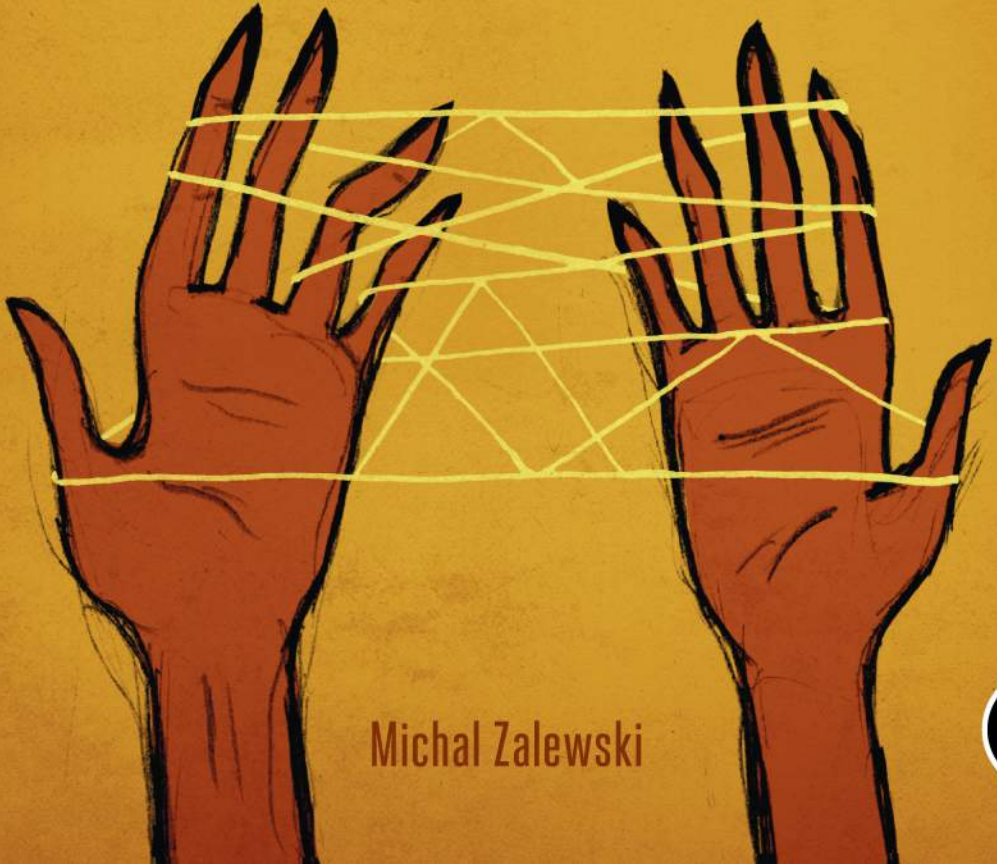
the Tangled Web

A Guide to Securing Modern Web Applications



Michal Zalewski

no starch press

CONTENTS IN DETAIL

PREFACE **xvii**

Acknowledgments xix

1
SECURITY IN THE WORLD OF WEB APPLICATIONS **1**

Information Security in a Nutshell 1
- Flirting with Formal Solutions 2
- Enter Risk Management 4
- Enlightenment Through Taxonomy 6
- Toward Practical Approaches 7

A Brief History of the Web 8
- Tales of the Stone Age: 1945 to 1994 8
- The First Browser Wars: 1995 to 1999 10
- The Boring Period: 2000 to 2003 11
- Web 2.0 and the Second Browser Wars: 2004 and Beyond 12

The Evolution of a Threat 14
- The User as a Security Flaw 14
- The Cloud, or the Joys of Communal Living 15
- Nonconvergence of Visions 15
- Cross-Browser Interactions: Synergy in Failure 16
- The Breakdown of the Client-Server Divide 17

PART I: ANATOMY OF THE WEB **21**

2
IT STARTS WITH A URL **23**

Uniform Resource Locator Structure 24
- Scheme Name 24
- Indicator of a Hierarchical URL 25
- Credentials to Access the Resource 26
- Server Address 26
- Server Port 27
- Hierarchical File Path 27
- Query String 28
- Fragment ID 28
- Putting It All Together Again 29

Reserved Characters and Percent Encoding 31
- Handling of Non-US-ASCII Text 32

Common URL Schemes and Their Function 36
- Browser-Supported, Document-Fetching Protocols 36
- Protocols Claimed by Third-Party Applications and Plug-ins 36
- Nonencapsulating Pseudo-Protocols 37
- Encapsulating Pseudo-Protocols 37
- Closing Note on Scheme Detection 38

© 2011 by Michal Zalewski

Resolution of Relative URLs 38
Security Engineering Cheat Sheet 40
When Constructing Brand-New URLs Based on User Input 40
When Designing URL Input Filters 40
When Decoding Parameters Received Through URLs 40

3
HYPERTEXT TRANSFER PROTOCOL 41

Basic Syntax of HTTP Traffic 42
The Consequences of Supporting HTTP/0.9 44
Newline Handling Quirks 45
Proxy Requests 46
Resolution of Duplicate or Conflicting Headers 47
Semicolon-Delimited Header Values 48
Header Character Set and Encoding Schemes 49
Referer Header Behavior 51
HTTP Request Types 52
GET 52
POST 52
HEAD 53
OPTIONS 53
PUT 53
DELETE 53
TRACE 53
CONNECT 54
Other HTTP Methods 54
Server Response Codes 54
200–299: Success 54
300–399: Redirection and Other Status Messages 55
400–499: Client-Side Error 55
500–599: Server-Side Error 56
Consistency of HTTP Code Signaling 56
Keepalive Sessions 56
Chunked Data Transfers 57
Caching Behavior 58
HTTP Cookie Semantics 60
HTTP Authentication 62
Protocol-Level Encryption and Client Certificates 64
Extended Validation Certificates 65
Error-Handling Rules 65
Security Engineering Cheat Sheet 67
When Handling User-Controlled Filenames in Content-Disposition Headers 67
When Putting User Data in HTTP Cookies 67
When Sending User-Controlled Location Headers 67
When Sending User-Controlled Redirect Headers 67
When Constructing Other Types of User-Controlled Requests or Responses 67

The Tangled Web
© 2011 by Michal Zalewski

4
HYPERTEXT MARKUP LANGUAGE 69

Basic Concepts Behind HTML Documents 70
- Document Parsing Modes 71
- The Battle over Semantics 72

Understanding HTML Parser Behavior 73
- Interactions Between Multiple Tags 74
- Explicit and Implicit Conditionals 75
- HTML Parsing Survival Tips 76

Entity Encoding 76

HTTP/HTML Integration Semantics 78

Hyperlinking and Content Inclusion 79
- Plain Links 79
- Forms and Form-Triggered Requests 80
- Frames 82
- Type-Specific Content Inclusion 82
- A Note on Cross-Site Request Forgery 84

Security Engineering Cheat Sheet 85
- Good Engineering Hygiene for All HTML Documents 85
- When Generating HTML Documents with Attacker-Controlled Bits 85
- When Converting HTML to Plaintext 85
- When Writing a Markup Filter for User Content 86

5
CASCADING STYLE SHEETS 87

Basic CSS Syntax 88
- Property Definitions 89
- @ Directives and XBL Bindings 89
- Interactions with HTML 90

Parser Resynchronization Risks 90

Character Encoding 91

Security Engineering Cheat Sheet 93
- When Loading Remote Stylesheets 93
- When Putting Attacker-Controlled Values into CSS 93
- When Filtering User-Supplied CSS 93
- When Allowing User-Specified Class Values on HTML Markup 93

6
BROWSER-SIDE SCRIPTS 95

Basic Characteristics of JavaScript 96
- Script Processing Model 97
- Execution Ordering Control 100
- Code and Object Inspection Capabilities 101
- Modifying the Runtime Environment 102
- JavaScript Object Notation and Other Data Serializations 104
- E4X and Other Syntax Extensions 106

The Tangled Web
© 2011 by Michal Zalewski

Standard Object Hierarchy 107
The Document Object Model 109
Access to Other Documents 111
Script Character Encoding 112
Code Inclusion Modes and Nesting Risks 113
The Living Dead: Visual Basic 114
Security Engineering Cheat Sheet 115
When Loading Remote Scripts 115
When Parsing JSON Received from the Server 115
When Putting User-Supplied Data Inside JavaScript Blocks 115
When Interacting with Browser Objects on the Client Side 115
If You Want to Allow User-Controlled Scripts on Your Page 116

7
NON-HTML DOCUMENT TYPES 117

Plaintext Files 117
Bitmap Images 118
Audio and Video 119
XML-Based Documents 119
Generic XML View 120
Scalable Vector Graphics 121
Mathematical Markup Language 122
XML User Interface Language 122
Wireless Markup Language 123
RSS and Atom Feeds 123
A Note on Nonrenderable File Types 124
Security Engineering Cheat Sheet 125
When Hosting XML-Based Document Formats 125
On All Non-HTML Document Types 125

8
CONTENT RENDERING WITH BROWSER PLUG-INS 127

Invoking a Plug-in 128
The Perils of Plug-in Content-Type Handling 129
Document Rendering Helpers 130
Plug-in-Based Application Frameworks 131
Adobe Flash 132
Microsoft Silverlight 134
Sun Java 134
XML Browser Applications (XBAP) 135
ActiveX Controls 136
Living with Other Plug-ins 137
Security Engineering Cheat Sheet 138
When Serving Plug-in-Handled Files 138
When Embedding Plug-in-Handled Files 138
If You Want to Write a New Browser Plug-in or ActiveX Component 138

The Tangled Web
© 2011 by Michal Zalewski

PART II: BROWSER SECURITY FEATURES 139

9
CONTENT ISOLATION LOGIC 141

Same-Origin Policy for the Document Object Model 142
document.domain 143
postMessage(...) 144
Interactions with Browser Credentials 145
Same-Origin Policy for XMLHttpRequest 146
Same-Origin Policy for Web Storage 148
Security Policy for Cookies 149
Impact of Cookies on the Same-Origin Policy 150
Problems with Domain Restrictions 151
The Unusual Danger of "localhost" 152
Cookies and "Legitimate" DNS Hijacking 153
Plug-in Security Rules 153
Adobe Flash 154
Microsoft Silverlight 157
Java 157
Coping with Ambiguous or Unexpected Origins 158
IP Addresses 158
Hostnames with Extra Periods 159
Non–Fully Qualified Hostnames 159
Local Files 159
Pseudo-URLs 161
Browser Extensions and UI 161
Other Uses of Origins 161
Security Engineering Cheat Sheet 162
Good Security Policy Hygiene for All Websites 162
When Relying on HTTP Cookies for Authentication 162
When Arranging Cross-Domain Communications in JavaScript 162
When Embedding Plug-in-Handled Active Content from Third Parties 162
When Hosting Your Own Plug-in-Executed Content 163
When Writing Browser Extensions 163

10
ORIGIN INHERITANCE 165

Origin Inheritance for about:blank 166
Inheritance for data: URLs 167
Inheritance for javascript: and vbscript: URLs 169
A Note on Restricted Pseudo-URLs 170
Security Engineering Cheat Sheet 172

11
LIFE OUTSIDE SAME-ORIGIN RULES 173

Window and Frame Interactions 174
Changing the Location of Existing Documents 174
Unsolicited Framing 178

The Tangled Web
© 2011 by Michal Zalewski

Cross-Domain Content Inclusion 181
A Note on Cross-Origin Subresources 183
Privacy-Related Side Channels 184
Other SOP Loopholes and Their Uses 185
Security Engineering Cheat Sheet 186
Good Security Hygiene for All Websites 186
When Including Cross-Domain Resources 186
When Arranging Cross-Domain Communications in JavaScript 186

12
OTHER SECURITY BOUNDARIES 187

Navigation to Sensitive Schemes 188
Access to Internal Networks 189
Prohibited Ports 190
Limitations on Third-Party Cookies 192
Security Engineering Cheat Sheet 195
When Building Web Applications on Internal Networks 195
When Launching Non-HTTP Services, Particularly on Nonstandard Ports 195
When Using Third-Party Cookies for Gadgets or Sandboxed Content 195

13
CONTENT RECOGNITION MECHANISMS 197

Document Type Detection Logic 198
Malformed MIME Types 199
Special Content-Type Values 200
Unrecognized Content Type 202
Defensive Uses of Content-Disposition 203
Content Directives on Subresources 204
Downloaded Files and Other Non-HTTP Content 205
Character Set Handling 206
Byte Order Marks 208
Character Set Inheritance and Override 209
Markup-Controlled Charset on Subresources 209
Detection for Non-HTTP Files 210
Security Engineering Cheat Sheet 212
Good Security Practices for All Websites 212
When Generating Documents with Partly Attacker-Controlled Contents 212
When Hosting User-Generated Files 212

14
DEALING WITH ROGUE SCRIPTS 213

Denial-of-Service Attacks 214
Execution Time and Memory Use Restrictions 215
Connection Limits 216
Pop-Up Filtering 217
Dialog Use Restrictions 218
Window-Positioning and Appearance Problems 219
Timing Attacks on User Interfaces 222

The Tangled Web
© 2011 by Michal Zalewski

Security Engineering Cheat Sheet 224
When Permitting User-Created <iframe> Gadgets on Your Site 224
When Building Security-Sensitive UIs 224

15 EXTRINSIC SITE PRIVILEGES 225

Browser- and Plug-in-Managed Site Permissions 226
Hardcoded Domains 227
Form-Based Password Managers 227
Internet Explorer's Zone Model 229
Mark of the Web and Zone.Identifier 231
Security Engineering Cheat Sheet 232
When Requesting Elevated Permissions from Within a Web Application 232
When Writing Plug-ins or Extensions That Recognize Privileged Origins 232

PART III: A GLIMPSE OF THINGS TO COME 233

16 NEW AND UPCOMING SECURITY FEATURES 235

Security Model Extension Frameworks 236
Cross-Domain Requests 236
XDomainRequest 239
Other Uses of the Origin Header 240
Security Model Restriction Frameworks 241
Content Security Policy 242
Sandboxed Frames 245
Strict Transport Security 248
Private Browsing Modes 249
Other Developments 250
In-Browser HTML Sanitizers 250
XSS Filtering 251
Security Engineering Cheat Sheet 253

17 OTHER BROWSER MECHANISMS OF NOTE 255

URL- and Protocol-Level Proposals 256
Content-Level Features 258
I/O Interfaces 259

18 COMMON WEB VULNERABILITIES 261

Vulnerabilities Specific to Web Applications 262
Problems to Keep in Mind in Web Application Design 263
Common Problems Unique to Server-Side Code 265

The Tangled Web
© 2011 by Michal Zalewski

EPILOGUE 267

NOTES 269

INDEX 273

The Tangled Web
© 2011 by Michal Zalewski