Personal DBA

Günter Unbescheid
Database Consult GmbH
Jachenau

Schlüsselworte

Datenbank, Security, DBA, Personalisierung, neueste Generation der Datenbank

Einleitung

In den letzten Jahren sind die Anforderungen an die IT-Security kontinuierlich gestiegen. Im Kontext von Oracle-Datenbanken ist in diesem Zusammenhang vor allem ein Thema in den Vordergrund gerückt: Personalisierte DBA-Accounts.

Die vielfältigen und vielschichtigen Aktivitäten von Datenbank-Administratoren erfordern naturgemäß umfangreiche Privilegien, nicht nur im Kontext der Datenbanken. Hinzu kommt, dass die Aufgaben-Profile von DBAs sich von Projekt zu Projekt unterscheiden können. Umso stärker wird daher der Ruf nach der Nachvollziehbarkeit und Zuordnung administrativer Aktionen sowie der Eingrenzung der unumschränkten Zugriffsrechte.

Die vielen technischen Möglichkeiten moderner Infrastrukturen müssen mit den konkreten funktionalen Anforderungen abgeglichen werden, um ein Gesamtkonzept zu entwerfen, welches den gesteckten Zielen gerecht wird. Dies ist nicht immer leicht. Der Vortrag zeigt unterschiedliche Lösungsszenarien mit ihren Pros und Cons.

Präludium

Die Menge und Relevanz der Daten, die in Datenbanken gehalten werden, wächst beständig. In diesem Zusammenhang hat auch der Gesetzgeber reagiert und die Rahmenbedingungen verschärft. Gleichzeitig werden administrative Dienstleistungen immer häufiger an externe Anbieter im In- und Ausland ganz oder teilweise vergeben. Die Ausschreibungen in diesem Kontext müssen noch dazu in vorgegebenen Zeitintervallen erneuert werden, so dass es nicht selten zu regelmäßigen Personalwechseln im administrativen Bereich kommen kann.

Diese Entwicklungen bereiten den Boden für eine geforderte Personalisierung administrativer Zugriffe, mit dem Ziel, die Sicht auf vertrauliche Daten weitestgehend einzuschränken, durchgeführte Aktionen nachvollziehbar zu machen und konkreten Personen zuordnen zu können.

„Personal DBA“ auf die personifizierte Anmeldung an den Datenbanken zu reduzieren, hieße jedoch, dass komplexe Thema sträflich zu unterschätzen. Vielmehr kommt es darauf an, das gesamte Aufgabenspektrum eines „DBA“ im Auge zu behalten und vor diesem Hintergrund eine sichere, aber auch praktikable Lösung zu finden. Dies bedeutet: Die Authentifizierung und Autorisierung der Administratoren muss sowohl den Anforderungen im Kontext der Datenbank als auch denen des jeweiligen Betriebssystems genügen, denn ein beträchtlicher Teil dieser Aufgaben spielt sich im letzteren Umfeld ab. Um die Nachvollziehbarkeit der Aktionen revisionssicher zu machen, müssen darüber hinaus Prozesse angepasst und Verantwortlichkeiten verlagert und personell getrennt zugeordnet werden.

Jedes Sicherheitskonzept hat sich an dem konkreten Aufgabenspektrum der „DBAs“ zu orientieren. Dieses weist zwar typischen Merkmale auf – Installation und „Patchen“ der Software, Parametrierung und Troubleshooting etc. – kann aber auch Konzern-spezifische Besonderheiten zeigen, beispielsweise bei der Konfiguration der Storage-Schicht, beim Auditing oder dem Deployment von Applikationen. Mit anderen Worten: Sicherheitskonzepte lassen sich nicht „von der Stange“ implementieren, können aber durchaus von dort inspiriert werden. Die Herausforderung besteht dann in der bedarfsgerechten Kombination an sich überschaubarer technischer Komponenten.

Bei der Implementierung der hier diskutieren Konzepte lassen sich drei unterschiedliche Herangehensweisen ausmachen:

- „Permanent-umfassend“ – Die Privilegien werden den Betroffenen dauerhaft und in vollem Umfang gewährt. Das Risiko des Missbrauchs ist dem entsprechend hoch und sollte kontrolliert werden (*auditing*).
- „Permanent-eingeschränkt“ – Privilegien werden eingeschränkt und zugeschnitten auf die jeweiligen Kernaufgaben vergeben. Das Risiko des Missbrauchs reduziert sich. Es kann hierbei jedoch zu gelegentlichen Privilegien-Engpässen kommen, die dann wie folgt umschifft werden:
- „Temporär-umfassend“ – umfassende Rechte werden nur temporär und gezielt vergeben und nach den Aktionen wieder entzogen. Auch hier reduziert sich das Risiko des Missbrauchs. Diese Strategie erfordert jedoch einen erhöhten Administrationsbedarf durch die Verwaltung von Passwörtern und Rechten.

Authentifizierung

Vor dem Hintergrund des DBA-Aufgabenspektrums muss das Anlegen von Benutzern und die Authentifizierung sowohl im Kontext des Betriebssystems als auch in dem der Datenbank geklärt werden. Folgende Benutzertypen sind in der Regel nötig:

- Software-Owner – d.h. Benutzer unter denen die Oracle-Software installiert und „gepatcht“ wird. Standard ist hier häufig `oracle` für das RDBMS und `grid` für Grid Infrastructure. Diese Benutzer können z.B. nur temporär zugänglich gemacht werden.
- Technische- und Batch-User – für Aufgaben, die keine persönliche Verantwortung möglich machen, beispielsweise Batch-Jobs.
- Persönlich Benutzer – für die auf Anforderung durchgeführten administrativen Arbeiten mit persönlicher Verantwortung.

Jeder Typ sollte stets nur in dem für ihn nötigen und geplanten Umfeld genutzt werden.

Die Kontexte „Betriebssystem“ und „Datenbank“ sind als aufeinander aufbauend zu betrachten

- Ein im Betriebssystem authentifizierter Benutzer kann durchaus in der Datenbank hochprivilegiert (SYSDBA) operieren, ohne dass die Nachvollziehbarkeit seiner Aktionen darunter leidet. Audit-Records geben bekanntlich auch den OS-User mit aus.
- Zusätzlich kann der personifizierte OS-User durch einen maßgeschneiderten DB-User in seinen Rechten innerhalb der Datenbank eingeschränkt werden.
- Ein Gruppenbenutzer des Betriebssystems, der noch dazu als SYS in der Datenbank agiert, ist aus Security-Sicht wertlos. Seine Aktionen können nicht eindeutig zugeordnet werden.

Die Methode der Authentifizierung hingegen kann den Erfordernissen und Gegebenheiten vor Ort angepasst werden. Die meisten Betriebssysteme wie auch die Datenbank selbst bieten in dieser Hin-

sicht vielfältige Möglichkeiten: Definieren von Passwort-Regeln, Nutzung von Zertifikaten oder biometrischen Merkmalen (*strong authentication*), Einbinden von Verzeichnissen (LDAP) oder Kerberos-Diensten.

Darüber hinaus lassen sich erlaubte aber hoch privilegierte Aktionen im Unix-Umfeld per **`sudo`** regeln.

Neben der Authentifizierung muss auch der Netzzugang zu den betreffenden Systemen geregelt und kanalisiert werden. In der Praxis haben sich ein oder mehrere „Jumpserver" etabliert, von denen aus per **`ssh`**-Tunnel der Zugriff auf die Zielsysteme und dort auf die betreffenden persönlichen Benutzer erfolgt. Die Benutzer selbst sind für lokale Authentifizierungen gesperrt. „Seitliche" Einstiege auf die Zielserver werden darüber hinaus durch Firewall-Regeln unterbunden.

Für die Authentifizierung innerhalb der Datenbank bieten sich auch die Verfahren der „Enterprise User Security" an, bei denen „Enterprise User" des LDAP-Dienstes auf *shared schemas* der Zieldatenbank „gemappt" werden. Über dieses Verfahren können auch „Enterprise Roles" die Privilegien zuordnen.

Autorisierung

Die Zuweisung von Rechten an die persönlichen Benutzer erfolgt ebenfalls auf den genannten Ebenen des Betriebssystems und der Datenbank. Im Kontext des Betriebssystems sind bekanntlich die Gruppenzugehörigkeiten maßgeblich. Teilweise darauf aufbauend bieten sich in der Datenbank unterschiedliche Verfahren an:

- „Externe" Methoden für administrative Rechte (SYSDBA/SYSOPER/SYSASM etc.)
    - die Zugehörigkeit zu OS-Gruppen (SYSDBA/SYSOPER etc.) für lokale Autorisierung
    - Oracle Passwordfiles zur lokalen und „remoten" Autorisierung
    - LDAP-Verzeichnisse (Oracle Internet Directory) für die Zuweisung von SYSDBA
- „Interne" Methoden nutzen die Rollen, System- und Objektprivilegien der Datenbank und erlauben eine feingliedrige Privilegienvergabe.

Die genannten administrativen Rechte werden über das Linken des Oracle Kernel an bestimmte OS-Gruppen gekoppelt und sind somit für alle Datenbanken relevant, deren Home-Verzeichnis dem entsprechend zugeordnet ist. Die Planung administrativer Benutzer mit ihren administrativen Rechte kann sich demnach kreativ an den folgenden „Dimensionen" orientieren:

- SW-Owner des Home-Verzeichnisses
- Anzahl der Home-Verzeichnisse
- Mapping des betreffenden Oracle-Kernel
- Gruppenzugehörigkeit des Benutzers

In diesem Spielfeld gibt es zahlreiche Kombinationsmöglichkeiten. Auch hier gilt: Sollte die Autorisierung über SYSDBA nicht wünschenswert sein (weil der Betreffende dann in der Datenbank als SYS agiert und damit unbegrenzte Möglichkeiten hat) können Aktionen über SUDO gesteuert werden.

Die neueste Generation der Datenbank bietet darüber hinaus weitere interessante Möglichkeiten.

Nachvollziehbarkeit

Die Kontrolle der Nutzung von Privilegien und der Zugriffe auf Daten ist besonders dort sinnvoll, wo „üppige“ Privilegien vergeben werden müssen und ein nachträgliche Analyse der Operationen geboten scheint. Dieses Thema kann in der Regel nur in Zusammenarbeit mit dem Betriebsrat und vor dem Hintergrund der jeweiligen Konzernvorgaben geplant und durchgesetzt werden. Nachvollziehbarkeit muss jedoch stets einhergehen mit Revisionssicherheit und der daraus resultierenden Gewaltenteilung (*segregation of duties*).

Auch hier gilt es, alle Ebenen der Infrastruktur mit einzubeziehen und für eine geplante Archivierung der zum Teil umfangreichen Datenbestände zu sorgen. Blockierte Filesysteme und/oder Tablespaces müssen vermieden werden.

Sowohl in der Datenbank, als auch auf Ebene des Betriebssystems stehen umfangreiche Möglichkeiten zur Verfügung: Bei Unix-System steht ab der Kernelversion 2.6 der **`auditd`**-Daemon zur Verfügung, um Zugriffsdaten in das SYSLOG zu schreiben. Darüber hinaus gibt es – Kernel-basierte – Keylogger für weitere, detaillierte Protokolle. In der Regel empfiehlt sich auch die Weiterleitung der SYSLOG-Daten auf einen zentralen Log-Server, um sie vor Manipulationen durch die betreffenden *root user* zu schützen. In der Datenbank lassen sich weitere Audit-Optionen auf Privilegien- und Objektebene einstellen. Diese Daten können in das SYSLOG des Betriebssystems geleitet werden (Achtung *Housekeeping*!), in eigenen Verzeichnissen als **`aud`**-Dateien abgelegt oder in die interne **`aud$`** Tabelle geschrieben werden. Zusätzlich können die Daten über Agenten des Oracle Audit Vault in einem zentralen Repository abgelegt werden. Letztes ist für übergreifende Analysen genauso wichtig wie für die Revisionssicherheit.

Unabhängig davon protokolliert die Datenbank grundsätzlich SYSDBA/SYSOPER Verbindungen in Audit-Dateien oder dem SYSLOG. Es lässt sich auch ein komplettes Befehlsprotokoll aktivieren (**`audit_sys_operations`**).

Der Zugriff auf Log- und Trace-Dateien steht neben dem Owner (**`oracle/grid`**) nur noch lesend der Inventory-Gruppe **`oinstall`** oder der ASM-Gruppe **`asmadmin`** offen.

Das Produkt „Database Activity Monitoring“ (DAM) ist nicht von den in der Datenbank eingestellten Audit-Optionen abhängig, sondern beobachtet mit Hilfe eines Regelwerks über Agenten den SGA-Speicher (*polling*) und leitet die *findings* an ein zentrales Repository weiter. Die Regeln lassen auch den Abbruch eines Session bei besonderen Aktionen zu. Die Daten des Repository lassen sich für Analysen und Auswertungen nutzen.

Einen anderen Weg geht „Oracle Database Vault“, das ab 10g Release 2 als kostenpflichtige Option der Enterprise Edition verfügbar ist. Hier werden „Schutzzonen“ (*realms*) in der Datenbank angelegt, die darin befindliche Objekte nur für explizit autorisierte Benutzer (**`grant`**) zugänglich machen und vordefinierte Standardrollen, wie die Rolle DBA, in ihren Privilegien beschneiden.

Weitere Werkzeuge wie „Oracle Privileged Account Manager“ (OPAM), „Oracle Identity Manager“ (OIM) oder „User Management for Databases“ (UM4DB) unterstützten bei der zentralen Verwaltung und Verteilung von Privilegien. Sie können bei der anfangs erwähnten „temporären, umfassenden“ Strategie zum Einsatz kommen.

Zusammenfassung

Es gibt viele technische Bausteine um „personal DBA“ zu implementieren. Die einzelnen Bausteine sind in der Regel überschaubar und unkompliziert zu konfigurieren. Entscheidend für die Sicherheit und Akzeptanz des gesamten Konzeptes ist jedoch, eine den gesamten Anforderungen gerechte Mischung dieser Bausteine zu finden. Dies kann eine genaue Analyse und Planung notwendig machen und ist dann nicht trivial.

Kontaktadresse:

Dr. Günter Unbescheid

Database Consult GmbH

Laich 9 1/9

D-83676 Jachenau

Telefon:	+49 (0)172 853 0790
Fax:	+49 (0) 8043 1011
E-Mail	g.unbescheid@database-consult.de
Internet:	www.database-consult.de