UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

COS 330

August 19, 2016

Practical 2

Release Date: 17 August 2016

Due Date: 31 August 2016

Submission Deadline: 30 August 2016

Note: There will be two practical sessions on the 31st August 2016 (First session: 08h00 - 09h30 and Second session: 17h30 - 19h00)

Instructions

In this practical you are required to create a simple web application (SQL-based) on the Apache/MySQL/PHP stack on the virtual machine. The application needs to have table(s) that include 'user' table that contain usernames and hashed passwords. The application must have two types of users, the administrator with write priviledges and the normal user who can only view the content on the web page(s) of the application. Like the first practical, the application must have the functionality of registering new users. First you are required to find and exploit specific vulnerabilities in this web application. Then you will need to modify the web application appropriately to prevent these vulnerabilities.

Note: The aim of the practical is not to test anything other than Computer Security concepts that you are expected to know by now.

Upload a zip archive of your screenshots and code, your answers and your modified code onto the CS website by the above-mentioned due date. If you do not demonstrate your work in one of the practical sessions, then you will not be allocated any marks (even if you did upload), i.e. you must be present in person at the demo session to be able to receive a mark.

Everything that you submit might be checked for plagiarism. Instances of plagiarism will be dealt with in a serious manner.

Background

Incomplete mediation of input data can lead to terrible security failures. When a web page posts critical information visible on the address bar as part of the URL to another page or system, the information can be manipulated. The exploitation of this common flaw is known as URL hacking.

Other types of vulnerabilities that occur due to incomplete mediation of input are SQL injection and Cross-site scripting (XSS). Wikipedia has a good description of each with examples.

If no adequate care is taken when transferring critical information in a system, an attacker can easily tamper with the information to get what they want – a failure of integrity.‘

Task 1 [5 Marks]

In this task you are to perform different SQL injection attacks to achieve different goals:

1. Attempt an SQL injection attack to gain entry to the web application as a normal user without the knowledge of any usernames or passwords. [2]
2. Modify the previous SQL injection attack to gain entry to the web application as an administrator. [1]
3. After gaining access as an administrator perform another SQL injection attack to get access to all the user details including the password hashes. [2]

Hint: SQL UNION is your friend.

Task 2 [5 Marks]

In this task you are to perform two different kinds of Cross-scripting (XSS) attacks:

1. Perform a non-persistent XSS attack to simply deface the web application. [2]
2. Perform a persistent XSS attack after you gain access with SQL injection to steal the administrator's session. You need to ex-filtrate the required information to a different web site or application. [3]

Task 3 [10 Marks]

Assume you are a security professional and the problems in task 1 and 2 above are presented to you. In this task you are to alter the web application to provide a viable solution to these problems (i.e. solve the SQL injection and XSS vulnerabilities).

Hint: See the countermeasures listed in Chapter 3 of the prescribed textbook.