

Database Auditing Best Practices

INTRODUCTION

This document has been produced based on the stated requirements from a number of major companies with whom we have worked to review and document their requirements for database auditing and controls.

The purpose of this document is to make this set of generic best practices available as a checklist to other companies to assist them to define their business and technical requirements for database auditing. This should help to:

- Meet internal and external audit requirements
- Meet compliance requirements
- Reduce operational risk
- Reduce ongoing costs of controls through effective automation
- Provide the capability to investigate the detail and causes of any data change

NOTE: The scope of this document is specifically focused on the implementation of technical controls and systems to achieve automated auditing of actual database activity. It does not cover the procedural controls around database security i.e. user access and security controls, account privileges, operational procedures etc.

-
-
-

Keith Hutton

Cervello Consultants 2009

CONTENTS

Introduction	1
Auditing Capability	3
Audit Methods	3
Audit Capability – activity type	3
Completeness of the Audit Record	3
Audit result collection and storage	4
Audit Coverage – DBMS Platforms.....	4
Forensic Audit	4
Audit Configuration and Control	4
Data audit policies/filters	4
Data audit policy/filter conditions	4
Data audit policy/filter deployment.....	5
Audit Automation.....	5
Audit Actions and Alerts.....	5
Reporting capability	5
Reports – Pre-defined	5
Audit Investigation - Forensics	5
Workflow Management	5
Retrospective investigation.....	5
Vulnerability Assessment	5
System Integrity	6
System Security	6
System Vulnerability	6
Separation of Duty	6
Reliability and Impact.....	6
Archive.....	6
Recoverability.....	6
System Deployment and Operation.....	6
Ease of installation	6
Management Console	6
User Access and Control.....	6
Usability.....	6
Enterprise.....	7
Performance Impact.....	7
Scalability	7
Integration.....	7
Communication	7
Footprint	7
System Technical Requirements	7

AUDITING CAPABILITY

Audit Methods

Capture of data change instructions (SQL statement)

- The system must capture the source SQL command which creates the database change. E.g. an instruction to double the salary of some of the employees:

```
update employees set salary = salary * 2 where dept = 'HR'
```

- The system must capture all the execution-time parameters (see “Completeness of the audit record”).

Capture of actual data changes (Change Log)

- The system must capture the ACTUAL data changes resulting from a SQL command. E.g. in respect of the SQL example above which changed 3 rows:

```
for employee_id = 23; salary before = 20000, salary after = 40000
```

```
for employee_id = 54; salary before = 30000, salary after = 60000
```

```
for employee_id = 18; salary before = 16000, salary after = 32000
```

- The system must capture all the execution-time parameters (see completeness of the audit record).
- The system must capture the full details of any data row which is **inserted**.
- The system must capture the full details of any data row which is **updated**.
- The system must capture the full details of any data row which is **deleted**.

Configurability

- The system must have the capability to allow the user to configure any target database to be audited by any one or more of the above methods, and to change this configuration seamlessly if auditing requirements change.

Audit Capability – activity type

- The system must be able to capture all of the following activity types:

User logins and login failures

- Full details of the login to the DBMS, including date, time, user account, host name and host login account.

Structural changes DDL (schema changes)

- All commands that alter the structure of the database e.g. ALTER TABLE, DROP TABLE etc.

Data modification DML (data change)

- All commands that modify the data itself. i.e. INSERT, UPDATE, DELETE, EXECUTE PROCEDURE etc.

Data access SELECT

- All commands that read or access data i.e. SELECT, EXECUTE PROCEDURE etc.

Database configuration changes

- All commands that alter the configuration of the DBMS or database.

User permission changes

- All commands that change the security of the database by creating user accounts or by changing the access privileges for user accounts.

Completeness of the Audit Record

The system must be able to record, for each data change:

- Source user (OS-level) login account
- Source host name or IP address
- Source application
- Source user (DBMS-level) login account
- Host name of the data server
- Instance name of the data server
- DBMS type of the data server
- Change operation (INSERT/UPDATE/EXECUTE etc)
- Full object name of the item changed (database-schema/object-owner/object-name)
- Exact time and date of the change
- Exact values of data changed in every column - for every data row changed

Audit result collection and storage

- The system must be able to collect audit data from a number of disparate and unrelated systems and store all the audit records in one consolidated audit data store for automated analysis and reporting.

Audit Coverage – DBMS Platforms

- The system must be able to audit all of the key DBMS platforms in use in the Company (e.g. Oracle, Microsoft SQL Server, Sybase, DB2 etc). (... insert Company specific requirements ...)

Forensic Audit

- The system must have the capability to retrospectively audit the trail of changes which have been made to a database, even if this audit trail was not captured by the auditing filters at the time. For example, by re-inspecting a restored backup copy of the database and the transaction logs from a previous point in time, the tool should be able to identify all changes made by a specified user or to a specified table.

AUDIT CONFIGURATION AND CONTROL

Data audit policies/filters

Normal usage - violation

- The system must allow the user to configure a set of policies/filters by which 'normal' data usage can be defined, so that the system automatically triggers if data usage outwith these conditions occurs.

Unauthorised usage

- The system must allow the user to configure a set of policies/filters which define unauthorised data usage, so that the system automatically triggers if data usage matching these conditions occurs.

Data audit policy/filter conditions

- The system must allow policy/filter conditions to be defined by configuring any one or more of the following filter conditions to operate in conjunction.

By Object

- Name of object(s) (e.g. table/view/procedure) to watch
- Name of object(s) to ignore

By Individual column

- Name of specific column(s) in table to watch
- Name of specific column(s) in table to ignore

By User

- Name of user(s) to watch
- Name of user(s) to ignore

By Operation

- Type of operation(s) to watch
- Type of operation(s) to ignore

By Source Host

- Name(s) or IP address of the source host(s) to watch
- Name(s) or IP address of the source host(s) to ignore

By Source Application

- Name(s) of the source application or tool to watch
- Name(s) of the source application or tool to ignore

By Day of Week

- Specified days of the week to watch or to ignore

By Time Range

- Specified time range to watch or to ignore

Data audit policy/filter deployment

- The system must facilitate the deployment of defined audit policies/filters to multiple heterogeneous servers at the same time.
- The system must allow the user to edit and save policy/filter templates; to copy existing policies/filters; and to create new ones.

Audit Automation

- The system must enable 24/7 automated operation of the auditing policies/filters and associated defined alerts and reports in order to ensure that the Company's data usage policies are being continuously enforced.

Audit Actions and Alerts

The system must permit the user to define what action to take following a violation of an audit policy/filter:

- Optionally send an alert to an identified person
- Optionally record the data activity in the audit data store
- Optionally mark the data activity record with a defined level of severity

Reporting capability

- The system must have a management dashboard that provides a high-level view of the audit status.
- The system must provide the capability to drill-down from a dashboard or high-level report in order to view more detailed information.
- The system must enable users to design their own reports.
- The system must enable users to define schedules for specified reports to run.
- The system must provide report output to common formats e.g. PDF, HTML, Excel

Reports – Pre-defined

The system must contain pre-defined reports to fulfill standard reporting requirements for:

- Sarbanes Oxley (SOX) 404 controls
- PCI

Audit Investigation - Forensics

- The system must provide reporting and analysis tools that allow the user to undertake ad-hoc search for data changes that match any given conditions. E.g. find all changes to Mrs. J Smith's account balance in the last 30 days; or find all changes to the payroll table, or find all changes made by a specified privileged user.
- The schema for the audit data store (and appropriate training) must be made available to the customer in order to facilitate development of forensic queries.

Workflow Management

- The system must have the capability to record and track comments, investigation details and sign-off of audit reports.
- The system must have the capability to define workflows and roles to route audit reports and notifications through a structured process.

Retrospective investigation

- The system must be able to perform an audit of an operational system retrospectively. I.e. where the audit system was not deployed during live operation, it should be possible to perform a retrospective audit at a point in time by use of a backup copy of the operational database as at that time and the database change logs for the previous period.

Vulnerability Assessment

- The system must have the capability to assess the security profile of the operational DBMS instances and proactively report on security settings which are sub-optimal and DBMS security patches which need to be applied.
- The system must have the capability to define a 'baseline' snapshot of the operational systems, and show changes to the configurations from that baseline.

SYSTEM INTEGRITY

System Security

- The audit information must be fully secure at all stages, so that there is absolutely no ability for it to be accessed or tampered with by unauthorized users – particularly users who have privileged operational accounts.
- The audit records must be encrypted.
- The audit data must be stored securely in an enterprise-class database.

System Vulnerability

- There must be no vulnerability or ‘backdoor’ by which a privileged user, who is a member of the group being audited, could alter the operation of the system, make changes that are not detected, or alter the records so that the audit trail is compromised in any way.

Separation of Duty

- The system must provide true separation of duty as required by many compliance regulations (e.g. SOX).
- Absolutely no member or part of the organizational unit which is being audited should have any control whatsoever over the design, development or operation of the auditing system.
- All components of the auditing solution must be separate from the operational systems.

Reliability and Impact

- If the system becomes unavailable for a period, that should not result in unrecoverable loss of audit data.
- If the system becomes unavailable this must not impact the operation of the production systems.
- The system must monitor itself for failed processes and re-initiate these.
- The system must be stand-alone from the production systems being audited.
- The system must not ‘drop packets’ in periods of high load, and thus have a gap in the audit trail.

Archive

- The system must provide the capability to archive the audit records for several years, as legislation requires, without risk of the files being modified, overwritten or deleted.
- The system must provide the capability to easily restore archived data for inspection.

Recoverability

- It must be easy to restore the system following failure, using standard backup and recovery processes.

SYSTEM DEPLOYMENT AND OPERATION

Ease of installation

- The system must not require changes to be made to the operational systems in order to implement the auditing solution. In particular it should not require table triggers to be installed onto the operational system.
- The system must not require downtime of the operational systems in order to implement the operational systems.

Management Console

- The system must have a browser-based management console which can be made accessible from any remote location. Access to the system via the browser console must be fully secure.

User Access and Control

- The system must allow appropriate permissions to be given to different sets of users, so that each set of users has the appropriate level of access and cannot access or change aspects which are not appropriate for their role.

Usability

- The system must have a graphical user interface which allows non-technical users to implement and control the auditing policies/filters and reports.

ENTERPRISE

Performance Impact

- The system must not significantly impact the performance of the operational systems.
- It should be possible to schedule the system to operate outside normal business hours.

Scalability

- It should be possible to scale the deployment of the system in order to audit several hundred servers.
- Scaling up should not require the introduction of additional system components or changes to the architecture or method.

Integration

- The system must have a published API conforming to open standards (e.g. SOAP) which can be used to control the system and/or to integrate with other systems.
- The system must have a published data model of the audit data to facilitate customer-defined links, reports and analysis to be created for this data.

Communication

- The system must be able to communicate alerts automatically to other enterprise systems, e.g. BMC Patrol / Tivoli etc.
- The system must be able to send alert notifications via email and SMS.

Footprint

- The system should be able to operate within the strategic technical environment defined by the customer, without introducing additional HW or SW technologies or components that need to be managed and supported.

System Technical Requirements

HW Platforms

- The system should be able to run on the common industry-standard HW platforms: i.e. Intel 32-bit, 64-bit, Itanium, Sparc, RISC etc.
- The system should not require the installation of specific hardware, the client should be able to select the HW that conforms to their strategic platform.

OS Platform

The system should be able to operate on any of the following industry-standard platforms (... **insert Company specific requirements ...**):

- Microsoft Windows – all 32-bit and 64-bit versions
- Solaris
- AIX
- LINUX

DBMS Platform

The system should be able use any of the following enterprise-class DBMS platforms for storage of the audit data (... **insert Company specific requirements ...**):

- Oracle
 - Microsoft SQL Server
 - IBM DB2
 - Sybase ASE
-