Security

Safeguard Sensitive Data Against Insider Threats and External Attacks

Top 5 Scenarios

ENTER »

« BACK | NEXT »

1

Preventing external attacks

2

Blocking privileged users from access to sensitive data

3

Identifying fraud at the application layer

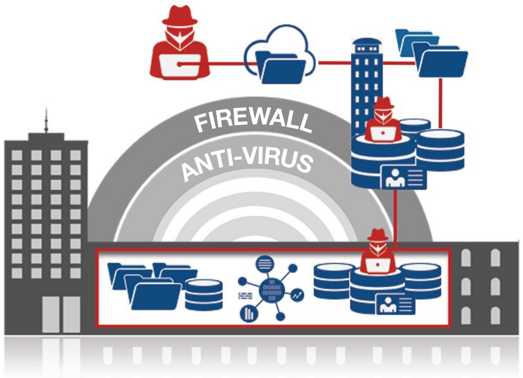
4

Ensuring authorized access

5

Detecting unauthorized changes

The Challenge: Complexity of Safeguarding Sensitive Data



70% of your company's value likely lies in intellectual property

Customer data, product designs, sales information, proprietary algorithms, communications, etc.

Source: TechRadar

Issue: To stop internal and external threats, security must be provided at the data level.
Are you doing enough to protect the data that's running your business?

Data security breaches are more common than ever—and more expensive. The year 2014 saw the term "data breach" become part of the broader public vernacular with *The New York Times* devoting more than 700 articles related to data breaches, versus fewer than 125 the previous year.[2] Global studies show that the average total cost of a data breach is now USD3.8 million. What's more, the loss of trade secrets, product designs or other intellectual property can spell financial ruin for an organization. Sensitive data is not only at the core of business interactions, it is also a highly attractive target for attack.

Securing sensitive data presents a multi-dimensional challenge where complex environments—which often include a wide range of heterogeneous database management systems (DBMS), enterprise applications, big data platforms, file systems, OS platforms with multiple access paths and permission levels—have created a seemingly endless array of security risks and violation scenarios.

Traditional perimeter "fortress approaches" such as firewalls and IDS/IPS systems are no longer sufficient to protect against 21st-century attackers who can easily bypass perimeter defenses … particularly as the boundary of what the perimeter is becomes increasingly blurred as Cloud environments become more and more popular. These security measures don't have the ability to differentiate between traffic that 'appears' legitimate and actually is legitimate. For example, if DBA credentials appear, the firewall will let that person through. But what if your data security layer is able to recognize abnormal behavior by this 'DBA and flat it, and maybe even quarantine the user for further investigation or block it entirely?

Whether you need to protect data in your databases or in other areas of your environment, you must be able to identify and stop not just external threats, but also internal threats as well. In fact, according to the latest IBM Security Services 2015 Cyber Security Intelligence Index, the insider threat continues to hold a top place in comparison to other attack types. While outsiders were found to be responsible for

Preventing external attacks

Blocking privileged users from access to sensitive data

Identifying fraud at the application layer

Ensuring authorized access

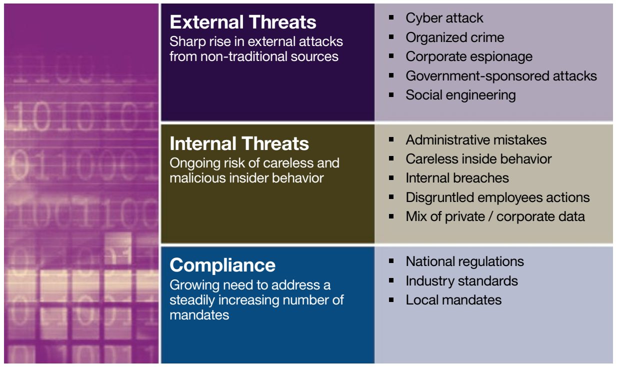
Detecting unauthorized changes

« BACK | NEXT »

45 percent of the attacks recorded in 2014, 55 percent of attacks were carried out by those who had insider access to organizations' systems.[1]

Insider threats are not only from malicious employees who want to do harm: Insider threats also come from compromised corporate IDs and credentials—for example, from a user that inadvertently clicks on a suspicious email attachment that exposes their system (and possibly the corporate network) to malware. Today, most spam is created by for-profit operators, who can attach any sort of malware to the spam message. From criminal intent to financial gain, any sort of adversary with the right motivation can hire a spam operator that will build a custom campaign to trick users to open an attachment or click on a link—thus, infecting the corporate network with ransomware or malware. From here, it's just a short step to coopt or corrupt user IDs.

Additionally, trusted third-party contractors also count as "insiders", as they have access and entitlements to systems and data that mirror those of direct employees.[1] This can include electricians, construction workers, phone or other type of repair personnel who come into physical locations or have access to networks. In the Target retail breach in the United States, abusing this type of third-party access demonstrated that attackers often steal credentials and gain access into networks.



External Threats Sharp rise in external attacks from non-traditional sources	▪ Cyber attack ▪ Organized crime ▪ Corporate espionage ▪ Government-sponsored attacks ▪ Social engineering
Internal Threats Ongoing risk of careless and malicious insider behavior	▪ Administrative mistakes ▪ Careless inside behavior ▪ Internal breaches ▪ Disgruntled employees actions ▪ Mix of private / corporate data
Compliance Growing need to address a steadily increasing number of mandates	▪ National regulations ▪ Industry standards ▪ Local mandates

Given the complexity of securing sensitive data against internal and external risks, data security is not a once-and-done event: It's an ongoing process that must be continuously managed, monitored, enhanced and audited across the entire organization. And, data security must be deployed as a process that integrates with other security processes (in particular, IAM and vulnerability management) as well as other critical business processes.

This eBook examines the top 5 scenarios and the essential best practices for preventing attacks and defending against insider threats. Organizations who adopt a proactive approach will require a broad and scalable solution that helps automate and centralize data security, so that it's able to help reduce compliance complexity while meeting key regulations such as SOX/COBIT, PCI DSS, data privacy laws, SCAP, FISMA and HITECH—and going beyond, to support comprehensive data security.

Top 5 scenarios

1. Preventing external attacks

2. Blocking privileged users from access to sensitive data

3. Identifying fraud at the application layer (Oracle EBS, PeopleSoft, SAP, etc.)

4. Ensuring authorized data access

5. Detecting unauthorized database changes

Preventing external attacks

Blocking privileged users from access to sensitive data

Identifying fraud at the application layer

Ensuring authorized access

Detecting unauthorized changes

« BACK | NEXT »

Preventing external attacks

Much of the world's sensitive data is stored in commercial databases/data warehouses such as Oracle, Microsoft SQL Server, IBM DB2, Informix, Sybase, MySQL, Netezza and Teradata—making databases a favorite target for cybercriminals. Web applications have transformed and improved the ways companies do business but have also left the database more exposed. The exposure exists because the application which uses the data now caters to a much wider audience and because users are not limited to inside employees. Attackers now have a direct pipe—through the application, past perimeter defenses—into the database.

Then you must stop and consider that close to 80 percent of data is unstructured—and a great portion of that data is sensitive data that's stored in file systems. That sensitive unstructured data may live in spreadsheets, word documents, PDF or CSV file formats—and the list goes on. Consequently, it's every bit as important to protect sensitive data in file systems as it is to protect sensitive data in database. And then you must go further, and consider the sensitive data that's also fueling big data analytics and that's being used in cloud deployments—and you quickly realized that all sensitive data needs to be secure, not just sensitive data in the database.

High-profile data breaches include external attacks by:

Cyber-criminals, highly motivated attackers associated with crime syndicates willing to pay hard cash for personal information. For example, healthcare data is highly sought after.

Cyber-espionage targeting intellectual property (IP) such as new product designs, algorithms, strategic plans, and information about strategic resources such as oil, energy, and infrastructure.

Hacktivism, a phenomenon targeting sites for political reasons.

Frequently, these attacks bypass traditional perimeter defenses by exploiting web application vulnerabilities such as SQL injection or by leveraging stolen administrative credentials to compromise backend databases.

Implementing a real-time data activity monitoring (DAM) and protection solution will help prevent outsider attacks such as SQL injection in several ways, all of which can be used simultaneously to provide a layered defense. This is accomplished by creating and enforcing real-time, proactive rules, for example:

- *Access rules* define conditions, which can be quite general or very granular, against which all accesses to the databases or files are compared an appropriate action taken—logging, alerting, blocking, etc.
- *Exception rules*, which are based on definable thresholds, such as an excessive number of failed logins or SQL errors. SQL errors can indicate that an attacker is "looking around" for names of key tables by experimenting with SQL commands using different arguments—such as "Credit_Card_Num" or "CC_Num"—until he finds a valid table name that does not result in a database error.
- *Exception rules*, which are based on specific SQL error codes from the database, such as "ORA-00903: Invalid table name" or "ORA-00942: Table or view does not exist." Such error codes may indicate hacking behavior.
- *Extrusion rules* that examine data leaving the database or file for specific data value patterns such as credit card numbers, or a high volume of returned records that might indicate a breach.

Blocking privileged users from access to sensitive data

In most IT environments, privileged users—such as DBAs, developers and outsourced personnel—can have unfettered access to sensitive data, with little or no monitoring controls around their activities. These “super users” can easily bypass application- or network-level security control points.

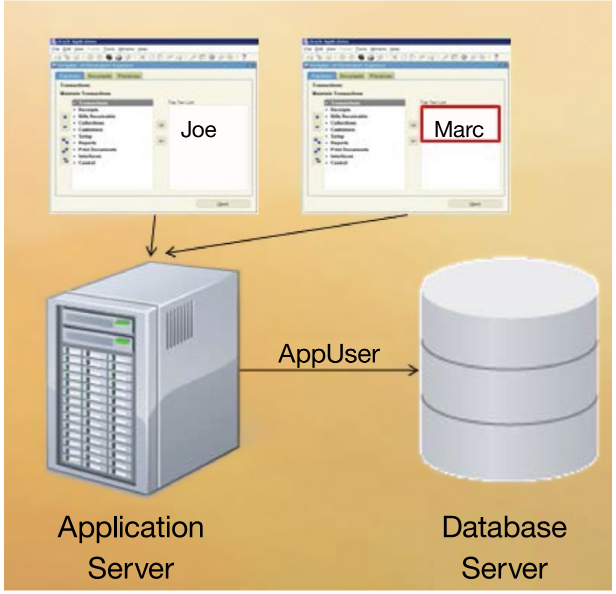
The 2015 Verizon report (cited above) found that, as ‘with prior years, the top action (55 percent of incidents) was privilege abuse—which is the defining characteristic of the internal actor breach. We see individuals abusing the access they have been entrusted with by their organization in virtually every industry’: And a recent example of this was the Korea Credit Bureau breach.

The Verizon report goes on, commenting that “it’s all about grabbing some easy Benjamins [money] for these mendacious malefactors, with financial gain and convenience being the primary motivators (40 percent of incidents), whether they plan to monetize stolen data by selling it to others (such as with financial data) or by directly competing with their former employer. Coming in a not-so-distant second is the motive of convenience (basically using an unapproved workaround to speed things up or make it easier for the end user), and while this is not something that is intended to harm the organization, it certainly often has the same result.”

Many organizations have formal data security policies that govern how and when privileged users access systems. However these organizations lack enforcement controls or granular visibility into what’s really going on, ultimately exposing vulnerabilities.

Role-based access and other built-in controls are designed to prevent *end-users* from accessing sensitive data in databases, but they cannot prevent DBAs or other *privileged users* who have the ability to execute any database command, on any database object, as part of their daily jobs.

For instance, application developers may be able to login to the database using the account that the Application itself uses (see figure 2), but without all of those sometimes annoying security measures built into the application. ***How do you know and handle someone misusing credentials?*** To make matters worse, accountability is difficult to achieve because privileged users often share the credentials used to access database systems.



Issue: The application server uses a generic service account (AppUser) to access the database – which doesn’t identify WHO (Joe or Marc) initiated transaction (connection pooling)

« BACK | NEXT »

Those organizations using native database logging find this approach to be impractical because it requires database changes that affect the performance and stability of business-critical applications such as ERP, CRM, and credit card processing systems. It also fails auditors' requirements for separation of duties, because database logging is not controlled by IT security personnel and can easily be circumvented by DBAs.

By deploying a combination of monitoring and blocking capabilities provided by a DAM solution, activity at the network layer and on the database server, file server, or big data platform, etc., can be controlled, preventing information leakage at the source as well as unauthorized changes to critical data. In addition, using real-time monitoring technology can empower the security team to immediately thwart unauthorized or suspicious access, even by privileged users, through:

- Monitoring all transactions to create a continuous, normalized, fine-grained audit trail that identifies the "who, what, when, where, and how" of each transaction. Implementing fine-grained access policies is required for key regulations such as Sarbanes-Oxley (SOX), the Payment Card Industry Data Security Standard (PCI-DSS), HIPAA/HITECH, FISMA/NIST 800-53 and state/local data privacy and protection laws.
- Continuously analyzing monitored data in real-time to identify unauthorized or suspicious activities, and executing responsive actions ranging from blocking the transaction in real-time, to dynamically masking sensitive data, to generating alerts for the security team.

Preventing external attacks

Blocking privileged users from access to sensitive data

Identifying fraud at the application layer

Ensuring authorized access

Detecting unauthorized changes

« BACK | NEXT »

Identifying fraud at the application layer

Multi-tier enterprise applications such as Oracle EBS, PeopleSoft, J.D. Edwards, SAP, Siebel, Business Intelligence, and in-house systems contain an organization's most sensitive financial, customer, employee, and intellectual property information. These systems are the most difficult to secure because they are highly distributed and designed to allow Web-based access from insiders and outsiders, such as customers, suppliers, and partners.

In addition, multi-tier enterprise applications mask the identity of end-users at the database transaction level using an optimization mechanism known as "connection pooling." Using pooled connections, the application aggregates all user traffic within a few database connections that are identified only by a generic service account name. Connection pooling makes it challenging to associate specific transactions with particular end-users. As a result, fraudulent transactions are difficult to trace.

Without application-layer monitoring, organizations are finding it challenging to associate specific database transactions with particular application end-users, leaving a huge gap in their ability to detect fraud (and other abuses of legitimate access) that occurs via enterprise applications. This level of monitoring is often required for data governance requirements such as SOX. New auditor guidance from the Public Company Accounting Oversight Board for Sarbanes-Oxley compliance has also increased the emphasis on anti-fraud controls.

With a real-time monitoring and auditing solution, organizations have the ability to capture both direct and indirect transactions. Capabilities are provided to ensure transactions executed via pooled connections include the associated application layer user IDs in the audit trail and other identifying information such as IP address, Domain login name etc. These capabilities, along with automated compliance workflow, ensure policy violations are easily traced, investigated and remediated.

Last of all the data associated with enterprise applications can also be accessed directly by privileged users via developer tools like SQL *Plus, bypassing controls within the application.

« BACK | NEXT »

Ensuring authorized access

Unsanctioned database access often goes undetected, exposing sensitive data and potentially causing billions of dollars in damage. A strong information security and privacy strategy should ensure that the organization can identify who has access to information, as well as ensure that access is required by the individual to execute their job.

Two primary supporting capabilities are Authentication and Authorization controls.

Authentication is the process of uniquely identifying a person or system. The most typical approach is using a user ID and a password. Authenticating users ensure full accountability per user, and manage privileges to limit access to data and must be enforced—even for the most privileged users.

Authorization is the mechanism to control what information and actions are available to a particular individual, and under which circumstances. Regulatory mandates and security requirements are requiring organizations to adopt strong, multifactor authentication methods to protect against unauthorized and unidentified access. For instance, DBAs should be authorized only to perform functions they need for their job. However a DBA doesn't need access to actual data.

Within data sources, there's so much complexity it's very difficult to understand who has access to what information through what roles and grants. Auditors typically require these entitlements to be reviewed periodically. So, one way to help meet this need is to automate the scanning of selected databases, file systems, cloud environments or other platforms for the collection, organization, distribution and review of user right data—including those granted through roles and group membership.

A purpose-built data or file activity solution includes reporting capabilities that automatically aggregate user entitlement information across the entire heterogeneous database infrastructure as well as identify which users have particular special privileges, what new rights have been granted by whom and what entitlements particular users have. Additionally, real-time activity monitoring can capture 'grants' and 'revokes' as they occur.

Preventing external attacks

Blocking privileged users from access to sensitive data

Identifying fraud at the application layer

Ensuring authorized access

Detecting unauthorized changes

« BACK | NEXT »

Detecting unauthorized changes

Detecting changes to databases or other data repositories is important from the perspective of placing controls around privileged users. However, it is also important from an external security perspective. These changes can be indicators that a repository has been compromised, since hackers often make changes while extracting data or embedding malware.

As a result, many organizations are looking to monitor their infrastructures using automated and centralized controls. This enables them to track all changes, even across complex, heterogeneous and distributed environments, including changes to:

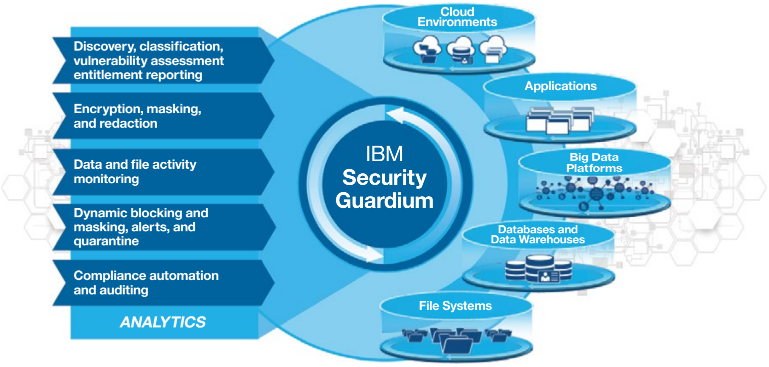
- **Database structures, data warehouses, Hadoop and NoSQL environments, and files.** such as tables, triggers, and stored procedures. For example, you can detect accidental deletions or insertions of critical tables that impact data governance and the quality of business decisions. You can also proactively identify malicious acts such as "logic bombs" planted by disgruntled employees.
- **Critical data values or resources** such as data that affects the integrity of financial transactions, or files that contain intellectual property.
- **Security and access control objects** such as users, roles, and permissions. For example, an outsourced contractor could create a new user account with access to critical databases and then delete the entire account, eliminating all traces of his activity.
- **Database configuration files** and other external objects that can affect your database security posture, such as environment/registry variables, configuration files (e.g., NAMES.ORA), shell scripts, OS files, and executables such as Java programs.

Automated, real-time controls implemented through a well-designed DAM solution can prevent unauthorized actions such as: executing queries on sensitive tables; changing sensitive data values; accessing critical files, adding or deleting critical tables (schema changes) outside change windows; and creating new user accounts and modifying privileges.

Another best practice is to use encryption to render sensitive data unreadable, so an attacker cannot gain unauthorized access to data from outside. This can be accomplished either by encrypting data and files "in place" via the operating system or dynamically, as the data is in flight.

« BACK | NEXT »

Conclusion


Discovery, classification, vulnerability assessment entitlement reporting
Encryption, masking, and redaction
Data and file activity monitoring
Dynamic blocking and masking, alerts, and quarantine
Compliance automation and auditing
ANALYTICS
IBM Security Guardium
Cloud Environments
Applications
Big Data Platforms
Databases and Data Warehouses
File Systems

Organizations must build strong security programs to defend and protect against new and emerging threats—such as SQL injection, cross-site scripting and privileged insider breaches, just to name a few—based on best practices for database security and compliance.

IBM® Security Guardium® is a comprehensive data security solution that helps clients secure the sensitive data at the heart of their business.

IBM Security Guardium is modular, so that you can deploy the capabilities you need, when you need them. In addition to providing functions such as sensitive data discovery, classification, entitlement reporting, vulnerability assessment, encryption, masking, etc., this highly scalable and integrated solution also enables continuous monitoring to keep track of access to enterprise databases, big data platforms, cloud environments and file systems to ensure comprehensive data protection and allow security teams to take action—such as alerting, blocking, and/or quarantining—when they see unusual behavior. Monitoring also can help simplify compliance audits by providing automated and centralized controls for heterogeneous environments.

IBM Security Guardium software helps organizations:

- Prevent data breaches, insider risk and fraud, and unauthorized changes to (or destruction of) sensitive data
- Monitor privileged users such as database administrators, developers, IT administrators, outsourced personnel, etc.
- Virtually eliminate the overhead and complexity of native DBMS, Big Data, and file system audit logs
- Automate compliance reporting, vulnerability and configuration assessments, and data discovery
- Encrypt files
- Mask confidential data in test, training and development systems
- Redact unstructured data in documents, forms and graphics at rest or dynamically

Preventing external attacks

Blocking privileged users from access to sensitive data

Identifying fraud at the application layer

Ensuring authorized access

Detecting unauthorized changes

For more information

To learn more about IBM Security Guardium please contact your IBM representative or IBM Business Partner, or visit the following website: **ibm.com**/guardium

Additionally, IBM Global Financing can help you acquire the IT solutions that your business needs in the most cost-effective and strategic way possible. For credit-qualified clients we can customize an IT financing solution to suit your business requirements, enable effective cash management, and improve your total cost of ownership. IBM Global Financing is your smartest choice to fund critical IT investments and propel your business forward. For more information, visit: **ibm.com**/financing

Statement of Good Security Practices: IT system security involves protecting systems and information through prevention, detection and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed, misappropriated or misused or can result in damage to or misuse of your systems, including for use in attacks on others. No IT system or product should be considered completely secure and no single product, service or security measure can be completely effective in preventing improper use or access. IBM systems, products and services are designed to be part of a lawful, comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective. IBM DOES NOT WARRANT THAT ANY SYSTEMS, PRODUCTS OR SERVICES ARE IMMUNE FROM, OR WILL MAKE YOUR ENTERPRISE IMMUNE FROM, THE MALICIOUS OR ILLEGAL CONDUCT OF ANY PARTY.

[1] http://www.ibm.com/common/ssi/cgi-bin/ssialias?subtype=WH&infotype=SA&htmlfid=WGL03076USEN&attachment=WGL03076USEN.PDF

[2] http://securityintelligence.com/cost-of-a-data-breach-2015/#.VbJ6p0bG9E0

Security
Route 100
Somers, NY 10589

Produced in the United States of America
November 2015

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary. It is the user’s responsibility to evaluate and verify the operation of any other products or programs with IBM products and programs.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

The client is responsible for ensuring compliance with laws and regulations applicable to it. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the client is in compliance with any law or regulation.

Statements regarding IBM’s future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

♻ Please Recycle