Tipps & Tricks: Unified Auditing

Bereich:	DBA	Erstellung:	07/2013 MP
Versionsinfo:	12.1	Letzte Überarbeitung:	07/2013 MP

Unified Auditing

Der folgende Beitrag beschäftigt sich mit dem Thema: Überwachen der Datenbank.

Oracle hatte schon seit Version 7.0 ein umfangreiches Auditing, das aber bis zur Version 11.1 nicht standardmäßig aktiviert war.

Oracle 12c hat diese Schnittstelle nun nochmal extrem erweitert und die Änderungen unter dem Namen **Unified Auditing** herausgebracht. Vereinfacht erklärt werden hier Snippets zur Überwachung von mehreren Objekten, Rechten, Tools (wie RMAN oder DataPump) oder Rollen zur Verfügung gestellt, die auch an Bedingungen geknüpft werden können.

Das Feature UNIFIED AUDITING fasst sämtliche Audit-Informationen an einer einzigen Stelle zusammen. Dadurch ergeben sich einige Vorteile:

Keine Parametereinstellung mehr notwendig
Bessere Übersicht und einfachere Sichtung der Audit-Informationen
Bessere Geschwindigkeit, was die Überwachung betrifft

Als Default verwendet eine Oracle 12.1.0.x DB den Mixed Audit Mode. Hier werden sowohl die alte als auch die neue Audit Syntax parallel verwendet. Sie können prüfen, ob die Unified Audit Option alleinig eingeschaltet ist (TRUE oder FALSE):

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

Bei FALSE wird die alte Audit Form nicht mehr unterstützt und es kann nur noch Unified Auditing verwendet werden. Wenn Sie das möchten, ist es notwendig die Instanz und Listener heruntzuerfahren.
Unter Unix Wechsel in `$ORACLE_HOME/rdbms/lib,` dann:

```
make -f ins_rdbms.mk uniaud_on ioracle ORACLE_HOME=$ORACLE_HOME
```

Listener und Instanz wieder starten.

Wenn Sie Unified Auditing wieder ausschalten möchten:

```
make -f ins_rdbms.mk uniaud_off ioracle ORACLE_HOME=$ORACLE_HOME
```

Der Initialisierungsparameter `UNIFIED_AUDIT_SGA_QUEUE_SIZE` regelt die Speichermenge, die der Hauptspeicher für die Audit Einträge zur Verfügung stellt: Speichergröße 1-30MB.
Benutzer, die Unified Auditing administrieren wollen, müssen über das Recht AUDIT SYSTEM verfügen. Die Auditdaten werden zuerst gecached und periodisch in Tabellen gespeichert. Bei einem SHUTDOWN ABORT

könnten jedoch Einträge verloren gehen, deswegen unterstützt Oracle zwei Modi:

- `Immediate` (sofort schreiben)
- `Queued Write` (verzögert schreiben)

Wenn Sie die Audit Daten sofort auf Platte schreiben möchten, verwendet man: `EXEC DBMS_AUDIT_MGMT.FLUSH_UNIFIED_AUDIT_TRAIL;`

Dann fangen wir mal mit dem Überwachen an (Natürlich muss man das immer schön mit dem Betriebsrat abstimmen, aber was die NSA darf ...).

Syntax des Unified Auditing:

```
CREATE AUDIT POLICY policy
  [ privilege_audit_clause ]
  [ standard_or_component_clause ]
  [ role_audit_clause ]
  [ WHEN 'audit_condition'
  EVALUATE PER { STATEMENT | SESSION | INSTANCE } ]
  [ CONTAINER = { ALL | CURRENT } ] ;
  privilege_audit_clause:
  privilege_audit_clause := PRIVILEGES privilege1 [, privilege2]
  role_audit_clause:
  role_audit_clause := ROLES role1 [, role2]
  standard_or_component_clause:= {standard_actions | component_actions}
  [, component_actions ]
```

Standard Actions:

```
ACTIONS { { object_action | ALL }
  ON { DIRECTORY directory_name |
  MINING MODEL [ schema. ] object_name | [ schema. ] object_name } |
  { system_action | ALL } }
  [ { object_action | ALL }
  ON { DIRECTORY directory_name |
  MINING MODEL [ schema. ] object_name | [ schema. ] object_name } |
  { system_action | ALL } ]...

Component Actions:
  ACTIONS COMPONENT = { DATAPUMP | DIRECT_LOAD | OLS | XS }
  component_action [, component_action ]... | DV component_action ON
  object_name
  [, component_action ON object_name ]...
```

`STATEMENT`: Für jedes ausgeführte Statement wird ein Auditeintrag erstellt
`SESSION`: Innerhalb einer Session wird nur einmal für ein ausgeführtes Statement ein Auditeintrag erzeugt
`INSTANCE`: Solange die Instanz läuft, wird ein Vorgehen nur einmal aufgezeichnet
`CONTAINER = ALL` (Alle Container einer Plugable Database)
`CURRENT` (nur der aktuelle Container)

Einfacher wird es, wenn wir uns ein paar Beispiele ansehen:

Beispiel 1: SELECT ANY TABLE und CREATE VIEW für zwei Benutzer überwachen

```
CREATE AUDIT POLICY osusers_tab_view_pol
  PRIVILEGES SELECT ANY TABLE, CREATE VIEW
  WHEN q'!SYS_CONTEXT ('USERENV', 'OS_USER') IN ('MARCO', 'HANS')!'
  EVALUATE PER SESSION;
```

Danach muss das Audit noch aktiviert werden:

```
AUDIT POLICY osusers_tab_view_pol;
```

Beispiel 2: Alle Anmeldungen durch SQL*Plus überwachen

```
CREATE AUDIT POLICY sqlplus_logon_pol
  ACTIONS LOGON
  WHEN q'!INSTR(UPPER(SYS_CONTEXT('USERENV','CLIENT_PROGRAM_NAME')),
    'SQLPLUS') > 0!' EVALUATE PER SESSION;
```

Audit wieder scharf schalten durch:

```
AUDIT POLICY sqlplus_logon_pol;
```

Beispiel 3: INS/UPD/DEL auf Emp Tabelle und alle Aktivitäten (auch SELECT) auf DEPT Tabelle überwachen:

```
CREATE AUDIT POLICY dml_poli ACTIONS
  DELETE on scott.emp,
  INSERT on scott.emp,
  UPDATE on scott.emp,
  ALL    on scott.dept;

AUDIT POLICY dml_poli BY scott;
```

Beispiel 4: Zwei DDL Statements überwachen:

```
CREATE AUDIT POLICY table_poli PRIVILEGES
  CREATE ANY TABLE,
  DROP   ANY TABLE;

AUDIT POLICY table_poli BY scott;
```

Beispiel 5: Überwachen, ob SYS Rechte an `UTL_FILE/UTL_TCP/UTL_SMTP` vergibt:

```
CREATE AUDIT POLICY dbms_utl_grants
  ACTIONS
  GRANT ON UTL_FILE,
  GRANT ON UTL_TCP
  GRANT ON UTL_SMTP;
```

```
AUDIT POLICY dbms_utl_grants BY SYS;
```

Beispiel 6: Datapump Export überwachen:

```
CREATE AUDIT POLICY audit_expdp_pol
  ACTIONS
  COMPONENT=DATAPUMP EXPORT;

AUDIT POLICY audit_expdp_pol;
```

Beispiel 7: Benutzung von Packages überwachen:

```
CREATE AUDIT POLICY muso_pack_pol
  ACTIONS
  EXECUTE ON sys.utl_file,
  EXECUTE ON sys.utl_smtp,
  EXECUTE ON sys.utl_http,
  EXECUTE ON sys.utl_tcp,
  EXECUTE ON sys.utl_inaddr,
  EXECUTE ON sys.dbms_fga,
  EXECUTE ON sys.dbms_rls,
  EXECUTE ON sys.dbms_crypto,
  EXECUTE ON sys.dbms_job,
  EXECUTE ON sys.dbms_scheduler;

AUDIT POLICY muso_pack_pol;
```

Nun möchten wir die Überwachung des DataPump auswerten:

```
SELECT dp_text_parameters1, dp_boolean_parameters1
  FROM unified_audit_trail
 WHERE audit_type = 'DATAPUMP';
```

Oder allgemein auch für die anderen Beispiele:

```
SELECT TO_CHAR(event_timestamp,'DD.MM.YY HH24:MI:SS') EVENT_TIMESTAMP,
     os_username, dbusername, substr(client_program_name,1,30)
     client_prg, action_name, object_name, sql_text, system_privilege_used
  FROM unified_audit_trail
 ORDER BY 1 desc;
```

Syntax zum Audit aktivieren:

```
AUDIT { POLICY policy [ { BY user [, user]... } |
  { EXCEPT user [, user]... } ]
  [ WHENEVER [ NOT ] SUCCESSFUL ] } |
  { CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]...
  [, CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]
```

```
  ... ]... [ BY user [, user]... ] } ;
```

Beispiele:

```
AUDIT POLICY table_poli;
AUDIT POLICY dml_poli BY scott,marco;
AUDIT POLICY dml_poli EXCEPT hans,uli;
```

Syntax zum Audit deaktivieren:

```
NOAUDIT { POLICY policy |
  CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]... [,
  CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]... ]... }
  [ BY user [, user]... ] ;
```

Beispiel:

```
NOAUDIT POLICY table_poli;
```

Sie können mit einem DROP die AUDIT Policy löschen:

```
DROP AUDIT POLICY policy
```

Wie man die AUDIT Daten löscht, werden wir in einem weiteren Tipp behandeln.

Das ist nur ein kleiner Teil aus unserem Security Kapitel des Oracle 12c Neuheiten Kurses. Schauen Sie doch mal rein!