SYNGRESS®

FREE E-BOOK DOWNLOAD

Web Application Vulnerabilities

DETECT, EXPLOIT, PREVENT

Learn to Attack and Defend Mission-Critical Web Applications!

- Defend Web-Based Applications Developed with AJAX, SOAP, XMLPRC, and More
- See why Cross-Site Scripting Attacks Are So Devastating
- Download Working Code from the Companion Web Site.

Michael Cross
Steven Kapinos
Haroon Meer
Igor Muttik PhD
Steve Palmer
Petko "pdp" D. Petkov

Web Application Vulnerabilities Detect, Exploit, Prevent

Michael Cross
Steven Kapinos
Haroon Meer
Igor Muttik PhD
Steve Palmer
Petko "pdp" D. Petkov
Roger Shields
Roelof Temmingh

KEY	SERIAL NUMBER
001	HJIRTCV764
002	PO9873D5FG
003	829KM8NJH2
004	BAL923457U
005	CVPLQ6WQ23
006	VBP965T5T5
007	HJJJ863WD3E
008	2987GVTWMK
009	629MP5SDJT
010	IMWQ295T6T

PUBLISHED BY
Syngress Publishing, Inc.
Elsevier, Inc.
30 Corporate Drive
Burlington, MA 01803

Web Application Vulnerabilities Detect, Exploit, Prevent

Printed in the United States of America
1 2 3 4 5 6 7 8 9 0

ISBN 13: 978-1-59749-209-6

Publisher: Andrew Williams
Page Layout and Art: SPi
Copy Editor: Audrey Doyle and Judy Eby

Visit us at

www.syngress.com

Syngress is committed to publishing high-quality books for IT Professionals and delivering those books in media and formats that fit the demands of our customers. We are also committed to extending the utility of the book you purchase via additional materials available from our Web site.

SOLUTIONS WEB SITE

To register your book, visit www.syngress.com/solutions. Once registered, you can access our solutions@syngress.com Web pages. There you may find an assortment of valueadded features such as free e-books related to the topic of this book, URLs of related Web sites, FAQs from the book, corrections, and any updates from the author(s).

ULTIMATE CDs

Our Ultimate CD product line offers our readers budget-conscious compilations of some of our best-selling backlist titles in Adobe PDF form. These CDs are the perfect way to extend your reference library on key topics pertaining to your area of expertise, including Cisco Engineering, Microsoft Windows System Administration, CyberCrime Investigation, Open Source Security, and Firewall Configuration, to name a few.

DOWNLOADABLE E-BOOKS

For readers who can't wait for hard copy, we offer most of our titles in downloadable Adobe PDF form. These e-books are often available weeks before hard copies, and are priced affordably.

SYNGRESS OUTLET

Our outlet store at syngress.com features overstocked, out-of-print, or slightly hurt books at significant savings.

SITE LICENSING

Syngress has a well-established program for site licensing our e-books onto servers in corporations, educational institutions, and large organizations. Contact us at sales@syngress.com for more information.

CUSTOM PUBLISHING

Many organizations welcome the ability to combine parts of multiple Syngress books, as well as their own content, into a single volume for their own internal use. Contact us at sales@syngress.com for more information.

SYNGRESS®

Contributing Authors

Michael Cross (MCSE, MCP+I, CNA, Network+) is an Internet Specialist/Computer Forensic Analyst with the Niagara Regional Police Service (NRPS). He performs computer forensic examinations on computers involved in criminal investigation. He also has consulted and assisted in cases dealing with computer-related/Internet crimes. In addition to designing and maintaining the NRPS Web site at www.nrps.com and the NRPS intranet, he has provided support in the areas of programming, hardware, and network administration. As part of an information technology team that provides support to a user base of more than 800 civilian and uniform users, he has a theory that when the users carry guns, you tend to be more motivated in solving their problems.

Michael also owns KnightWare (www.knightware.ca), which provides computer-related services such as Web page design, and Bookworms (www.bookworms.ca), where you can purchase collectibles and other interesting items online. He has been a freelance writer for several years, and he has been published more than three dozen times in numerous books and anthologies. He currently resides in St. Catharines, Ontario, Canada, with his lovely wife, Jennifer, his darling daughter, Sara, and charming son, Jason.

Igor Muttik PhD is a senior architect with McAfee Avert™. He started researching computer malware in 1980s when anti-virus industry was in its infancy. He is based in the UK and worked as a virus researcher for Dr. Solomon's Software where he later headed the anti-virus research team. Since 1998 he has run Avert Research in EMEA and switched to his architectural role in 2002. Igor is a key contributor to the core security technology at McAfee. He takes particular interest in new emerging malware techniques, and in the design of security software and hardware appliances. Igor holds a PhD degree in physics and mathematics from Moscow University. He is a regular speaker at major international security conferences and a member of the Computer Antivirus Research Organization.

Haroon Meer is the Technical Director of SensePost. He joined SensePost in 2001 and has not slept since his early childhood. He has played in most aspects of IT Security from development to deployment and currently gets most of his kicks from reverse engineering, application assessments, and similar forms of pain. Haroon has spoken and trained at Black Hat, Defcon, Microsoft Tech-Ed, and other conferences. He loves "Deels," building new things, breaking new things, reading, deep find-outering, and making up new words. He dislikes sleep, pointless red-tape, dishonest people, and watching cricket.

Steve Palmer has 14 years of experience in the information technology industry. Steve has worked for several very successful security boutiques as an ethical hacking consultant. Steve has found hundreds of previously undiscovered critical vulnerabilities in a wide variety of products and applications for a wide variety of clients. Steve has performed security assessments and penetration tests for clients in many diverse commercial industries and government agencies. He has performed security assessments for companies in many different verticals such as the entertainment, oil, energy, pharmaceutical, engineering, automotive, aerospace, insurance, computer & network security, medical, and financial & banking industries. Steve has also performed security assessments for government agencies such as the Department of Interior, Department of Treasury, Department of Justice, Department of Interior, as well as the Intelligence Community. In 2001, Steve's findings contributed to the entire Department of Interior being disconnected from the Internet during the Cobel vs Norton lawsuit. Prior to being a security consultant Steve worked as a System Administrator, administering firewalls, UNIX systems, and databases for the Department of Defense, Department of Treasury, and the Department of Justice. Prior to that, Steve served 6 years in the United States Navy as an Electronics Technician. Steve has also written several security tools which have yet to be released publicly. Steve is also a member of the Department of Justice's Infragard organization.

Petko "pdp" D. Petkov is a senior IT security consultant based in London, United Kingdom. His day-to-day work involves identifying vulnerabilities, building attack strategies and creating attack tools and penetration testing

infrastructures. Petko is known in the underground circles as pdp or architect but his name is well known in the IT security industry for his strong technical background and creative thinking. He has been working for some of the world's top companies, providing consultancy on the latest security vulnerabilities and attack technologies.

His latest project, GNUCITIZEN (gnucitizen.org), is one of the leading web application security resources on-line where part of his work is disclosed for the benefit of the public. Petko defines himself as a cool hunter in the security circles.

He lives with his lovely girlfriend Ivana, without whom his contribution to this book would not have been possible.

Roelof Temmingh Born in South Africa, Roelof studied at the University of Pretoria and completed his Electronic Engineering degree in 1995. His passion for computer security had by then caught up with him and manifested itself in various forms. He worked as developer, and later as a system architect at an information security engineering firm from 1995 to 2000. In early 2000 he founded the security assessment and consulting firm SensePost along with some of the leading thinkers in the field. During his time at SensePost he was the Technical Director in charge of the assessment team and later headed the Innovation Centre for the company. Roelof has spoken at various international conferences such as Blackhat, Defcon, Cansecwest, RSA, Ruxcon, and FIRST. He has contributed to books such as *Stealing the Network: How to Own a Continent, Penetration Tester's Open Source Toolkit,* and was one of the lead trainers in the "Hacking by Numbers" training course. Roelof has authored several well known security testing applications like Wikto, Crowbar, BiDiBLAH and Suru. At the start of 2007 he founded Paterva in order to pursue R&D in his own capacity. At Paterva Roelof developed an application called Evolution (now called Maltego) that has shown tremendous promise in the field of information collection and correlation.

Contents

Chapter 1

Introduction to Web Application Hacking

Solutions in this chapter:

- What is a Web Application?
- How Does the Application Work?
- The History of Web Application Hacking and Evolution of Tools
- Modern Web Application Hacking Methodology and Tools
- Automated Tools: What they are good at and what they aren't
- A Brief Tutorial on how to use WebScarab

☑ Summary

Introduction

What is hacking? To me, the act of hacking is the tinkering, studying, analyzing, learning, exploring and experimenting. Not just computers, but anything. One of the great outcomes of this activity is discovering ways to make the object of your attention bend to your will for your benefit, under your control. An accountant who discovers a new tax loophole can be considered a hacker. Through out time tinkerers, thinkers, scholars and scientists who created things like the wheel, lever and fulcrum, capacitor, inductor, polio vaccine, the light bulb, batteries, phone, radio, air plane, and of course the computer, in a sense, were all hackers. All of the individuals behind most every great invention had a relentless pursuit to bend the will of whatever force they could leverage to a desired outcome. Very few innovations were created by accident, and even if the result of an accident was the inspiration, a great degree of tinkering, studying, analyzing, learning, exploring and experimenting was most certainly necessary to obtain or perfect the desired goal. Most great innovations came from an almost unnatural amount of tinkering, studying, analyzing, learning, exploring and tinkering... or hacking. The act of hacking when applied to computer security typically results in making the object of your desire (in this case, usually a computer) bend to your will. The act of hacking when applied to computers, just like anything else, requires tenacity, intense focus, attention to detail, keen observation, and the ability to cross reference a great deal of information, oh and thinking "outside of the box" definitely helps.

In this book, we aim to describe how to make a computer bend to your will by finding and exploiting vulnerabilities specifically in Web Applications. We will describe common security issues in web applications, tell you how to find them, describe how to exploit them, and then tell you how to fix them. We will also cover, how and why some hackers (the bad guys) will try to exploit these vulnerabilities to achieve their own end. We will also try to explain how to detect if hackers are actively trying to exploit vulnerabilities in your own web applications.

In this book the examples will being teaching how to find vulnerabilities using "Black Box" methods (where the user does not have the source code, documentation or web server logs for the application). Once the black box methods have been described, source code and audit trail methods of discovering vulnerabilities will also be mentioned.

It should also be noted that it is not possible to document every possible scenario you will run into and fit all of that information into one moderately sized book, but we will try to be as broad and encompassing as possible. Also this book more aims to teach the reader how to fish by defining a methodology of web application hacking and then describes how to find common vulnerabilities using those methodologies.

To begin our lessons in web application hacking it is important that you (the reader) are familiar with what a web application is and how one works. In this chapter, the next few sections describe how a web application works and the later sections in this chapter describe web hacking methodologies.

Web Application Architecture Components

Basically a web application is broken up into several components. These components are a web server, the application content that resides on the web server, and typically there a backend data store that the application accesses and interfaces with. This is a description of a very basic application. Most of the examples in this book will be based on this model. No matter how complex a Web application architecture is, i.e. if there is a high availability reverse proxy architecture with replicated databases on the backend, application firewalls, etc., the basic components are the same.

The following components makeup the web application architecture:

- The Web Server
- The Application Content
- The Datastore

The Web Server

The Web Server is a service that runs on the computer the serves up web content. This service typically listens on port 80 (http) or port 443 (https), although often times web servers will run on non standard ports. Microsoft's Internet Information Server and Apache are examples of web servers. It should be noted that sometimes there will be a "middleware" server, or web applications that will access other web or network applications, and we will discuss middleware servers in future chapters.

Most web servers communicate using the Hyper Text Transfer Protocol (HTTP) context and requests are prefixed with "http://". For more information about HTTP please refer to RFC 2616 (HTTP 1.1 Specification) and RFC 1945 (HTTP 1.0 Specification).

Ideally web applications will run on Secure Socket Layer (SSL) web servers. These will be accessed using the Hyper Text Transfer Protocol Secure (HTTPS) context and requests will be prefixed with "https://". For more information about HTTP please refer to RFC 2818 (HTTP Over TLS Specification). (We'll cover hardening a Web server in Chapter 7.)

The Application Content

The Application Content is an interactive program that takes web requests and uses parameters sent by the web browser, to perform certain functions. The Application Content resides on the web server. Application Content is not static content but rather programming logic content, or content that will perform different actions based on parameters sent from the client. The way the programs are executed or interpreted vary greatly. For example with PHP an interpreter is embedded in the web server binary, and interactive PHP scripts are then interpreted by the web server itself. With a Common Gateway Interface (CGI) a program resides in a special directory of the web server and