Publication Date: 11 May 2010

Data Obfuscation for Test Environments

By Randy Raymond

Introduction

Sixty percent (60%) of data breaches are initiated by insiders or are a result of organizational mismanagement, according to a recent report[1]. About one-third of data breaches are from hackers.

Accordingly, governments have been creating regulations for years imposing penalties on companies that have data breaches. Civil litigation further exposes companies to financial risk of not protecting sensitive information.

To curb these problems, businesses have focused on protecting the perimeter of systems to keep outsiders at bay. But what about protecting test environments. Hackers typically do not target test environments, but disgruntled insiders often do. Using production data in test environments provides hundreds or even thousands of employees and contractors working onsite, offsite, and offshore with access to your sensitive production data.

Sensitive data in non-production environments is rarely held to the same security standards as production data. The inability to protect sensitive data in test environments keeps many companies from leveraging the cost savings of outsourcing and offshoring.

Most of the high-profile and notorious data thefts reported in the news and trade publications are breaches of production systems. Outsiders hack their way into a production system, storage media is lost or stolen (or in one high-profile case tapes literally fell off a truck), laptops are lost or stolen, or other lapses occur in corporate security for production systems. While enterprise data protection is critical, organizations must pay heed to the alarming number of internal security threats such as those occurring during development and testing of applications.

This whitepaper focuses exclusively on protecting sensitive data in test environments.

Government and Industry Regulations

Government Oversight

Many countries have laws and/or regulations to protect the private information of citizens. .

The US Gramm-Leach-Bliley Act[2] requires the protection of personal consumer information, the Health Insurance Portability and Accountability Act (HIPAA) of 1996[3] has requirements to protect personal information, and the Right to Financial Privacy Act of 1978[4] requires protection of financial records. Several US states have enacted additional regulations above and beyond these federal measures. For example, California's SB 1386[7] requires companies to notify customers of data breaches. This model legislation is being pursued in several other states.

European nations are also taking a stand against data theft European Union Directive 95/46/EC[5] outlines strict guidelines on protecting individual private data and describes the responsibilities of data holders to shield from misuse.
The United Kingdom Data Protection Act of 1998[6] extends the EU directive while placing greater legal obligations on those storing personal, private or sensitive data.

Industry Oversight

Industry oversights complement and extend the effectiveness of government regulations. The Payment Card Industry Data Security Standard (PCI DSS), for example, is a comprehensive standard to help organizations proactively protect customer account data[8]. Compliance is not mandatory but companies accepting credit cards or processing credit card transactions must comply in order to do business with the five major credit card payment networks American Express, Master Card Worldwide, Visa, Inc., Discover Financial Services, and Japan Credit Bureau (JCB)[9].

Compliance is audited by the credit card networks and includes an inventory of test facilities[9]. Business will need to identify all points where credit card data can enter or leave the company, and examine data reports, log files, servers, e-mail, and file transfers[9] all of which take place in test environments. Failing to comply can result in fines being issued by the credit card networks[9].

Sensitive Data in Test Environments?

Should sensitive data be used in test environments? The answer is a resounding no. Test data pulled directly from production is still sensitive although it's no longer in a live situation.

Test environments are never as secure as production, and there is little legitimate business reason testers should have access to sensitive data. Nondisclosure agreements with employees and contractors do not protect the data. They just make it easier to litigate to recover costs when data is breached. The ultimate objective is to not have a data breach in the first place.

Testers also need expanded access rights to perform a variety of testing activities, including:

- Creating test data for specific test scenarios
- Destroying data during testing
- Investigating data problems or application defects
- Having stabilized and reliable data for test automation activities both in development and execution

Using sensitive data in test environments also prevents organizations from gaining the cost-savings benefits of utilizing outsourced or offshore labor.

Obfuscation Techniques

Obfuscation generally refers to the various techniques used to hide sensitive data. The term "scrubbing" may be used interchangeably with obfuscation. The techniques listed here have two main goals: Protect sensitive data from disclosure and create usable test data with the same data shape as the original. It is ideal to use more than one technique to bring added protection.

Masking

Data masking replaces sensitive characters or fields with a meaningless character such as "X." Masking preserves the data shape for display on screens and reports. Everyday examples of masking include using Xs on all but the last 4 digits of a credit card number when printing a receipt.

Substitution

Substitution replaces fields of data with similar content that is unrelated to the original data. An example of substitution is to replace the actual first and last names with names randomly picked from a large list of valid first and last names that has been created specifically for use in substitution. Substitution preserves the original data shape while hiding the actual sensitive information.

Shuffling records

Shuffling and substitution are similar except that shuffling uses the source data itself instead of an external list. Shuffling moves data between rows so that the data shape is preserved but the original details of the sensitive information are hidden.

Number and date variance

Variance modifies number or date information by replacing the field with similar information that is a random percentage of the original. The percent variance is chosen to keep the new data within valid ranges for the field and the field's use. Variance keeps the data shape while hiding the original sensitive information.

Gibberish generation

Gibberish generation is needed when the sensitive data you wish to hide has associated data such as correspondence that can identify the original data. An everyday example would be bank records. You can obfuscate the account information of customers in the database tables but the records are linked to images or other facsimiles (.pdf files) of the monthly statements sent to those customers. Those stored statements contain all of the information you wish to hide. To prevent this sensitive information from being revealed, gibberish generation replaces the confidential data with random "junk" data files of equivalent size.

Encryption

Encryption keeps the original data in place and available to anyone with the decryption key. This is not very desirable since the data will be rendered data unusable for development and testing purposes.

Data Generation

Data generation creates fictitious or mock data from scratch or other sources that is usable for testing purposes.

There are many more obfuscation techniques not listed here and the approach for each of these listed can range from very simple to highly mathematically.

Data Obfuscation Plan

To secure data in development and test environments it is critical to create a data obfuscation plan (DOP) that governs how production data is obfuscated for use in testing or how mock data is created. For very large relational databases this exercise will not be simple or short but it by far preferable to facing security breaches, regulatory fines, or and civil litigation.

The data obfuscation plan should be different than your test data management Plan (TDMP). The TDMP describes how you plan to manage the data within the test environment to maintain the validity of your tests and keep different testing teams from "stomping" on one another and invalidating each other's tests. Responsibility for the TDMP belongs to the testing organization.

The DOP is a security plan for protecting sensitive information while providing testing with large volumes of valid test data. It should be an integral part of the organization's overall data security plan managed by the chief compliance officer or risk management officer.

Simply stated, the DOP governs scrubbing and creating the data while the TDMP governs the use of the data in the test environment.

The DOP should have the following general goals:

- Prevent actions from being reversed. Those who access the data should never be able to reconstitute the original sensitive data
- Maintain the same shape as the source data. "Shape" can include human readability, geographic distribution for reporting purposes, mock or masked data the same field length as the original, etc.
- Maintain referential integrity. If you don't maintain relationships within the database then your test data will not be of much use.
- Mask non-sensitive data that can be used to recreate sensitive data. Some data may not be sensitive in itself, but combined with other non-sensitive data sensitive data may be reverse engineered. Non-sensitive data may also tie back to sensitive data. Such an example would be in bank statements. Account owner information could be scrubbed but the record ties back to PDF copies of monthly statements stored in the database.
- Ensure measures are repeatable. Single-use transformation is a throw-away exercise. To be useful, test data must represent the constantly changing data in your production environment. Additionally, eventually your testing team may use all of the data in your test database and you will need to refresh this from the production environment.

Industry Tools for Implementation

A wide variety of tools have been created to allow companies to protect sensitive test data. The biggest players in this tools space are Compuware, Oracle, and IBM.

Compuware File-AID Enterprise-wide Data Management

The File-AID family of tools allows you to manage test data in both the mainframe and distributed (client server) environments. The following File-AID products are available:

- File-AID/CS
- File-AID/MVS
- File-AID/Data Solutions
- File-AID/RDX
- File-AID for DB2
- DBA-XPERT for DB2

The File-AID family can perform many obfuscation techniques, including:

- Masking
- Substitution
- Number and date variance
- Encryption
- Data generation

See http://www.compuware.com/products/fileaid/default.htm for additional details on File-AID from Compuware.

Oracle® Data Masking Pack

Oracle’s Data Masking Pack uses masking rules to provide production-like yet scrubbed data for testing where the test organization requires accurate and realistic data for application testing. Regulatory compliance is enabled via consistent rules-based masking formats across enterprise-wide databases. Search “Data Masking Pack” at www.oracle.com for additional information.

IBM Data Masking Solution

The IBM Data Masking Solution protects sensitive data without disrupting the testing process. IBM’s solution prevents sensitive information from being exposed while allowing internal and external (both onshore and offshore) testers to perform software product testing and quality assurance. Search “Data Masking Solution” at www.ibm.com for greater details.

Conclusion

Protecting sensitive data is serious business as evidenced by laws that have been enacted around the world and industries taking self-regulatory action to enforce data security. Securing sensitive data also allows companies to take advantage of the worldwide IT labor industry, utilizing reduced costs overseas testing without exposing or losing sensitive data.

This white paper has discussed data obfuscation for the testing organization. All of the techniques described here can and should be used for development, training, analytical teams, and anyone needing access to non-production data.

References

[1] A Case of Mistaken Identity? News accounts of hacker, consumer, and organizational responsibility for compromised digital records, *Journal of Computer-Mediated Communication*, 12(4), article 5, Erickson, K. and Howard P., 2007.
http://jcmc.indiana.edu/vol12/issue4/erikson.html

[2] Privacy Initiatives, Financial Privacy, Federal Trade Commission
http://www.ftc.gov/privacy/privacyinitiatives/glbact.html.

[3] *Fact Sheet Administrative Simplification under HIPAA: National Standards for Transactions, Privacy and Security*, United States Department of Health and Human Services
http://www.hhs.gov/news/press/2002pres/hipaa.html

[4] Federal Deposit Insurance Corporation, *Financial Institutions Regulatory and Interest Rate Control Act Of 1978 cited as the "Right to Financial Privacy Act of 1978"*
http://www.fdic.gov/regulations/laws/rules/6500-2550.html.

[5] Data Protection in the European Union, European Commission, Justice and Home affairs, Data Protection Guide http://ec.europa.eu/justice_home/fsj/privacy/guide/index_en.htm.

[6] The Data Protection Act 1998, Schedule 1, Part 1, Section 8http://www.hmso.gov.uk/acts/acts1998/19980029.htm.

[7] *California Raises the Bar on Data Security and Privacy,* Brelsford. J., September 2003http://library.findlaw.com/2003/Sep/30/133060.html

[8] About the PCI Data Security Standard (PCI DSS), PCI Security Standards Council,https://www.pcisecuritystandards.org/tech/index.htm.

[9] Mastering the Payment Card Industry Standard, *Journal of Accountancy Online*, January 2008, http://www.aicpa.org/PUBS/JOFA/jan2008/payment_card.htm