Return to Home Page

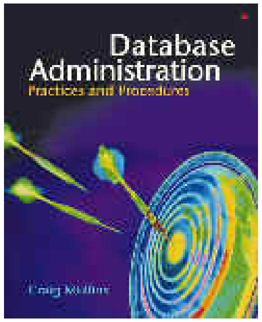
November 2007

DATABASE
TRENDS AND APPLICATIONS

The DBA Corner

by Craig S. Mullins

Data Access Auditing: A Compliance Requirement



We are all surely aware of the increasing burden of complying with government regulations. Nowhere is the pressure to comply greater than on data stored in corporate databases. Organization must be hyper-vigilant as they implement controls to protect and monitor their data, and the access to it.

One of the more useful techniques to protect your company’s database data is via data access auditing, which is a facility for tracking the use of, and modifications made to, database resources and authority. A data access auditing capability enables companies to produce an audit trail of information with regard to their database data. This audit trail can contain information such as what database objects were impacted, who performed the operations, and when. A comprehensive audit trail of database operations, coupled with an analysis engine allows DBAs and auditors to perform in-depth analysis of access and modification patterns against data in the DBMS. Only when armed with such details is it possible to comply with regulations, pass security audits, and drill-down into the details to review potential vulnerabilities for effective issue resolution.

Why would you need a fine-grained audit trail? Well, many of the PCI Data Security Standard requirements emphasize the importance of real time monitoring and tracking of access to cardholder data, as well as continuous assessment of database security health status. And HIPAA, the Health Insurance Portability and Accountability Act, directs health care providers to protect individual’s health care information going so far as to state that the provider must be able to deliver a list of everyone who even so much as looked at their patient’s information. Could you produce a list of everyone who looked at a specific row or set of rows in any database you manage?

Data access auditing is important because there are many threats to the security of your data. External agents trying to compromise security and access your company’s data are rightly viewed as a security threat. But industry studies show that the majority of security threats are internal – within your organization. Indeed, internal threats can comprise 60% to 80% of all security threats. The most typical security threat comes from a disgruntled current or ex-employee that has valid access to the DBMS. Auditing is crucial because you may need to find unauthorized access emanating from an authorized user.

A typical auditing facility permits auditing at different levels within the DBMS, for example, at the database, database object, and user levels. One of the biggest problems with existing internal DBMS audit facilities is performance degradation. The audit trails that are produced must be detailed enough to capture before- and after-images of database changes. But capturing so much information, particularly in a busy system, can cause performance to suffer. Furthermore, this audit trail must be stored somewhere which is problematic when a massive number of changes occur. Therefore, a useful auditing facility must allow for the selective creation of audit records to minimize performance and storage problems.

There are several popular techniques that can be deployed to audit your database structures. The first technique is trace-based auditing, which is usually built directly into the native DBMS. Commands are set to turn on auditing and the DBMS begins to cut trace records when activity occurs against audited objects. The problems with this technique include a high potential for performance degradation when audit tracing is enabled, a high probability that the database schema will need to be modified, and insufficient granularity of audit control, especially for reads.

Another technique is to scan and parse the database transaction logs. Every DBMS uses transaction logs to capture every database modification for recovery purposes. Software exists that interprets these logs and identifies what data was changed and by which users. The drawbacks to this technique: reads are not captured on the logs, there are ways to disable logging that will cause modifications to be lost, performance issues scanning volumes and volumes of log files looking for only specific information to audit and the difficulty of retaining logs over long periods.

The third data access auditing technique is proactive monitoring of database operations at the server. This technique is the best because it captures all SQL requests as they are made. Beware because some proactive monitors sniff network traffic, but it is important that all SQL access is audited, not just network calls because not every SQL request goes over the network. Proactive audit monitoring does not require transaction logs, does not require database schema modification, and should be highly granular in terms of specifying what to audit.

Database auditing can be a crucial component of database security and compliance with government regulations. Be sure to study the auditing capabilities of your DBMS and to augment these capabilities with additional tools to bolster the auditability of your databases.

From Database Trends and Applications, November 2007.

Home.