SECURITY

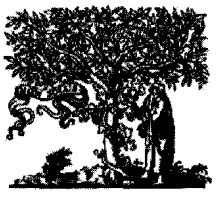
Ron Ben Natan

Implementing Database Security and Auditing

Includes Examples For:
ORACLE
SQL SERVER
DB2 UDB
SYBASE

Implementing Database Security and Auditing

A guide for DBAs, information security administrators and auditors



ELSEVIER
DIGITAL
PRESS

Amsterdam • Boston • Heidelberg • London • New York • Oxford
Paris • San Diego • San Francisco • Singapore • Sydney • Tokyo

Elsevier Digital Press
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA
Linacre House, Jordan Hill, Oxford OX2 8DP, UK

Library of Congress Cataloging-in-Publication Data
Application submitted.

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ISBN: 1-55558-334-2

For information on all Elsevier Digital Press publications
visit our Web site at www.books.elsevier.com

Printed in the United States of America
05 06 07 08 09 10 10 9 8 7 6 5 4 3 2 1

Preface

This book is a guide on implementing security and auditing for database environments. It is meant to be used by database administrators, security administrators, system administrators, auditors, and operational owners—anyone who manages or oversees the database environment, data/database security, or the process by which database security and database audits are accomplished.

The book shows you how to secure and audit database environments which include the major relational products: environments, which include the major relational database products: Oracle, Microsoft SQL Server, IBM DB2, Sybase, and even a bit of MySQL. It is useful if you have a single database product and is even more useful if you need to secure and/or audit heterogeneous environments that include more than one database version. The methods you will learn apply to all modern relational database environments.

This book is meant to show you *methods* and *techniques* that will help you elevate the security of your database infrastructure. Each chapter in the book focuses on a certain area of database administration and usage and shows you what you need to do in that domain, as well as how to do it. Because educated administrators are sure to be more effective than those that follow checklists with a limited understanding of what each item does and why, each chapter details anatomies of vulnerabilities in addition to the remedies. By understanding how attackers may try to compromise the database, you will be better able to invest your limited resources where they count most. You may even be able to address issues that are not mentioned in this book and that may not even be known at this point in time.

I mentioned that the aim of this book is to make your database environment more secure and that the focus is often both administration and usage. Many database vulnerabilities and security issues are caused by misconfigurations and inappropriate usage of the database by application serv-

ers and other clients (or even other databases in replicated and other distributed environments). In addressing this topic, many of the chapters take a broader look of database security and show you how to resolve problems by improving the way the database interacts with applications and with other elements in the infrastructure. Without understanding these techniques, you may invest a lot of time in securing "your island," only to learn that you have a gaping hole—one that you could have easily addressed if you weren't too busy investing in perfecting your corner of the world. The book is therefore not only meant to be a practical guide, but it also means to be an *effective* guide and address real-world problems.

This book is not a checklist. Detailed instructions are included in almost all chapters, but the book is not a reference text for each of the database products. I will include pointers to relevant checklists and reference texts and instead focus on ensuring that you invest your time wisely. Security is a never-ending battle against would-be attackers, and if you don't pick your fights wisely, you can lose to attrition. Auditing is another area that can easily overwhelm you in terms of work. Therefore, I will try to highlight the most important areas in which you should invest your time, show you what to do, and how to do it.

I mentioned that each chapter addresses a certain area—or category of techniques. This means that in most cases you can read the book sequentially or skip directly to a particular chapter when you are starting an initiative that has a specific focus. As an example, if you plan to start an initiative focused on database encryption, you should read Chapter 10; if you are concerned with database links, synonyms, nicknames, or replication, skip to Chapter 8; and if you are concerned with Web application access to your database, you can start with Chapter 5. The chapters that discuss auditing (Chapters 11 through 13) are a bit different. Rather than discussing categories of *techniques* as do Chapters 3 through 10, each chapter on the topic of auditing focuses on database auditing from a different *perspective*: Chapter 11 from the perspective of mapping of business requirements or regulations to actionable audit tasks, Chapter 12 from a content perspective, and Chapter 13 from an architectural perspective. Chapters 1 and 2 are introductory chapters. Chapter 1 details some starting points you should always have in place, and Chapter 2 gives you a brief overview of enterprise security and domains from which you can get many implementation ideas.

Finally, I'd like to thank the many people who have helped me understand, prioritize, implement, and navigate the complex topic of database security and audit, including George Baklarz, Moshe Barr, Roy Barr, Rodrigo Bisbal, Heather Brightman, Nir Carmel, Mike Castricone,

Stephen Chaung, Curt Cotner, Peggy Fieland, Gilad Finkelstein, Bobbi Fox, Guss Frasier, Guy Galil, Jerrilyn Glanville, Richard Gornitsky, Yaffi Gruzman, Evan Hochstein, Memy Ish-Shalom, Nate Kalowski, Dario Kramer, Kai Lee, Mike Lee-Lun, Michael MacDonald, Art Manwelyan, Jack Martin, Charles McClain, Ram Metser, Ola Meyer, Bruce Moulton, Gary Narayanan, Alex Narinski, Fred Palmer, Themis Papageorge, Jason Patti, Jennifer Peng, Daniel Perlov, Bob Picciano, Harold Piskiel, Jonathan Prial, James Ransome, Leonid Rodniansky, Elliott Rosenblatt, Mojgan Sanayei, Ury Segal, Pat Selinger, Nati Shapira, Mark Shay, Izar Tarandach, David Valovcin, Holly Van Der Linden, and John Young. I would also like to thank Tim Donar, Alan Rose, Theron Shreve, and Stan Wakefield for making this book fun to write.

Contents

Getting Started

This book is about database security and auditing. By reading it you will learn many methods and techniques that will be helpful in securing, monitoring, and auditing database environments. The book covers diverse topics that include all aspects of database security and auditing, including network security for databases, authentication and authorization issues, links and replication, database Trojans, and more. You will also learn of vulnerabilities and attacks that exist within various database environments or that have been used to attack databases (and that have since been fixed). These will often be explained to an "internals" level. Many sections outline the "anatomy of an attack" before delving into the details of how to combat such an attack. Equally important, you will learn about the database auditing landscape—both from a business and regulatory requirements perspective as well as from a technical implementation perspective.

This book is written in a way that will be useful to you—the database administrator and/or security administrator—regardless of the precise database vendor (or vendors) that you are using within your organization. This is not to say that the book is theoretical. It is a practical handbook that describes issues you should address when implementing database security and auditing. As such, it has many examples that pertain to Oracle, SQL Server, DB2, Sybase, and sometimes even MySQL. However, because detailing every single example for every database platform would have meant a 2,000-page book, many of the examples are given for a single database or a couple of them. The good news is that all techniques (or almost all of them) are relevant to all database platforms, and I urge you to read through all sections even if the example code snippets are taken from a database environment that you are not running. In all of these cases, it will be easy for you to identify the equivalent setting or procedure within your own environment.

More important, many of the techniques you will see in this book will never be described in a manual or a book that is devoted to a certain database product. As you'll learn throughout this book, good database security cannot always be implemented solely within the database, and many of the most serious security issues that you may face as the database owner (or the server owner) have to do with the way applications use a database and the way various interacting systems are configured. Addressing these complex issues must take into account more than just the database, and focusing on capabilities that are provided only by the database vendor is not always enough.

At this point you may be asking yourself a few questions:

- Doesn't the database have many security and auditing features? Isn't a database merely a file system with a set of value-added services such as transaction management and *security*? Isn't my database secure?
- Why now? The database has been part of the IT environment for many years (relational databases for at least 20 years); why should we suddenly be overly concerned with security and auditing?

The answer to the first set of questions is that while such features exist, they are not always used and are not always used correctly. Security issues are often a matter of misconfiguration, and the fact that the database implements a rich security model does not mean that it is being used or that it is being used correctly. If you are like 90% of database administrators or security administrators, you are probably aware that your database has big gaping holes—disasters waiting to happen. In fact, here are some examples that made the headlines (and rest assured that for every incident that makes headlines there are 100 that are kept quiet):

- In early 2000, the online music retailer CD Universe was compromised by a hacker known as "Maxus." The hacker stole credit card numbers from the retailer's database and tried to extort money from the retailer. When his demands were refused, he posted thousands of customers' credit card details to the Internet. (Go to http://databases.about.com/gi/dynamic/offsite.htm?site=http://www.pc%2Dradio.com/maxus.htm to see what Maxus' Web site looked like.)

- In December 2000, the online retailer Egghead.com announced that its customer database may have been compromised and warned that more than 3.5 million credit card numbers may have been stolen. Egghead.com later announced that the credit cards were not compromised but the investigation cost millions and few customers were willing to continue to do business with the retailer. The company went out of business shortly thereafter.
- In 2001, Bibliofind, a division of Amazon.com that specialized in rare and out-of-print books, was attacked and details for almost 100,000 credit cards were stolen. Even worse, the attackers maintained free access to the database for four months before being discovered! As a result, Bibliofind stopped offering buy/sell services and ended up as a matching service only (i.e., had to forgo a large portion of its revenues).
- In March 2001, the FBI reported that almost 50 bank and retail Web sites were attacked and compromised by Russian and Ukrainian hackers.
- In November 2001, Playboy.com was attacked and credit card information was stolen. In fact, the hackers sent e-mails to customers that displayed the credit card information.
- In the course of 2001, Indiana University was successfully attacked twice and private information, such as social security numbers and addresses, was stolen.
- A study conducted by Evans Data (a market research firm) in 2002 sampled 750 companies and reported that 10% of databases had a security incident in 2001! More than 40% of banking and financial services companies reported "incidents of unauthorized access and data corruption" and 18% of medical/healthcare firms reported similar types of incidents.
- In Oct. 2004 a hacker compromised a database containing sensitive information on more than 1.4 million California residents. The breach occurred on Aug 1 but was not detected until the end of the month. The database in question contained the names, addresses, Social Security numbers, and dates of birth of caregivers and care recipients participating in California's In-Home Supportive Services (IHSS) program since 2001. The data was being used in a UC Berkeley study of the effect of wages on in-home care and was obtained with authorization from the California Department of Social Services. The hacker had reportedly taken advantage of an unpatched system and

while officials declined to state which vendor's database was the subject of the attack they did report that it was a "commercially available product with a known vulnerability that was exploited."

- In Jan 2005 the following was reported by Security Focus (http://www.securityfocus.com/news/10271):

> A sophisticated computer hacker had access to servers at wireless giant T-Mobile for at least a year, which he used to monitor U.S. Secret Service e-mail, obtain customers' passwords and Social Security numbers, and download candid photos taken by Sidekick users, including Hollywood celebrities, SecurityFocus has learned... by late July [of 2004] the company had confirmed that the offer was genuine: a hacker had indeed breached their customer database

The answer to the second set of questions—why now?—is a convergence of several factors—almost a "perfect storm." True, the database has been around for a long time, but the following trends are dominating the last few years:

- E-commerce and e-business
- New and wonderful ways to use databases
- Increased awareness among the hacker community
- Widespread regulations that pertain to IT and to security

E-commerce and e-business have changed the way we live. We buy from online retailers, we pay our utility bills using online banking sites, and more. Businesses have optimized their supply chains and use Customer Relationship Management (CRM) software to manage relationships with their clients. In doing so, systems have become much "closer" to each other and much "closer" to the end users. Sure, we use firewalls to secure our networks and we don't connect databases directly to the Internet, but you'll see in Chapter 5 that there is more than one way to skin a cat and that databases are far more exposed than they used to be. Ten years ago the database was accessed by applications that were only available to internal employees. Now it is (indirectly through the application) accessed by anyone who has access to the Web site (i.e., everyone in the world).