Name: Jun Wang 000791814

Lab 4: Attacking Session Management

Web Application Security

Marks: \_\_\_\_\_

Bonus: \_\_\_\_\_

Table of Contents

[Lab Outcome 2](#_Toc524601642)

[Background Reading 2](#_Toc524601643)

[Required Hardware/Software 2](#_Toc524601644)

[Introduction 2](#_Toc524601645)

[1.0 Spoofing 2](#_Toc524601646)

[2.0 Session Hijacking 2](#_Toc524601647)

[3.0 Session Fixation 2](#_Toc524601648)

[4.0 Sign-Off – Lab 4: Attacking Session Management 3](#_Toc524601649)

Web Application Security

Lab 4: Attacking Session Management

# Lab Outcome

Exploit session management controls.

Background Reading

Read the textbook sections listed in the Course Schedule.

Architecture Diagram

|  |
| --- |
|  |

Required Hardware/Software

* VM Ubuntu 18.04 – 2 CPU, 4GB Ram, 20 GB hard disk
  + Docker
    - WebGoat v7.1
    - DVWA (Dawn Vulnerable Web App)
* VM Kali
  + Burp or other Web proxy (scanner)

# Introduction

Session management controls what users can do and how they can interact with the system. It is crucial in maintaining a user’s identity across multiple requests.

# 1.0 Spoofing

In WebGoat, complete the **Session Management Flaws > Spoof an Authentication Cookie** lesson.

**Step1>login as webgoat, we found the cookie is “65432ubphcfx”**

**Step2>login as aspect , found the cookie is “65432udfqtb”**

**These two cookies have the same digits65432, and the following characters are the reversed order and shift right one character of the username.**

**alice ->reverse order ->ecila**

**ecila -> right shift one character -> fdjmb**

**alice’s authentication cookie should be “65432fdjmb”**

**Step3> login as webgoat, change the cookie from “65432ubphcfx” to “65432fdjmb” and refresh.**

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | |  |      |  | | --- | |  |      |  | | --- | |  | |

# 2.0 Session Hijacking

(Tool: BurpSuite)

In WebGoat, complete the **Session Management Flaws > Hijack a Session** lesson.

Vulnerability: The server skips authentication if you send the right cookie.

Step1>Set proxy both on browser and burp.

Step2> login on browser, capture the response packet with keyword”set cookie.” Right click on the package and “send to sequencer”.

Step3>sequencer -> start live capture, and then analyze the live capture, we found The cookie value is made of a set of 2 values. The first part of the cookie is a sequential number, the second part is milliseconds. The cookie value is not random but predictable.

Step4> save the live capture to “cookie.txt”, sort the file so we can find some gap between two consecutive cookies. ( in this case: 40628-1571083623745 40630-1571083623763)

Step5> Pick the cookie request packet and send it to intruder.

Step6>intruder->positions, change the WEAKID to the sequence number gaped in our list, which might be the session id for another user. Find the difference range between the number before and after the gaped number. And iterate through the range one number a step. The different response packet length shows the the number is we are looking for( 40629-1571083623754)

Step7> copy this cookie and paste in browser to replace the previous cookie,

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

# 3.0 Session Fixation

In WebGoat, complete the **Session Management Flaws > Session Fixation** lesson.

Step1> add session id “SID=1500”( any value the hacker can pick) to the URL. And send the email.

Step2> Jane receive the email and click the link “ Goat HIlls Financial”.

Step3> Jane logs in. We will find the session id on the URL is the value the hacker assigned SID=1500.

Step4> On the login page, the hacker Joe changed his URL to SID =1500. Joe will take the session which belongs to Jane and log in without username or password.

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

# 4.0 Sign-Off – Lab 4: Attacking Session Management

Detach this page and submit it to your instructor to indicate that you have completed each section.

Name:

Student ID:

|  |  |
| --- | --- |
| **Section** | **Instructor Initials** |
| 1.0 Spoofing |  |
| 2.0 Session Hijacking |  |
| 3.0 Session Fixation |  |