Review of the book

"Architecting Secure Software Systems"

by Asoke K. Talukder and Manish Chaitanya

CRC Press, Taylor & Francis Group, 2009

ISBN: 1-4200-8784-3, 978-1-4200-8784-0

Dr. Emin Islam Tatli
IBM Germany

April 2011

1 Summary of the review

The book "Architecting Secure Software Systems" focuses on both theoretical and practical aspects of designing secure software systems for different technologies, platforms and programming languages.

In the first part of the book, the theory part, the authors explain the need for security, security attributes and various attack types. A secure software development lifecycle methodology with its required activities for constructing secure software systems is given in the theory part as well. In the second part, they focus on practical aspects of software security. They provide concrete architecture examples and code snippets to explain how to construct secure software systems.

In this review, I will summarize the chapters, provide positive and negative aspects of the book and define the target reader group from my perspective.

2 Summary of the book

The book is divided into 10 chapters. The following will summarize the content of each chapter and give an overview of the book.

- **Chapter 1 - Security in Software Systems:** In this chapter, the need for computer security, security attributes (i.e. confidentiality, integrity, availability, authentication, authorization, accounting, anonymity), different attack types (e.g. passive/active attacks), various attacks (e.g.brute-force, authentication attacks, DoS), database security and security programming are explained in details.
- **Chapter 2 - Architecting Secure Software Systems:** This chapter is the heart of the book. It focuses on a generic methodology for building secure software systems. It defines the methodology for secure SDLC (software development lifecycle) and explains the required activities for realizing this methodology successfully in practice. Security requirement analysis (e.g. use and misuse cases), threat modeling (i.e. STRIDE, attack tree, DREAD, attack surface), security design (e.g. patterns, anti-patterns), security coding, safe programming, security review, security testing (e.g. penetration testing, fuzzing, ethical hacking), secured deployment, security documentation and security response planning are the main activities that belong to the methodology and explained in this chapter in details.
- **Chapter 3 - Constructing Secured and Safe C/Unix Programs:** Starting from this chapter, the authors explain constructing secured systems for specific platforms. Writing secure C/Unix

programs is the main topic in this chapter. Unix privileges, string operations, exception handling and memory management in C/C++ are explained in details as well.

- **Chapter 4 - Constructing Secured System in .NET:** The .NET runtime security and .NET security architecture are the main topics of this chapter. Additionally, ASP.NET security and Windows Security are covered.
- **Chapter 5 - Networking and SOA-Based Security:** The main topics of this chapter are using SSL (secure socket layer) in programming, SOA (service-oriented architecture) security, RPC (remote procedure call) security, RMI (remote method invocation) security, CORBA (common object request broker architecture) security, ActiveX security and DCOM (distributed component object model) security.
- **Chapter 6 - Java Client-Side Security:** This chapter explains Java security framework, Java cryptography application programming interface, Java secure sockets, authentication and authorization concept in Java, Java sandbox model, security of Java applets and Java Swing security.
- **Chapter 7 - Security in Mobile Applications:** This chapter explains NGN (next generation network) security, Java Micro Edition security, Java Card security, WAP (Wireless Application Protocol) security, security of mobile agents, mobile ad hoc network security and Digital Rights Management for mobile devices.
- **Chapter 8 - Security in Web-Facing Applications:** Web security, identity management (e.g. Single-Sign-On, identity federation etc.) and PKI (public key infrastructure) are the main topics in this chapter.
- **Chapter 9 - Server-Side Java Security:** Security of servlets, JSPs (java server pages), JSFs (java server faces), Struts and Enterprise JavaBeans and web application development rules are explained in this chapter in details.
- **Chapter 10 - Constructing Secured Web Services:** In the final chapter, Web Services Security is explained by illustrating XML and XPATH injection attacks.

3 What is the book like (style)?

The theory part of the book is quite well-written and gives a clear understanding of the need for constructing secure software systems. The explained methodology is well-structured and its activities that need to be carried out in each phase of software development are explained with good examples. In general, the book gives a good overview for various platforms and technologies from secure design perspective. Besides, a detailed reference section is included at the end of each chapter. These are the positive points in the book.

On the other hand, the practical part is not well-structured as opposed to my expectations from the title of the book. It brings a very broad set of different technologies and platforms together and tries to explain their security design concerns. But this prevents the authors from getting into details of each individual topic and makes many topics incomplete. For example, Java cryptography is summarized only in 5 pages and this critical section cannot be used as a reference for design or implementation. Besides, the book contains sometimes unnecessary details (e.g. description of OSI layers, competition history of AES algorithm) which are already known by an average reader with some network/security background and/or should not be contained within the book.

From my point of view, putting all these together, the book is like an overview book covering a number of topics just briefly or a terminology book explaining many security or technology terms, rather than a reference design and implementation book for security experts.

4 Would you recommend this book?

I would strongly recommend this book to people who need to get only an *overview* of different topics regarding secure software design. Therefore, the book is more suitable for students, people having less security knowledge or security experts that are in need of having an overview for various technologies.

I would not recommend this book as a *reference* design or implementation book for security experts to study a specific topic in details. If you need to go into details of SOA design security, for example, this book would be insufficient since the topic is incompletely covered in 4 pages. However, chapter 2 is an exception here. It can be very helpful for security experts working on secure design or implementation of software systems.

The reviewer is working as a Security Architect at IBM Germany.