Security Software Development

Assessing and Managing Security Risks

Douglas A. Ashbaugh

CRC Press
Taylor & Francis Group

SECURITY SOFTWARE DEVELOPMENT

Assessing and Managing Security Risks

DOUGLAS A. ASHBAUGH, CISSP, CISA

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

AN AUERBACH BOOK

Auerbach Publications
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Auerbach is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works
Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number-13: 978-1-4200-6380-6 (Hardcover)

Library of Congress Cataloging-in-Publication Data

Ashbaugh, Douglas A.
Security software development : assessing and managing security risks / Douglas A. Ashbaugh. -- 1st ed.
p. cm.
Includes bibliographical references and index.
ISBN 978-1-4200-6380-6 (alk. paper)
1. Computer security. 2. Application software--Development. 3. Computer networks--Security measures. I. Title.

QA76.9.A25A8246 2008
005.8--dc22 2008015213

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the Auerbach Web site at
http://www.auerbach-publications.com

SECURITY SOFTWARE DEVELOPMENT

Assessing and Managing Security Risks

Preface

Application security is a relatively new, yet very exciting field. It is being driven by a number of open source, government, regulatory, and industry organizations, but the need for application security is, sadly enough, the fact that software continues to be developed that isn't secure. For example, buffer overflows continue to plague software development despite the fact that buffer overflows and the methods for preventing them have been known for more than 20 years. The author believes that the primary reasons that secure software hasn't been developed lies with two factors:

- First, software development teams have not been sufficiently trained in how to identify vulnerabilities associated with their software development projects.
- Second, software development teams falsely believe that if perimeter security controls are in place, then the software they develop will also be secure, or at least will not affect the perimeter security.

The author was one of those developers who believed that as long as perimeter security (i.e., firewalls, intrusion detection and prevention, anti-virus, etc.) was in place, flaws in his code could not possibly affect the security of that perimeter. That may have been the case when applications were primarily mainframe- or client/server-based. However, the paradigm shifted with the introduction of Web-based applications, as the author painfully discovered.

Traditional firewalls must let Web-based traffic through the perimeter in order for Web-based applications to function. Therefore any attacker who can exploit flaws in the code of a Web application is already within the perimeter! There are additional controls that may be added to secure this perimeter including application and database firewalls, but many organizations have not yet recognized the need for such controls, as headlines sadly continue to point out. When you couple this with the fact that organizations are often slow to adopt new security controls because security is often seen as another expense, it becomes even more imperative for software development teams to understand the vulnerabilities associated with their software development efforts.

The author believes that education is truly the key. Software development teams, including project managers, technical analysts, business analysts, business managers, developers, quality assurance analysts, and testers must all be aware of the vulnerabilities that could plague any software development effort. However, with more than 3400 new vulnerabilities discovered in the first half of 2007 alone, this becomes an almost impossible task.

That is why the author believes in the process of assessing risks within the software development process. Through techniques such as threat modeling, software development teams can quickly begin to learn how to measure the risk associated with their software development projects. Once potential risks are understood management can at least make informed decisions on how to deal with those risks.

It is the sincere hope of the author that you can improve the security of the applications that you develop by following the techniques outlined in this book.

Acknowledgments

The author would like to thank R. J. Droll and Sue Horsman for their assistance in pulling all of the desperate ramblings of the author together to meet the submission deadlines for this manuscript. There is no way I could have done it without your assistance. You're the best!

The author also wishes to thank Jim Bridges, president of Software Engineering Services for taking a chance on a fellow Air Force veteran. You provided me with the chance to prove myself and my abilities at a time in my life when few others would. I truly appreciate all of the faith that you have placed in me and my abilities to get the job done right for you and SES. Thanks Jim!

The author would also like to thank his step-children, Stephanie Bennett and Brody Bennett, for putting up with his surly attitude and closed doors when attempting to put some serious thought into this volume!

Finally, the author would like to thank his wonderful wife Debi for all of the love, support, chiding, and comfort provided during the writing of this book. My love, I could not have done this without your love, patience, understanding—and most of all prodding—to finish writing the book.

And the author also wishes to thank almighty God for providing him with the wisdom and knowledge required to produce this work.

Author Biography

Douglas A. Ashbaugh is a Certified Information Systems Security Professional (CISSP) and member of the International Information Systems Security Certification Consortium (ISC^2), as well as a Certified Information Systems Auditor (CISA) and member of the Information Systems Audit and Control Association (ISACA). Mr. Ashbaugh is also the manager of information assurance for Software Engineering Services (SES) where he leads a team of dedicated information security analysts in providing security strategy and solutions, evaluation and assessment services, application security services, and security remediation services to corporate as well as various federal, state, and municipal government clients.

A dedicated information security professional, Mr. Ashbaugh has extensive experience in project management, application development, and information security. His 18+ years of information systems experience in both government and commercial environments provides a solid foundation to achieve outstanding results. He has a Bachelor of Science in engineering operations from Iowa State University. He served eight years in the U.S. Air Force as an acquisition project officer performing project management duties on a number of different development projects ranging in size from $50,000 to $3 billion. He has also worked as a software developer/analyst for the financial services industry for a period of more than six years. For the past five years, Mr. Ashbaugh has been providing information security services to a number of clients for SES. SES provides leading-edge IT solutions to DoD, government, state agencies, and the private sector. Mr. Ashbaugh may be reached through the Iowa branch office of Software Engineering Services.

Mr. Ashbaugh is married to a wonderful woman named Debi and lives in the great Midwest with her and a menagerie consisting of two yellow Labrador retrievers, Sam and Barbara Jean, a lop-eared rabbit named Fast-Girl, a red-eared slider (turtle) named Moses, and the head of the household, a black and white tabby named Sassy Marie.

Contents

Chapter 1

Current Trends in Application Security

Information is among the most important assets in any organization. Organizations are constantly building more complex applications to help them accomplish their mission; they are entrusting their sensitive information assets to those applications. But are those information assets secure as they are transmitted, modified, stored, and displayed by those applications? One only has to look at today's headlines to realize that information stored by organizations is not as secure as it could, or should, be.

1.1 Recent Data Security Breaches

Let's look at some of the recent data security breaches in the news today:

- July 27th, 2007: City Harvest, New York. Improper access to systems that contained donor credit card information resulted in the improper exposure of approximately 12,000 records.
- July 26th, 2007: Names and Social Security numbers of 10,554 U.S. Marines were found through the Google Internet search engine.
- July 25th, 2007: The private medical information, including Social Security numbers and treatment details of 25 people who sought medical assistance from the county was posted on the Hidalgo County, Texas Web site.

- July 24th, 2007: A security lapse compromised names, addresses, and Social Security numbers of more than 51,000 employees and patients of St. Vincent's Hospital in Indianapolis, Indiana.
- July 23rd, 2007: A security hole on a Fox News Web server exposed sensitive content to the public, including log-in information that allowed hackers to access names, phone numbers, and e-mail addresses of at least 1.5 million people.
- July 21st, 2007: University of Michigan databases were hacked. More than 5500 names, addresses, Social Security numbers, birth dates, and in some cases, the school districts where former students were teaching were exposed.
- July 20th, 2007: A Pentagon contractor may have compromised personal information, such as names, addresses, birth dates, Social Security numbers, and health information about 580,000 military personnel and their relatives because it did not encrypt data transmitted online.

These incidents represent just one week's worth of recent incidents reported by the Privacy Rights Clearinghouse (PRC) (http://www.privacyrights.org/) with a grand total of 2,159,079 records that were potentially compromised. The PRC has reported a total of 218,621,856 compromised records since the beginning of 2005. In reality, the number is probably much larger, because for many of the breaches listed by the PRC, the number of records actually compromised is unknown, and there are many data breaches that go unreported.

By conservative estimates at least 230 million records held by private companies, private and public organizations, universities, state and local governments, and the federal government have been compromised over the past three years. How were these records compromised? What security controls have failed to protect this valuable resource called information? The answer is that many different controls have failed. Some breaches are caused by the simple loss or theft of media containing confidential information. Theft (or misplacement) of laptops, hard drives, flash drives, backup tapes, and CD/DVD ROM account for many of the data breaches. Still others are caused by operator error, improperly configured or protected systems, improperly or poorly trained people, and transmission of information in the clear or just plain ignorance. Finally, many of these record losses can be attributed at least indirectly to poor, inconsistent, or nonexistent application security.

What is application security? Inasmuch as this book is about secure software development—which means that it is really all about application security—a definition is in order.