

An Oracle White Paper
July 2010

Oracle Database Auditing: Performance Guidelines

Introduction

Database auditing has become increasingly important as threats to applications become more sophisticated. In fact, the use of Oracle database auditing has steadily increased over the past decade and today is mandatory in many organizations. Surveys conducted with the Independent Oracle User Group (IOUG) have consistently shown over 50% of the respondents use some level of native Oracle database auditing.

Database auditing is generally used to:

- Provide proof of monitoring internal controls to auditors
- Provide reports on changes to the database environment to auditors
- Act as a deterrent to unauthorized activity
- Assist with investigations of data breaches or other suspicious activity
- Detect when an attempt is made to bypass a security control

Database auditing monitors and records activity that occurs in the database. Oracle database auditing has been enhanced with each successive release of the database and today provides highly customizable auditing that can be fine tuned to specific security requirements. Oracle9i Database introduced the Fine Grained Auditing (FGA) feature, enabling audit policies to be associated with application tables. Oracle Database 11g enhancements include the ability to audit based on connection factors such as IP address and new built-in manageability features that eliminate the storage and administration costs associated with audit data on the production server.

Perhaps the most common question that arises when it comes to any security solution is “*What about performance overhead?*” This is especially true in the case of commercial organizations where response times and availability impact profitability. As with any security control, database auditing does require additional system resources. This paper helps you determine application throughput and CPU usage when enabling auditing policies in the Oracle database. To understand the impact of Oracle database auditing, let’s first review how Oracle auditing is configured.

Database Auditing Overview

Oracle database auditing can be highly customized based on the granularity of auditing desired. Oracle FGA enables even more customized auditing, enabling audit conditions to be associated with specific columns within an application table, such as a credit card or social security number.

Oracle auditing can be divided into two basic categories: standard auditing and FGA. Standard auditing provides the ability to audit based on user, privileges, schemas objects, and statements. For example, it can be based on a specific type of SQL statement (create, alter, update, delete,...). FGA provides the ability to audit access to specific application table columns conditionally based on factors such as IP address or the program name used to connect to the database.

Starting with Oracle Database Release 11, the Oracle Database Configuration Assistant (DBCA) can automatically configure Oracle recommended minimum audit settings for compliance and internal controls. These audit settings are associated with important security relevant SQL statements and privileges and are also listed in the Oracle security documentation. After creating a database with DBCA, the database will audit the following privileges and SQL statement short cuts by default:

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	
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	
ROLE	SYSTEM AUDIT	PUBLIC SYNONYM
DATABASE LINK	PROFILE	SYSTEM GRANT

Table 1 – Oracle Database 11g Default Audit Settings

Oracle Database Audit Trails

The Oracle database can write records to a database table or an operating system file. If you chose to write the audit record to an operating system file, you can direct it to be text based, XML formatted, or written directly to the SYSLOG on Unix and Linux or the Event Viewer on Windows. The Oracle database OS audit files can be text based with an extension of ‘.aud’ or XML format with an extension of ‘.xml’.

Configuring the Oracle Database Audit Trail

The database parameters (<sid>init.ora) that direct standard audit records are as follows:

Parameter	Value	Description
audit_trail	DB	The standard audit content to sys.aud\$
	DB, EXTENDED	Write standard audit content to sys.aud\$ and include the SQL text and bind variable content that was executed for that SQL
	OS	Write the standard audit content to text files
	XML	Write the standard audit content and FGA audit content to an XML formatted file
	XML, DB	Write the standard audit content and FGA content to an XML formatted file along with SQL text and bind variable content.
audit_sys_oper		Indicates all top-level SYSDBA and SYSOPER activity to be recorded
audit_syslog_level		Provides level information to write text of the audit records into the syslog
audit_trail_dest		Specifies the OS directory location to write the OS and XML audit files

Table 2 – Oracle Database Auditing Parameters

FGA policies are created use the DBMS_FGA package. FGA is conditional auditing to limit when an audit record is created based on values for a user session. FGA audit records can be written to a database table or an XML formatted file by specifying the value DB or DB,Extended or XML or XML,Extended for the DBMS_FGA.ADD_POLICY procedure.

For optimal performance Oracle recommends the following “auditing best practices” for standard auditing when writing to database tables and OS files.

- Optimize Space Management - create an audit specific, user-defined tablespace, for example AUDSYS, with pre-created sized (1GB) extents for audit trail tables.
- Load Distribution - move audit trail tables, sys.aud\$ and sys.fga_log\$ to the new user-defined tablespace.
- Reduce number of OS audit files - set larger file-size (100MB) for OS audit files using the DBMS_AUDIT_MGMT package.

Performance Testing Results

To demonstrate the resources used by the Oracle database with auditing turned on, testing was performed based on the destination where the audit records are being written (database table sys.aud\$ or operating system file (OS)) with a TPC-C like workload generating approximately 250 audit records per second. A TPC-C like workload is used since it provides standardized OLTP database activity for complex application environments. The workload is characterized by simultaneous execution of multiple transaction types that span breadth of complexity that are common across all industries.

Machine Configuration

The Oracle Database Release 11.2.0.1 was installed on hardware with the following configuration:

- 4 x 3.40 GHz Xeon CPUs
- 4 GB memory
- x86_64 GNU/Linux

The database also included the following one-off patches:

- Remove Oracle database OS flush call after every XML audit write and rely on the Operating System to flush the writes to disk. (9078032)
- Remove XML Index file. (8880803)

Oracle performed individual testing based on the location of the audit file being written as specified by the Oracle database audit_trail value and the DBMS_FGA.ADD_POLICY audit_trail parameter setting.

Audit Performance Overhead

Before the audit test was run, a standard workload was introduced to use 50% of system resource before auditing was initiated. For each test run the following results were recorded:

- Throughput: - Additional time used by the transaction after auditing was turned on
- Additional CPU Usage – Measured additional CPU after auditing was turned on

For standard database auditing, a test was created to generate approximately 250 audit records per second using the Oracle database standard audit command.

CPU Initial Load	Audit Trail Setting	Additional Throughput Time	Additional CPU Usage
50%	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%
50% + Auditing Best Practices	OS	1.39%	1.75%
	XML	1.70%	3.51%
	XML, Extended	3.22%	4.26%
	DB	3.64%	8.16%
	DB, Extended	13.68%	16.33%

Table 3 – Oracle Database 11.2.01 Standard Audit Trail

For fine-grained auditing (FGA), a test was created to generate approximately 200 audit records per second using the DBMS_FGA package. The condition of the audit policy creates an audit record when an UPDATE or SELECT occurs on the TPCC.ORDL table and the client_identifier value is equal to NULL.

```

dbms_fga.add_policy (
object_schema => 'TPCC',
object_name   => 'ORDL',
policy_name   => 'Config_A',
audit_condition =>
  'SYS_CONTEXT(''USERENV'', ''CLIENT_IDENTIFIER'') IS NULL',
statement_types => 'UPDATE, SELECT',
audit_trail    => DBMS_FGA.XML +DBMS_FGA.EXTENDED);

```

CPU Initial Load	Audit Trail Setting	Additional Throughput Time	Additional CPU Usage
50% + Auditing Best Practices	XML	3.66%	4.35%
	XML, Extended	4.62%	9.09%
	DB	6.60%	11.11%
	DB, Extended	9.61%	20%

Table 4 – Oracle Database 11.2.01 Fine Grained Audit Trail

Conclusion

Threats to applications have become more sophisticated and database auditing plays an important part not only in helping detect suspicious behavior but providing proof of controls to auditors. Oracle database auditing has minimal impact on performance even for very high audit trail loads. For optimal performance, Oracle recommends writing database audit records to the operating system (OS) and setting larger file sizes for the OS audit files. Oracle security documentation provides best practice recommendations on minimal audit settings that should be turned on for both compliance and internal controls. Auditing inside the database, whether Oracle or non-Oracle databases, should be part of your defense-in-depth architecture.



Oracle Database Auditing: Performance
Guidelines

JULY 2010

Author: Tammy Bednar

Contributing Authors:

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd. 0110