Mittuniversitetet
MID SWEDEN UNIVERSITY

The Complete Study Guide for DT144G Web Application Security

Daniel Bosk*

studyguide.tex 2062 2014-10-29 18:15:24Z danbos

Contents

1 Course Aim

The course is a preparatory course for development of secure web applications, to give an awareness of the need for security in web applications. The course covers the most frequent attacks against web applications and methods of preventing these attacks.

More specifically, after taking this course you should be able to:

- state the most common attacks against Web applications.
- explain how these attacks work.
- apply methods for preventing these attacks in Web applications.
- audit application code to find vulnerabilities.

2 Course Structure

The course is built around the material from the Open Web Application Security Project (OWASP). The OWASP project works generally with security which concerns the Web in some way, hence it covers a bit of e.g. mobile security as well. Among other things, every three years, OWASP publishes a top-ten list, the latest one being in 2013 [OWASP13], which contains a ranking of the ten most common vulnerabilities on the Web.

The tutoring of the course consists of a set of lectures introducing the field of security. As they progress, they are more and more focused towards secure software development and finally the focus is on Web application security. The main literature used in this part is Gollmann's *Computer Security* [Gol11] and Anderson's *Security Engineering* [And08]. After that, the lectures will change into workshops. Workshops are more practice oriented, in them you will have new information but also use the theory from the lectures. The main literature for this part is OWASP's material [e.g. OWASP13; ASVS14; MKC08]. The details of the contents of lectures and workshops is covered in section 2.1.

The examination of the course is a project and an audit of someone else's project (see the separate instruction for details). For the own project a report will be written, this report is first handed to the auditor. The auditor writes a report of his or her findings and hands it back. Then the auditor presents the project and audit report in a presentation session, the project developer may also reply during this session. The project developer may then fix the problems found by the auditor before finally handing the project in for examination.

The supporting literature for the auditing is *OWASP Testing Guide* [MKC08]. It is recommended to use *OWASP Application Security Verification Standard 2014* [ASVS14] before handing over the project to the auditor. There are also some tools available which can do some basic checking, see the OWASP website[1] for those.

It is recommended that you start working on this project as soon as possible. Within the first three weeks you must do the introductory assignment, which, in this course, consists of writing a project proposal (see that instruction for details).

[1] URL: `http://www.owasp.org/`.

There are also some non-mandatory labs available for you to do as exercises to gain experience in the area of security. Since these are not a part of the examination of the course, you are allowed to discuss them freely in the forums and during tutoring sessions.

2.1 Schedule

You can find a summary of the course schedule in Table 1 on the next page. The detailed reading instructions for each item in the schedule can be found in the subsections below. You are assumed to follow these instructions and timetable.

2.2 Introduction to Security

To introduce you to the area of security, you should first read Chapter 1 in *Security Engineering* [And08] and then read Chapters 1 and 2 in *Computer Security* [Gol11]. These chapters cover the history of the security field and an introduction to what it is all about.

After reading this material you are recommended to do exercises 1.2, 1.3 and 1.7 in [Gol11].

2.3 Foundations of Security

Gollmann's chapter on "Foundations of Computer Security" [Gol11, Ch. 3] attempts at a definition of Computer Security and related terms, e.g. confidentiality, integrity, and availability, which we need for our treatment of the topic. After reading this chapter you you are encouraged to do exercises 3.2, 3.5, 3.6, 3.7 and 3.8 in [Gol11].

Anderson also covers this in Chapter 1 of [And08]. However, he treats a wider area than just *computer* security, he covers many aspects of security in different examples.

2.4 Identification and Authentication

Identification and authentication of principals have always been a central part of computer security. Why we want to do this, and how we can accomplish this is treated in Chapter 4 in [Gol11].

Anderson also treats this topic (Chapter 2 in [And08]), although in a wider perspective with less technical details.

When you have read this chapter you should do exercises 4.2, 4.3, 4.4 and 4.6 in [Gol11]. (Also apply your knowledge of entropy to these exercises.)

2.5 Cryptographic Mechanisms

To fully understand how many security mechanisms can be implemented we need cryptography. Cryptography has a central role for many security mechanisms. Chapter 5 in Anderson's *Security Engineering* [And08] and Chapter 14 in Gollmann's *Computer Security* [Gol11] cover the aspects of cryptography we need in this course.

To practice your understanding of these mechanisms it is recommended to do exercises 14.2, 14.3 and 14.7 in [Gol11].

Course Week	**Work**
1	Course Start/Introduction to Security Lecture on Foundations of Security
2	Lecture on Identification and Authentication Lecture on Cryptographic Mechanisms, Part I Lecture on Cryptographic Mechanisms, Part II Lecture on Secure Protocols Tutoring session
3	Lecture on Security Usability Lecture on Access Control Lecture on Reference Monitors Workshop on Sensitive Data Exposure Workshop on Broken Authentication and Session Management Workshop on Missing Function-Level Access Control Tutoring session Deadline for the introductory assignment
4	Lecture on Accountability and Non-Repudiation Lecture on Software Security Lecture on Database Security Workshop on Injection Attacks and Cross-Site Scripting Workshop on Cross-Site Request Forgery Tutoring session
5	Workshop on Unvalidated Redirects and Forwards Workshop on Security Misconfiguration and Using Components with Known Vulnerabilities Tutoring session
6	Tutoring session
7	Tutoring session
8	Tutoring session
9	Tutoring session
10	Project audit presentation
+3 months	Project audit presentation
+10 months	Project audit presentation

Table 1: A summary of the course timetable. It is adjusted to a study rate of 20 hours per week.

2.6 Secure Protocols

As soon as two principals need to interact, there is need for a protocol which secures the communication, be it inside or between systems – even one principal communicating with itself in different points in time, which is the case when storing something for use at a later time.

Anderson gives an overview of this area in *Security Engineering* [And08], Chapter 3 "Protocols". Gollmann has a more technically detailed treatment in Chapter 15 of *Computer Security* [Gol11].

2.7 Security Usability

One important aspect of security, which traditionally is forgotten, is the users' weaknesses. The psychology of the human mind is therefore an important subject to discuss in the context of security. Anderson gives a short summary of the psychology of users, their strengths and weaknesses, in Chapter 2 "Usability and Psychology" in [And08].

Also treated in this lecture is the ever-recurring problem of password policies. The material covering this area is the article "Of passwords and people: Measuring the effect of password-composition policies" [Kom+11] and its follow-up article "Can long passwords be secure and usable?" [Sha+14].

2.8 Access Control

Once you have authenticated users you can support access control – and this is also one of the main reasons to authenticate them in the first place. Access control aims at controlling who may access what, and how they may access it. This is treated by Chapter 5, followed by Chapters 11 and 12, in *Computer Security* [Gol11]. You are also recommended to read Anderson's treatment of the subject, he treats this in Chapters 4, 8, and 9 in *Security Engineering* [And08].

To establish your newly gained knowledge in this area, you should do exercises 5.1, 5.2, 5.5, 5.6, 5.8 and 5.9 in [Gol11].

2.9 Reference Monitors

The area of reference monitors covers enforcing access controls, it also covers trusted computing base and enforcing access control on the lower layers in the system architecture. Gollmann treats this area in Chapter 6 of his book *Computer Security* [Gol11].

Exercises 6.1, 6.3 and 6.5 in [Gol11] are recommended for your learning.

2.10 Sensitive Data Exposure

The workshop treats OWASP risk "A6 Sensitive Data Exposure" in the OWASP Top 10 [OWASP13]. The content of the workshop is thus applying cryptographic mechanisms in Web applications to protect sensitive data.

Before attending the workshop, you should also have read "V7 Cryptography at Rest Verification Requirements", "V9 Data Protection Verification Requirements", "V10 Communications Security Verification Requirements" and

"V11 HTTP Security Verification Requirements" in *OWASP Application Security Verification Standard 2014* [ASVS14].

2.11 Broken Authentication and Session Management

The treatment of this workshop is the OWASP Top 10 risk "A2 Broken Authentication and Session Management" [OWASP13]. All access control and accountability properties depend on proper authentication, i.e. that you can ensure the user is who he or she claims to be. Since HTTP(S) is a stateless protocol, we need session management to keep track of a user between the different requests to the server; we don't want the user to have to send his or her username and password with every request, hence we need sessions.

Due to this, it is important to get authentication and session management right – which, unfortunately, is rarely the case! If implemented incorrectly, an attacker can compromise passwords, keys, sessions, or other flaws to assume other users' identities.

Before attending the workshop, you should also read "V2: Authentication Verification Requirements" and "V3: Session Management Verification Requirements" in [ASVS14].

2.12 Missing Function-Level Access Control

This workshop treats OWSAP risk "A7 Missing Function Level Access Control" in OWASP Top 10 [OWASP13]. The workshop emphasizes on the need for proper access control functionality, namely that this is needed on the function level of the application. Without this, there is a risk that the application might end up with presentation-layer access control, which is very easy to circumvent.

The risk "A4 Insecure Direct Object References" in the OWASP Top 10 [OWASP13] is also covered, since it is very much related, in fact a good example of the above. Direct object references are sometimes used in applications to be able to link to a specific part of the application, e.g. a specific comment or post in a community, through a file or using a database key. However, sometimes these direct object references can yield vulnerabilities which can be exploited by attackers to get data they are not authorized to access.

Before attending the workshop, you should also have read "V4 Access Control Verification Requirements" in [ASVS14].

2.13 Accountability and Non-Repudiation

The need for accountability has been apparent in civilisations for as long as they have existed. One of today's institutions which are most renowned for keeping accounts are banks, it is quite natural therefore that Anderson describes accountability with start in the experience from banks. He treats this subject in Chapter 10 "Banking and Book-keeping" in [And08].

Gollmann also describes the Clark-Wilson Security Policy Model in Section 12.3 of his book [Gol11]. This is a model of how to securely enforcing a security policy.

Further, Schneier and Kelsey describes a system for secure audit logs in their paper "Secure audit logs to support computer forensics" [SK99]. The construction described therein is a method to safely store audit logs in an untrusted

machine; in the scheme, all log entries generated prior to a compromise will be impossible for the attacker to read, modify, or destroy undetectably. This is not interesting because you very probably will implement this scheme, because you will proably not. It is interesting because it is a bit counter-intuitive at first, it is an example of application of crypto mechanisms, and having seen it will help you to "keep your heads out of any boxes".

2.14 Software Security

Perhaps the part of security most people intuitively associate with security, and computer security in particular, is software security. This part of computer security treats vulnerabilities in software, e.g. possibility of buffer overruns or code injections. Gollmann treats this area in Chapter 10 of his book, *Computer Security* [Gol11]. The recommended exercises to do after reading this material are 10.1, 10.3 and 10.4 [Gol11].

Anderson also treats this subject—in Chapter 4.4 and Chapter 18 of [And08]—albeit with less technical details.

2.15 Database Security

Databases are used to store user and application *data*, however, it does provide *information* to its users. Database security thus constitues a long range of problems, from who may access what data to what information a user can get by combining different data. An example of the latter being that a user may only access averages of tables, not individual values. However, by combining the average of all entries and the average of all entries except one, that one entry can be inferred.

Gollmann treats this area in Chapter 9 of his book, *Computer Security* [Gol11]. After reading this material you are recommended to do exercises 9.1, 9.3 and 9.4 in [Gol11].

2.16 Injection Attacks and Cross-Site Scripting

The workshop focuses on OWASP Top 10 risk "A1 Injections" [OWASP13]. In most (if not all) Web applications the user provide input in some form. This input data cannot be trusted, even if it is from something like a "unmodifiable" drop-down selection box (the attacker can write a custom request totally ignoring the limitations of that box). This is one of the most serious and common problems on the Web, that applications use user-provided data in interpreted code or queries. The result is that the attacker can execute arbitrary code in the vulnerable component.

The risk "A3 Cross-Site Scripting" in the Top 10 [OWASP13] is also covered. Cross-Site Scripting is a special case of injection. The aim in this attack is to inject script code which is then served by the Web application to other users.

Before attending the workshop you should also read "V5 Malicious Input Handling Verification Requirements" in [ASVS14].

2.17 Cross-Site Request Forgery

This workshop treats "A8 Cross-Site Request Forgery" in OWASP Top 10 [OWASP13]. This class of exploits utilizes that a user is signed in to a Web application in the same browser as used to visit a malicious site. This way the attacker can forge requests from the user since the browser will automtically include session cookies etc.

2.18 Unvalidated Redirects and Forwards

This workshop concerns "A10 Unvalidated Redirects and Forwards" in the OWASP Top 10 [OWASP13]. It is common for Web applications to redirect users; e.g. in online payment or single sign-on systems. In some cases an attacker can use this to redirect the user to a malicious URL, other times redirect him- or herself to pages he or she is not authorized to access.

2.19 Security Misconfiguration and Using Components with Known Vulnerabilities

This workshop considers two related risks in the OWASP Top 10 [OWASP13], namely "A5 Security Misconfiguration" and "A9 Using Components with Known Vulnerabilities". One of the cornerstones to a secure Web application is the components used and the configuration of these components. Every now and then vulnerabilities are discovered in the libraries available to developers, thus it is important to keep these up-to-date. Also, the default configuration of these libraries, frameworks or other components, usually come with insecure default configurations.

Before attending the workshop, you should also read "V8 Error Handling and Loggin Verification Requirements", "V16 Files and Resources Verification Requirements" in [ASVS14].

2.20 Project

The main literature for the development of the project is the books by Gollmann [Gol11] and Anderson [And08], the documents from the OWASP project [OWASP13; MKC08; ASVS14] and the papers by Komanduri et al. [Kom+11; Sha+14]. Of course the same literature applies for the audit of a Web application as well.

3 Examination

The introductory assignment, i.e. the project proposal, will be reported as I101 in Ladok. The grading scale is pass and fail.

The audit is reported as G101 in Ladok, it gives 1.5 credit points. The grading scale is pass and fail.

The project is examined through the report and the audit presentation. The report is graded A-E for pass or F-Fx for fail. This is reported in Ladok as P101 and gives 6.0 credit points.

3.1 What if I'm not done on time?

The deadlines are very important in this course, and there are several you should be aware of which are covered here.

First, you must have completed the introductory assignment (I101) within the first three weeks of the course, otherwise you will be deregistered from the course and your place will be available to other applicants.

Second, for the presentations there will be a total of three scheduled presentations for the duration of a year, the first one will be held in the end of the course. All these events will be in the course schedule in the University's schduling function in the Student Portal.

The deadlines for the hand-in of the report are strict. If you miss the deadline for the report you are not eligible to participate, as a possible auditor have not had enough time for auditing your project. Hence, you are referred to the next presentation event. After the third round of presentations you are referred to the next time the course is given.

Third, there is no tutoring scheduled after the end of the course, i.e. the final tutoring session in the schedule. Hence, if you cannot finish the course on time and want to be guaranteed tutoring you should reregister on the next time the course is given.

Unfortunately, reregistering on a course is of lower priority than registering for the first time. Thus, everyone applying for the course for the first time will be prioritized, this includes the reserve lists as well. As such it is much better for you do deregister from the course yourself. This can be done within three weeks of the course start and has the benefit that you will be counted as a first time applicant for the course.

References

[ASVS14] Sahba Kazerooni, Daniel Cuthbert, Andrew van der Stock, and Krishna Raja, eds. *OWASP Application Security Verification Standard 2014*. 2014. URL: `https://www.owasp.org/images/5/58/OWASP_ASVS_Version_2.pdf`.

[And08] Ross J. Anderson. *Security Engineering. A guide to building dependable distributed systems*. 2nd ed. Indianapolis, IN: Wiley, 2008. ISBN: 978-0-470-06852-6 (hbk.) URL: `http://www.cl.cam.ac.uk/~rja14/book.html`.

[Gol11] Dieter Gollmann. *Computer Security*. 3rd ed. Chichester, West Sussex, U.K.: Wiley, 2011. ISBN: 9780470741153 (pbk.)

[Kom+11] Saranga Komanduri, Richard Shay, Patrick Gage Kelley, Michelle L. Mazurek, Lujo Bauer, Christin Nicolas, Lorrie Faith Cranor, and Serge Egelman. "Of passwords and people: Measuring the effect of password-composition policies". In: *CHI*. 2011. URL: `http://cups.cs.cmu.edu/rshay/pubs/passwords_and_people2011.pdf`.

[MKC08] Matteo Meucci, Eoin Keary, and Daniel Cuthbert, eds. *OWASP Testing Guide*. 2008. URL: `http://www.owasp.org/images/5/56/OWASP_Testing_Guide_v3.pdf`.

[OWASP13] The Open Web Application Security Project. *OWASP Top 10 - 2013. The Ten Most Critical Web Application Security Risks*. June 2013. URL: http://owasptop10.googlecode.com/files/OWASP%20Top%2010%20-%202013.pdf.

[SK99] Bruce Schneier and John Kelsey. "Secure audit logs to support computer forensics". In: *ACM Transactions on Information and System Security (TISSEC)* 2.2 (1999), pp. 159–176.

[Sha+14] Richard Shay, Saranga Komanduri, Adam L Durity, Phillip Seyoung Huh, Michelle L Mazurek, Sean M Segreti, Blase Ur, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. "Can long passwords be secure and usable?" In: *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*. ACM. 2014, pp. 2927–2936. URL: http://lorrie.cranor.org/pubs/longpass-chi2014.pdf.