Lab Report

SECURITY INSIDER LAB II PART 3: WEB APPLICATION VULNERABILITIES - 3

Group 5

Abhijeet Patil
Mohammad Saiful Islam
Thejeswi Preetham Nagendra Kamatchi

Exercise 1: Session Fixation

The security team of the bank is working overtime to fix all the wholes you discovered and exploited. Assume that they were successful in disabling all vulnerabilities which you exploited using JavaScript. Further, assume that you canâĂŹt steal cookies anymore. Is there still a way to get access to an account of a user?

1. Sketch an attack that allows you to take over the session of a bank user.

a When we visit the bank application with a browser a cookie with the key "USECURITYID". This key has a random hexadecimal value assigned to it.

b We send a money transfer to the victim's account with a meta tag in the remark. The meta tag looks like: <meta http-equiv="Set-Cookie" content="USECURITYID=abc path=/">

c Now when the victim visits his accounts page and his session has been replaced with a different session or "USECURITYID".

d The attacker now visits the bank application with his "USECURITYID" manually set to the same one as the victim and thus can access his account.

2. 2. How can you generally verify that an application is vulnerable to this kind of attack?

a Copy the security token from one browser and paste it in another browser.

b Login from the second browser.

c Refresh the page in the first browser, if the page is logged in to the account with which user logged in to the second browser, the application is vulnerable to session fixation.

3. 3. Does https influence your attack?

https does not influence this attack.

4. Accordingly, which countermeasure is necessary to prevent your attacks? Patch your system and test it against session fixation again.

A counter measure to prevent this attack is:

i Generating new tokens every request.

ii Removing tags from user inputs.