A Data-driven Security Game to Facilitate Information Security Education

***Abstract*— Many universities have started to teach information security to educate students on how to develop secure software and systems. One challenge of teaching information security is that the curriculum can easily be outdated, because new attacks and mitigation approaches arise. It is therefore necessary to provide software developers with methods and tools that is attractive (e.g., computer games) for self-study up-to-date information security knowledge during and after the university education. This paper presents an on-going study to develop an educational game to facilitate information security education. The game is developed as a single player Tower Defense (TD) game. The educational goal of the game is to teach developers, who are not security experts, how to choose proper mitigation strategies and patterns to defend against various security attack scenarios. One key benefit of our game is that it is data driven, meaning, it can continuously fetch data from relevant security-based online sources (e.g., Common Attack Pattern Enumeration Classification CAPEC) to stay up to date with any new information. This is done automatically. We evaluated the game by letting students play it and give comments. Evaluation results show that the game can facilitate students learning of mitigation strategies to defend against attack scenarios. Our future work will focus on improving the fun part and usability of the game to attract students to learn information security and at the same time have fun while playing the game.**

Keywords—Information security, serious game, game-based education

I. INTRODUCTION

Gamification has on many occasions proven itself as a useful tool for teaching [1]. It has also been tried within the domain of information security knowledge. Examples of popular games in this domain are Elevation of Privilege (EoP) [2], and Protection Poker (PP) [3]. However, the current security educational games are mostly multiple-player collectible card games, which require more than one person to play, and require high information security expertise and experience among the players. Such constraints could severely restrict who, when, and where these games can be played. In addition, new information about information security such as new vulnerabilities, attacks, and patterns are usually not updated in the card games like EoP [2] and PP [3]. As a result, current security educational games are limited as they do not provide up-to-date security information to the end-users. In addition, EoP [2] is reported to have adoption challenges when studied in a university setting such as the game dynamics, relevance of hints on the cards, and the time needed to play the game [4].

Our study aims to address these limitations by using data-driven security and educational computer game to teach amateurs and beginners in information security in a fun way. We hope this game will contribute to educate more people, especially software engineering students and developers, who have an interest in information security but lack an engaging and fun way to learn about it.

To develop such an educational game, we have focused on answering two Research Questions (RQs):

- RQ1: What are the functional and non-functional requirements for a single-player information security computer game to be fun and to provide educational benefits to the players?
- RQ2: How to make this game data-driven so that no or minimum human labor is needed to update or add new security information to make the educational content up to date?

To answer RQ1, we used questionnaire to collect a set of requirements for the game from 31 university students, who are studying or who are interested in information security. Based on the requirements, we implemented a prototype of the game named Data-driven Security Game (DdSG), which is a TD type game. DdSG is a single-player game. When the game starts, the assets (e.g., web clients, network, and servers) will show in the game view. Then various patterns of attacks will be generated randomly to attack the assets. The game player is expected to choose the proper mitigation patters or strategies, which are listed in the game view, and place the mitigation patterns or strategies on path of the attack in the game view before the attack reaches the asset to defend against the attacks. If no proper mitigation patter is chosen in time, the assets will be attacked, and the blood of the assets, which is initialized as 100%, will be reduced. If the blood of all assets is down to 0%, the game will finish. To address RQ2 on how to make the game data-driven, we organized the game entities of DdSG based on the Structured Threat Information eXpression (STIX) [5] concept. As the information of the CAPEC repository [6] is organized according to STIX, DdSG can refresh its game entities periodically through synchronizing with information in the CAPEC repository. To our knowledge, DdSG is the first information security computer educational game which can easily be played by a single amateur and beginner in information security. DdSG is also the first information security education game that can automatically extract information from security information repositions to update its education content. To evaluate the prototype of the game, we have asked 10 students from the initial 31 students who were available when the prototype was

ready to test DdSG. We evaluated the prototype of DdSG from learning impacts and usability aspects. Results show that students are positive about the learning outcome of the game, especially learning about the mitigation strategies. Since the game is still at the prototype stage, the usability and fun features of the game are found to need improvement.

The rest of this paper is organized as follows. Section 2 introduces related information security educational games. Section 3 describes the design and implementation of DdSG. Section 4 presents and discusses our evaluation results, and Section 5 concludes.

II. Related work

There are a number of games for teaching information security knowledge, such as EoP [2], PP [3], , Control-Alt-Hack [7], Cyber Threat Defender [8], and Open Web Application Security Project (OWASP) Cornucopia [9].

A. Elevation of Privilege (EoP)

EoP is a game suitable for 3-6 players and to teach and reason about threat modeling. The EoP game requires the model or architectural diagram of the system before play starts. There are 74 playing cards, divided into six suits based on the STRIDE threat mnemonic [10]. Each suit consists of cards numbered in similar way to normalize playing cards. The game is played by players taking turns playing one card. For every card that is played, the card is read out and the threat on the card is discussed. If a player cannot link the threat to the system, play proceeds. One point is awarded for a relevant threat on the card played. The game requires information security experts to be present, either as players themselves, or as facilitators.

B. Protection Poker (PP)

PP is intended to be played by a software development team during some form of iteration planning meeting. PP is to help teams already developing projects to reflect and plan their risks, rather than to teach information security knowledge. Gameplay consists of letting players discuss some potential security risk in their system, or to discuss a new feature they are planning to implement and its potential risks. After the discussion, all players vote with a certain amount of points on probability and impacts of the attacks to prioritize them. PP requires several players, and every player must at least be somewhat familiar with information security concepts to be able to discuss the risks. At least one information security expert needs to be present as a moderator or facilitator when playing PP.

C. Control-Alt-Hack

Control-Alt-Hack is a multiplayer role-playing tabletop card game, which usually needs to involve 3 to 6 players. The teaching goals of the Control-Alt-Hack game are to raise awareness of the importance of information security, to raise awareness about the wide spectrum of technologies where information security is relevant, and to highlight the diversity of attacks and the creativity of attackers.

D. Cyber Threat Defender

Cyber Threat Defender is a multiple-player collectible card game which usually needs 2 players. The game tries to introduce basic concepts of the security domain to very beginners of information security. The game is supposed to be easy to learn and easy to enjoy, while not introducing any hands-on skills or abilities in security. The educational content of the game is somewhat thin and will only give a superficial introduction to information security. It means that the amount of educational content is limited, and players will have "learnt everything" rapidly. The game makes good use of already popular game-types to attract and interest players.

E. Cornucopia

Cornucopia is also multiple-player collectible card came which usually needs 3 to 6 players. The teaching goal of the game is to introduce threat modelling ideas to the players. The target players are software development teams, preferably teams using agile and are comprised of several different roles. Cornucopia especially targets at web development teams creating e-commerce systems. To play this game, at least one player should have reasonable knowledge of web application vulnerability terminologies.

The current computer games for teaching information security usually need to have several players, and all or some players are expected to have decent information security knowledge to educate or guide other players. In addition, the security information of the games, which are usually presented in cards, is mostly static and need manual intervention to be updated to follow the up-to-date security knowledge. As many developers, who want to learn information security, may not have the opportunity to play a multiple-player card game with security experts and to learn from them, we want to develop a single-player computer game for amateurs and beginners in information security. The teaching goals of the game are:

- Game players can learn, identify and explain state of the art attacks that may be targeted at their applications, e.g., web applications.
- The game players can learn and choose proper mitigation patterns and strategies to defend against corresponding attacks.

III. Game Design and Implementation

As the targeted players of DdSG are amateurs or beginners of the information security, we collected requirements of the game by using questionnaire. The respondents are 22 university students who are taking the software security course, and 9 students, whom we know, are interested in learning information security knowledge but have not taken any related courses. The questionnaire included close-ended and open-ended questions, and the questions were divided into categories, namely background information of the respondents, respondents' experience of playing information security educational games, and their key expectations from DdSG. Before distributing the questionnaire, we introduced our preliminary high-level idea of DdSG to the respondents.

A. Key Expectations of DdSG

We hereby present some key findings from the answers of the questionnaire.

- 83.4% of the respondents would try the information security educational game we want to develop.

- The top three focus areas that we elicited as the most important include fun, intuitive gameplay, and real-world educational content. Fun means that the game should provide entertainment value to the users. Intuitive gameplay means that the game needs to be quickly and easily understood, and the UI needs to be intuitive and easy to use for all users in the target group. Real-world educational content means that the learning value, provided by the game needs to reflect the real-world state of information security knowledge. For the students who are taking the software security course, they rated educational content and ease of use higher than the fun factor.

B. Game Type and Framework Selection

To design DdSG, we have evaluated two game types, namely Role-Playing Games (RPGs) and Tower Defenses (TDs). In RPGs, the player controls the actions of one or more characters that they role-play as in one universe or another. This is often an imagined universe. RPGs are often connected to a rich lore and a greater narrative of the universe it takes place in. Popular themes within the genre are typically fantasy and science fiction.

The main idea of TDs is letting the player defend their base against enemies with some form of defenses they can place on the board. The base usually has a set amount of integrity or blood, that is gradually compromised when enemies can get past all the player's defenses and enter or attack the base. The success of the RPGs is heavily dependent on the designers' ability to write and implement an engaging story.

As we want to use DdSG to teach general information security knowledge rather than knowledge limited to a specific scenario or story, RPG may not fit our purpose. TDs however, match quite perfectly with the concepts in information security knowledge. TDs' defenses, bases, and enemies are easily compared to the concepts of information security, such as assets, attacks and mitigations.

C. Game Entities and its Data-Driven Design

To implement the TD type game for teaching information security, we translated the real-world information security concepts to DdSG game entities. To make DdSG data-driven, i.e., fetch the security information from online repository and convert the information to game entities, we implemented DdSG using client-server architecture, as shown in Figure 1. At the server side of DdSG, the assets, attack patterns, and mitigation pattern information are fetched from the CAPEC repository every 24 hours. The hardest part about implementing a completely data-driven implementation was parsing the unstructured data received from CAPEC. They have a strict structure as to what fields an attack pattern can contain. However, within the fields of CAPEC, the raw data is very unstructured and is not precise. To be able to parse and categorize this data to comprehensible entities for use in the game client, the server had to implement some form of categorization algorithm. The relationship between real-world information security concepts, the DdSG entities, and the CAPEC repository content are show in Table 1.

D. Implementation of Game Views and Game Play Options

DdSG is developed using the Unity game engine [9]. As shown in Figure 2, the initial game view presents the user with the game board, containing the injection vectors where attacks may enter the user's system client, network, and server. The initial game view also contains the assets of the users. The assets are randomly chosen from the procured game entities at the start of each match. A wave is the game's way of breaking up the incoming attacks into groups of random sized subsets that try to attack the assets with a set amount of time between them. The time between waves allows the user to implement mitigations and to prepare, to some extent, for the incoming attacks. During each wave, a random amount of attacks is spawned. The attacks, which are spawned, are chosen from a subset of possible attack patterns that are selected at the start of the match. After potential attack patterns are chosen, all course of actions that mitigate these attack patterns are made available to the user, as can be seen in the bottom middle of the game view in Figure 2. The player can click these mitigation options to implement a corresponding mitigation on the board. The placement of these mitigations is restricted by the green grid, as we can see in Figure 2, and are not allowed to be placed on the paths. DdSG is data-driven, which means that all interactable entities, such as mitigations, assets, and incoming attack patterns, are pulled from CAPEC. The user can hover over and right-click the entities to read the full source information from CAPEC.

All incoming attacks will try to get to one of the assets in game view. Once an attack reaches an asset, the attack will damage the asset. If the asset's blood reaches 0 %, it will be destroyed. This will prevent further attacks on the asset from spawning anymore. However, it will penalize the user by reducing score points. If all assets are destroyed, the user will lose, and the game will be over. To make DdSG fun to play, DdSG is made configurable, meaning that players can configure the kind of music, the sound effect, and the speed of the attack waves to use during play.

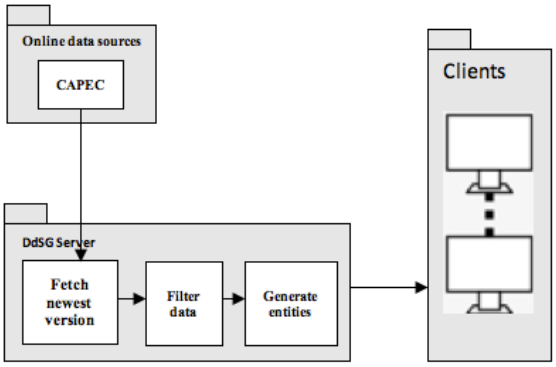

Fig. 1. High-level architecture of DdSG

TABLE I. RELATIONSHIP BETWEEN SECURITY CONCEPTS, GAME ENTITIES, AND CAPEC CONTENT

Information security knowledge concepts	DdSG Game entities	CAPEC concepts
Attack pattern Attack threat	Attack pattern	Attack Pattern Catalog
Mitigation solution	Mititgation or Course of Action	Mitigations Type
Asset System asset Activation zone	One out of three options, i.e., client, network, or server	Asset

IV. GAME EVALUATION AND DISCUSSIONS

To evaluate DdSG, we invited all students who participated in answering the questionnaire to pilot DdSG. Ten students accepted the invitation to evaluate DdSG. The evaluations included three parts. First, to test the usability of DdSG, we listed some tasks (e.g., start the game with 25% difficulty level and 1x game speed) and asked the students to finish the tasks, and then asked the students how easy the task was to perform. After that, we asked the students to play DdSG as much as they wanted, and then we asked them to rate their knowledge of certain aspects, e.g., attack patterns, assets, and mitigation patterns. In addition, we evaluated the game by using System Usability Scale (SUS) [12] and asked the students to give us feedbacks on any positive and negative aspects of DdSG. Over 80.0% students reported a significantly improved score of their mitigation knowledge and over 60% reported a significantly improved score of their attack pattern knowledge. The comments on DdSG can be boiled down to one of three categories, namely missing features, too simple mechanics, and information overload and balancing issues.

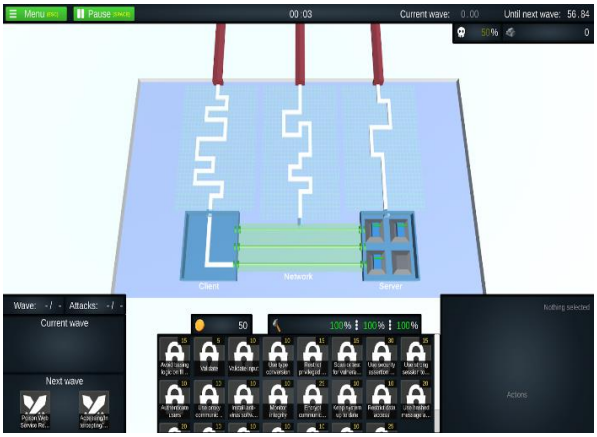

Fig. 2. Initial game view of DdSG

DdSG is at the prototype stage. We have not implemented many supporting features to help new beginner of the game to learn to play it and we have not made user manuals for the players in the evaluation. Thus, some players felt it was difficult to play the game and gave negative feedback on the usability of the game. The main purpose of the DdSG prototype is to test the idea of using data-driven single-player TD game to teach information security. Thus, the current version implements only features related to attack patterns and mitigation patterns. The limited functions may make the players feel bored after playing the game for a while. There are four types of fun gamers have while gaming [13]: 1) hard fun, which is about challenge and mastery; 2) easy fun, which is about imagination and exploration; 3) serious fun, which is about changing the player's internal state or doing real work; and 4) people fun, which is about social interaction. For the DdSG prototype now, we focused mainly on the hard fun and serious fun to let the players learn through thinking and playing. We did not cover social fun, as the game is designed as a single-player game. The easy fun is overlooked now, which might be the reason that players felt they were overloaded. In the future, the amount of information could be limited to a smaller subset at first, and then grows as the player progresses and becomes more familiar with the game itself and the educational content.

V. CONCLUSIONS AND FUTURE WORK

With today's ever-changing threat landscape, more and more developers need to learn information security to develop secure software. Attempts have been made to use gamification and educational games to engage and raise interest of information security. However, previous games have not been able to provide the fun and engaging successes needed to sufficiently raise the popularity of information security. We have identified requirements for creating a single-player TD game to teach information security and have implemented the prototype of the game. To ensure the security knowledge taught by the game is always up-to-date, a server application was implemented to fetch and parse the newest data from online sources like CAPEC at regular intervals. Testing the game with the targeted players revealed that the game concept is good, and the game prototype could help players increase their knowledge of information security after playing the game. As the game is still at prototype stage, we will need to put more effort to develop the game further to make it easy, fun, and more attractive to play.

REFERENCES

[1] Nah Fiona Fui-Hoon et al. "Gamification of Education: A Review of Literature," Springer LNCS vol. 8527, pp. 401-409, 2014.

[2] Adam Shostack, "Elevation of Privilege: Drawing Developers into Threat Modeling". In: USENIX Summit on Gaming, Games, and Gamification in Security Education, pp. 1–15, 2014.

[3] Laurie Williams, Andrew Meneely, and Grant Shipley, "Protection Poker: The New Software Security "Game"". In: IEEE Security and Privacy, vol. 8, no.3, pp. 14–20. DOI: 10.1109/MSP.2010.58, 2010.

[4] Inger Anne Tøndel et al., "Understanding Challenges to Adoption of the Microsoft Elevation of Privilege Game, " In Proc. of the 5th Annual Sympo. and Bootcamp on Hot Topics in the Science of Security. ACM, New York, NY, USA, Article 2, 10 pages, DOI: https://doi.org/10.1145/3190619.3190633, 2018.

[5] Sean Barnum, "Standardizing Cyber Threat Intelligence Information with the Structured Threat Information eXpression (STIX™)". In: MITRE Corporation, July, pp. 1–20. ISSN: 1011-6702, 2014.

[6] CAPEC repository. https://capec.mitre.org/.

[7] Tamara Denning , Tadayoshi Kohno , and Adam Shostack, "Control-Alt-Hack: the Design and Evaluation of a Card Game for Computer Security Awareness and Education". In Proc. of the 2013 ACM SIGSAC conference on Computer & communications security, pp. 915-928, DOI: 10.1145/2508859.2516753, 2013.

[8] Gregory B. White, "The Cyber Security Collectable Card Game (Version 1.0)," Tech. rep. The University of Texas at San Antonio. URL: http://cias.utsa.edu/ctd_rules.html, 2016.

[9] Colin Watson et al. "Cornucopia". In: OWASP Cornucopia Ecommerce Website Edition. URL: https://www.owasp.org/index.php/OWASP_Cornucopia, 2012.

[10] Dianxiang Xu et al. "Automated Security Test Generation with Formal Threat Models". In: IEEE Transactionson Dependable and Secure Computing 9.4, pp.526–540, DOI: 10.1109/TDSC.2012.24, 2012.

[11] Unity game engine. https://unity3d.com/unity.

[12] Aaron Bangor, Philip Kortum, and James Miller, "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale". Journal of usability studies, vol. 4, no. 3, pp. 114-123, 2009.

[13] Nicole Lazzaro, "The Four Fun Keys". In: Game Usability: Advice from the Experts for Advancing the Player Experience. Burlington: Taylor & Francis, pp. 317–343. DOI: 10.1016/b978-0-12-374447-0.00020-2, 2008.