ORACLE | CERTIFIED SOLUTION PARTNER

All About Oracle Auditing – Everything You Need to Know

Mike Dean

Database Specialists, Inc.

www.dbspecialists.com

RMOUG

February 12, 2013

DatabaseSpecialists

Who Am I?

- Oracle 11g Certified Professional DBA
- More than 15 years working with Oracle
- 9 years working on classified Dept of Defense databases
- Currently with Database Specialists, Inc
- Email: mdean@dbspecialists.com
- On the Web: www.dbspecialists.com

Agenda

- Mandatory, Standard and Fine-Grained Auditing
- What should you audit?
- What to look for in the Audit Trail?
- Changes in Oracle 11
- Examples of Audit Trail use for the DBA
- Managing the Audit Trail
- Performance impact of Auditing

What is Auditing? (and why do I need it?)

"Auditing is the monitoring and recording, either in the database or in OS files, of selected database actions, from both database users and non-database users"

- Regulatory Compliance (Sarbanes-Oxley, HIPAA, DOD, GSA etc)
- Accountability
- Deter bad behavior
- Investigate security incidents
- Monitor database activities
- Makes life easier for the DBA (???)

Mandatory Auditing

Happens automatically and cannot be turned off. Records startup/shutdown and SYSDBA/SYSOPER logins. Writes to `AUDIT_FILE_DEST` on Unix or the Event Viewer on Windows.If Oracle cannot write to this directory, you won't be able to start the database or connect as SYSDBA.

```
SQL> alter system set audit_file_dest='/bogus'
SQL> startup
ORA-09925: Unable to create audit trail file
Linux-x86_64 Error: 2: No such file or directory
Additional information: 9925
```

Auditing Actions By SYS

By default, SYS does not generate any audit records other than startup, shutdown and connect. You can enable SYS auditing by setting AUDIT_SYS_OPERATIONS=true. This will audit ALL statements by SYS and send to the OS audit trail, or if specified by AUDIT_SYSLOG_LEVEL, the unix syslog.

AUDIT_SYSLOG_LEVEL

Allows you to integrate the database audit trail with the Unix syslog and even send it to a centralized syslog server.

```
AUDIT_SYSLOG_LEVEL =
  'facility_clause.priority_clause‘

AUDIT_SYSLOG_LEVEL = 'LOCAL1.WARNING';
```

This will cause all SYS audit records to go to the Syslog and also standard audit records but only if audit_trail=OS

```
/etc/syslog.conf:
LOCAL1.WARNING /var/log/dbaudit.log
```

AUDIT_SYSLOG_LEVEL

- Jan 8 09:43:49 mdlinux Oracle Audit[26936]: LENGTH: "266" SESSIONID:[6] "691499" ENTRYID:[1] "1" STATEMENT:[2] "10" USERID:[4] "MIKE" USERHOST:[7] "mdlinux" TERMINAL:[5] "pts/1" ACTION:[1] "1" RETURNCODE:[3] "955" OBJ$CREATOR:[4] "MIKE" OBJ$NAME:[5] "TEST1" OS$USERID:[6] "oracle" **DBID:[10] "4057968456**" PRIV$USED:[2] "40"

- Jan 8 09:42:02 mdlinux Oracle Audit[22100]: LENGTH: "266" SESSIONID:[6] "417957" ENTRYID:[1] "6" STATEMENT:[2] "73" USERID:[4] "MIKE" USERHOST:[7] "mdlinux" TERMINAL:[5] "pts/2" ACTION:[1] "1" RETURNCODE:[3] "955" OBJ$CREATOR:[4] "MIKE" OBJ$NAME:[5] "TEST1" OS$USERID:[6] "oracle" **DBID:[10] "1635432707**" PRIV$USED:[2] "40"

Standard Auditing

Can audit statements, privileges and objects. Enabled by setting the AUDIT_TRAIL parameter. Configured with AUDIT/NOAUDIT commands

NONE = disables standard auditing

OS = writes audit records to a file in AUDIT_FILE_DEST

DB = writes audit records to the SYS.AUD$ table

DB, EXTENDED = Includes SQL statement and bind values

XML = writes audit records to an OS file in XML format

XML_EXTENDED = writes to an OS file in XML format plus records the SQL statement and bind values to SYS.AUD$

Standard Auditing

Privilege auditing:

```
audit select any table;
(dba_priv_audit_opts)
```

Statement auditing:

```
audit select table;
(dba_stmt_audit_opts)
```

Object auditing:

```
audit select on SCOTT.SALARY;
(dba_obj_audit_opts)
```

Standard Auditing

```
audit select table;

audit select table by session | by access;

audit select table whenever not successful

audit select table by SCOTT;

noaudit select table;

noaudit all;

noaudit all privileges;
```

Fine Grained Auditing (FGA)

Defines specific conditions that must take place for the audit to occur. It provides granular auditing of database operations based upon content. Writes records to SYS.FGA_LOG$ or to OS files.

For example:

- Accessing a table outside of normal working hours
- Logging in from a particular IP address
- Selecting or updating a particular table column
- Modifying a value in a table column

Fine Grained Auditing (FGA)

```
DBMS_FGA.ADD_POLICY(
object_schema VARCHAR2,
object_name VARCHAR2,
policy_name VARCHAR2,
audit_condition VARCHAR2,
audit_column VARCHAR2,
handler_schema VARCHAR2,
handler_module VARCHAR2,
enable BOOLEAN,
statement_types VARCHAR2,
audit_trail BINARY_INTEGER IN
  DEFAULT,
audit_column_opts BINARY_INTEGER IN
  DEFAULT);
```

Fine Grained Auditing (FGA)

Audit all updates to the SALARY column of EMPLOYEE if made by anyone other than MIKE

```
begin
DBMS_FGA.ADD_POLICY(
object_schema       => 'MIKE',
object_name         => 'EMPLOYEE',
policy_name         => 'salary_change',
audit_column        => 'SALARY',
audit_condition     =>
  'SYS_CONTEXT(''USERENV'',''SESSION_USER''
  ) <> ''MIKE'' ',
enable              =>  TRUE,
statement_types     => 'UPDATE'
audit_trail         =>  DBMS_FGA.DB);
end;
/
```

What should you be auditing?

In general, you should audit changes to the database (alter database, alter system), DDL, system/object privileges, logon/logoff and unsuccessful operations.

More specifically, you should audit data that is sensitive and/or important to your organization. (salaries, classified data, financial info, etc). This requires you to understand your data

What to look for in the Audit Trail?

- Look for anomalies
  - DDL not during a scheduled build
  - Activity outside of normal working hours
  - Failed attempts to access data or exceed privileges
    - ORA-00942: "table or view does not exist"
    - ORA-02004: "security violation"
    - ORA-01031: "insufficient privileges“
- Excessive failed login attempts
- Alter system/Alter database
- Unauthorized privilege/object grants
- Unsuccessful operations – returncode != 0

What To Look For In The Audit Trail

```
select action_name "Action", priv_used "Privilege",
returncode "ReturnCode",count(*) "Count"
from dba_audit_trail
 where timestamp > sysdate-90 group by action_name,
  priv_used, returncode order by 1;

Action          Privilege                 Code      Count
--------------- ------------------------ -----   ----------
ALTER USER      ALTER USER                   0          17
ALTER USER      ALTER USER               28007           1
CREATE TABLE    CREATE ANY TABLE             0           1
CREATE USER     CREATE USER                  0           1
GRANT ROLE      GRANT ANY ROLE               0           2
LOGOFF                                       0      243955
LOGON           CREATE SESSION               0      246067
LOGON                                     1017          10
SET ROLE                                     0           1
SYSTEM GRANT    GRANT ANY PRIVILEGE          0           2
```

DatabaseSpecialists

What To Look For In The Audit Trail

```
select username,timestamp,sys_privilege,
grantee from dba_audit_trail where
action_name='SYSTEM GRANT';

USERNAME   TIMESTAMP SYS_PRIVILEGE          GRANTEE
---------- --------- --------------------- ---------
JOSANESR   24-OCT-12 CREATE SESSION        PTHOMPSON
JOSANESR   24-OCT-12 SELECT ANY DICTIONARY PTHOMPSON
```

New in Oracle 11G

- Default Auditing! (11.1)
- Changes to AUDIT BY SESSION (11.2.0.1)
- Separate records for LOGON, LOGOFF (11.2.0.1)
- DB_EXTENDED replaced by DB, EXTENDED (11.2.0.1)
- BY ACCESS now the default (11.2.0.2)
- Audit Trail Cleanup process (11.2.0.1)
- Caution when upgrading to 11G with auditing turned on. Refer to Note 1329590.1

Oracle 11 Default Privilege Auditing

ALTER ANY PROCEDURE	CREATE ANY LIBRARY	DROP ANY TABLE
ALTER ANY TABLE	CREATE ANY PROCEDURE	DROP PROFILE
ALTER DATABASE	CREATE ANY TABLE	DROP USER
ALTER PROFILE	CREATE EXTERNAL JOB	EXEMPT ACCESS POLICY
ALTER SYSTEM	CREATE PUBLIC DATABASE LINK	GRANT ANY OBJECT PRIVILEGE
ALTER USER	CREATE SESSION	GRANT ANY PRIVILEGE
AUDIT SYSTEM	CREATE USER	GRANT ANY ROLE
CREATE ANY JOB	DROP ANY PROCEDURE	

Oracle 11 Default Statement Auditing

ROLE

DATABASE LINK

SYSTEM AUDIT

PROFILE

PUBLIC SYNONYM

SYSTEM GRANT

In Addition To 11G Default, I recommend:

- `audit not exists;`
- `audit select any table;`
- `audit select any dictionary;`
- `audit SELECT TABLE whenever not successful;`
- `audit INSERT TABLE whenever not successful;`
- `audit UPDATE TABLE whenever not successful;`
- `audit DELETE TABLE whenever not successful;`

In Addition To 11G Default, I recommend:

- audit table; (shortcut for create, drop, truncate)
- audit alter table;
- audit procedure;
- audit trigger;
- audit view;
- audit index;
- audit grant procedure;
- audit grant table;

Examples of Audit Trail Use for the DBA

- Performance issues caused by excessive logon activity.
- “Table Or View Does Not Exist” – which table?
- Who dropped/altered this table/index/procedure?
- Missing table grants
- Unhandled Oracle errors
- Inappropriate system privileges (`SELECT ANY TABLE` vs direct grants)

Managing the Audit Trail

- Security
  - Restrict access to AUDIT_FILE_DEST, AUD$ and FGA_LOG$ and Audit Config
  - Audit access to AUD$ and FGA_LOG$
- Space Management
  - Define a retention policy and automate purging
  - Move the Audit Trail tables to dedicated tablespaces
  - Include the files in AUDIT_FILE_DEST
  - DBMS_AUDIT_MGMT (>=10.2.0.5 or 10.2.0.3 + patch)

Managing the Database Audit Trail with DBMS_AUDIT_MGMT

1. Move audit tables to dedicated tablespaces

```
DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_LOCATION
( AUDIT_TRAIL_TYPE =>
DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD,
AUDIT_TRAIL_LOCATION_VALUE =>
'AUD_DATA');
```

2. Initialize the Cleanup process and set the Cleanup interval

```
DBMS_AUDIT_MGMT.INIT_CLEANUP(
AUDIT_TRAIL_TYPE =>
DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD,
DEFAULT_CLEANUP_INTERVAL => 12 );
```

Managing the Database Audit Trail with DBMS_AUDIT_MGMT

3. After reviewing/archiving the audit data, set the archive timestamp with

```
DBMS_AUDIT_MGMT.SET_LAST_ARCHIVE_TIMESTAMP(
AUDIT_TRAIL_TYPE =>
DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD,
LAST_ARCHIVE_TIME => sysdate-90);
```

4. Create the Purge job with

```
DBMS_AUDIT_MGMT.CREATE_PURGE_JOB (
AUDIT_TRAIL_TYPE =>
DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD,
AUDIT_TRAIL_PURGE_INTERVAL => 12,
AUDIT_TRAIL_PURGE_NAME => 'Audit Purge',
USE_LAST_ARCH_TIMESTAMP => TRUE );
```

Performance impact of auditing?

Overhead of Standard Auditing during TPC-C OLTP benchmark

Audit Trail Setting	Additional Throughput Time	Additional CPU Usage
OS	1.39%	1.75%
XML	1.70%	3.51%
XML, Extended	3.70%	5.26%
DB	4.57%	8.77%
DB, Extended	14.09%	15.79%

Source:
Oracle Database Auditing: Performance Guidelines

Performance impact of auditing?

Overhead of Fine Grained Auditing during TPC-C OLTP benchmark

Audit Trail Setting	Additional Throughput Time	Additional CPU Usage
XML	3.66%	4.35%
XML, Extended	4.62%	9.09%
DB	6.60%	11.11%
DB, Extended	9.61%	20%

Source:
Oracle Database Auditing: Performance Guidelines

Conclusion

Oracle provides a wealth of functionality that can be used to audit various aspects of your database. When properly implemented and reviewed on a regular basis, auditing is an important and useful tool for securing your database and maintaining compliance with your organizational security requirements.

Questions?

?