Cybersecurity

## Activity 2.3.1 Securing E-Commerce Data

Copy and paste screenshots and/or answer questions from the activity.

5. Explain what you see when the page loads in the browser, and tie each observed

behavior to the corresponding line of code.

Alerts the user with the alert function and then some text is displayed in the body using the paragraph tag

6. What do you think would happen if you load this web page in a browser, enter the name

jdoe, and click the button?

It will say it back with some other text

What do you think would happen if you click the button without entering a name?

Just empty text

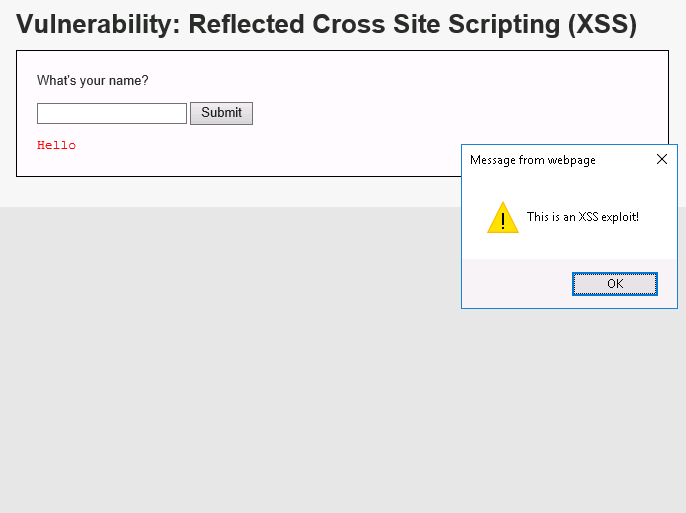
7. Double-click the file HelloWorldButton.html to render the HTML in the web browser. Did

it behave the way you thought it would?

Yes

14. Document the XSS reflected exploit script and then describe what you observed. Save a

screenshot of the results.



15. Document the XSS reflected exploit script and then describe what you observed. Save a

screenshot of the results.

Graphical user interface, text, application, email

Description automatically generated

17. Describe what you observed with security set back to High.

Graphical user interface, text, application, email

Description automatically generated

20. Read through the code of each, focusing on the line containing echo 'Hello'.

Select/highlight the few lines of code that prevent the XSS reflected exploit (but not the

entire script) and save a screenshot of the script.

High: echo 'Hello ' . htmlspecialchars($\_GET['name']);

Graphical user interface, text, application, email

Description automatically generated

26. Type the number 1 in the User ID text box and click Submit.

What do you observe? Predict what columns (categories of data) are in the users table.

Text

Description automatically generated with medium confidence

29. Document this first SQL injection exploit code and then describe what you observed.

Text

Description automatically generated with medium confidence

Save a screenshot of the result.

30. Observe the results after the first authentic result (where user\_id is 1). Document the

second SQL injection exploit code and then describe what you observe.

Save a screenshot of the first page of the results.

Table

Description automatically generated with low confidence

31. Document the third SQL injection exploit code and describe what you observe.

Save a screenshot of the result

Table

Description automatically generated with medium confidence

33. Describe what you observe when you set the site security level to High.

Putting in 1’ or 1’ OR 1=1 does nothing

36. Select/highlight the few lines of code that prevent the SQL injection exploit (but not the

entire script) and take a screenshot of the script.

$id = $\_GET['id'];

$id = stripslashes($id);

$id = mysql\_real\_escape\_string($id);

CONCLUSION

2. Data cleansing is a common protection measure that companies implement today. Do

you think it’s acceptable to try out XSS and SQL injection for fun thinking it won’t harm

anyone? Explain your thinking.

No, if they are monitoring traffic you could get in big trouble for stealing those credit cards