Lab – 10 – Assessed Lab II

INFO3165

Proverbs 13:4

A sluggard's appetite is never filled,

but the desires of the diligent are fully satisfied.

The goal of this lab is to develop a Python script using Selenium that tests a webpage for common web vulnerabilities, including HTML Injection, Command Injection, SQL Injection, and XSS Attacks. The script allows users to choose which vulnerability to test for and reports whether the page is vulnerable or not. If a vulnerability is found, it provides details on how to exploit it and the specific location of the vulnerability.

Requirements:

1. **Python**: Ensure that Python 3.x is installed on your system.
2. **Selenium**: Install the Selenium library for Python to automate browser actions.
3. **WebDriver**: You will need the Chrome WebDriver to interact with Chrome in the Selenium script.
4. **OWASP DVWA**: For testing vulnerabilities such as Command Injection and SQL Injection.
5. **bwapp**: For testing XSS vulnerabilities.
6. **Chrome with Remote Debugging**: Run Chrome with a remote debugging port to interact with it using Selenium.

Your Script should test for the following vulnerabilities:

1. **HTML Injection:** You can test this on the following page: Go to *OWASP 2013 → A1 Injection (Other) → HTML via DOM Injection → HTML5 Storage.*
2. **Command Injection:** Test using *Damn Vulnerable Web Application (DVWA).*
3. **SQL Injection:** Test using *Damn Vulnerable Web Application (DVWA).*
4. **XSS Attacks:** Use *bWAPP* to test both *Reflected* and *Stored XSS.*

For this lab, you *might* find it useful to run Chrome on a specific port and access that instance using Selenium:

```
chrome_options = Options()

chrome_options.debugger_address = "localhost:9222"  # If attaching to a remote session
```

To run Chrome on a specific port for remote debugging, use the **`--remote-debugging-port`** command-line switch followed by the desired port number. Here's a more detailed breakdown:

Command-Line Switch:

Use the **`--remote-debugging-port`** switch when launching Chrome to enable remote debugging.

Port Number:

Choose a port number that is not already in use by another application. Port **`9222`** is commonly used, but any non-reserved port will work.

Examples:

To run Chrome on port **`9222`**:

```
chrome --remote-debugging-port=9222
```

To run Chrome on port **`8080`**:

```
chrome --remote-debugging-port=8080
```