Web Security: Hands on Activity

This activity will give you some hands on experience with pentesting a real vulnerable web application and allow you to explore the impact of various security policies. We will be using DVWA – the Damn Vulnerable Web Application – as the web app to attack. You will perform the steps below in groups of two, and then submit a lab report in pdf format containing short descriptions and screenshots of all your steps. What to include in the report is written in green font in the steps below. Screenshots alone will get zero credit. The naming convention that you should follow for the submission is **DVWA1_yourSection_yourRollNumber_yourPartnerRollNumber.**

Part 1: Setup

Please follow the separately provided setup manual to get DVWA up and running on your machine. Any errors you run into will usually appear at Step 5 of the manual. Fixes for some common errors can be found on this link: https://www.blackmoreops.com/2018/11/13/configure-your-web-application-pentesting-lab/2/ and in Step 6 of the manual. Whenever you make any change to the configuration files, remember to restart the Apache server from the XAMPP control panel.

Once you get to the login page, setup is complete and you can now move to the pentesting part.

In your report, make a heading “Part 1” and report any errors that came up and how you removed them.

Log in using username “admin” and password “password” (without quotes).

Part 2: SQL Injection Attacks

The first attack we will try is SQL Injection. DVWA has 4 security settings, i.e. low, medium, high and impossible. We will attempt SQL Injection attacks to steal the password of users stored in the database in all four security settings.

Part 2A: SQLi attack under low security:

First set the security to Low (from the DVWA Security button in the menu on the left).

Click the View Source button on the bottom right of the DVWA page to see the code behind the submit button. Observe the portion where the SQL is being constructed. Based on this code, write (1) a query that will return first name and surname of all users in the database, and (2) a query that will return the list of user names and passwords from the users table. To view the table and attribute names in the database for constructing your query, you can go to *Admin* in front of MySQL in the XAMPP control panel.

In your report, make the heading “Part 2A” and paste a screenshot of the results of both the above queries. Make sure the query you wrote is also visible.

Part 2B: SQLi attack under medium security:

Now change the security to Medium (from the DVWA Security button in the menu on the left).

In your report, under the heading “Part 2B”, briefly describe what has changed, in both the webpage display and the code on the backend? Also describe all the security techniques that have been implemented compared to the low security setting.

Now you can close tha DVWA tab in your browser. You instead need to open up Burpsuite and use the integrated browser (Chromium) to run your DVWA app. In Burpsuite, toggle intercept HTTP traffic to ON. Once again login to DVWA, set security to Medium, and go to SQL Injection. Now submit the ID 1. Go to Burpsuite and find the last request in HTTP History. You will see that it is a post request. When you view the request, you will see the part where you are submitting ID 1. Now right click the request and select "Send to Repeater". Go to the repeater tab. Now you can edit this request. Now instead of the "1" you were submitting, add a query that will return all usernames and passwords. Send it using the Send button and see if it worked. If you get an SQL syntax error, you need to write a new query. Once again you should use the View Source button on the bottom right of the DVWA page to see how the query should be formed.

In your report, show screenshots of Burpsuite capturing the query when you submitted ID 1, your edit to the query in Burpsuite that results in a successful SQLi attack, and the response you get in Burpsuite when you finally successfully obtain the usernames and passwords. In the end, discuss why the medium security added was insufficient.

Part 2C: SQLi attack under high security:

Now change the DVWA security setting to High.

In your report, under the heading "Part 2C", briefly describe what has changed, in both the webpage display and the code on the backend? Also describe all the security techniques that have been implemented compared to the low and medium security setting. Comment especially on the "LIMIT 1" part of the SQL query on the backend; will that indeed limit multiple attacks from an attacker probing the database?

Once again, perform a SQLi attack against the application to obtain all usernames and passwords from the database. Attach screenshots of your query and the results obtained and write a short description in your report of how you did it.

Part 2D: SQLi attack under impossible security:

Now change the DVWA security setting to Impossible. This is meant to be impossible to attack, so you already know that you cannot perform a successful SQLi attack in this setting. Look at the source code and understand why it is secure. In your report, under the heading "Part 2D", describe your understanding.