SearchSecurity

TechTarget

E-Guide

Best Practices for Database Security

***Databases contain a large amount of highly** sensitive data, making database protection extremely important. But what about the security challenges that can pose a problem when it comes to keeping attacks at bay? Read this expert E-Guide to learn about the best practices for managing databases and the steps your enterprise should take to secure them.*

Contents

Best Practices for Enterprise Database Compliance

By: Charles Denyer, Compliance, Frameworks

Databases are a treasure trove of data, often highly sensitive data, and not surprisingly are an important area of emphasis for compliance programs. Almost all enterprise compliance regulations feature requirements concerning who can access what database and when, and managing these permissions can easily be a full-time job. In this article, we'll cover the basic database security requirements necessary for database compliance with major regulations such as PCI DSS and HIPAA, as well as best practices for managing database permissions and upkeep in order to maintain compliance with those regulations.

All five of the most common enterprise core database environments (1. Microsoft SQL Server; 2. IBM DB2; 3. MySQL; 4. Oracle; 5. Postgres) have the ability to be appropriately provisioned, hardened, secured and locked down when conducting an initial installation. The challenge is understanding the important components that actually need to be in place. It's not just the database itself; it's the server the operating system and the database reside on.

PCI DSS currently requires the following explicit controls for databases:

- All users are to be authenticated prior to accessing any databases.

Contents

- All user access to any databases, user queries and user actions (such as move, copy and delete) are done so through programmatic methods only (such as stored procedures).

- Database and application configuration settings restrict only direct user access or queries to database administrators (DBA).

- For database applications and the related application IDs, application IDs can only be used by the applications, not by individual users or other processes.

Regarding HIPAA, the above measures are not specifically stated as requirements for HIPAA compliance, but should be looked upon as best-of-breed security controls for complimenting, and ultimately, helping meet the needs for the “security” provisions within HIPAA. Specifically, those provisions of HIPAA require the following:

- Ensure the confidentiality, integrity, and availability of all e-PHI created, received, maintained or transmitted;

- Identify and protect against reasonably anticipated threats to the security or integrity of the information;

- Protect against reasonably anticipated, impermissible uses or disclosures; and

- Ensure compliance by their workforce.

Additionally to compliment the regulatory compliance initiatives, such as PCI DSS, the following are considered best practices for securing all database environments listed above.

Regarding the host operating system on the server that supports the database, the following best practices should be in place:

Contents

1. System administrators and other relevant IT personnel should have adequate knowledge, technical skill-sets and an understanding of all critical operating system security requirements.

2. Industry-leading configuration standards and supporting internal documentation should be utilized when deploying operating systems into the managed services environment.

3. Only necessary and secure services, protocols, daemons and other essential functions should be enabled on the operating systems.

4. All unnecessary functionality and all insecure services and protocols should be effectively disabled on the operating systems.

5. Root accounts should be appropriately secured with the selection of a unique password that is changed on a regular basis.

6. Root accounts should be restricted to the fewest number of personnel necessary.

7. Syslog should be configured for sending and copying syslog data to a central syslog server, for which log information is reviewed.

8. The principle of "least privilege," which states users are only given privileges that are required to efficiently and properly perform their job function, should be in place regarding operating system access rights.

9. All relevant and critical security patches should be applied to operating systems as warranted.

For the actual database itself, the following best practices are recommended:

1. A list of authorized users who have access to databases within the managed application services environment should be maintained and kept current by appropriate personnel.

2. System administrators and other relevant IT personnel should have adequate knowledge, technical skill-sets and an understanding of all critical database security requirements.

3. Industry-leading configuration standards and supporting internal documentation should be utilized when deploying databases into the managed services environment.

4. Default user accounts that are not necessary for database functionality should be locked and expired.

5. For all user default accounts that remain in use, passwords should have been effectively changed to invoke strong password measures.

6. Administrative accounts within the databases should have different passwords assigned to them, with no shared or group passwords being used for these accounts.

7. Measures should be in place for protecting the data dictionary and the supporting metadata that describes all objects in the database.

8. For any host-based authentication measures in place for accessing the database, appropriate procedures should be in place for ensuring the overall security of this type of access.

9. Database monitoring should be in place consisting of tools that alert appropriate personnel as needed.

10. All relevant and critical security patches should be applied to the databases as warranted.

Thus, companies should first and foremost have an IT staff that is well-trained, knowledgeable in database security, and has the necessary provisioning guidelines and hardening documents for implementing effective database security. For all existing database platforms in place and for future database installs, a highly structured and standardized approach needs to be in place for effectively provisioning, hardening, securing and locking down the database environment. Follow these best practices and with any luck your enterprise database compliance efforts will pay off handsomely when your next assessment rolls around.

About the author:
Charles Denyer is a member of NDB Accountants & Consultants, a nationally recognized boutique CPA and advisory firm specializing in Regulation AB, SAS 70, SSAE 16, ISAE 3402, FISMA, NIST, HIPAA, ISO and PCI DSS compliance, along with other regulatory compliance initiatives. Mr. Denyer is actively involved in numerous professional associations and organizations for a wide range of industries and business sectors. He is also an advanced social media expert, having spent years working in the field of search engine optimization (SEO) and various forms of online marketing and social media. Mr. Denyer holds numerous accounting and technology certifications along with a Masters in Information and Telecommunication Systems from the Johns Hopkins University and a Masters in Nuclear Engineering. He is also currently an MBA candidate for the Johnson School of Business at Cornell University.

Database Monitoring Best Practices: Using DAM Tools

By: Adrian Lane, Contributor

There are few IT security challenges more difficult than protecting databases and the data they store, especially from the most common database and Web application attack: SQL injection. Even though relational database management system (RDBMS) vendors, IT security professionals and application developers are all aware of these attacks, they remain a problem because the attacks are difficult to detect and stop without compromising business operations.

What's more, SQL injection is one of many common avenues of assault that allows attackers to take complete control of a relational database. The

Contents

complexity of the relational platform, coupled with multiple applications moving data in and out of the system -- each supporting a variety of business functions -- makes it difficult to differentiate good from bad. When attacks look like normal database commands, you have to do more than a casual inspection of events.

For those who may not be familiar with the technology, database activity monitoring (DAM) systems are advanced security platforms used to detect misuse of relational databases. DAM is unique in that it analyzes database queries in near real time to differentiate between normal operations and attacks. The systems collect information from different sources, provide several forms of advanced analytics, alerting or even halt malicious activity. No other security product focuses on database activity in this way, or offers the level of granular inspection provided by DAM.

Prioritize which data, transactions to protect

The first step in any DAM deployment is deciding what you want to protect. In any discussion of database monitoring best practices, it's not practical to monitor every event because your monitoring system would be larger than the databases it was designed to protect. You need to have some understanding of what data and/or transactions are important. There are three ways to do this:

1. **Interview** database administrators and application developers who set up the databases because they typically know where sensitive data resides, and which databases support critical business functions.

2. **Inspect** database contents using data discovery and database crawlers. Monitors can, at the very least, detect sensitive data as it moves in and out of the database. Some vendors also provide crawlers to search database content as an add-on option. These inspection tools locate sensitive information through a combination of meta data and content analysis techniques to determine what needs to be protected.

Contents

3. **Observe** SQL statements and database transactions. Most DAM systems are deployed in monitor-only mode for the first few weeks so the organization can develop an understanding of what is happening with the database. In essence, you profile how applications use the database, and what typical queries look like. Then policies can be developed and implemented to detect misuse.

Based upon what you find, you can define rules of what activities are allowed, and what suspicious activity should generate an alert.

How to capture database events

Now that you know what types of transactions are important, you need to decide how you want to collect database events. Every database monitor offers multiple methods for data collection, and each comes with advantages and disadvantages.

Agents installed on the database platform are common because they capture all SQL activity, which is desirable for understanding if a query was intended to be malicious without compromising database performance.

Native audit features collect events, but don't always gather the original SQL queries, and cost a great deal more in terms of performance overhead.

Network collectors offer a quick and easy way to collect a majority of SQL activity, but miss some transactions and activity performed by administrators working at the console.

Agents are the de facto deployment for security of critical databases. Native auditing for compliance and network monitoring of non-critical databases is common, but used more in special cases.

Defining basic database security

Now that you are collecting events from critical systems, it's time to implement your security policies. DAM works by analyzing database queries, and you have many options to specify which statements are examined and how. The fundamental features you can expect a database activity monitoring solution to provide are:

- Monitoring and discovery
- Alerts and reports
- Verification of work orders
- Catching misuse
- SQL capture for auditing

Most policies are enforced by examining attributes of the database query: whom the user is; what columns the user is viewing; what application they are using; how much data did they touch; and time of day are all commonly used to define security policies. You assign an arbitrary value to each of these attributes, and the monitoring system will generate alerts when the user exceeds the defined threshold. For example, you may want to alert on all queries after midnight, or after 3 failed login attempts, or any time someone accesses credit card data.

Advanced monitoring

Database activity monitoring systems have greatly advanced their capabilities in the last couple of years. What used to be purely monitoring and altering now provides a reliable method of blocking attacks and actively resisting misuse. The advanced features you should find with most DAM products include:

- SQL injection detection
- Blocking and virtual patching
- Identifying specific application users
- Session termination
- Behavioral monitoring and insider threat detection

But advanced analysis means advanced policies that are specific to your environment, and these don't come from your vendor. In order to detect and block SQL injection attacks, you need to define legitimate SQL queries you wish to allow from the application. If you can't implement a database patch in a timely fashion, you need to write a policy to detect the attack and deploy DAM to block the threat. Hopefully, your DAM vendor will help with these policies if your database vendor does not.

In order to perform behavioral monitoring, in essence detecting abnormal behavior, you need to define what's considered normal. To identify specific application users -- not just the generic accounts that connect from the application to the database -- you need to provide the means to lookup IP addresses or pass user credentials. If a serious threat is detected, you need to determine if you want to disconnect¬ the user or lock the account from being accessed. All of these advanced features require custom work on your part; while DAM vendors provide templates and tools from which to build your policies, detection and enforcement are specific to your environment, so the policies must be customized by you.

Deploying DAM

There are several operational aspects to managing DAM platforms that you want to employ from the outset to save yourself time and trouble in the long run. These include:

Separation of duties: Both for security and compliance reasons, the people who write policies and review the reports should not be the database administrators that manage the monitored databases. Similarly, a DBA for one group should not be allowed to use DAM tools to peer into other groups of databases. The idea is to provide checks and balances to detect fraud, so divide up roles and responsibilities within the DAM product.

Long-term storage: Database activity monitoring platforms don't commonly have the storage capacity of security information and event management (SIEM) and log management platforms. While they provide some correlation capabilities, these products focus on statement-level analysis, not long-term storage and management. Events are seldom kept within a DAM product for more than 30 days. If you need to perform forensic analysis on a 90- or 180-day window, plan on adding storage to an existing DAM server or leveraging log management systems help in this area.

Scalability: There are three critical issues to scalability of DAM; transactional volume, network topology and the resources dedicated to the monitoring. DAM architectures are comprised of local data collectors, appliances that analyze events and generate alerts, and management appliances that provide central policy management and reporting. You need to ensure each of these pieces can reliably communicate with one another, and that you can collect events from databases deployed in virtual environments. To maximize your DAM investment, it's best to understand the type of events you want to analyze, and filter everything you are not worried about. The fewer events means reduced storage and processing overhead. When monitoring databases that process millions of requests per hour, filtering can dramatically cut your investment in appliances or servers to support monitoring.

Database activity monitoring is a mature technology, and its focus on database transactions provides an effective means of securing data and blocking malicious actions. But getting value out of a monitoring platform requires investment in building rules, alerts and reports to address the security issues you face.

Further, careful section of deployment options is needed to avoid creating administration headaches or performance problems. There are many uses for DAM, so you will need to have a clear idea of what your priorities are going in, and focus on getting the basics working first before you move onto more advanced security features.

About the author:
Adrian Lane is CTO and security strategist for information security research and analysis firm Securosis.

TechTarget

Free resources for technology professionals

TechTarget publishes targeted technology media that address your need for information and resources for researching products, developing strategy and making cost-effective purchase decisions. Our network of technology-specific Web sites gives you access to industry experts, independent content and analysis and the Web's largest library of vendor-provided white papers, webcasts, podcasts, videos, virtual trade shows, research reports and more —drawing on the rich R&D resources of technology providers to address market trends, challenges and solutions. Our live events and virtual seminars give you access to vendor neutral, expert commentary and advice on the issues and challenges you face daily. Our social community IT Knowledge Exchange allows you to share real world information in real time with peers and experts.

What makes TechTarget unique?

TechTarget is squarely focused on the enterprise IT space. Our team of editors and network of industry experts provide the richest, most relevant content to IT professionals and management. We leverage the immediacy of the Web, the networking and face-to-face opportunities of events and virtual events, and the ability to interact with peers—all to create compelling and actionable information for enterprise IT professionals across all industries and markets.

Related TechTarget Websites

SearchCloudSecurity

SearchSecurityChannel

SearchFinancialSecurity

SearchMidmarketSecurity