Mandatory Access Control mit Oracle Label Security

Heinz-Wilhelm Fabry, Oracle Deutschland B.V. & Co. KG

Oracle Label Security (OLS) ist eine Option der Enterprise Edition der Datenbank. Es gibt sie seit der Datenbankversion 8.1.7. OLS ist im Grunde nichts anderes als eine sehr komfortable und leistungsstarke Virtual Private Database (VPD) out of the box, deren Einsatz - auch nachträglich - keinerlei Änderungen an Anwendungen erfordert. Aber obwohl es sich bei OLS um ein ausgereiftes Produkt handelt und es sogar mindestens einmal in jedem major release der Datenbank nach EAL4+ oder vergleichbaren Standards evaluiert wurde, ist es vielen Kunden unbekannt. Der hier vorliegende Beitrag soll Datenbank Security Experten in die Lage versetzen, das Produkt für ihre Sicherheitsstrategie einordnen und bewerten zu können.

Um den Zugriff auf Tabellen oder Views zu kontrollieren, bietet SQL als sogenanntes *discretionary access control* die GRANTs. Diese GRANTs zielen aber nicht auf die Zeilen einer Tabelle, sondern immer auf die gesamte Tabelle. Zum Beispiel bietet das Objekt Privileg SELECT immer die Möglichkeit, auf alle Zeilen einer Tabelle zuzugreifen. Will man nun den Zugriff nur auf bestimmte Zeilen zulassen, was mitunter auch als *mandatory access control* bezeichnet wird, stehen drei Lösungsansätze zur Verfügung:

1. Definition von Views und Steuerung des Zugriffs auf die Daten über GRANTS auf diese Views. Sind die Parameter, die den Zugriff steuern, komplex, stößt man bei dieser Lösungsvariante irgendwann an Grenzen.
2. Programmieren einer eigenen Zugriffssteuerung mit VPD. Das benötigt Expertise und Zeit.
3. Oracle Label Security.

Installation

OLS wird bei einer Installation der Datenbanksoftware bis zur Version 11.1 nicht automatisch installiert, sondern muss im Rahmen der Advanced Installation zur Installation ausgewählt werden. Eine nachträgliche Installation ist nur über den Oracle Universal Installer (OUI) möglich.

Ab der Version 11.2 der Datenbank werden grundsätzlich alle Softwarekomponenten der Datenbank installiert, also auch OLS. Die Auswahl unter Advanced Installation

führt dann dazu, dass OLS nicht nur installiert, sondern dass, sofern bei der Installation auch eine Datenbank erstellt wird, OLS in dieser Datenbank auch konfiguriert wird. Hat man das bei der Installation nicht veranlasst, kann OLS vom Eigentümer der Oracle Software nach dem Herunterfahren der Datenbank und des Listeners über folgende Eingabe auf der Kommandozeile nachträglich installiert werden:

```
chopt enable lbac
```

Zusätzlich müssen dann noch mit dem Database Configuration Assistant (dbca) die von OLS benötigten Datenbankobjekte erstellt werden. Zu diesen Objekten gehören Tabellen, Prozeduren und Funktionen, aber auch der Eigentümer aller OLS Objekte, der Benutzer LBACSYS (**L**abel **B**ased **A**ccess **C**ontrol **SYS**). Der Account ist zwar gesperrt, aber das OLS Handbuch empfiehlt, ihn für Notfälle zu entsperren. Zum täglichen Arbeiten mit OLS sollte der Account hingegen nicht genutzt werden. Stattdessen sollte dafür ein spezieller Benutzer angelegt oder ausgewählt werden, dem die Rolle LBAC_DBA zugewiesen wird. Wenn dieser Benutzer weitere Privilegien benötigt - zum Beispiel um die graphische Schnittstelle zu OLS im Oracle Enterprise Manager Database Control (OEM) zu nutzen - müssen ihm diese Privilegien zusätzlich zugewiesen werden, für das OEM Beispiel also das Privileg SELECT ANY DICTIONARY.
Nach der erfolgreichen Konfiguration von OLS liegt übrigens die Tabelle AUD$ nicht mehr im Schema SYS, sondern im Schema SYSTEM. Diese Verlagerung geschieht, weil AUD$ um einige OLS spezifische Spalten ergänzt wird.
Wie bei der Version 11.2 wird OLS bei der Installation der Version 12 der Datenbanksoftware ebenfalls immer installiert. Das Aktivieren von OLS erfolgt aber grundsätzlich nicht automatisch, sondern immer durch den Benutzer SYS in 3 Schritten:

```
-- 1. Konfiguration
EXEC LBACSYS.CONFIGURE_OLS;
-- 2. Einschalten
EXEC LBACSYS.OLS_ENFORCEMENT.ENABLE_OLS;
-- 3. Aktivieren
```

Shutdown und Neustart der Datenbank

Diese Vorgehensweise erleichtert auch die nachträgliche Installation von OLS in einer existierenden Datenbank.

Überblick

Das Vorgehen zum Einsatz von OLS folgt stets dem gleichen Muster:
Zuerst wird eine sogenannte Policy (Regel) angelegt, eine Art Container für die in einem konkreten Fall benötigten OLS Objekte. Diese Policy kann einer Tabelle oder auch mehreren Tabellen zugewiesen werden.
Im zweiten Schritt werden die Labels definiert, die man verwenden möchte. Offizielle Labels in der Bundesrepublik sind zum Beispiel "streng geheim", "geheim", "Verschlusssache - vertraulich" und "Verschlusssache - nur für den Dienstgebrauch". Diese Labels und ihre Anzahl sind jedoch nicht zwingend, man kann auch beliebige eigene Labels definieren.
Im dritten Schritt werden die Tabellen identifiziert, die mit OLS zusätzlich geschützt werden sollen. Ihnen wird automatisch eine neue, in der Regel Anwendern und Anwendungen verborgene Spalte hinzugefügt, in der dann später die Labels gespeichert werden.
Im abschliessenden Schritt weist man den Benutzern Labels zu. Beim Einloggen werden die Labels Teil des jeweiligen *session context*.
Da GRANTs nach wie vor die Voraussetzung für jeden Zugriff auf eine Tabelle oder View sind, passiert beim Zugriff auf eine mit OLS geschützte Tabelle dann Folgendes: Es wird zunächst geprüft, ob der Benutzer über die Objekt- oder Systempriviliegien verfügt, um überhaupt auf die Tabelle zugreifen zu dürfen. Erst wenn das der Fall ist, werden Benutzerlabel und Zeilenlabel abgeglichen, um festzustellen, auf welche Sätze der Benutzer tatsächlich zugreifen darf.

Policy anlegen und mit einer Tabelle verbinden

Im Folgenden werden die vier Schritte zum Einsatz von OLS näher betrachtet.
Eine Policy kann als Benutzer mit der Rolle LBAC_DBA entweder im OEM unter dem Reiter SERVER im Bereich SECURITY oder zum Beispiel in SQL*PLUS mit der Prozedur SA_SYSDBA.CREATE_POLICY angelegt und verwaltet werden.

```
SA_SYSDBA.CREATE_POLICY(
   POLICY_NAME     => 'doag2013',
   COLUMN_NAME     => 'spalte2013',
   DEFAULT_OPTIONS => 'read_control, write_control');
```

Zur Erläuterung: Da die Policy ein Datenbankobjekt ist, benötigt sie einen Namen, hier DOAG2013. Oben wurde bereits darauf hingewiesen, dass ein Zeilenlabel in einer speziell dafür angelegten Spalte gespeichert wird. Der Name dieser Spalte ist frei zu vergeben und wird hier mit SPALTE2013 angegeben. Die Policy soll sowohl beim Lesen der Daten als auch beim Schreiben angewendet werden, also für die Aktionen SELECT, INSERT, UPDATE und DELETE.
Mit folgenden Prozeduraufruf wird die Policy nun einer Tabelle zugewiesen. Zugleich wird diese Tabelle um die Spalte SPALTE2013 erweitert.

```
SA_POLICY_ADMIN.APPLY_TABLE_POLICY(
   POLICY_NAME => 'doag2013',
   SCHEMA_NAME => 'einschemaname',
   TABLE_NAME  => 'eintabellenname');
```

Labels

Ein Label in OLS kann aus bis zu drei Komponenten bestehen. Jedes Label MUSS ein LEVEL (Ebene) haben. Das Level hat jeweils einen Lang- und einen Kurznamen sowie einen numerischen Wert. Da Levels hierarchisch sind, ist der Wert um so höher, je höher das Level ist. Die Datenbank arbeitet nur mit diesen Werten, während die Namen dem DBA beziehungsweise Anwendungsentwickler das Arbeiten erleichtern. Benutzer haben Zugriff auf alle Sätze, deren Labelwerte kleiner oder gleich dem Wert sind, für den diese Benutzer zugriffsberechtigt sind.
Labels KÖNNEN zusätzlich um sogenannten COMPARTMENTs (Abteilungen) erweitert werden. Compartments sind nicht hierarchisch, sondern dienen lediglich der Differenzierung. Gibt es in einer Tabelle Datensätze, die vom Level her alle GEHEIM sind, die aber unterschiedlichen Compartments zugeordnet sind, muss ein Benutzer nicht nur das Level GEHEIM in seinem eigenen Laben haben, sondern auch noch das richtige Compartment, um auf die entsprechenden Datensätze zugreifen zu können.

Schliesslich KÖNNEN Labels als dritte Komponente auch noch sogenannte GROUPs (Gruppen) enthalten. Gruppen sind wie Leveles hierarchisch und fungieren als weiteres Differenzierungsmerkmal für den Zugriff.
Die drei Komponenten (level, compartment, group) könnten also zum Beispiel zu folgenden Labels zusammengestellt werden: geheim.verteidigung.heer, geheim.verteidigung.marine oder - im nicht-militärischen Bereich - vertraulich.personal.bayern, vertraulich.personal.hessen.

Anlegen von Labels

Das Anlegen eines Labels erfolgt in zwei Schritten: Zunächst werden die Komponenten angelegt, aus denen ein Label bestehen kann, also die Levels, Compartments und Groups. Aus den angelegten Komponenten werden dann die Labels selbst definiert. Es müssen nicht alle denkbaren Kombinationen angelegt werden, sondern nur die wirklich benötigten. Natürlich erfordert dieser Schritt eine gewisse Sorgfalt, damit nicht Kombinationen entstehen, die das Bearbeiten der entsprechenden Datensätze durch die dafür eigentlich freigegebenen Benutzer verhindert.
Für diesen Beitrag werden nur Labels aus den beiden Komponenten Levels und Compartments angelegt. Auf Groups wird verzichtet. Die Ebenen sollen FINANZEN, FORSCHUNG und VERKEHR heissen, die Compartments BUND, BAYERN und BERLIN. Hier als Beispiel das Anlegen des Levels FINANZEN

```
LBACSYS.SA_COMPONENTS.CREATE_LEVEL(
   POLICY_NAME => 'DOAG2013',
   LEVEL_NUM   => 1000,                 -- zwischen 0 und 9999
   SHORT_NAME  => 'FIN',              -- maximal 30 Zeichen
   LONG_NAME   => 'FINANZEN'          -- maximal 80 Zeichen);
```

Für die Definition des Levels ist die Angabe des Wertes LEVEL_NUM wichtig: Je höher die Geheimhaltungsstufe oder Vertraulichkeit, die der Level bezeichnen soll, desto höher wird dieser Wert sein. Der SHORT_NAME wird für den Aufruf weiterer Prozeduren verwendet, der LONG_NAME hat nur erläuternden Charakter.
Nach gleichem Muster werden nun die Compartments angelegt. Beispielhaft das Compartment BUND.

```
LBACSYS.SA_COMPONENTS.CREATE_COMPARTMENT(
   POLICY_NAME => 'DOAG2013',
   COMP_NUM    => 100,
   SHORT_NAME  => 'BRD',
   LONG_NAME  => 'Bund');
```

Aus den beiden angelegten Komponenten kann nun mit einem weiteren Prozeduraufruf ein Label erzeugt werden.

```
SA_LABEL_ADMIN.CREATE_LABEL(
   POLICY_NAME => 'DOAG2013',
   LABEL_TAG   => 1100,            -- zwischen 0 und 99999999
   LABEL_VALUE => 'FIN:BRD');
```

Der Wert für das LABEL_TAG bestimmt den Sortierwert des Labels, der LABEL_VALUE zeigt, welcher Level und welche Compartments dem LABEL_TAG zugeordnet sind.

Benutzern und Zeilen Labels zuweisen

Das Zuweisen der Benutzerlabels und der Zeilenlabels erfolgt ebenfalls über Prozduraufrufe. Dabei werden die Benutzerlabelkomponenten separat zugewiesen.

```
SA_USER_ADMIN.SET_LEVELS(
   POLICY_NAME => 'DOAG2013',
   USER_NAME   => 'einbenutzer',
   MAX_LEVEL   => 'FIN');
...
SA_USER_ADMIN.SET_COMPARTMENTS(
   POLICY_NAME => 'DOAG2013',
   USER_NAME   => 'einbenutzer',
   READ_COMPS  => 'BRD,BY,B',
   WRITE_COMPS => 'BY');
```

Die Zuweisung von Labels an Tabellenzeilen erfolgt über ein UPDATE der Labelspalte unter Verwendung der Funktion CHAR_TO_LABEL durch einen entsprechend privilegierten Benutzer, also zum Beispiel

```
UPDATE einschemaname.eintabellenname
SET spalte2013 = CHAR_TO_LABEL('DOAG2013', 'FIN:BRD')
WHERE einespalte = 'einwert'
```

Sofern sinnvoll könnten auch die Werkzeuge Data Pump Import oder SQL*Loader eingesetzt werden, um Daten einschliesslich der Labels zu laden.

Fazit

Damit sind die Grundlagen für den Einsatz von OLS abgeschlossen. Es fehlt bewusst die Darstellung weitergehender Möglichkeiten, die das Arbeiten mit OLS komplettieren. Dazu gehört zum Beispiel wie man ein Default Level für das Einloggen festlegt, welches Label beim Einfügen neuer Sätze verwendet wird, wie man ein vorhandenes Label ändert und wie OLS und das Oracle Internet Directory zusammenarbeiten. Die kurzen Demos zum mündlichen Vortrag bei der Jahreskonferenz werden zum Teil darauf eingehen.

Am Ende soll noch der Hinweis darauf stehen, dass sich OLS selbstverständlich auch mit sehr komplexen Anwendungen verträgt. Das wird deutlich, wenn man sich klar macht, dass zum Beispiel OLS zum Einsatz mit der Oracle EBusinessSuite durch den Oracle Support unterstützt wird.

Kontakt:

Heinz-Wilhelm Fabry

heinz-wilhelm.fabry@oracle.com