October 2007

Group Practice Journal

Tracking Data Access for Compliance

By Craig S. Mullins

In a world replete with regulations and threats, organizations today have to go well beyond just securing their data. Protecting this most valuable asset means that companies have to perpetually monitor their systems in order to know who did exactly what, when and how - to their data.

Increased regulations demand the implementation of policies and procedures to protect sensitive healthcare data. To ensure compliance, medical professionals must be ever vigilant in the techniques used to manage and protect the data under their care.

Many healthcare records are electronically stored in database management systems. As such, to be compliant, companies must pay attention to how data in those databases is governed. One important technique is to audit access to database data. This can be accomplished using database auditing software.

It is not enough to simply set up database authorization using the controls within the database software. This is true for several reasons. First of all, it is impossible to guarantee that surreptitious access to database data is blocked with simple database authorization mechanisms. And secondarily, it is possible for authorized users to nefariously access data.

Robust database auditing software can comprehensively track the usage of database resources and authority. When auditing is enabled, each database operation produces a detailed audit trail of information tracking what data was accessed, who accessed it and when. Operators can analyze the audit trail and generate reports showing access and modification patterns against the healthcare data in the DBMS.

Database auditing helps answer questions such as, “Who accessed the Mr. Jones’ patient details?” and “When was Ms. Smith’s appointment time changed?” and “Who changed that appointment time?” It is even possible to answer more detailed questions like “What was the old appointment time prior to the change?”

The ability to answer such questions is very important for regulatory compliance. Sometimes it may be necessary to review certain audit data in greater detail to determine how, when, and who changed the data.

Regulatory Compliance

Why would one need to ask such questions? Well, one obvious answer to that question is to be in compliance with the Health Insurance Portability and Accountability Act (HIPAA). This legislation contains language specifying that healthcare providers must protect individual’s healthcare information, even going so far as to state that the provider must be able to track everyone who even so much as looks at an individual’s healthcare data.

HIPAA audits frequently require the examination of the processes used to create, document and

review exception reports and logs. When confronted with a HIPAA audit, organizations can be required to produce a list of exceptions to policy, such as, “When were patient records accessed during off hours and by whom?” Without database auditing software, it is impossible to produce a list of users who looked at a specific row or set of rows in any database.

Tracking who does what to each piece of regulated data is important because there are many threats to database data security. External agents trying to compromise security and access company data are rightly viewed as a threat to security. But industry studies have shown that the majority of security threats are internal – within an organization. Indeed, some studies have shown that internal threats may comprise as much as 80 percent of all security threats. The most typical security threat comes from a disgruntled current (or ex-) employee with valid authorization access to data. In these instances, auditing is crucial to find an unauthorized access emanating from an authorized user.

Audit trails help promote data integrity by enabling the detection of security breaches, also

referred to as intrusion detection. An audited system can serve as a deterrent against data tampering because infiltrators are more easily identified, and caught.

Things to Watch For

A typical auditing facility permits auditing at different levels within the DBMS, for example, at the database, database object level, and user levels. But capturing so much information, particularly in a busy system, can cause performance to suffer. Production of the required audit details must be accomplished without diminishing the operations of the computerized systems that keep the practice functioning.

The detail and authenticity of the audit trail produced is just as important as the operational systems' performance. The audit trails must be detailed enough to capture before- and after-images of database changes. If the mechanism capturing the audit details is not comprehensive and efficiently engineered, it ceases to be a compliance solution. Furthermore, the audit trails must be stored somewhere that protects the authenticity of the audited information while allowing seamless access for reporting.

Due to the potential volume of changes made to database data, a useful auditing facility must allow for the selective creation of audit records to minimize performance and storage problems. The general rule of thumb is that only data which must be audited to be in compliance should be audited, and nothing more.

Auditing Techniques

There are several popular techniques that can be deployed to audit database data. By far, the best technique engages proactive monitoring of database operations directly at the database server. This technique captures **all** requests for data as they are made. By capturing the audit details at the server level the software can guarantee that all access is monitored. Other techniques, such as trace-based auditing or parsing database logs can miss certain types of database activities.

A robust database access auditing solution that addresses regulatory compliance should be able to provide answers to at least the following questions:

1. Who accessed the data?
2. At what date and time was the access?

3. What program or client software was used to access the data?
4. From what location was the request issued?
5. What SQL was issued to access the data?
6. Was the request successful; and if so, how many rows of data were retrieved?
7. If the request was a modification, what data was changed? (A before and after image of the change should be accessible)

Of course, there are numerous details behind each of these questions. A robust database access auditing solution should provide an independent mechanism for long-term storage and access of audit details. The solution should offer canned queries for the most common types of queries, but the audit information should be accessible using industry standard query tools to make it easier for auditors to customize queries as necessary.

Author Biography:

Craig S. Mullins is an executive and data management strategist with NEON Enterprise

Software, Inc. He has extensive experience in the field of database management and has authored of two books on the subject: “DB2 Developer’s Guide” and “Database Administration: The Complete Guide to Practices & Procedures.” Contact Craig via his web site at http://www.craigsmullins.com.

From Group Practice Journal, October 2007.

Home. Phone: 281-494-6153 Fax: 281-491-0637