| Cybersecurity |
| --- |
| Module 15 Challenge Submission File |

## Testing Web Applications for Vulnerabilities

Make a copy of this document to work in, and then respond to each question below the prompt. Save and submit this completed file as your Challenge deliverable.

### Web Application 1: *Your Wish is My Command Injection*

Provide a screenshot confirming that you successfully completed this exploit:

|  |
| --- |

Write two or three sentences outlining mitigation strategies for this vulnerability:

| One way you could mitigate the command injection vulnerability is to limit user input when calling for files from the web application. Also, you could mitigate this potential attack by implementing input validation to limit the user’s ability to modify the file. Lastly, running web servers under a special service user account that only has access to the web folder is another good way to mitigate command injection. |
| --- |

### Web Application 2: *A Brute Force to Be Reckoned With*

Provide a screenshot confirming that you successfully completed this exploit:

|  |
| --- |

Write two or three sentences outlining mitigation strategies for this vulnerability:

| When mitigating a brute force attack, you could make it so you are required to have complex usernames and passwords to gain access into the system. Making the process more tedious by having users include special characters or capital letters mixed with lower-case letters in the username and password will hinder attackers from achieving progress. Another good way to mitigate this attack would be to use a multi-factor authentication that requires users to have a password and a secondary form of authentication. Locking out accounts after a number of failed attempts is a great strategy too because if a user only has three attempts before being locked out, it will make gaining access even more difficult if the correct password is not guessed in time. |
| --- |

### Web Application 3: *Where's the BeEF?*

Provide a screenshot confirming that you successfully completed this exploit:

| Social Engineering> Pretty Theft:        Social Engineering> Fake Notification Bar:    Host> Get Geolocation (Third Party): |
| --- |

Write two or three sentences outlining mitigation strategies for this vulnerability:

| The most common way to mitigate cross-site scripting is the use of input validation, which is used to authenticate the data input with a predefined logic to ensure that the input is what the application is expecting. Using client-side input validation, a malicious user would not be allowed to input scripts, while using server-side input validation, a malicious user would not be able to store scripts on an application’s web server. |
| --- |

© 2023 edX Boot Camps LLC. Confidential and Proprietary. All Rights Reserved.