Steffen Schneckenburger

Uwe Dittmann, Alfred Schätter (Eds.)

Optimization of Web Application Security

Analysis of Common Threats, Countermeasures and Impact on the Software Development Lifecycle

Steinbeis-Transferzentrum Marketing, Logistik und Unternehmensführung an der Hochschule Pforzheim

Steinbeis-Edition

Steffen Schneckenburger

Uwe Dittmann, Alfred Schätter (Eds.)

Optimization of Web Application Security

Analysis of Common Threats, Countermeasures and Impact on the Software Development Lifecycle

Steinbeis-Transferzentrum Marketing, Logistik und Unternehmensführung an der Hochschule Pforzheim

Imprint

Steffen Schneckenburger | Uwe Dittmann, Alfred Schätter (Eds.)
Optimization of Web Application Security
Analysis of Common Threats, Countermeasures and Impact on the Software Development Lifecycle

1st edition 2011 | Steinbeis-Edition, Stuttgart
ISBN 978-3-941417-69-4

Layout: Steinbeis-Edition
Cover: ©iStockphoto.com/Baris Simsek
Production: Frick Werbeagentur / Frick Digitaldruck, Krumbach

Steinbeis is an international service provider in knowledge and technology transfer. The Steinbeis Transfer Network is made up of about 800 Steinbeis Enterprises and project partners in 50 countries. Specialized in chosen areas, Steinbeis Enterprises' portfolio of services covers consulting; research and development; training and employee development as well as evaluation and expert reports for every sector of technology and management. Steinbeis Enterprises are frequently attached to research establishments, universities, universities of applied sciences and universities of cooperative education.

Founded in 1971, the Steinbeis-Stiftung is the umbrella organization of the Steinbeis Transfer Network. It is headquartered in Stuttgart, Germany. Steinbeis-Edition publishes selected works mirroring the scope of the Steinbeis Network expertise.

146956-2011-06 | www.steinbeis-edition.de

Foreword

The intention of the book is all about creating awareness in terms of web application security and to support the reader with several examples as well as best practices through the development of secure web applications. Software developers and their customers often do not realize the importance of these requirements within a contract or at least define them superficially. For this reason the objective of the book is to develop an annex comprising common threats and countermeasures as well as necessary adjustments of the software development lifecycle in terms of security to establish a common basis of security understanding between developers, managers, customers and other stakeholders. As a result this book is directed to anyone from developer to decision-maker who wants to get an overview of current web application security flaws and corresponding countermeasures.

The book introduces current web application security threats and elaborates countermeasures in order to avoid or at least to reduce the impact of these flaws. In addition the security software development lifecycle of Microsoft is evaluated in order to avoid flaws in the first place.

Several critical web application vulnerabilities are identified based on intensive research. They were individually ranked according to the related risks. The top five risks elaborated are the following:

- Social Engineering
- (Blind) SQL Injection
- Brute Force
- Insecure Direct Object Reference
- Security Misconfiguration

Management summary

This book introduces current web application security threats based on literature research. Countermeasures are elaborated in order to avoid or at least to reduce the impact of these flaws. In addition the security software development lifecycle of Microsoft is evaluated in order to avoid flaws in the first place. The intention of this book is all about creating awareness in terms of web application security and to support the reader with several examples as well as best practices through the development of secure web applications.

Table of Contents

List of Abbreviations 9

List of Illustrations 11

1 Introduction 15
1.1 Problem Statement 15
1.2 Objectives 16
1.3 Topic Outline 17
1.4 Structure of the Work 18

2 Theoretical Foundations 19
2.1 Terminology Delimitation 19
2.2 Classification of Vulnerabilities 21
2.3 Architectural Overview 24
2.3.1 Client-Server Architecture 24
2.3.2 Communication Principles 27
2.3.3 Security Aspects 34
2.4 Threat Agents 35
2.4.1 Classification of Threat Agents 35
2.4.2 Targets of Threat Agents 37
2.5 Economic Consequences 40

3 Software Development Lifecycle 43
3.1 Software Development Lifecycle 43
3.2 Security Software Development Lifecycle 45
3.3 Threat Risk Modeling 53

4 Common Attack Scenarios 65
4.1 Social Engineering 67
4.2 Injection 72
4.2.1 SQL-Injection 73
4.2.2 Blind SQL Injection 79
4.2.3 Cross-Site Scripting 81
4.3 Cross-Site Request Forgery 87
4.4 Broken Authentication and Session Management 94
4.4.1 Session Hijacking 95
4.4.2 Session Fixation 99
4.4.3 Brute Force 103
4.5 Insecure Direct Object References 106
4.6 Insecure Cryptographic Storage 110
4.7 Failure to Restrict URL Access 113
4.8 Insufficient Transport Layer Protection 118
4.9 Invalidated Redirects and Forwards 123
4.10 Malicious File Execution 128
4.11 Information Leakage and Improper Error Handling 133
4.12 Security Misconfiguration 137
4.13 Executive Summary 140

5 Conclusion 145
5.1 Best Practices 145
5.2 Outlook and Summary 147

6 List of Literature 151

List of Abbreviations

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
AS / NZS	Australian / New Zealand Standard
ASCII	American Standard Code for Information Interchange
CEO	Chief Executive Officer
CGI	Common Gateway Interface
CIA	Confidentiality, Integrity and Availability
CMS	Content Management System
COM	Component Object Model
CSI	Computer Security Institute
CSRF	Cross-site Request Forgery
CSS	Cascading Style Sheets
CVSS	Common Vulnerability Scoring System
CWE	Common Weakness Enumeration
DBMS	Database Management System
DMZ	Demilitarized Zone
EJB	Enterprise Java Beans
FBI	Federal Bureau of Investigation
FSR	Final Security Review
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
HW	Hardware
IPS	Intrusion Prevention System
ISO	International Organization for Standardization
IT	Information Technology

JAAS	Java Authentication and Authorization Service
JDBC	Java Database Connectivity
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
MS	Microsoft®
NIST	National Institute of Standards and Technology
OWASP	Open Web Application Security Project
PHP	Hypertext Preprocessor
RFI	Remote File Include
SANS	SysAdmin, Networking and Security Institute
SDL	Security Software Development Lifecycle
SMB	Server Message Block
SQL	Structured Query Language
SSL	Secure Sockets Layer
SW	Software
TLS	Transport Layer Security
UML	Unified Modeling Language
URL	Uniform Resource Locator
US	United States
WASC	Web Application Security Consortium
WWW	World Wide Web
XSS	Cross-site Scripting

List of Illustrations

Figure 1: IT-Security Overview....17
Figure 2: Terminology Delimitation....20
Figure 3: Classification of Vulnerabilities....23
Figure 4: Logical Layer vs. Physical Layer....25
Figure 5: Three-Tier-Architecture....25
Figure 6: Layers and Technologies of a Typical Java Application....26
Figure 7: Client-Server Communication Principles....28
Figure 8: HTTP-Request....29
Figure 9: HTTP-Response....30
Figure 10: HTTP Status Code Categories....31
Figure 11: Server-Side and Client-Side Technologies....32
Figure 12: Three-Pillar Concept....34
Figure 13: Outcomes of Web Hacking....38
Figure 14: Top Attack Sources....38
Figure 15: Top Five Organizations Attacked Most Often....39
Figure 16: Average Loss Due to Web Application Security Incidents....41
Figure 17: General Lifecycle Model....44
Figure 18: Software Development Lifecycle of MS....44
Figure 19: Microsoft® Software Development Lifecycle....46
Figure 20: Process of the Software Development Lifecycle....48
Figure 21: Software Assurance Activities in the SDL....51
Figure 22: Security Best Practices of the Clusif Organization....52
Figure 23: Relative Cost of Removing Software Defects....53
Figure 24: Key Development Roles in the Threat Modeling Process....54
Figure 25: Microsoft® Threat Modeling Process....55
Figure 26: Threat Graph Example....57
Figure 27: STRIDE Threats Per Element....59
Figure 29: CVSS Metric Groups....61
Figure 30: Access Complexity Metric....62
Figure 31: CVSS Metrics and Equations....63
Figure 32: Evaluation of the CVSS Threat Modeling System....63

Figure 33: Basic Functionality of Social Engineering 70
Figure 34: Social Engineering on Valentines Day 71
Figure 35: Total Risk Ranking of Social Engineering....72
Figure 36: Common SQL Statement 75
Figure 37: Detectability of SQL Injection Flaws 75
Figure 38: Example of a SQL Injection Attack....76
Figure 39: Drop Table SQL Statement....76
Figure 40: Constrain SQL Input....77
Figure 41: SQL Parameter 78
Figure 42: Total Risk Ranking of SQL Injection 79
Figure 43: Example of a Blind SQL Injection 80
Figure 44: XSS Basics 84
Figure 45: Malicious Link....84
Figure 46: Persistent XSS Attack....85
Figure 47: Encoded Output....86
Figure 48: Escape Function in JavaScript....86
Figure 49: Total Risk Ranking of XSS 87
Figure 50: XSS vs. CSRF 88
Figure 51: Basic Principle of CSRF....90
Figure 52: Standard HTTP-Request....91
Figure 53: Malicious Image Tag, Compiled by the Author 91
Figure 54: Malicious HTTP-Request....92
Figure 55: Bold Tag in HTML....92
Figure 56: Malicious BBCode....92
Figure 57: Total Risk Ranking of CSRF 94
Figure 58: Sniffing....97
Figure 59: XSS Session Attack 97
Figure 61: Basic Session Fixation Attack Scenario....100
Figure 62: Session Fixation Attack Using XSS 101
Figure 63: META Tag Attack....102
Figure 64: HTTP-Header Response Attack 102
Figure 65: Brute Force Attack....104
Figure 66: Total Risk Ranking of Brute Force 106

Figure 67: Path Traversal Attack Scenario....108
Figure 68: Malicious Scripts in Path Traversal Attacks....108
Figure 69: Original CGI Request....109
Figure 70: Path Traversal CGI Attack....109
Figure 71: Total Risk Ranking of Path Traversal....110
Figure 72: Total Risk Ranking of Insecure Cryptographic Storage....113
Figure 73: Basic Principle of Forced Browsing....115
Figure 74: Forced Browsing Calendar Attack Part One....116
Figure 75: Forced Browsing Calendar Attack Part Two....116
Figure 76: Total Risk Ranking of Forced Browsing....118
Figure 77: Insecure Connection....120
Figure 78: Firesheep Add-On....121
Figure 79: Invalid SSL Certificate....122
Figure 80: Total Risk Ranking of Insufficient Transport Layer Protection....123
Figure 81: Open Redirect Attack....125
Figure 82: Open Forward Attack....126
Figure 83: Total Risk Ranking of Invalidated Redirects and Forwards....127
Figure 84: Malicious File Execution Vulnerability Example Part 1....129
Figure 85: Malicious File Execution Vulnerability Example Part 2....130
Figure 86: Malicious File Execution Vulnerability Example Part 3....130
Figure 87: Language Dependent Recommendations....132
Figure 88: Total Risk Ranking of Malicious File Execution....132
Figure 89: Comments in HTML....135
Figure 90: Error Handling Example....135
Figure 91: Total Risk Ranking of Information Leakage and Improper Error Handling....136
Figure 92: Directory List....138
Figure 93: Total Risk Ranking of Security Misconfiguration....140
Figure 94: Scale for the Attack Scenario Characteristics....141
Figure 95: Executive Summary of the Attack Scenarios....142

1 Introduction

1.1 Problem Statement

"Information is the currency of the new millennium."
(Crowell 2001)

William P. Crowell, former president and chief executive officer (CEO) of the Cylink Corporation, already identified information as the currency of the millennium in 2001. This fact tightens now nine years later by the increasing popularity of social networks and other web 2.0 developments where people divulge a large quantity of private information to the service provider whereof the related business model depends. On the one hand trust will be the critical success factor in the web 2.0 environment (McClure 2008, 36). On the other hand attacks against web applications have expanded and become even worse with the recent trends towards richer web 2.0 applications (Mehta 2008, 26). The focus has moved to application layer vulnerabilities because of the increasing security level of operating systems (NSA 2007, 1). Furthermore security weaknesses in web applications are often easy to exploit and not just feasible for professional hackers. The National Institute of Standards and Technology (NIST) recorded over 6,600 vulnerabilities with an upward trend already in 2006 (NSA 2007, 1).

The attackers' motives have changed over time from personal prestige to financial fraud today. The possible impact on the e-commerce is remarkable according to a survey conducted in the United States (US) which discovered that over 60 percent of clients would neglect doing business with a company if their personal data were at risk due to unsecure web applications (Clusif 2010, 6).

There are numerous reasons for the necessity of web application security. Web applications offer new and valuable ways to interact with customers but they also expose organizations to significant risks. 50 percent of all web applications have major vulnerabilities according to the SysAdmin, Networking and Security (SANS) Institute and 80 percent of successful attacks against organizations are caused by the exploitation of these flaws (Mehta 2008, 27). The extent of the problem is hard to measure. The parties involved are often even unaware when such attacks occur until the financial