CyBOK v1.1 Linkage for a Secure Coding Game-Based Lab

Report 2022-03-11

Project funded by CyBOK (small projects to develop resources around CyBOK v1.1)

Heriot-Watt University

Investigators: Manuel Maarek, Sheung Chi Chan

Contributors: Léon McGregor, Szymon Wlodarczyk, Callum Jones

Table of Contents

Introduction

To promote cybersecurity education, we have developed a virtual laboratory, the Citadel Programming Lab, for secure programming education that combines a tower defence game to emulate a realistic attack and defence environment, and a GitLab online coding environment to perform secure programming tasks. In the game, potentially adversarial vehicles (attackers) are targeting a bank (asset to protect) and towers are defending it (defenders). The player acts as the controller of the tower to manage the defence strategy. This attack/defence relationship in the game emulates real attacks and defences in cybersecurity and is designed to make such digital adversarial context tangible. The game was co-designed by researchers in software security and in serious games to gamify the cybersecurity processes at play when coding security applications. The initial development took place as part of a RISCS/NCSC funded project [2-3] and its programming tasks were based on exercises from a prior study on software developers [1] which was not related to game or game-based learning. An extension to the game and platform into an educational lab was later supported by the EPSRC Secrious project.

This report focuses on the outcome of the CyBOK-funded project to link this Secure Coding Game-Based Lab with CyBOK v1.1. The goal of this project was to link the cybersecurity content of the game-based lab to the CyBOK v1.1 knowledge base, and to prepare its distribution and dissemination. These two aspects are covered in this report in the following two sections, respectively.

We named the Secure Coding Game-Based Lab the “Citadel Programming Lab”.

Website: https://citadel-programming-lab.gitlab.io/

Sources: https://gitlab.com/citadel-programming-lab/citadel-programming-lab

CyBOK v1.1 Linkage

The main components of the lab are the game with its creeps and towers, and the exercises the player needs to unlock to upgrade the towers. In this section, we describe the CyBOK v1.1 mapping we developed: from the exercises to CyBOK and from the game elements to CyBOK. We then explain how we have put into action this mapping with in-game help menus and within the exercises with dedicated lab sheets.

Lab to CyBOK v1.1 Mapping

The following table is showing how the lab's secure programming exercises connect to Knowledge Areas and Topics of CyBOK v1.1.

Coding Tasks	CyBOK Knowledge Areas	CyBOK Topics
All exercises	Software Security	coding practices
	Secure Software Lifecycle	motivations for secure software lifecycle
	Risk Management & Governance	risk assessment
	Applied Cryptography	Cryptographic Libraries
PGP	Applied Cryptography	Managing Public Keys and Public Key Infrastructure
	Applied Cryptography	Digital Signatures
	Web & Mobile Security	web PKI and HTTPS
	Network Security	Public Key Infrastructure
	Law & Regulation	electronic signatures and indentity trust services
Certificates	Applied Cryptography	Managing Public Keys and Public Key Infrastructure
	Applied Cryptography	Binding Public Keys and Identities via Certificates
	Web & Mobile Security	web PKI and HTTPS
	Network Security	Public Key Infrastructure
SSL	Network Security	TLS (Transport Layer Security)
	Applied Cryptography	Difie-Hellman Key Exchange
URL shortener	Privacy & Online Rights	obfuscation-based inference control
	Privacy & Online Rights	privacy engineering
	Risk Management & Governance	risk assessment
Credentials	Software Security	SQL injection
	Software Security	query generation
	Applied Cryptography	Hash functions
	Law & Regulation	prescriptive jurisdiction and data protection
	Web & Mobile Security	input sanitisation
	Web & Mobile Security	SQL-injection
	Web & Mobile Security	password leaks
String encryption	Privacy & Online Rights	privacy engineering
	Privacy & Online Rights	cryptography-based access control
	Applied Cryptography	Authenticated Encryption(AE)schemes
	Applied Cryptography	Cryptographic Libraries

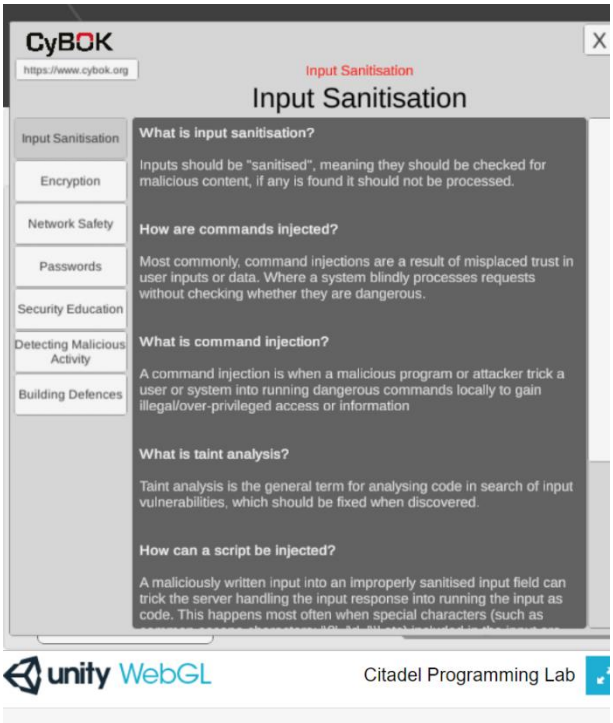
The serious game approach of the lab means that the game as well as the accompanying programming exercises provide cybersecurity content and support cybersecurity learning. The game elements

(vehicles and towers) provide metaphors of real-life attacks and defences. The linkage of these game elements with the Knowledge Areas and Topics of CyBOK v1.1 is shown in the following table.

Game Elements	CyBOK Knowledge Areas	CyBOK Topics
Simple Vehicle	Software Security	coding practices
	Risk Management & Governance	risk assessment
Tank	Privacy & Online Rights	obfuscation-based inference control
	Privacy & Online Rights	privacy engineering
Hacker	Adversarial Behaviours	Hacktivists
	Web & Mobile Security	password leaks
	Web & Mobile Security	SQL-injection
Interceptor	Applied Cryptography	Managing Public Keys and Public Key Infrastructure
	Applied Cryptography	Binding Public Keys and Identities via Certificates
	Network Security	TLS (Transport Layer Security)
	Web & Mobile Security	web PKI and HTTPS
	Network Security	Public Key Infrastructure
Standard Turret	Software Security	coding practices
	Risk Management & Governance	risk assessment
	Web & Mobile Security	input sanitisation
	Applied Cryptography	Hash functions
Communication Tower	Network Security	Networking Infrastructure Security
	Network Security	Cloud and Data Center Security
	Distributed System Security	reliable and secure group communication
Missile Turret	Network Security	device fingerprints
	Adversarial Behaviours	Attribution
	Security Operations & Incident Management	cyber-threat intelligence (CTI)
Laser Turret	Network Security	TLS (Transport Layer Security)
	Web & Mobile Security	web PKI and HTTPS
Watch Tower	Applied Cryptography	public-key schemes with special properties
	Applied Cryptography	Binding Public Keys and Identities via Certificates
	Authentication, Authorisation & Accountability	Accountability
Radar	Authentication, Authorisation & Accountability	Authentication
	Applied Cryptography	public-key schemes with special properties
	Applied Cryptography	Binding Public Keys and Identities via Certificates

In-game CyBOK v1.1 Menu

As part of the project, we developed a dedicated in-game help menu. The help menu available from the upgrade panel displays questions/answers discussions on each major topics covered by the game. These questions are organised in terms of attack, defence, and vulnerability dimensions. The menu is repeated within the in-game pause panel accompanying additional explanations on the role of the towers and vehicles within the game.



Tasks Lab Sheets with CyBOK v1.1 Links

Lab sheets are provided for each of the six programming tasks of the game. Note that three of the programming tasks originate from a study [1] conducted to evaluate how resources used by developers impact the security of their code. The study was replicated [2-3] with a serious game wrapping to explore the impact such serious gamification could have.

The lab sheets provide documentations for the programming tasks. Each task contains in their starting source code a target specification, these lab sheets are therefore intended to provide additional guidance and context to the task. The structure of a task lab sheet is as follows.

- **Introduction**. Overview of the topic of the task and its motivation.
- **Practical exercise**. A technical guidance with steps to complete the practical exercise.
- **Relation to the game**. An explanation making explicit the relation between the task and the game elements and metaphors.
- **Learning Objectives**. A comprehensive list of learning outcomes the task and its related game elements and metaphors are designed to encompass, as well as a list of related CyBOK v1.1 Knoweldge Areas and Topics.

Evaluation of the CyBOK v1.1 Linkage

As part of the project, we planned for focus group discussions with participants to evaluate the CyBOK linkage. The focus group discussion is structured along the following five topics of discussion with their guiding questions.

- **Topics/knowledge areas of CyBOK v1.1 related to Secure Coding Game-based Lab.**

  What topics and knowledge areas of CyBOK v1.1 relate to the Secure Coding Game-based Lab? Which are fully covered, which are partially covered, which are missing?

- **Secure Coding Game-based Lab elements relate to topics/knowledge areas of CyBOK v1.1.**

  Which aspects and elements of the Lab relate directly to CyBOK v1.1, which ones relate indirectly, which do not relate to CyBOK v1.1, which do not fit in the CyBOK v1.1 knowledge tree?

- **Links, embedding, guidance, explanations.**

  What aspects of the CyBOK v1.1 linkage are useful and which could be improved? In particular, with regards to where the links and embedding are, and how the explanations and guidance are provided?

- **CyBOK v1.1 and other cyber-security knowledge base.**

  How would alternative or complementary cyber-security knowledge base be suitable? For instance, threats models, vulnerability databases and scoring: CVE, CWE, CVSS, OWASP, STRIDE, etc. Would the lab benefit from including these, Why / Why not? How would this compare or repeat or complement with CyBOK?

- **Prompts, messaging, security labelling.**

  How appropriate is the CyBOK advice in the lab? How easy is it to follow and understand? How does it compare to any other messaging found in the lab? How does the messaging prompt your behaviour?

Distribution and Dissemination

An essential motivation for this project to link the Citadel Programming Lab to CyBOK v1.1 is to disseminate it. The link to CyBOK v1.1 makes the lab's cybersecurity knowledge content explicit. As part of the project, we prepared the distribution of the source code of the lab, a public website for the lab and a video showcasing the lab.

Web Page and Distribution

We prepared a public facing Web page for the game together with an open distribution of the sources of the game-lab platform. We intentionally decided to host the Web pages describing the lab platform on gitlab.io and to host the source code of the platform on gitlab.com as the lab uses an instance of GitLab for coding and running code. This public distribution of the game provides instructions on how to deploy the lab on a Linux server using Docker, GitLab and a GitLab runner with Python and Java. The source code includes the deployment scripts as well as the coding tasks and a WebGL build of the Unity game. The source code of the Unity game is distributed separately.

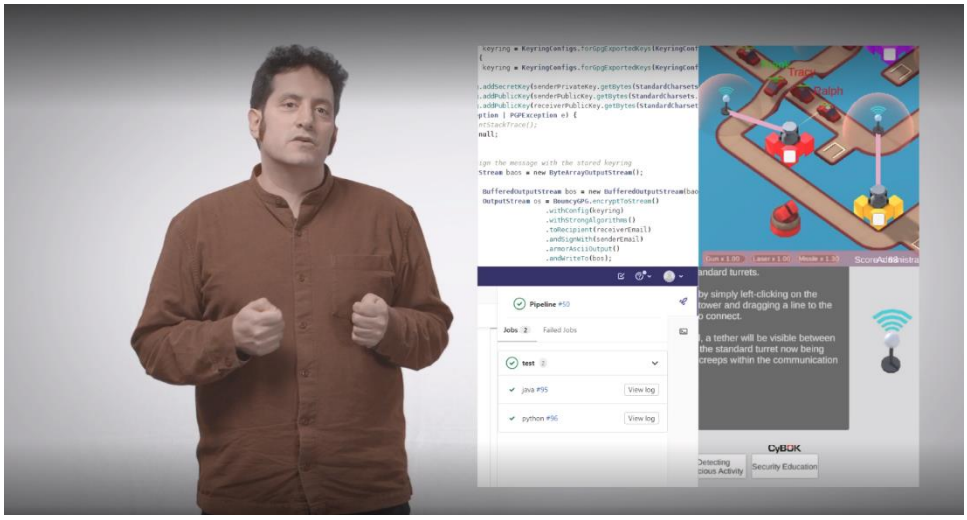
Website: https://citadel-programming-lab.gitlab.io/

Sources: https://gitlab.com/citadel-programming-lab/citadel-programming-lab

Video Dissemination

A video showcasing the lab was produced as part of the project. While explaining the lab and its CyBOK linkage, the video demonstrates the gameplay and how the player could complete exercises to unlock more tower upgrade options to help build up the defence. The video has been designed as a general introduction to the platform to the benefits of potential users whether they are coders, gamers, students, or teachers. The script of the video and a screenshot are reproduced below. A link and embedding of the video are available on the public page of the lab.

Video: https://www.youtube.com/watch?v=LOb9sYrtnsQ



“

Security is serious and gaming is playful. In the Citadel Programming Lab, secure coding becomes playful, and you seriously learn cybersecurity by playing.

We developed a virtual lab, the Citadel Programming Lab, for secure coding with a serious game. It is a tower defence game where a flow of vehicles is trying to reach your asset. You will need to defend it. You have different towers to help you in your defence against the attacks.

The tower defence game renders tangible attacks that can occur in a digital context. In the game, you activate your towers with your mouse and your keyboard. You enter the code names of vehicles to target them. But be aware that a name can hide an attack. You will also use your keyboard to build up your digital defence. To upgrade your towers, you can undertake secure programming tasks. The game transforms into a GitLab Computer Lab where you write secure programmes and test your solutions.

The programming tasks are available in both Java and Python. They are about securing data, execution, or communications. Your communication tower is getting hijacked, take the programming task to strengthen their communications. The gameplay explains security processes and throughout the gameplay, game elements lead to the coding. The security coding reinforces the game metaphors.

The cybersecurity of the game and tasks are aligned to the CyBOK, the Cybersecurity Body of Knowledge. The relations with the Knowledge Areas and Topics of CyBOK are made explicit in the lab. These links to CyBOK map the scope of the learning outcomes of the lab.

The Lab can be self-hosted for the purpose of training or running a course. Users play and code while learning about cybersecurity. They also review codes and compete for higher scores. The lab teachers can make use of the lab sheets provided. They access the students’ coding for potential assessment or direct feedback.

The development of the game and platform has been supported by RISCS, NCSC, the Secrious EPSRC project and CyBOK. The game and platform were developed in Scotland jointly at Heriot-Watt University and the Glasgow School of Art. Get in touch if you want to play, if you want to learn, if you want to use the lab in your own teaching.

“

Conclusion

As part of this CyBOK v1.1 project, we undertook the linkage of the Citadel Programming Lab with CyBOK v1.1 and prepared dissemination and distribution materials for the platform. At the time of writing this report, the evaluation of the CyBOK v1.1 linkage is ongoing but a first focus group discussion with participants highlighted the meaningfulness of the linkage, its suitability to support the game-lab's progressive introduction to complex cybersecurity topics.

References

[1] Acar, Y., Backes, M., Fahl, S., Kim, D., Mazurek, M.L., Stransky, C., 2016. You Get Where You're Looking for: The Impact of Information Sources on Code Security, in: 2016 IEEE Symposium on Security and Privacy (SP). Presented at the 2016 IEEE Symposium on Security and Privacy (SP), pp. 289–305. https://doi.org/10.1109/SP.2016.25

[2] Maarek, M., Louchart, S., McGregor, L., McMenemy, R., 2019. Co-created Design of a Serious Game Investigation into Developer-Centred Security, in: Gentile, M., Allegra, M., Söbke, H. (Eds.), Games and Learning Alliance, Lecture Notes in Computer Science. Springer International Publishing, pp. 221–231. https://doi.org/10.1007/978-3-030-11548-7_21

[3] Maarek, M., McGregor, L., Louchart, S., McMenemy, R., 2019. How Could Serious Games Support Secure Programming? Designing a Study Replication and Intervention. Presented at the EuroUSEC European Workshop on Usable Security, Stockholm, Sweden.