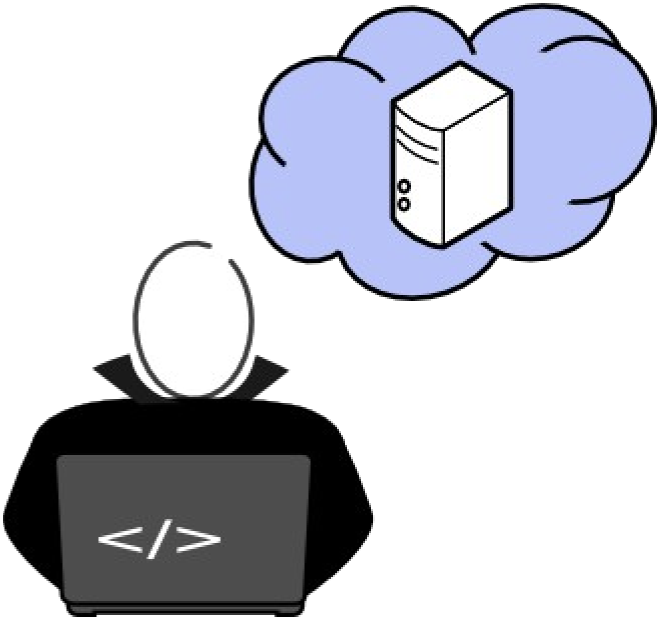
Frank Hissen



Secure Programming of Web Applications

Web Application Security for Software Developers and Project Managers

Contents

Motivation: Web Application Attacks

We can read about numerous successful attacks on well-known web applications on a weekly basis. Reason enough to study the background of "Web Application Security" of custom-made / self-developed applications - no matter if these are used only internally or with public access.

This book DOES NOT cover related topics like secure (network) infrastructures, operating system security, patch management, firewall architectures etc. but instead focuses only at the application level - the central field of activity of a software developer.

Web applications are a generic expression for

- Internet applications
- Intranet applications
- Cloud services
- Web portals
- Web services
- Web APIs

Some of these types of applications are not affected by certain attack patterns. For instance, a pure backend for a mobile app usually cannot be attacked through clickjacking - but (No)SQL code injections are extremely relevant.

Typical Attack Patterns

The most common / typical attacks against web applications are:

- [01] Code/Command Injection in general
    - e.g. e-mail, header injection
- [02] (No)SQL Code Injection
- [03] Cross-Site Request Forgery (CSRF)
- [04] Cross-Site Scripting (XSS)
    - i.a. JavaScript, HTML
- [05] Open Redirection
- [06] Remote File Inclusion (RFI) and Local File Inclusion (LFI) resp. Directory/Path Traversal
- [07] Clickjacking
- [08] Session-Hijacking
    - i.a. manipulation of transactions
- [09] Information Disclosure
- [10] Attacks on Weaknesses of the Authentification

    - password handling, hashing, reset, configuration
    - alternative authentication methods / multifactor

Moreover, the application level is related to the following attacks which, however, can only be influenced in a small way by the developer-side:

- [11] Denial of Service
- [12] Middleware
    - exploits, (TLS-) configuration, ...
- [13] Third-Party Software
    - Libraries (client/server)
    - Browser plugins (client-side)

Causes

The causes of successful attacks are (security-)weaknesses within the software. These can be found in the whole application stack:
Meaning the application itself, 3rd-party libraries / frameworks that are being used, middleware, operating system, virtualization stack, hardware, network components, …

→ Focus in this book is on the application level and its interfaces

In general, we speak of "programming errors". However, secure programming is much more complex than pure application programming! Deep technical and interdisciplinary understanding is the foundation of secure software.

Unfortunately, today's applications are highly complex in comparison to applications of 20 years ago. Due to this complexity, secure programming is a huge challenge - even having corresponding expert knowledge! Even 20 years ago, one of the most famous security and cryptography experts of our time, Bruce Schneier, said:

"Complexity is the Worst Enemy of Security"

Hacking Anatomy

First, let's try to understand which basic practices of attacks on web applications exist: