Anwendungs-User genutzt werden. Diese Benutzer werden von der Anwendung genutzt, die dann die Autorisierung vornimmt, aber auch von Endbenutzern, die mit Tools wie SQL-Developer auf den Daten operieren. Diese Endbenutzer können dann alle mächtigen Privilegien anwenden, da die Autorisierung in der eigentlichen Anwendung nicht mehr stattfindet. Zusammenfassend:

- Kein Least-Privilege-Konzept implementiert.
- Massive Nutzung von Datenbank-Links, die unsicher konfiguriert sind.
- Anlage einer Password-Datei mit vielen SYSDBAs; eine Remote-Administration der Datenbank wird aber nicht ausgeübt und eine Standby-Umgebung existiert nicht.
- Autorisierung übernimmt die Anwendung und die Datenbank liefert ihr mächtige Zugriffsrechte.

In der Zwecktrennung fordern die Gesetze eine Personalisierung der DBAs, sodass Admins tatsächlich eindeutig identifizierbar sein sollten. Stattdessen werden „SYS“ und „SYSTEM“ genutzt. Diese Standardbenutzer werden dann als sogenannte „Shared Accounts“ genutzt und das von vielen DBAs. Auch werden die Privilegien für das Account-Management vermischt, also die Erstellung von Usern und Rollen, sodass viele Personen, auch über Shared-Accounts, für das Account-Management zuständig sind. Die Datenbank 12*c* führt eine gewisse Zwecktrennung für DBAs ein, ein sogenanntes „DBA Segration of Duty (DBA SoD)“.

Maßnahme: Einheitliche Konzepte müssen durchgängig implementiert sein. Die Zugriffskontrolle sollte regelmäßig nach dem „least“-Privileg überprüft werden. Hierfür kann man mit der Datenbank-Version 12*c* die Funktionalität „Privilege Analysis“ verwenden und somit feststellen, welche Privilegien tatsächlich gebraucht werden. Bei der Zwecktrennung gilt, entsprechend dem „Top Issue 1“,Verantwortlichkeiten zu schaffen und eigene Konzepte in der Datenbank zu erzwingen. Man kann in der Datenbank auch eine DBA-Zwecktrennung implementieren. Zu hinterfragen gilt es Konzepte, die unnötig die Zugriffskontrolle verkomplizieren, wie etwa viele Schema-Kopien mit gleichen Tabellen-Strukturen – „Keep it simple“. Wichtig ist, die notwendige Transparenz zu schaffen, damit die Kontrolle nicht verloren geht. Auch eine regelmäßige Kontrolle und Attestierung der Zugriffskontrolle ist notwendig. Hierfür sind solche Prozesse geeignet, die aus dem Identity-Management-Umfeld kommen.

Top Issue 4: Erhöhte Komplexität und Anwendung falscher Konzepte

Komplexe Konzepte in der Datenbank können dazu führen, dass die Sicherheit unnötig leidet. Lieblings-Beispiel des Autors ist ein Konzept zur Erhöhung der Performance, wenn es um sehr viele Daten-Zeilen in einer Tabelle geht. Dafür werden veraltete Daten aus einer Tabelle „A“ in einem Schema „A“ in eine gleiche Tabelle „AA“ eines neuen Schemas „AA“ verschoben. Die Anwendung operiert auf Schema „A“, also auf den aktuellen Daten. Die Trennung der Daten soll dazu führen, dass die Abfragen auf Objekte im Schema „A“ schneller durchlaufen, da nicht mehr so viele Daten gelesen werden müssen. Dieses komplexe Konzept schafft aber Redundanzen in den Objekten (es existieren gleiche Objekte in unterschiedlichen Schemata) und natürlich Redundanzen in der Zugriffskontrolle, denn das gleiche Zugriffskonzept für Objekte im Schema „A“ ist nun auch auf das Schema „AA“ abzubilden.

An diesem Beispiel sollte man erkennen, dass die Zugriffskontrolle so komplex werden kann, dass man irgendwann die Übersicht verliert und damit auch die Kontrolle. Je mehr Schemata man aufbaut, desto komplexer wird das ganze Konstrukt. Abhilfe würde hier das Information-Lifecycle-Management-Konzept von Oracle helfen, eine transparente Lösung, die auf Datenbank-Objekte wie Tabellen abgebildet wird und das gleiche Ergebnis

hat, nämlich Performance, ohne die Komplexität zu erhöhen (siehe http://www.oracle.com/technetwork/database/enterprise-edition/index-090321.html).

Themen wie Datenschutz außerhalb der Datenbank sind auch selten im Fokus. So werden Exports und Backups von Daten durchgeführt, ohne den Datenschutz zu betrachten. Sobald die Daten die Datenbank verlassen, sind sie nicht mehr geschützt. Wenn doch, werden teilweise komplexe organisatorische Konzepte angewendet, die letztendlich aber nicht ausreichend für einen geeigneten Datenschutz sind. Obwohl man mit sehr einfachen Mitteln jeden Dump und jedes Backup mit Datenbankmitteln durch Verschlüsselung schützen kann.

Maßnahme: Die Sicherheitskonzepte der Datenbank kennen, die man transparent anwenden kann, ohne die Komplexität zu erhöhen und ohne die Anwendung in der Arbeit zu behindern. Diese Konzepte nützen auch oft der Sicherheit in der Datenbank.

Top Issue 5: Verfügbarkeit entsprechend Anforderungen?

In Bezug auf die Maßnahmen, die die Verfügbarkeit erhöhen, sind die Datenbank-Betreiber meist gut aufgestellt. Auffallend ist aber, dass Vermutungen angestellt werden. Es werden also zum einen Konzepte benannt, die so nicht umgesetzt sind, wie eine Retention Policy von zehn Tagen im Backup, oder es werden massive Anstrengungen implementiert, ohne genau die Anforderungen an die Verfügbarkeit zu kennen.

Hier sind wir wieder beim „Top Issue 1“: Man muss den Schutzbedarf, also auch die Verfügbarkeitsanforderungen, kennen und entsprechende Maßnahmen implementieren. Wenn Kunden für einen 24x7-Betrieb sehr gute Lösungen implementiert haben wie Dataguard, RAC und ein gutes „Backup&Recovery“-Konzept, dann meistens zum Schutz vor ungeplanten Ausfallzeiten. Ein Schutz vor geplanter Ausfallzeit wird, obwohl die notwendigen Lösungen dafür bereits aktiviert sind, nicht praktiziert. Hier fehlt erneut das Wissen um Konzepte wie dem Transient Logical Standby für Rolling Upgrades (Oracle-HA-Konzepte siehe http://www.oracle.com/technetwork/database/features/availability/twp-dataguard-11gr2-1-131981.pdf, Oracle Patch Assurance – Data Guard Standby-First Patch Apply, Rolling Upgrades (Doc ID 1265700.1), siehe http://www.oracle.com/technetwork/database/features/availability/maa-wp-11g-upgrades-made-easy-131972.pdf und http://www.oracle.com/technetwork/database/features/availability/maa-wp-11g-transientlogicalrollingu-1-131927.pdf).

Maßnahme: Genau die Anforderungen mit entsprechenden Maßnahmen erfüllen und das Wissen über die Anforderungen und über die implementierten Konzepte haben.

Top Issue 6: Schwaches oder gar kein Auditing implementiert

Hier geht es um die Überwachung der Datenbank mittels Audit. Die Gesetzeslage ist eindeutig: Ohne eine geeignete Protokollierung ist die Erfüllung von §9 des Bundesdatenschutzgesetzes nicht möglich. Trotzdem wird das Audit der Datenbank nicht immer aktiviert oder die Aktivierung nur halbherzig durchgeführt. Wichtige SYS-Objekte („sys.user$“ etc.), wichtige Anwendungsobjekte, wichtige und gefährliche Privilegien sowie „SYSDBA“-Aktivitäten sollten immer protokolliert werden. Ausreden, dass mit einer Audit-Protokollierung Unmengen an Daten entstehen, sind falsch. Man muss das Audit nur richtig konfigurieren. Hierbei ist es wichtig zu erkennen, wann gegen die eigenen Konzepte verstoßen wird, und nur dann diese Verstöße zu protokollieren. Dazu ein Beispiel: Bei Zehntausenden von Zugriffen pro Tag auf die Datenbank sollte man nicht „CREATE SESSION“ protokollieren, sondern „CREATE SESSION WHENEVER NOT SUCCESSFUL“, damit zumindest mögliche Brute-Force-Attacken erkannt werden.

Maßnahme: Ein sinnvolles Audit-Konzept aufsetzen. Es ist wichtig, Verstöße gegen eigene Policies (unerlaubte Zugriffe) zu erkennen und nicht alle Aktivitäten zu protokollieren.

Fazit

Wer den Sicherheitszustand nicht kennt, wird auch keine Maßnahmen ergreifen. Daher ist es wichtig, regelmäßig den Sicherheitszustand zu überprüfen. Wie man das macht, welche Erkenntnisse man aus den Einstellungen der Datenbank gewinnen kann und welche Best Practices existieren, beschreibt der Autor ausführlich in seinem neuen Buch „Oracle Security in der Praxis. Vollständige Sicherheitsüberprüfung Ihrer Oracle Datenbank“.

Die Sicherheit der Datenbank ist ernst zu nehmen. Dabei sollte man den wahren Zustand der Datenbank kennenlernen. Wissen über reale Zustände und Wissen über geeignete Konzepte schützt. Erst dann kann man entscheiden, welche Maßnahmen tatsächlich notwendig sind.

Carsten Mützlitz
carsten.muetzlitz@oracle.com

Vorschau

Schwerpunkt-Thema der nächsten Ausgabe ist

„Big Data“

Wir berichten über

- Datenquellen (Hadoop Distributed File System, NoSQL
- Analyse, Data Scientist
- M2M-Kommunikation
- Praktische Erfahrungen

Sie erscheint am
14. Februar 2014