Raising Cybersecurity Resilience With Millionaire's Challenge

1st Dhani Kataria
Dept. of Electrical and Computer Engineering (MTIS)
University of Victoria
Victoria, Canada
dhanikataria@uvic.ca

2nd Hailay Teklemariam
Dept. of Electrical and Computer Engineering (MTIS)
University of Victoria
Victoria, Canada
haitek@uvic.ca

3rd Shreya Goyal
Dept. of Electrical and Computer Engineering (MTIS)
University of Victoria
Victoria, Canada
shreyagoyal@uvic.ca

4th Anuinder Sekhon
Dept. of Electrical and Computer Engineering (MADS)
University of Victoria
Victoria, Canada
anuindersekhon@uvic.ca

5th Vatsala Arora
Dept. of Electrical and Computer Engineering (MADS)
University of Victoria
Victoria, Canada
vatsala09@uvic.ca

***Abstract*—With all the advancements occurring in the IT world each year, cybercriminals are also devising new attack vectors making it difficult to predict how they will target and impact organizations. With the high rise in cyber crimes such as social engineering, phishing, ransomware, etc., it has become important that every individual is aware of possible cyber attacks. It is a common notion in the security world that most of the cyber attacks can be prevented by getting just the basics right. Thus, having an idea about the fundamentals of information security would tremendously help individuals in recognizing if they are getting caught up in the web of cybercriminals or not. Learning sometimes can be time-consuming, exhausting, and boring. Therefore, to overcome this, our team (Security Knights) has discovered a way to gamify the whole process of learning about cyber crimes and security. We have decided to develop a game that would explain and help the users know more about the cybersecurity world through a modified version of "Who Wants to be a Millionaire".**

***Index Terms*—Cybersecurity, Information Security, Millionaire Quiz, Gamification, Security Awareness, Adobe Captivate, Web Vulnerabilities, Section 508, WCAG, HTML5, Mobile Gestures**

I. Why Gamification is Important?

Students, children, millennials and even engineers, all are caught up and influenced by games in general. Gamification means incorporating game design aspects and ideas into non-game situations such as education, marketing, and staff training to make them more engaging. Research [1] shows that gamification could be considered a vital tool for creating sustainable higher educational institutions by improving underline motivation and performance. Gamified learning includes the aspects like high scores and virtual rewards which increases the players' engagement by motivating them with the achievements they earn. This is because people like to feel and see that they're actually making progress. Learning by playful means helps in better information retention which is yet another reason which shows why gamification can be considered as an essential part of any learning. Our game tends to target all these essential components of gamification, making the learning process of information security a fun and exciting journey for the user.

II. Tool for Game Development

The tool to be used for the game development is "Adobe Captivate" which is a kind of Digital Learning Solution by Adobe [2]. This is a GUI-based tool which allows the creation of e-learning content in the form of a quiz, online course etc. The tool allows the creation of content in the form of interactive slides. The creators can easily incorporate other media such as images, videos etc. into the slides to make the overall experience more immersive. The game should web-based and Adobe Captivate allows to author HTML5-based content without the need for any programming. Such ease of use has made this tool quite popular among large enterprises and it is no surprise that companies like Amazon, J.P. Morgan, and Boeing are using it for their Learning Management Systems [2].

III. Game Design

This section of the report will describe the various components of the game. The details would be divided into various subsections highlighting the game content and other features of the GUI which adds to the overall dynamics of the game. The motive of the project is to spread security awareness playfully so that users can grasp the much-needed cyber security concepts. Therefore, the dynamics of the game would be designed in a way such that it provides a captivating and intellectually stimulating experience to its players.

A. *Type of Questionnaire*

The game's questionnaire will cover various aspects of information security such as phishing, password security, device security, and IoT security. These topics are just the exemplary areas but the actual game will include a broad spectrum of questions covering multiple domains starting from physical security to implementing security at both network and software levels. The questions will not just be from the perspective of a potential victim, rather, they will cover the viewpoint of both an attacker and a defender. This will allow the players to think about how the attackers usually target their victims and how they can avoid getting exploited. Further, the underline focus of the game would be to aware the users regarding those cyber-attacks that are more prevalent in the IT world. Apart from teaching the users about various attacking and defensive techniques, the questionnaire may also test the players regarding the known best practices. Such knowledge is critical because there are always multiple ways to solve a security problem, however, there is only one best choice which can ensure the security posture of the underlying infrastructure.

B. *Interface Design and Features*

a) Presentation: To sustain the player's interest, the GUI of the game will be loaded with various features such that the user can easily interact with them. Fig. 1 [3] is an exemplary illustration depicting how players will be shown most of the questions. There will be 15 questions in total and for answering each question, corresponding virtual dollars are awarded. The reward amount increases along with increasing difficulty of questions after each correct answer. For each question, there would be 4 options out of which only one of the answers will be correct. The questions can be in form of text, images or a video which simulates the attack. The benefit of including the images or videos as part of the questions is that it would provide the users with much clarity regarding how these attacks may look in real life. For example, if the question is regarding phishing and the video shows the victim clicking on the phishing email, then, the player can at least have an idea from the video regarding what a phishing email looks like. Moreover, adding such media will also retain the player's interest in the game.

b) Assistance: For answering each of the questions, the interface will provide a “hint” or “context” to the user which will help the user in answering the questions. This is an important aspect of the game because the motive of the project is to create awareness among its players and not to test the player's pre-existing knowledge. To better understand the usage of these hints in the game, let us assume a scenario in which the player is presented with a screenshot of 4 sample emails. Among 4 of these, one of the screenshots represents a phishing email and the player needs to select that email as an answer. For answering this question, as a hint to the question, the player is provided with information in form of ”hint” regarding how to spot a phishing email. Based on this information, the player can then select the correct answer.

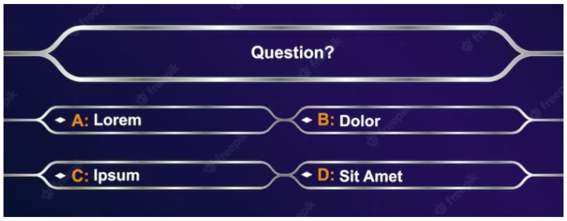


Fig. 1. Example of How Question will be displayed on the interface.

Apart from the hints, the interface includes the selection of lifelines which can assist the players in choosing the correct answer even if they are not able to select any option using the information provided as a hint. These lifelines would be one-time use and cannot be renewed. Following are some of the lifelines which the team has planned so far for the project:

- 50:50 – This lifeline will remove two of the wrong answers such that the user is left with two options on the screen. Among these two, one of the options is correct and the other is incorrect.

- Voting Booth – The lifeline provides the probability for each of the options to be correct. Higher probability represents more chances that the corresponding option will be correct. The user can choose the option which has the highest probability. The game would be developed in such a way that the logic automatically assigns the highest probability to the correct answer as compared to other options.

- Switcheroo – This lifeline will change the current question with some other question.

C. *Audio Design*

For creating a more immersive experience, the game will include different sound effects at different instances. Moreover, for each of the questions, audio samples can be added to the game which can dictate the questions and the corresponding options to the players. Such sound effects will create an atmosphere for users which will make the game more fun to play with.

D. *Educational Content*

The main motive for designing this game is to aware people regarding the importance of cybersecurity. Therefore,

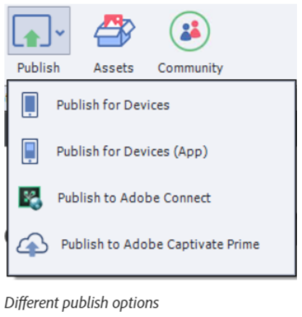

Publish
Assets
Community
Publish for Devices
Publish for Devices (App)
Publish to Adobe Connect
Publish to Adobe Captivate Prime
Different publish options

Fig. 2. Publish Options supported by Adobe Captivate.

the major aspect of the game is to educate the players such that they can realize and exercise best security practices in their day-to-day routine. Whenever the user answers any question, the game will provide a brief regarding why the correct option is correct and other options are wrong. This will improve the decision-making for the players if they encounter some security incident in future. As mentioned previously the game will be providing some context or hints for most of the questions. These hints may introduce the players to common technical terms that are being used in the cybersecurity industry. The knowledge of these terms will not only educate the users, rather it will provide a base for normal users to perform further research on the security concepts. Also, the focus of the questionnaire would be to teach users how they can recognize potential attacks and if they encounter such attacks, then, how can they prevent or mitigate them to reduce the involved risks.

IV. Accessibility

Another major aim of developing this project is that the game should be accessible on most of the common platforms. This will increase the reach of the game so that more and more people can be educated. Following subsections would provide a brief regarding some of the important features which would improve the accessibility of the game.

A. Compatible Platforms

The game would be web-based (HTML5), therefore, it will be accessible on both mobile and personal computers. Fig. 2 [4] shows the "Publish for Devices" option in the "Adobe Captivate" tool. Selecting this option provides the publisher with multiple options including support for mobile gestures. As most smartphones have touchscreen displays, the publisher will need to enable support for mobile gestures to ensure game accessibility.

B. Compliance with WCAG

The game will be developed in such a way that it is accessible to all people including those with disabilities like visual impairment. To complete such accessibility requirements, Web Content Accessibility Guidelines (WCAG) [5] can be considered as a reference which provides a comprehensive list of measures that should be present in a web content. The WCAG provides guidelines at three different levels i.e., Level A, Level AA and Level AAA [5].

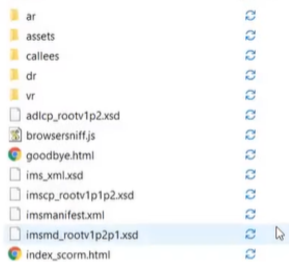
Adobe Captivate has an in-built feature which allows content creators to generate educational content which complies with section 508. As per the official web source [6], section 508 standards apply the WCAG 2.0 Level AA conformance requirements. Enabling the accessibility option in Adobe Captivate would provide content creators with different options that can be leveraged to design the game content which is in-compliant with WCAG Level AA requirements. Following are some of the exemplary features which would be implemented as part of the WCAG 2.0 guidelines [7]. Please note that, apart from these, the team would also be implementing other relevant guidelines into the final project.

- Each of the questions will be dictated by a pre-recorded voice. Therefore, captions will also be provided along with the video.
- There will be a dedicated button for audio control because as per the WCAG guidelines if the content includes audio of more than 3 seconds then a mechanism should be available through which the user can control the volume.
- All the non-text which is present in the game will have a text alternative.
- It will be ensured that the content displayed is available in both portrait and landscape modes. However, WCAG also allows the use of a single mode if either orientation does not suit the content.
- It will be ensured that there are no more than 3 flashes per second in the whole game. This will be beneficial for people with photo-sensitivity.

V. Publishing

a) On University of Victoria (UVic) Web server: As previously mentioned, Adobe Captivate [4] includes a built-in option to publish the developed content as HTML5-based web content. When the user clicks on the publish button, Adobe Captivate generates backend files on the local computer. Fig. 3 [8] provides a view of the backend files generated for some exemplary project. There are multiple folders and files including script files. The folder named "vr" contains all the video files and likewise, different folders can include other forms of media like images, audio clips etc. that have been used for building the game. If the UVic web server needs to host the game, then, all the generated backend files should be copied onto UVic's webserver and public access to index.html needs to be enabled. This way anyone who wants to play the game can play it on the UVic's hosted webpage.

b) On AWS Web server: Alternatively, there is another option to host the game online using AWS servers [9]. For



ar
assets
callees
dr
vr
adlcp_rootv1p2.xsd
browsersniff.js
goodbye.html
ims_xml.xsd
imscp_rootv1p1p2.xsd
imsmanifest.xml
imsmd_rootv1p2p1.xsd
index_scorm.html

Fig. 3. Backend files generated for an exemplary project.

that, all the backend files need to be copied onto the S3 storage on the cloud. The storage can then be integrated into a dedicated AWS web server for publically hosting the website. The AWS server will automatically generate a public link for the index.html file generated by Adobe Captivate. The admin of UVic's web server can just provide the public link on UVic's website rather than hosting the whole game on UVic's web servers. So whenever any user clicks the link on UVic's website, the user will be redirected to AWS servers where the user can start playing the game. The redirection is transparent to the user and does not require any technical expertise to access the game hosted on AWS server.

c) Recommended Choice: It is recommended that the game should be hosted on the AWS servers. This strategy has multiple advantages from the prospect of security. A complete security and privacy review of the game will be performed as part of the project, however, it is not feasible for the team to review the security using enterprise graded tools. Therefore, if the game is hosted on AWS servers and any security incident occurs, the private data of UVic's web server will not get affected.

VI. Security and Privacy Review

The game intends to provide the players with a playful and knowledgeable experience which will familiarize them with the basic concepts of cybersecurity that they can practice in their day-to-day routines. The motive of the game is to teach how users can protect their data and privacy. Therefore, it would be quite ironic if the game web application itself is vulnerable. Thus, one of the critical aspects of game development would be that the game itself should be secure and follows all the required best practices. The security measures of the application need to be reviewed and evaluated even before the game development starts [10]. The reason being that it is always a good practice to evaluate security from the get-go. In other words, all the security aspects should also be carefully considered during the planning stages of the project. Ensuring this will take online players more in confidence and will motivate them to play the game. Keeping all of this in mind, the team will ensure that the game is immune to most of the known web vulnerabilities.

The foremost emphasis is to be given to the development tool i.e., Adobe Captive. Online research on Common Vulnerabilities and Exposures (CVE) [11] could offer us knowledge regarding which versions of the Adobe Captivate are vulnerable or patched. For instance, the CVE database discloses a vulnerability CVE-2017-3098 [12] related to Adobe Captive version 9 and earlier. The attacker can leverage this vulnerability to execute a remote code execution attack, especially in the quiz reporting feature of Adobe Captivate. The successful exploitation of this vulnerability can lead to random reading and writing of files on the host server. Therefore, the team will ensure that a new release with an automatic update of Adobe Captive would be chosen as the hosting and development platform for this project.

In addition to the CVE vulnerability database, the Open Worldwide Application Security Project (OWASP) [13] foundation guidelines will also be considered as a benchmark for securing the application. Hosting web-based game applications for cyber security awareness can prove to be really effective because of ease of access and independence from location, time, and device compatibility. Even though hosting a web-based game application has various advantages, there can still be instances where users might not adhere to policies and recommendations presented by the owners of the application. This issue could be categorized as malicious intent or security ignorance. External attackers can exploit either the frontend or the backend of the web application by identifying weaknesses such as bad configurations or coding practices. This can result in gaining unauthorized access by the attacker, exposure of sensitive data or compromised functionality of web applications in general. Therefore, before hosting the game on web server, the team will perform a thorough manual penetration test of the application which will help us in identifying any of the common web-based vulnerabilities. Following are some of the most common vulnerabilities which are likely to be present in any new web application:

- Cross-Site Scripting (XSS): These attacks [14] are performed by injecting malicious code into user input fields. Therefore, if there are any input fields present in the final game, anything entered by the user into those fields needs to be validated against special characters vetted as string text variables.

- Directory Traversal: This attack [15] is quite common for web applications. The attacker can manipulate the GET parameters in the URL with ../../ to gain unauthorized access to arbitrary files on the host server. Therefore, if the developed game accepts any GET parameter, it will be ensured that each request data is validated and sanitized.

- Broken authentication and session management: Such attacks [16] take place when a website is not properly implemented with authentication and session management procedures which could lead to hijacking

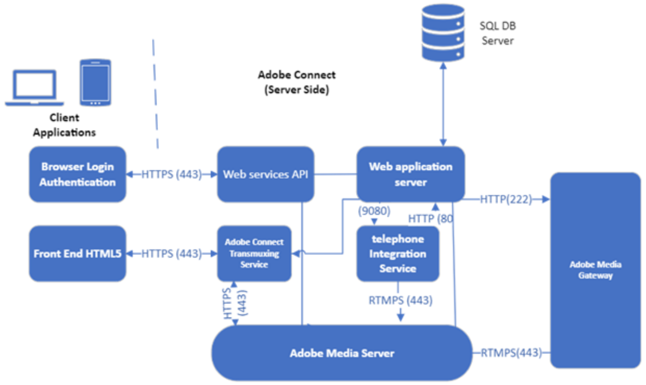


Fig. 4. Data Flow Architecture *(Created in Microsoft Visio [20])*

or stealing user credentials.

- File inclusion: This vulnerability [17] occurs when the attacker includes files from the local file system that trigger arbitrary remote code execution.

The above-mentioned are only some of the common vulnerabilities for which manual pen-testing can be performed. Apart from this, we can use tools like OWASP ZAP [18].

VII. Maintenance and Manageability

Most cybersecurity-related training and awareness programs are based on strict procedures and rules which rely on boring technical jargon. Adobe Captivate is a convenient e-learning and interactive media content development tool with automatic HTML5 generation capability. The application GUI is quite user-friendly which makes it easy to maintain and manage the contents of the game. The process of creating functionalities and applications involves modular action models which makes it easy to import, export and publish. The underlying content creation is completely based on the slides which can be edited by even a non-technical person to update the questions in the game. Before hosting the updated game, a sample preview can be generated by the administrator with just a click of a button for simulating the final game. In addition, the content could be saved in different formats according to developer choice such as MP4, HTML5, XML, PPT and so on. With different format output capabilities, the end-users could access the content through mobile, tablet or PC.

The following excerpts will describe the data flow for the content created using Adobe Captivate. The flow would be described using Adobe Connect on the server side. In actual implementation by UVic, Adobe Connect would be replaced by UVic's infrastructure. Adobe Captivate content can be published in three different ways i.e., on the Local Computer, Adobe Connect or Adobe Learning Manager. Adobe Connect is Java based web application running on the Tomcat servlet engine [19]. The server duties include access control, security quotas, licensing, auditing along with other management functionalities. It provides the transcoding of media files such as PowerPoint presentations and audio. The application server manages requests and content transfer requests over HTTP or HTTPS connection.

Adobe Media Gateway integrates Adobe Connect with Session Initiation Protocol (SIP)/Realtime Transport Protocol (RTP) setup. This helps phone communication requests from users in playing the game inside Adobe Connect space. For storing user and scoring details, an embedded database engine inside Adobe Connect is used for reducing the complexity and costs of a stand-alone SQL server [20].

Fig. 4 shows that the data can flow from client to server and vice-versa. Data communication between the front-end and back-end applications is secured. A secure connection across the network or internet data flow is established using encrypted HTTPS (443) protocol instead of unencrypted HTTP (80). Client-side web browser requests content by sending a uniform resource locator (URL) over a secured HTTPS (443) connection. The web server replies to the requested content through a secured connection. After receiving a response from the web server, the client requests a secure connection to Adobe Media Server over RTMPS:443. Adobe Media Server then opens a secured persistent connection to Adobe Connect Media Streaming.

VIII. Conclusion

Technical and logical controls protect and defend against attacks originating from external sources. Most infrastructure and personal data breaches occur due to the carelessness and cluelessness of users. Educating and training the insiders or users of any infrastructure helps strengthen the security of any company which ultimately minimizes the number of targeted attacks. The project will be a web-based cybersecurity awareness game which will offer a learning experience through fun and entertainment which is also the most effective way to teach. The tool provides a reporting feature also, which when enabled catalogues the score of each user's attempt. Based on that organizations can further decide if a particular user needs more training or not.

References

[1] J. Espinosa, M.Abellan, A.Moreno et al.“Gamification as a Promoting Tool of Motivation for Creating Sustainable Higher Education Institutions.” *International journal of environmental research and public health* vol. 19,5 2599. 23 Feb. 2022

[2] Adobe.“Digital Learning Solutions - Adobe Captivate”. https://www.adobe.com/ca/products/captivate.html (accessed April 19, 2023).

[3] grebeshkovmaxim.“Who wants to be a millionaire?.intellectual game template background”. Freepik.https://www.freepik.com/premium-vector/who-wants-be-millionaire-intellectual-game-template-background_5487912.htm (accessed April 18, 2023).

[4] Adobe.“Preview and publish responsive projects”.https://helpx.adobe.com/ca/captivate/using/preview-publishing.html (accessed April 18, 2023).

[5] Wikipedia.“Web Content Accessibility Guidelines”.https://en.wikipedia.org/wiki/Web_Content_Accessibility_Guidelines (accessed April 19, 2023).

[6] Section508.“Applicability and Conformance Requirements for Developers”.https://www.section508.gov/develop/applicability-conformance/#::text=The%20Revised%20508%20Standards%20incorporate,must%20be%20accessible%20(scoping). (accessed April 18, 2023).

[7] World Wide Web Consortium (W3C).“How to Meet WCAG (Quick Reference)”.W3.https://www.w3.org/WAI/WCAG21/quickref/?versions=2.0 (accessed April 19, 2023).

[8] Paul Wilson.“Publish Adobe Captivate eLearning for Your LMS”.YouTube.https://youtu.be/mYjwsAnk80M?t=3745 (accessed April 18, 2023).

[9] Paul Wilson.“Upload Your Adobe Captivate eLearning Project to the Web”.https://www.youtube.com/watch?v=qtF4vPNRWu0 (accessed April 19, 2023).

[10] Tom Venables.“Security Engagement”.Turnkey Consulting.https://www.turnkeyconsulting.com/keyview/security-engagement (accessed April 19, 2023).

[11] CVE.“Common Vulnerabilities and Exposures”.https://cve.mitre.org/ (accessed April 19, 2023).

[12] CVE.“CVE-2017-3098”.https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-3098 (accessed April 19, 2023).

[13] CVE.“OWASP Top Ten”.https://owasp.org/www-project-top-ten/ (accessed April 19, 2023).

[14] KirstenS.“OWASP Top Ten”.OWASP.CrossSiteScripting(XSS) (accessed April 19, 2023).

[15] OWASP.“Path Traversal”.https://owasp.org/www-community/attacks/Path_Traversal (accessed April 19, 2023).

[16] OWASP.“A2:2017-Broken Authentication”.https://owasp.org/www-project-top-ten/2017/A2_2017-Broken_Authentication (accessed April 19, 2023).

[17] OWASP.“Testing for Remote File Inclusion”.https://owasp.org/www-project-web-security-testing-guide/v42/4-Web_Application_Security_Testing/07-Input_Validation_Testing/11.2-Testing_for_Remote_File_Inclusion (accessed April 19, 2023).

[18] OWASP.“OWASP Zed Attack Proxy”.https://www.zaproxy.org/ (accessed April 19, 2023).

[19] Adobe.“Adobe Connect Technical Overview”.https://helpx.adobe.com/adobe-connect/installconfigure/preparing-install-connect.html (accessed April 19, 2023).

[20] Microsoft.“Visio in Microsoft 365 – Diagram and Flowchart Creator”.https://www.microsoft.com/en-ca/microsoft-365/visio/flowchart-software (accessed April 19, 2023).