SCC 363 - Web Security Assignment

Submission deadline: Monday 11 March 2019, 16:00

Submission format: Please submit one pdf file named <groupid>-attack.pdf with group id being your assigned group id. You may also submit a zip file <groupid>-attack.zip with any files used for attacks, such as web pages

Introduction

Assume you are a part of a pen-testing team in the security industry specializing in webservers. Your client, who runs an online marketplace website, wants you to thoroughly pen-test their webserver and find as many security defects as possible. Carry out a (blackbox) penetration test of the webserver and produce a pen-testing Report.

Instructions

The VM is deployable through Vagrant. Once you boot it, you do not need to log in (and should not). The VM will run on ip 192.168.56.101, which should give you access to the site when entered into a web browser on your host.

The VM is usable on your own laptops now using the file below, and will be installed on the lab machines in B76 in the overnight update, so you should be able to boot using "svagrant up scc-363-webapp" in the terminal.

The VM is available using the vagrant file

http://scc-vagrant.lancs.ac.uk/atlas/scc-363-webapp/Vagrantfile

How to run the VM on your own machine:

http://scc-vagrant.lancs.ac.uk/help.html

How to run the VM in the labs:

http://scc-vagrant.lancs.ac.uk/labs.html

When running on your own laptop, you will need to delete and remake the folder you open the vm in in-between runs. This will not need to be done in the labs.

Attacks and Vulnerabilities List

We classify an attack as any action that can be performed by a regular (non admin) user such as yourself which has the potential to cause the company harm financially, reputationally or any other way.

Each attack listed below is associated with a reward depending on its severity. You need to perform all of them and write a report. Attacks are grouped by type, though for the report you should write up each one individually. Attacks should be treated individually - treat each attack in isolation assuming no knowledge from other attacks. You should not use any automated tools, such as SQL map, to do these attacks, though software such as burp is okay.

- Access Control
  - **[Medium]** View the admin page as a non admin user
  - **[High]** Find the folder containing all the images used in the website
  - **[HCritical]** Successfully log into the mysql database remotely
- SQL Injection
  - **[High]** use SQL injection to log into the website without knowing the password
  - **[Critical]** Use SQL injection to register as admin
  - **[Critical]** Use SQL injection to change another user's password when buying a product.
  - **[HCritical]** Use SQL injection to print the contents of a table. You may need to perform multiple queries to do this.
- Person-in-the-middle/proxy/tampering
  - **[High]** Register as an admin user (without SQL injection)
  - **[High]** Get the store to owe you money
  - **[HCritical]** View the full php source code of any page
- CSRF
  - **[HCritical]** Perform a CSRF attack on the website to add a product to a victim's basket. You can demonstrate this by running your own local webserver using XAMPP or similar.
- XSS
  - **[High]** Stored XSS on the contact us page.
  - **[High]** Reflected XSS on the Search page.
  - **[HCritical]** Use a stored XSS attack to steal the cookie of the admin user, and use it to view the home page as the admin user when not logged in as a different user. You should collect the cookie as if you were an attacker, so assuming you are not the admin yourself.

Marking Scheme

For each attack (in no more than 500 words per attack):

1. **[2 marks]** Describe the following: what is the attack category, vulnerability exploited, technical impact (Loss of integrity, availability, confidentiality, or accountability), business impact (Financial damage, reputational damage, privacy violation), and its likelihood (Low, Medium, or High). You should justify your answers.
2. **[1 Mark]** Explain why each vulnerability exists and how it can be fixed. Fixes should be specific to the particular occurrence of the vulnerability. Vague answers such as "apply blacklisting" will not be sufficient - what should be blacklisted?
3. **[Medium severity – 1 mark, High severity – 3 marks, Critical – 5, and Highly Critical – 8 marks]** Document the code you developed to exploit the vulnerability. And, perform the attack and capture the attack (including any intermediate stages and post attack state) using screenshots. For example, if you use SQL injection to print the contents of the table, take screenshots of the final result, and any intermediate queries results.

Note that the marks add up to 111. This will be scaled down after marking.

Marking

You will be expected to demonstrate all of the attacks in the lab marking session in week 19. Group members will be chosen at random to carry out the attacks, and explain them. Therefore, while when working on the assignment you can delegate report writing and attacking based on skillset, you should all be familiar with your group's attacks.