Geht Security in Oracle Database 12c eigentlich anders?

Heinz-Wilhelm Fabry
ORACLE Deutschland B.V. & Co.KG
München

Schlüsselwörter

Auditing, unified auditing, separation of duties, Rollen

Einleitung

Der New Features Guide von Oracle Database 12*c* listet für das Thema Security der Datenbank ohne Berücksichtigung der Security Optionen etwa ein Dutzend Neuerungen und Änderungen auf. Dazu zählen zum Beispiel Hinweise auf Änderungen bei der Rolle RESOURCE und beim Privileg SELECT ANY DICTIONARY sowie auf die jetzt in der Tabelle USER$ gespeicherte Zeit des letzten Logins. Die beiden wesentlichsten Bereiche betreffen allerdings das Auditing sowie das *separation of duties*, das in Form neuer Betriebssystemrollen umgesetzt ist. Beide, das Auditing und die neuen Rollen, sind nicht in erster Linie für den Nerd interessant, sondern für jeden DBA elementar. Kenntnisse darüber lassen ihn / sie ein Upgrade auf Oracle Database 12*c* mit der Gewissheit durchführen, dass auch die Datenbank Security für ein zügiges Upgrade spricht. Der Artikel geht nicht auf Besonderheiten der Container Architektur ein, sondern geht davon aus, dass mit der klassischen Datenbankarchitektur (NonCDB) gearbeitet wird.

1. Das neue Auditing (*unified auditing*)

1.1 Hintergrund

Der Zugriff auf Daten in der Datenbank unterliegt in der Regel gesetzlichen oder betrieblichen Auflagen. Der Nachweis, dass diese Auflagen eingehalten werden, ist nur über ein Auditing in irgendeiner Form möglich. Eine ernstzunehmende Datenbank gänzlich ohne Auditing ist deshalb schlicht nicht vorstellbar.

Das Auditing vor Oracle Database 12c hat sich über Jahrzehnte entwickelt. Dabei wurden immer neue Auditingverfahren, sogenannte *audit trails,* eingeführt. Sie waren gedacht für unterschiedliche Einsatzszenarien - Default, Objekt, Privileg, SYS und Fine Grained Auditing (FGA), verwendeten unterschiedliche Formate - Datenbanktabellen, proprietäre oder XML Dateien und das Syslog des Betriebssystems - und Umfänge - zum Beispiel mit dem oder ohne den vollständigen Text der SQL Statements. Das hat das Auditing nach und nach unübersichtlich gemacht. Unübersichtlichkeit ist aber bei allen Security Themen problematisch, auch beim Auditing. Denn das Auditing zielt nicht in erster Linie auf das Sammeln, sondern auf das Auswerten von Informationen. Oracle Database 12*c* löst das Problem durch das Zusammenführen der diversen *audit trails* zu einem einzigen *audit trail.* Das Ergebnis wird als *unified auditing* bezeichnet. Es steht in allen Editionen der Datenbank zur Verfügung, ist flexibler, performanter und zusätzlich offen für die Nutzung durch weitere Oracle Werkzeuge wie SQL*Loader und RMAN.

1.2 Nach dem Upgrade: Altes oder neues Auditing?

Eine Abfrage auf die View V$OPTION zeigt, ob das *unified auditing* aktiv ist:

```
SELECT value FROM v$option
WHERE  parameter = 'Unified Auditing';
```

Erscheint hier als Rückgabe der Wert FALSE, heisst das, dass das *unified auditing* **nicht als Standardauditingverfahren** eingerichtet ist, sondern dass beide Verfahren, das klassische und auch das neue Auditing, nebeneinander aktiv sind. Dabei werden Audit Daten im klassischen Format und im individuell definierten Umfang sowie zusätzlich in den neuen *unified audit trail* geschrieben. Diesen Zustand bezeichnet man als *mixed mode auditing.* Im Volumen entspricht das standardmässige *unified auditing* dem, was eine Datenbank der Version 11 mit der Standardeinstellung AUDIT_TRAIL=DB sammelt. Das *mixed mode auditing* soll den Wechsel vom alten zum neuen Auditing erleichtern.

Der DBA kann sich dafür entscheiden, das alte Auditing unverändert weiterzunutzen. Soll aber das *unified auditing* als alleiniges Verfahren genutzt werden, ist das nur über ein erneutes Linken der Datenbanksoftware zu erreichen. Dies geschieht unter Linux nach dem Stoppen der Datenbank und des Listeners aus dem Verzeichnis $ORACLE_HOME/rdbms/lib heraus mit der Eingabe

```
make -f ins_rdbms.mk uniaud_on ioracle
```

Nach dem Neustart der Datenbank ist das alte Auditing dann ausgeschaltet; die für das alte Auditing spezifischen Initialisierungsparameter werden nicht mehr berücksichtigt. Die gespeicherten Audit Daten stehen aber nach wie vor zur Verfügung. Sie können bei Bedarf gesichert und danach aus den *audit trails* gelöscht werden. Erwähnt sei noch, dass Befehle des alten Verfahrens ohne Fehlermeldung eingesetzt werden können. Sie werden allerdings nicht ausgeführt.

1.3 Konfigurieren

Für das neue Auditing benötigt man keine Initialisierungsparameter, sondern entweder das Privileg AUDIT SYSTEM oder die Rolle AUDIT_ADMIN. Das gilt auch für die Steuerung des Auditing auf eigene Objekte.

Um das *unified auditing* aktiv zu nutzen, muss eine sogenannte Policy angelegt werden. Ähnlich funktionierte schon das Fine Grained Auditing (FGA - muss nach wie vor verwendet werden, wenn man lediglich Zugriffe auf einzelne Spalten auditieren möchte). Es gibt mitgelieferte Policies, die wichtigste heisst ORA_SECURECONFIG. Sie ist grundsätzlich aktiviert - und sollte natürlich, wenn das alte Auditing weiterhin genutzt werden soll (siehe oben), deaktiviert werden. Sinnvollerweise sollte man sich eigene Policies anlegen, denn ORA_SECURECONFIG enthält zum Beispiel keinerlei Regeln für das Auditieren von Benutzerobjekten.

Die gesamte Verwaltung des Auditing kann über den Enterprise Manager Cloud Control oder über die Kommandozeile erfolgen. Auf der Kommandozeile legt man eine Policy mit dem Befehl CREATE AUDIT POLICY an.

```
CREATE AUDIT POLICY doag2015
PRIVILEGES     SELECT ANY TABLE
ACTIONS        CREATE USER, ALTER USER, SELECT ON SCOTT.EMP
ROLES          RESOURCE
WHEN           'SYS_CONTEXT(''USERENV'', ''MODULE'') <> (''HR'')'
EVALUATE       PER STATEMENT
```

Die Policy wird als Datenbankobjekt mit dem Namen DOAG2015 angelegt. Es folgt eine Auflistung der Systemprivilegien und Aktionen, die auditiert werden. Weiterhin wird die Nutzung aller Privilegien auditiert, die auf Berechtigungen zurückgehen, die über die Rolle RESOURCE erworben

wurden. Der Parameter WHEN bietet die ausgesprochen hilfreiche Möglichkeit festzulegen, unter welchen Bedingungen ein Audit Eintrag geschrieben wird. Im Beispiel wird davon ausgegangen, dass die Anwendung HR einen eigenen *audit trail* schreibt. Deshalb kann für diese Anwendung das Auditing ausgeschlossen werden. Das führt dazu, dass nur Zugriffe erfasst werden, die die Anwendung HR umgehen. Die Klausel EVALUATE legt fest, dass für jedes Statement ein Audit Eintrag geschrieben wird.

Mit dem Befehl CREATE AUDIT POLICY ist auch das Auditing Verhalten der Komponenten Database Vault (DV), Data Mining, Label Security (OLS), Real Application Security (RAS), SQL Loader (direct loads) und Data Pump zu steuern. Der Recovery Manager (RMAN) steuert sein Auditing Verhalten über seinen Aufruf selbst. Aber auch die Audit Daten des RMAN sind über den *unified audit trail* auszuwerten.

Nachdem die Policy angelegt ist, steht fest was und unter welchen Bedingungen auditiert wird. Das Auditing muss dann noch mit dem Befehl AUDIT gestartet werden.

```
AUDIT POLICY doag2015 EXCEPT SYS
```

Da das Auditieren hier nicht mit der Klausel BY auf bestimmte Benutzer eingeschränkt wird, gilt die Policy für alle Benutzer - auch für den Benutzer SYS. Mit der Klausel EXECPT - die es im alten Auditing nicht gibt - wird SYS vom Auditing durch die Policy DOAG2015 ausgenommen. Auf die Verwendung der Klausel WHENEVER (NOT) SUCCESSFUL wird hier verzichtet. Es wird also in beiden Fällen ein Satz mit Audit Daten geschrieben.

1.4 Audit Informationen auswerten und Auditing administrieren

Anwender, die die Daten des *unified auditing* auswerten möchten, benötigen entweder die Rolle AUDIT_ADMIN oder die Rolle AUDIT_VIEWER. Zur Auswertung stehen eine Reihe von Views zur Verfügung zum Beispiel die View UNIFIED_AUDIT_TRAIL.

Der *unified audit trail* wird ausschliesslich über das Package DBMS_AUDIT_MGMT administriert. Mit dem Package können zum Beispiel die Objekte zur Speicherung der Audit Daten aus dem Standardtablespace SYSAUX in ein anderes Tablespace verschoben werden oder auch Audit Daten nach ihrer Sicherung oder Aufbewahrungsfrist gelöscht werden. Das Löschen kann über spezielle Jobs zeitgesteuert ablaufen. Die Verwendung von DBMS_AUDIT_MGMT setzt entweder die Rolle SYSDBA oder die Rolle AUDIT_ADMIN voraus. Ausserdem werden jede Verwendung des Package, alle Befehle, die das Auditing in irgendeiner Form manipulieren, sowie alle Befehle, die die Konfiguration von Database Vault betreffen, auditiert.

Um die Performance des Auditing zu verbessern, werden die Daten des Auditing nicht mehr synchron in die Datenbank geschrieben, sondern asynchron über eine Queue in der SGA. Die Größe der Queue ist auf 1 MB eingestellt, kann aber über den statischen Initialisierungsparameter UNIFIED_AUDIT_SGA_QUEUE_SIZE auf bis zu 30 MB erhöht werden. Zusätzlich gibt es die Möglichkeit, das asynchrone Schreiben durch ein synchrones Schreiben der Audit Informationen zu ersetzen.

Wenn die Datenbank nicht verfügbar ist, müssen mitunter trotzdem Audit Informationen gesammelt werden können. Dazu wird in Dateien des Betriebssystems geschrieben. Die Informationen aus diesen Dateien können, wenn die Datenbank wieder zum Schreiben zur Verfügung steht, in den *unified audit trail* geladen werden - natürlich wieder mit dem Package DBMS_AUDIT_MGMT.

2. Privilegierte Benutzer kontrollieren

Schon lange wurde bemängelt, dass zu viele DBA Aufgaben die Privilegien des Benutzers SYS erforderten. Oracle Database 12*c* läßt nun über das Anlegen eigener Gruppen für die Bereiche Backup, Keystore / Wallet und Standby Management eine Differenzierung zu.

Das funktioniert unter UNIX / Linux auf die gleiche Weise, in der schon bisher die DBA Gruppe festgelegt wird: Es werden vor der Software Installation im Betriebssystem Gruppen für die einzelnen Bereiche eingerichtet. Diesen Gruppen können dann Benutzer zugewiesen werden, die ihre speziellen Funktionen ohne das Privileg SYSDBA ausführen können.

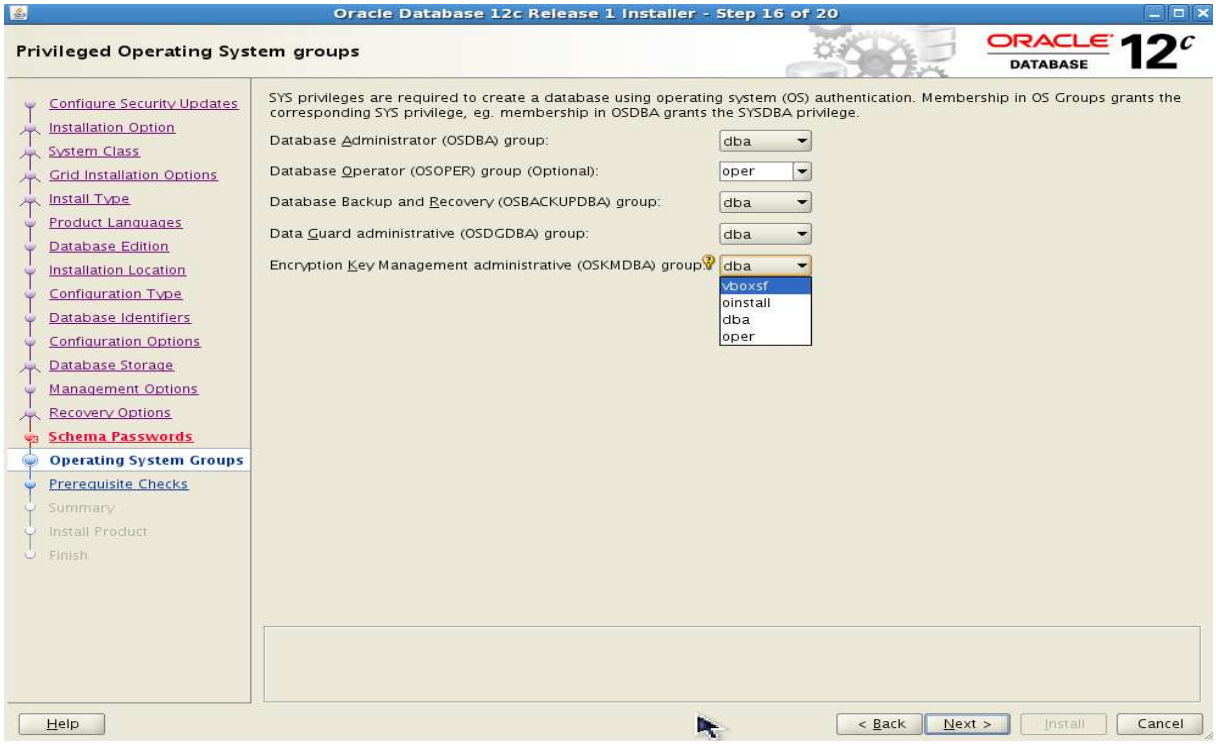

Abb. 1: Bezugnahme auf eingerichtete Gruppen bei der Software Installation

Die Person, die zum Beispiel für das Öffnen eines Keystore / Wallet für Advanced Security Option Transparent Data Encryption (TDE) verantwortlich ist, könnte sich durch die Zugehörigkeit zur Gruppe VBOXSF aus Abbildung 1 bei der Datenbank anmelden mit

```
orcl /home/oracle> whoami
oracle
orcl /home/oracle> sqlplus / as syskm

SQL*Plus: Release 12.1.0.2.0 Production on Thu Aug 20 12:06:25 2015 ...

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit ...

SQL orcl> show user
USER is "SYSKM"
SQL orcl> exit

Disconnected from ...
```

```
orcl /home/oracle> sqlplus scott as syskm

SQL*Plus: Release 12.1.0.2.0 Production on Thu Aug 20 12:15:24 2015 ...

Enter password:

Connected to:

Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit ...

SQL orcl> show user
USER is "SYSKM"
SQL orcl>
```

Die Person hat damit alle Berechtigungen zum Arbeiten mit den Keystores / Wallets, aber zum Beispiel keinerlei Berechtigungen zum Anlegen oder Ändern von Benutzern.

Neben diesen Rollen, die auf der Betriebssystemebene verankert sind, gibt es - analog zum SYSDBA Privileg - identische Privilegien namens SYSDG, SYSBACKUP und SYSKM. Diese Privilegien wirken allerdings nur bei geöffneter Datenbank. Auch das ist vom Privileg SYSDBA bekannt, das allein ja zum Beispiel ein Starten der Datenbank nicht ermöglicht. Dazu ist die Zugehörigkeit zur DBA oder OPER Gruppe erforderlich.

Abschliessend sei auch noch darauf hingewiesen, dass auch das Einschränken des Lesens der Audit Informationen auf Benutzer mit dem Privileg AUDIT_VIEWER, auf das oben ja schon verwiesen wurde, als weiterer Schritt in die zunehmende Kontrolle privilegierter Benutzer zu sehen ist.

3. Fazit

Beide Themen - schnelleres, flexibleres und übersichtlicheres Auditing sowie deutlich verbesserte Kontrolle privilegierter Benutzer - sollten gerade sicherheitsbewuste Anwender motivieren, das Upgrade auf Oracle Database 12*c* möglichst zügig durchzuführen.

Kontaktadresse:
Heinz-Wilhelm Fabry
ORACLE Deutschland B.V. & Co.KG
Riesstr. 25
D-80992 München

Telefon: **+49 (089) 1430 1534**
E-Mail **heinz-wilhelm.fabry@oracle.com**
Internet: **www.oracle.com**