Developing and Assessing a Web-Based Interactive Visualization Tool to Teach Buffer Overflow Concepts

Jinghua Zhang\textsuperscript{1}, Xiaohong Yuan\textsuperscript{2}, Jaris Johnson\textsuperscript{1}, Jinsheng Xu\textsuperscript{2}, Mounika Vanamala\textsuperscript{2}
\textsuperscript{1}Department of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
\{zhangjij\}@wssu.edu
\textsuperscript{2}Department of Computer Science
North Carolina A&T State University
Greensboro, NC, USA
\{xhyuan, jxu\}@ncat.edu

Abstract—This Innovative Practice Full Paper presents a new way to teach buffer overflow concepts. Historically, buffer overflow has been the number one security vulnerability in applications for many years. More recently, advances in protection methods including non-executable stack, canaries, ASLR, and Windows DEP have made buffer overflow attacks a much smaller security concern, but they are still a serious issue in embedded systems and micro-controllers. Therefore, it is still very important to teach students this topic. There are several tools available for teaching buffer overflow attacks, but there are no easily accessible interactive teaching tools to help students understand the concepts. We developed a web-based interactive visualization tool that aims to help students gain a deeper understanding of buffer overflow concepts. There are six learning components that build upon one another as well as an assessment after each component for immediate learning feedback. There is also a space shooter mini-game between each learning component. To evaluate the impact of this online visualization tool on students' learning, we developed in-game assessments, a pre-test, a post-test and a survey. This tool was used in two classes at Winston-Salem State University (WSSU) and North Carolina A&T State University (NC A&T) in Fall 2019. The classroom experience reports and focus group discussion show that this tool helped students improve their understanding of buffer overflow concepts.

Keywords—Cybersecurity; Visualization; Buffer Overflow

I. INTRODUCTION

Cybersecurity education is increasingly critical as we spend more of our everyday lives online. Buffer overflow is an important topic in the discipline of computer science, more specifically cybersecurity. A buffer overflow occurs when a program puts more data in a buffer than it can hold. The memory allocated may contain a string of characters, an array of integers, or some combination of both. Writing past the bounds of allocated memory can crash a program, cause data corruption, or allow for the execution of malicious code [9].

Historically, buffer overflow has been the number one security vulnerability in software applications. Although buffer overflow’s impact on code injection on the stack is diminished through many protection mechanisms, programmers are still making mistakes when using buffers and this can still cause significant damages. By teaching buffer overflow to students, we can teach them the importance of input validation and buffer boundary checking. Additionally, students can learn several important concepts, including how the call stack works, safe vs. unsafe libraries, safe vs. unsafe programming languages, integer overflow, and various protection methods.

In recent years, many efforts have been made aiming to enhance cybersecurity education through gaming [2, 3, 6, 7, 8, 12]. Most work related to the buffer overflow concepts is focused on detecting and prevention techniques [1, 4, 9]. For example, Chen and Mao aim to increase the effectiveness of manual buffer overflow detection with a game [1]. The effects of buffer overflow can be shown through the use of media; in this case, the interactive medium of visualization tools. Visualization tools with educational components can be very useful for higher levels of engagement in content and retention of information. There are no online game/visualization tools available to help students understand the effects of buffer overflow. Therefore, we developed a web-based interactive visualization tool that aims to help students gain a deeper understanding of buffer overflow concepts. There are six learning components with lessons and examples that build upon one another as well as a short quiz after each component for immediate learning feedback. There is also a space shooter game between each learning component. The buffer overflow online visualization tool has four learning objectives:

1) Be able to describe how buffer overflow happens.
2) Be able to explain the effects of buffer overflow.
3) Be able to describe the effects of overwriting return address with buffer overflow.
4) Be able to list countermeasures against buffer overflow.

To evaluate the impact of this online visualization tool on students' learning, we developed in-game assessments, a pre-test, a post-test and a survey. This tool was used in two classes at WSSU and NC A&T in Fall 2019. The classroom experience report and focus group discussion show that this tool did help students improve their understanding of buffer overflow concepts.

Section II provides a more in-depth description of the learning component design and development. Section III presents the detailed assessment results. Section IV describes the conclusion and future directions of this research.
II. LEARNING COMPONENT DEVELOPMENT

This web-based interactive visualization tool was developed using the Unity game engine. All scripts were written in C# programming language using Visual Studio Community IDE. Currently, this tool was built to the WebGL format through Unity and uploaded to a web server to be played as an online game. It can be deployed into multiple other platforms as well, such as PC and Mac.

Collecting and analyzing the data we receive from players provide us with powerful insight on the effectiveness of the buffer overflow tool on students’ learning. We used GameSparks [5] to manage player profile/data on the cloud. GameSparks is a game data service that provides a variety of options, such as game scaling, player management, marketing, and segmentation & analytics. We utilized Unity API offered by GameSparks to integrate, set up analytics dashboard, do player registration and validation as shown in Fig.1, and collect player game data. It also provides the ability to create leaderboards, which will be applicable to the high scores received from the Space Shooter game in the future.

A. Learning Component #1 - Call Stack Illustration

The first learning component is to introduce the basic concept of a stack structure and illustrate call stack with examples. It was important to start with this concept to give students a visual idea of what a stack is and what it consists of before data is copied into it. Player will understand the basic concepts such as memory, return addresses, frame pointers, parameters, and variables.

Upon reviewing the call stack, the player is prompted to begin their first quiz to help gauge their immediate understanding of the concepts presented. Each quiz consists of only 2-3 multiple choice questions presented in a dropdown menu format. Once the quiz has been completed, the player is transitioned to a Space Shooter minigame. The objective of each in-between game is slightly different. It is related to what has been learned in the previous component.

B. Learning Component #2 - Simple Buffer Overflow

The second learning component gives a simple buffer overflow example and is carried out as two tasks. First, the player enters their buffer size and input characters to see the result of characters copied into the buffer without going past the bounds of the buffer size as shown in Fig.2. This presents the opportunity to directly affect the outcome that is visualized on screen, giving the player a better sense of control while processing the information.

After completing both tasks, the player is presented with the next quiz shown in Fig.4 to reinforce the new information and earn coins for the next Space Shooter level. The goal of this space shooter game level shown in Fig.5 is to shoot the falling rocks with letters to fill the buffer within the time limit. The coins earned from each correct answer in the previous quiz may be used as an extra life to restart the game. Every coin that is unused will provide a 100-point boost in total score.

![Fig. 1. Player registration on the Welcome Screen](image1)

![Fig. 2. Simple Buffer Overflow Task 1: No Overflow](image2)

![Fig. 3. Simple Buffer Overflow Task 2: Buffer Overflow](image3)

![Fig. 4. Quiz for Learning Component #2](image4)
C. Learning Component #3 - Overwriting a Variable

The third learning component demonstrates how buffer overflow can change the value of a variable in memory. For this component, the player enters their buffer size and input, just as previously done in component #2. However, this time, there is an integer value in memory as well. Similar to the second component, this component contains two tasks: copy string input into the buffer without altering the integer value shown in Fig. 6 and copy string input that causes buffer overflow to change the integer value from 15 to 102 shown in Fig. 7.

After completing both tasks, the player is presented with a new quiz to receive learning feedback and earn coins for the next Space Shooter level. Beginning with this level, the Space Shooter game introduces a new obstacle: enemy ships. These ships come in progressively larger waves in the following levels.

D. Learning Component #4 - Overwriting a Return Address

The fourth learning component demonstrates buffer overflow causing a segmentation fault when overwriting a return address. This component differs from the previous ones. The player no longer enters a buffer size. Now the buffer size is set to 8 bytes by default, while the player still enters their string input. Additionally, there is a return address that is 4 bytes that the player must overwrite to produce a segmentation fault, which causes a program to crash, as shown in Fig. 8.

After successfully causing the segmentation fault, the player is presented with the next quiz to receive learning feedback and earn coins for the next Space Shooter level.

E. Learning Component #5 - Redirecting

Similar to the previous component, the fifth learning component presents the player the challenge of overwriting the f1 return address. However, in this instance the goal is to overwrite it with another address, f2, to print the message “Hello World” by inputting the correct string sequence. A hex editor may be used to accomplish the task. Initially, the player will try to use keyboard input to see if it is possible to accomplish the task in that manner shown in Fig. 9 and Fig. 10. After seeing the result, they are then presented the option to choose a file that can properly overwrite the f1 return address to cause a hop to function f2 as shown from Fig. 11 to Fig. 13. The f2 function has the cout statement, which prints “Hello World”. This demonstrates how a buffer overflow
attack could be used to redirect a pointer to an address containing malicious code.

This component is the culmination of all the lessons before it, combined with the new challenge of using a hex editor to create the correct address for f2. After completing this component successfully, the player is presented with the final quiz to receive learning feedback and earn coins for the final Space Shooter level.

Fig. 9. Overwrite f1 Return Address with f2 Address - Read from Keyboard

Fig. 10. Overwrite f1 Return Address with f2 Address - Read from Keyboard

Fig. 11. Overwrite f1 Return Address with f2 Address: Task 2 - Read from File

Fig. 12. Overwrite f1 Return Address with f2 Address: Choose Hex File

Fig. 13. Overflow f1 Return Address with f2 Address: Task 2 - Hello World!

F. Learning Component #6 - Countermeasures

The final component of this visualization tool is a list of countermeasures to help protect and prevent from buffer overflow attacks or from developing inadvertent programs vulnerable to buffer overflow attacks. This brings together the player’s newfound understanding of possible causes and effects of buffer overflow with ideas on how to mitigate those causes as well. This concludes the game, followed by an end screen congratulating the player on completing the interactive visualization experience and presenting with their final Space Shooter game score.

III. CLASSROOM EXPERIENCE REPORT

In Fall 2019, this web-based buffer overflow interactive visualization tool was tested in a senior-level class (Organization of Programming Languages) at WSSU and in a graduate class (Network Security) at NC A&T. Twenty students (3 female students, 17 male students) participated in this study at WSSU. 18 out of 20 students were minority students. 21 students (9 female students, 12 male students) participated in this study at NC A&T.

Students did the pre-test, played the game, did the post-test and completed a survey in one sitting. Students took about an hour to complete all the activities. Overall, they enjoyed this type of learning compared to the traditional lecture. According to the survey, 90.5% of the students at WSSU and 83.3% of the students at NC A&T agreed that the learning objectives of the Buffer Overflow visualization tool are met.

A. In-Game Quiz Assessment

There is an in-game quiz after each learning component, each with 2-3 multiple-choice questions shown in the Table 1. Each quiz received 20 responses at WSSU. Fig.14 shows the percentage of students who answered a question correctly for each question in the five quizzes at WSSU. Each quiz received 21 responses at NC A&T. Fig.15 shows the percentage of students who answered correctly for each question.
According to the survey, 45% of the students have taken cybersecurity courses for academic credits. Some of the courses taken are Intro to Information Security, Intro to Cryptography, and Hardware and Media Security. 10% of the students have received formal training on common cybersecurity practices via IBM employee cybersecurity training. The students were asked to rank their agreement on a list of statements on a 5-point Likert scale (where 5 means strongly agree, and 1 means strongly disagree) on their motivation and experience in learning with the buffer overflow game. The detailed results are presented in Table 2.

Table 2. Motivation/Experience in Learning with the game module (WSSU)

<table>
<thead>
<tr>
<th>Field</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning experience with the Buffer Overflow game is interesting.</td>
<td>4.33</td>
</tr>
<tr>
<td>Motivated in learning buffer overflow using the Buffer Overflow game.</td>
<td>4.33</td>
</tr>
<tr>
<td>The Buffer Overflow game engaged me in learning the concept of buffer overflow.</td>
<td>4.33</td>
</tr>
<tr>
<td>I wish more cybersecurity topics will be taught using games.</td>
<td>4.50</td>
</tr>
<tr>
<td>The Buffer Overflow game helped me to better understand the topic.</td>
<td>4.25</td>
</tr>
<tr>
<td>I am interested in learning about the topic of buffer overflow.</td>
<td>4.58</td>
</tr>
<tr>
<td>After using the Buffer Overflow game, I have more confidence in describing the concepts learned.</td>
<td>4.25</td>
</tr>
<tr>
<td>Enjoyed the learning experience with the Buffer Overflow game.</td>
<td>4.33</td>
</tr>
<tr>
<td>I think the learning experience with the Buffer Overflow game is effective.</td>
<td>4.16</td>
</tr>
<tr>
<td>I am satisfied with the level of effort the Buffer Overflow game requires for learning this topic.</td>
<td>4.25</td>
</tr>
</tbody>
</table>

According to the survey, 45% of the students have taken cybersecurity courses for academic credits. Some of the courses taken are Intro to Information Security, Intro to Cryptography, and Hardware and Media Security. 10% of the students have received formal training on common cybersecurity practices via IBM employee cybersecurity training. The students were asked to rank their agreement on a list of statements on a 5-point Likert scale (where 5 means strongly agree, and 1 means strongly disagree) on their motivation and experience in learning with the buffer overflow game. The detailed results are presented in Table 2.

According to the survey, 45% of the students have taken cybersecurity courses for academic credits. Some of the courses taken are Intro to Information Security, Intro to Cryptography, and Hardware and Media Security. 10% of the students have received formal training on common cybersecurity practices via IBM employee cybersecurity training. The students were asked to rank their agreement on a list of statements on a 5-point Likert scale (where 5 means strongly agree, and 1 means strongly disagree) on their motivation and experience in learning with the buffer overflow game. The detailed results are presented in Table 2.

Table 2. Motivation/Experience in Learning with the game module (WSSU)

<table>
<thead>
<tr>
<th>Field</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning experience with the Buffer Overflow game is interesting.</td>
<td>4.33</td>
</tr>
<tr>
<td>Motivated in learning buffer overflow using the Buffer Overflow game.</td>
<td>4.33</td>
</tr>
<tr>
<td>The Buffer Overflow game engaged me in learning the concept of buffer overflow.</td>
<td>4.33</td>
</tr>
<tr>
<td>I wish more cybersecurity topics will be taught using games.</td>
<td>4.50</td>
</tr>
<tr>
<td>The Buffer Overflow game helped me to better understand the topic.</td>
<td>4.25</td>
</tr>
<tr>
<td>I am interested in learning about the topic of buffer overflow.</td>
<td>4.58</td>
</tr>
<tr>
<td>After using the Buffer Overflow game, I have more confidence in describing the concepts learned.</td>
<td>4.25</td>
</tr>
<tr>
<td>Enjoyed the learning experience with the Buffer Overflow game.</td>
<td>4.33</td>
</tr>
<tr>
<td>I think the learning experience with the Buffer Overflow game is effective.</td>
<td>4.16</td>
</tr>
<tr>
<td>I am satisfied with the level of effort the Buffer Overflow game requires for learning this topic.</td>
<td>4.25</td>
</tr>
</tbody>
</table>

According to the learning objectives analysis at WSSU, 90.5% of students strongly agree or agree that the learning objectives of the buffer overflow game are met. Before using this tool, students were asked to complete a self-ranking of their knowledge related to the learning objectives of the buffer overflow game. They were asked to fill out a post-survey on the learning objectives after using the game. Table 3 shows students self-ranking of the learning objectives in the pre- and post-survey. There are 5 questions in the Pre-Quiz and Post-Quiz. All questions are equally weighted (1 point each). The mean is calculated and tabulated. From the pre-test and post-test results shown in Fig.17, we can see that the difference between the pre-test and post-test results at WSSU is small.
Survey Analysis at NC A&T

A pre-survey is conducted for the buffer overflow tool and the students’ cybersecurity experiences are analyzed. 20 students participated in the pre-survey and 15 students participated in the post-survey at NC A&T. During the pre-survey, Attitudes/Behavior about cybersecurity of the students are analyzed, which are presented in Fig. 18.

According to the survey, 80% of the students have taken courses for academic credits. Some of the courses taken are Network Security, Software Security Testing Information Security and Privacy, Secure Networking, Disk Forensics, Secure Web, and Web Security. 20% of the students have received formal training on common cybersecurity practices from software security courses at NC A&T such as Network Administration and Network Communication, and at conference workshops. The survey results on students’ motivation and experience in learning using the buffer overflow game at NC A&T are presented in Table 4.

According to the learning objectives analysis at NC A&T, 83.3% of the students strongly agree or agree that the learning objectives are met. Table 5 shows students self-ranking of the learning objectives in the pre- and post-survey. From the pre-quiz and post-quiz results shown in Fig. 19, we can see that the students scored higher on average (3.27) when compared to the pre-test results.

<table>
<thead>
<tr>
<th>Field</th>
<th>Pre-survey Mean</th>
<th>Post-survey Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Describe how buffer overflow happens.</td>
<td>2.36</td>
<td>4.16</td>
</tr>
<tr>
<td>Understand the effects of buffer overflow.</td>
<td>2.52</td>
<td>4.00</td>
</tr>
<tr>
<td>Describe the effects of overwriting return address with buffer overflow.</td>
<td>2.47</td>
<td>3.91</td>
</tr>
<tr>
<td>List countermeasures against buffer overflow.</td>
<td>2.21</td>
<td>3.58</td>
</tr>
</tbody>
</table>

Table 3. Pre- vs. Post-Survey on Student Learning Objectives (WSSU)

<table>
<thead>
<tr>
<th>Field</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning experience with the Buffer Overflow game is interesting.</td>
<td>4.57</td>
</tr>
<tr>
<td>Motivated in learning buffer overflow using the Buffer Overflow game.</td>
<td>4.23</td>
</tr>
<tr>
<td>The Buffer Overflow game engaged me in learning the concept of buffer overflow.</td>
<td>4.33</td>
</tr>
<tr>
<td>I wish more cybersecurity topics will be taught using games.</td>
<td>4.33</td>
</tr>
<tr>
<td>The Buffer Overflow game helped me to better understand the topic.</td>
<td>4.38</td>
</tr>
<tr>
<td>I am interested in learning about the topic of buffer overflow.</td>
<td>4.38</td>
</tr>
<tr>
<td>After using the Buffer Overflow game, I have more confidence in describing the concepts learned.</td>
<td>4.14</td>
</tr>
<tr>
<td>Enjoyed the learning experience with the Buffer Overflow game.</td>
<td>4.42</td>
</tr>
<tr>
<td>I think the learning experience with the Buffer Overflow game is effective.</td>
<td>4.47</td>
</tr>
<tr>
<td>I am satisfied with the level of effort the Buffer Overflow game requires for learning this topic.</td>
<td>4.38</td>
</tr>
</tbody>
</table>

Table 4. Motivation/Experience in Learning with the game module (NC A&T)

Figure 18. Cybersecurity experiences of students at NC A&T

According to the survey, 80% of the students have taken courses for academic credits. Some of the courses taken are Network Security, Software Security Testing Information Security and Privacy, Secure Networking, Disk Forensics, Secure Web, and Web Security. 20% of the students have received formal training on common cybersecurity practices from software security courses at NC A&T such as Network Administration and Network Communication, and at conference workshops. The survey results on students’ motivation and experience in learning using the buffer overflow game at NC A&T are presented in Table 4.

According to the learning objectives analysis at NC A&T, 83.3% of the students strongly agree or agree that the learning

Focus Group Discussion

After using the web-based buffer overflow visualization tool, a group of 5 students were selected to take part in a focus group session at two universities. Collectively, they were asked a series of questions regarding the game and their experience playing it. From the focus group discussions at WSSU and NC A&T, there were some points that emerged, which are presented below.

- The students liked the game and the concept of using games for learning.
The buffer overflow game module was regarded as being “pretty cool” and “interactive”. The game was also described as “fun and addictive”. 

The interactivity of the tool, as opposed to a lecture, was the most enjoyable aspect. More specifically, the ability to input your own characters into the buffer to see if an overflow was caused was a highlight feature of the module.

Having the Space Shooter game break up the learning components was “a nice break” between consuming new information.

A visual representation of that topic created interest and made it easy to understand when compared to reading textual description of the topic.

Before studying this topic, they had no idea about buffer overflow and the importance of it. After this module they were able to define what buffer overflow and code injection is and how it is done.

Though they liked the idea of learning through gaming they have some feedback that could increase the engagement of the student in the topic that are presented below.

- In the learning section, students were interested to study about some real-life attack scenarios performed using buffer overflow, which could help them understand the importance of this topic and also its real-life application.
- It would be beneficial to have a function that shows which question(s) was answered wrong. With that, having a tip explaining why the answer was wrong would be helpful as well.
- Regarding the space game sections, the students agreed that having more upgrades for the ship or an extra life multiplier if you destroy a given number of asteroids/enemy ships would be great.

In conclusion, students agreed that this buffer overflow game module/tool indeed helped them understand the concepts better.

Furthermore, we also look to make it accessible using the UI Accessibility Plugin for Unity. Another improvement we want to implement is allowing the player to restart the level instead of restarting the game when necessary, such as when the game crashes or when competency has been shown in previous levels but not the current level. Finally, we plan to run more extensive testing to ensure the best possible quality of this tool is delivered to all players.

ACKNOWLEDGEMENTS

This work is supported by NSF under the grant DUE-1821960 and 1821965. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF.

REFERENCES


