Masaryk University
Faculty of Informatics

Prerequisite testing of cybersecurity skills

Master's Thesis

Bc. Valdemar Švábenský

Brno, Spring 2017

Declaration

Hereby I declare that this paper is my original authorial work, which I have worked out on my own. All sources, references, and literature used or excerpted during elaboration of this work are properly cited and listed in complete reference to the due source.

Bc. Valdemar Švábenský

Advisor: RNDr. Jan Vykopal, Ph.D.
Consultant: Mgr. Martin Ukrop

Acknowledgement

I sincerely thank my supervisor, RNDr. Jan Vykopal, Ph.D. for offering me to work on this thesis and subsequently introducing me to the great team of people around the KYPO project. Without his invaluable advice during our frequent meetings, I would not be able to finish this work. His high demands, precision, and determination combined with his willingness to help created a highly motivating working environment, which was at the same time friendly and supportive. I am glad for the opportunity to work on my thesis with him.

I thank my consultant, Mgr. Martin Ukrop, for constructively criticizing my work and for his enthusiasm in explaining me the ways of scientific thinking. His zeal is almost contagious (even in written comments). Moreover, I thank my "unofficial consultant", doc. RNDr. Radek Pelánek, Ph.D. for independently assessing my progress and for identifying transgressions in methodology. The different points of view, which they both provided, were highly valuable for perfecting my work.

I also thank the following people who contributed to the improvement of this thesis: prof. RNDr. Václav Matyáš, M.Sc., Ph.D. for reminding me of the importance of checking the assessment's validity; Mgr. Maria Králová, Ph.D. and Mgr. Petr Mareška for consulting suitable statistical methods with me; Juraj Uhlár for creating a helpful visualization tool; anonymous reviewers from the AIED conference, Mgr. Vlasta Šťavová, Mgr. Petra Kalábová, and Bc. Karel Kubíček for their comments; and Mgr. Ina Sečíková for proofreading.

Last but not least, I want to express my gratitude to the people that are close to me. I thank my parents, Eva and Valdemar, for encouraging my interest in mathematics and computers ever since my early childhood, for trusting in me, and for always caring for my well-being. I thank my brother, Gabriel, for his graphic design advice and for sending me dank memes in the dark hours of despair, when I wanted to give up. I thank my friends for reminding me to go outside or have a beer from time to time. Finally, I thank my girlfriend, Pavlínka, for her unyielding support, patience, and love.

Many thanks to you all.

Abstract

Cybersecurity games are an attractive and popular method of active learning. However, the majority of current games are created for advanced players, which often leads to frustration of less experienced learners. Diagnostic assessment of participants' knowledge and skills before starting an educational game can increase the benefits of playing. The information acquired by prerequisite testing enables tutors or learning environments to suitably assist participants with game challenges and maximize learning in their virtual adventure.

To the best of our knowledge, this work is a pioneering attempt in researching prerequisite testing of cybersecurity skills. The thesis proposes a methodology for developing cybersecurity games and pretests, resulting from a thorough literature review and exploration of state-of-the-art cybersecurity platforms. The method is applied in practice to create the first prerequisite test for a cybersecurity game in the KYPO: Cyber Exercise and Research Platform at Masaryk University. Moreover, this work investigates the pretest's predictive value for identification of learners' readiness before playing the KYPO game.

The lessons learned from the experimental study are vast. A linear regression analysis confirmed that players' skill, expressed using the game score, can be predicted by the prerequisite test result. Furthermore, the model's accuracy and statistical significance improved after confidence assessment of certainty in one's answers was introduced. Interestingly, the qualitative study of in-game actions revealed several anomalies in the performance patterns of the participants. These findings uncovered numerous factors that may create noise in the model, bringing unique insights into the field and implying new opportunities for future research.

Keywords

active learning, cybersecurity games, diagnostic assessment, prerequisite testing, linear regression modeling, KYPO

Contents

1 Introduction

Cybersecurity games allow participants to test their knowledge and exercise their skills in different areas of computer security. Although carried out in a closed and controlled environment, the games often simulate practical, real-world situations. The players can, for example, attack and defend computer systems, analyze network traffic, or disassemble binaries, all without any negative consequences in reality.

Studies confirm multiple benefits of cybersecurity games [71, 76, 85]. They can inspire interest in computer security and motivate participants to explore the field further. Games designed specifically for education enrich the curriculum and test the learners' competence in an authentic setting, enabling them to discover their strengths and weaknesses. Moreover, cooperative games implicitly teach teamwork, management, and communication skills. Ranking well in competitive games often leads to peer recognition, (monetary) prizes, or job opportunities. Lastly, playing can aid in preparing for a future profession.

Competitions and games of various difficulty and focus are spreading widely, from informal online hacking communities to universities and professional security conferences. CTFtime [15], a popular game-announcing website, listed 63 competitions in 2014, then 90 in 2015, and 103 in 2016. In addition, the number of participants in cybersecurity games is growing exponentially [76].

At the same time, several authors argue that although high-quality games are available, they offer little educational value to learners [71, 84]. This is because the games often require substantial knowledge of the problem domain, as well as practical expertise, in advance. As a result, the majority of computer science students are unable to participate. Even worse, some students' interest and motivation may diminish after an unsuccessful attempt or after performing poorly against professionals [71]. Research suggests that games and contests are effective only for already skilled players, in other words, for players whose skills "closely match those required by the competition" [76].

Educational games are specifically created to aid learners from beginner to intermediate levels. One of the biggest difficulties in creating them is achieving *game balance*: assigning tasks that are just right for the player's skill, neither trivial nor impossible to solve [71, 61].

One approach to achieving the game balance is introducing methods of adaptive learning [28], which change tasks during the game to easier or more difficult based on the player's success rate. Another solution is a diagnostic assessment by prerequisite testing, the topic of this thesis. This approach, suggested in pedagogical theory [59, 44], refers to testing the player before (and possibly during) a cybersecurity game to determine whether the player's skills are sufficient to finish the tasks, thus, by extension, providing game balance [71].

This work's main motivation is the demand for timely identification of students who may require help while playing. As a result, their individual needs can be appropriately addressed. A diagnostic assessment provides useful information about learners before the game starts, when no other data about them are available. This information enables both human and automated tutors to assist specific players: for example, by providing more precise instructions, hints, or relevant study materials. Moreover, the results of the assessment can be used to create balanced teams in cybersecurity games and exercises.

To the best of our[1] knowledge, none of the state-of-the-art cybersecurity games implement prerequisite testing. Therefore, this work proposes a general methodology for creating prerequisite tests, which is based on a thorough literature review. The method is applied to develop the first pretest for a selected cybersecurity game at the KYPO: Cyber Exercise and Research Platform [86] at Masaryk University. This thesis also presents an experimental research investigating whether the proposed quiz and self-assessment can identify learners' readiness before playing.

The thesis is divided into seven chapters. Chapter 2 clarifies the key terms used throughout the text. Chapter 3 maps the current research and practice of cybersecurity games and using assessments, provides examples of cybersecurity games, and compares selected platforms for cybersecurity training. Chapter 4 defines a method for creating games and prerequisite tests. Chapter 5 explains the design of an experiment that implements the method in practice. Chapter 6 presents and discusses the experiment's results. Chapter 7 concludes by summarizing the topic, my contribution, and the key results. Finally, it suggests opportunities for future work.

1. Plural is used in the text when refering to me and the thesis supervisor.

2 Background

The thesis connects two major areas: cybersecurity games and educational assessment. This chapter lists and, if needed, compares the definitions of key terms in these two fields, along with examples. Section 2.1 defines cybersecurity terminology, which is not firmly established, as the field is still evolving. Section 2.2 serves as a glossary of assessment terminology, which might be unfamiliar to readers with a purely technical background. Whenever possible, general terms appear before specific terms, and terms are listed in the order of relevance.

2.1 Cybersecurity terminology

The basic concepts, such as cyberspace, cyber attack, and cybersecurity are defined in the NIST's Glossary of Key Information Security Terms [43] and are not repeated here. Other relevant terms follow.

Cybersecurity game

A *serious game* is a software application that uses computer game structure or includes game elements for a primary purpose other than entertainment [45], such as for learning, practicing, or competing. A *cybersecurity game* is a serious game designed to apply cybersecurity concepts. Note that a cybersecurity game differs from a *cybersecurity exercise*, which is a simulated training event. ISO norm 22398:2013 [39] defines the terms related to (cybersecurity) exercises.

Capture the flag (CTF)

Originally, *Capture the flag* is a traditional outdoor game for two teams. Each team has one physical flag in their base. The goal is stealing the other team's flag and bringing it to own base, while at the same time defending the own flag. Popular computer games, such as World of Warcraft or Team Fortress 2, also use this structure.

In this work, CTF is a specific cybersecurity game. To define it precisely, CTFtime [15], an archive and a roadmap for these games, lists three types of CTFs: Attack-defense, Jeopardy, and a mix of these two.

Attack-defense CTF

In an *Attack-defense* CTF, each team (having one or more players) controls a computer network with hosts running vulnerable services. The goal is attacking other teams' assets and stealing secret information: *flags* (usually long random strings), while at the same time defending the own assets. The teams normally receive time to prepare their exploits and patches in advance. Historically, Attack-defense is the first type of CTF games [15], and some authors [53, 54, 81] use the term CTF to mean Attack-defense CTF exclusively.

Attack-only or *Defense-only* CTFs [54] may be viewed as a subcategory. Still, there is no clear line between attacking and defending [28, 53], since offensive and defensive skills are closely related. Some suggest that learning to attack is required for learning to defend [28], for example, finding a security flaw in a program is the first step to repairing it.

Jeopardy CTF

In a *Jeopardy* CTF, each team (having one or more players) receives several tasks. The task topics are similar to Attack-defense CTFs and include web security, service exploitation, cryptography, network forensics, or reverse engineering. Since the tasks are usually of an offensive nature, Jeopardy CTFs can be regarded as a subcategory of Attack-only CTFs [17]. However, this text makes a distinction: the tasks in Attack-defense CTFs are carried out in an underlying network infrastructure; in Jeopardy CTFs, the tasks are often simply predefined in a web interface or a virtual machine. Completing a task yields a unique flag confirming the solution; the tasks' difficulty and score value gradually increase.

The need for finer distinction in cybersecurity terminology

Gondree et al. [28] argue for a more accurate terminology than simply Attack-defense and Jeopardy CTFs. The authors discuss a classification of cybersecurity games concerning task variety (whether the skill set needed to participate is narrow or extensive) and adversary dynamics (whether the game dynamic is determined by designers or influenced by players). Still, this classification omits other aspects, such as specifying learning outcomes, revealing correct solutions, and allowing to replay the challenges for practice after the game ends [28].

2.2 Assessment terminology

Formative assessment

Formative assessment is a practice of obtaining, analyzing, and using evidence about a student's performance with the aim to make better pedagogical decisions than without this evidence [8]. Formative assessment is crucial for promoting learning and separates educational games from play [71]. In teaching practice, it most often involves providing "feedback to learners while they are still learning" [69, p. 480] to present an opportunity for improvement. An example of formative assessment is informing students about their strengths and weaknesses after skill assessment, and recommending relevant literature to fill knowledge gaps.

Summative assessment

Summative assessment is a practice of ranking, grading, or otherwise categorizing learners based on their knowledge. The learning is usually proven by an exam awarded with a mark, grade, or score, unlike in formative assessment. While formative assessment is performed to monitor and improve knowledge, summative assessment evaluates knowledge.

Summative assessment of a learner can be *norm-referenced* (by comparison to other learners) or *criterion-referenced* (with respect to predefined criteria regardless of others' performance) [69, chap. 43].

Initial, Diagnostic, and Placement assessment

Initial or *diagnostic assessment* is a practice of discovering learners' background, including their prior learning and qualifications, before the start of a learning process [69, chap. 43 and 47]. *Initial assessment* places a learner against a standardized qualifications framework, while *diagnostic assessment* determines a learner's proficiency in a certain skill by breaking it down into parts and using tests, questionnaires, or interviews to discover the learner's competence in these parts [69, p. 539]. The information gained can be used formatively as a feedback for a teacher or the learner, or summatively to differentiate between learners. *Placement testing* is a related, similarly performed process. However, it has a different goal: to assign students to courses of different difficulty [58]; therefore, it is always used summatively.

Prerequisites, Specific entry characteristics

Prerequisites are knowledge and skills that a learner must have to benefit from an upcoming training. Morrison et al. use the term *specific entry characteristics*, which additionally include required attitudes [59, p. 58].

Pretest, Prerequisite test

Morrison et al. state that *pretest* is a test carried out before the instruction to check prerequisites, while *prerequisite test* is a part of a pretest measuring content or skill preparation [59, p. 476]. In this work, the term *pretest* refers to the whole test conducted before playing a game with the aim to determine learners' knowledge or skills, while *prerequisite test* is any non-empty subset of pretest's questions measuring prerequisites. Note that a question such as "What is your name?" can be a part of a pretest but is not a prerequisite test, since it does not measure prerequisites.

Assessment validity

Validity is evidence to support the "interpretation assigned to assessment results" [20]. Downing [20] argues an assessment itself cannot be proclaimed valid or invalid; instead, its outcome (such as a score) must be validly defined and interpreted to justify resulting conclusions (such as passing a course). Caution must be exercised about whether the assessment's outcome measures the knowledge or skills it was designed to measure [69, p. 500].

Assessment reliability

Reliability means that the assessment's outcome is consistent across multiple repetitions [21, 82], that is, retest scores exhibit low variance. When assessing a skill, written work, or an oral examination, it means that different examiners give the same score to the work of the same standards [69, p. 501] (*interrater reliability* [82]). Like validity, reliability is a characteristic of the assessment's outcome, not of the assessment itself [21]. Reliability is a necessary condition for validity [21].

3 State of the art

Research in cybersecurity education is fragmented and not yet widely established. We are not aware of any international scientific community that focused on the topic and existed before 2014, and there is no single comprehensive resource, such as a monograph or journal series.

This chapter maps current theoretical research and practice of cybersecurity games and using assessments. Section 3.1 presents works related to cybersecurity education. Section 3.2 provides motivation for prerequisite testing. Section 3.3 characterizes and compares selected platforms for cybersecurity training regarding prerequisite testing and game design. It also provides examples of cybersecurity games.

3.1 Cybersecurity education literature

In 2014, USENIX Summit on Gaming, Games, and Gamification emerged as the first conference also covering cybersecurity games. Two years later, it expanded into USENIX Advances in Security Education Workshop [79], which specializes in research and practice of computer security education. ACM International Computing Education Research conference [35] focuses on IT education in general. Related works were also presented at the ACM SIGCSE conference [67].

There are several works concerning CTFs in learning. Werther et al. describe creating and organizing an academic Attack-defense CTF focused on web application security, regarding "teaching methods, game design, scoring measures, logged data, and lessons learned" [84]. Vigna [81] describes in-class exercises and an Attack-defense CTF designed for a network security course, along with their execution. Similarly, Fanelli and O'Connor [25] describe their experience with organizing an educational Attack-defense CTF. None of these games used prerequisite testing; however, courses on related topics were conducted beforehand to narrow the knowledge gap. Chothia and Novakovic [13] developed Jeopardy CTF challenges for a university course. Among other topics, they describe post-assessment of individual students. Based on the students' written homework and final exam results, the authors concluded that success in CTFs indicates basic competence and knowledge of computer security (but not necessarily thorough understanding).

To practice for CTFs, the Cyber Security Awareness Week initiative [65] recommends a CTF Guide [64] and a Practice CTF List [70]. Hardikar [32] offers a roadmap for cybersecurity games and training platforms. Cybrary [16] is a free, extensive portal for cybersecurity education. Open Security Training [77] is a set of free courses on cybersecurity, where certain courses link to other courses as their prerequisites.

There is also a standardized effort of describing cybersecurity skills. The National Cybersecurity Workforce Framework [26] organizes cybersecurity fields of work into seven categories, each containing specialty areas with required knowledge and skills. However, the content is classified by the US Department of Homeland Security.

3.2 Motivation for prerequisite testing

Mirkovic et al. [54, 55] emphasize considering individual skills in team cybersecurity games to balance the teams and give all an equal chance to succeed. Before using CTFs in classrooms, Mirkovic and Peterson [54] surveyed the students about their skills to create balanced teams; unfortunately, the paper does not provide details about the process. The authors disclosed by e-mail that the students self-evaluated their familiarity with topics such as programming, hacking, or Linux on a four-step scale from "not familiar" to "expert". In another study [55], the participants reported their knowledge of programming, security, and tools that were to be used later. Again, the survey results were used to balance the teams. However, the authors concluded that this led to inequality among the teams, as the self-assessment was often inaccurate.

Bolívar-Cruz et al. [9] examined self-assessment of university students. Their summary of research in the area says that self-assessment's accuracy is frequently low or questionable but also warns about methodological errors in some of the previous studies. Interestingly, the authors point out a gender bias: men tend to self-assess their skills higher than women. This finding is consistent with previous results by Beyer [7].

In contrast, Allen and Van Der Velden [2] advocate using self-assessment, arguing that all people know the level of their skills the best. However, the authors warn about its issues, including misunderstood skill items, ambiguous rating scale, and the risk of an unreliable answer (either intentional or not). To address these problems, the authors sug-

gest defining clear and simple items, labeling the rating scale actively and concretely, and explaining the importance of accurate answers to the participants. Still, the authors advise using independent, objective tests alongside self-assessment to increase the reliability of the results.

To address the inaccuracy of self-assessment for balancing teams, Mirkovic et al. similarly recommend "conducting a short quiz-type assessment prior to the event" [55] but do not specify how to do this.

Nagarajan et al. [61] stress that measuring skills before and after playing is vital to determine the game's effectiveness. The authors report that security training programs do not implement this measurement.

Finally, the need for prerequisite testing arises not only in cybersecurity. Govindasamy [29] suggests using it in e-learning courses to test both minimal requirements and proficiency. Based on the results, the learner can be directed to a simpler or more difficult course, or skip the already familiar areas in the current course. Educational literature [69, 59, 44] advocates the use of prerequisite testing in teaching practice.

3.3 Comparison of selected training platforms

This section describes and compares five selected platforms for cybersecurity training. The list is not exhaustive; I chose the platforms based on differing types of tasks, differing game design, and, if the platforms had restricted access, availability for Masaryk University. This thesis also provides details about the KYPO environment used for experimentation and places it in the context of other state-of-the-art platforms.

The following seven questions are answered for all the platforms to help understand their design and characteristics:

1. What is the goal of the platform?
2. Is prerequisite testing performed before the start of the game, and if not, are prerequisites at least described in any way?
3. Are hints available during the game; if so, in what form?
4. In what environment is the game played?
5. Is the platform intended for individual use or teams?
6. If the players receive points, what scoring system is used?
7. How can the platform be accessed?

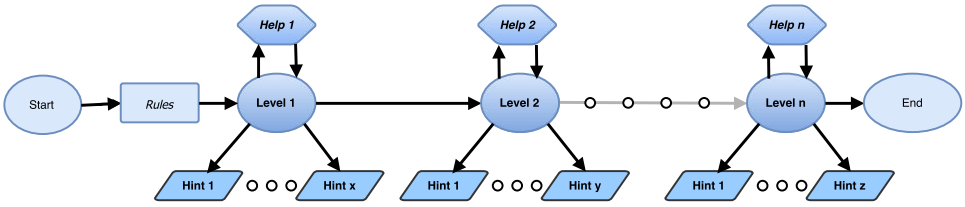

Figure 3.1: General structure of a KYPO cybersecurity game [5]

KYPO

The KYPO: Cyber Exercise and Research Platform developed at Masaryk University is an environment for cybersecurity research, simulation and analysis of cyber attacks, and cybersecurity education [86]. It offers various training scenarios performed in a realistic network environment, which can be modeled based on the desired purpose. The virtual network is emulated by KYPO cyber range [86] and can be accessed online by invited members.

KYPO provides a generic format of Attack-only CTFs and several game instances. Figure 3.1 shows the scheme of a game, which is structured into successive levels leading to the final objective, such as data theft. Before the start, each player (usually an individual) has access to limited network resources and brief information about the goal. Every level is finished by finding a correct flag; this accomplishment is awarded a specified number of points contributing to the player's total score. The game ends upon entering the last flag or when a predefined final check of the system's state succeeds.

Each level contains a time limit for finding the solution. A countdown in seconds is presented to the learner to simulate the real-life constraint of the scarcity of time. The time limit may also indicate the level's difficulty in comparison with other levels. After the time expires, the player can still finish the level without any penalty.

The game provides optional scaffolding by offering hints, which are usually ordered from general (for example, which tool to use) to specific (for example, how to use the tool). If the player struggles with a level, these hints can be used in exchange for penalization by negative points. Figure 3.2 shows an example of the time, scoring, and hint panels. There is evidence that these game elements can improve the overall

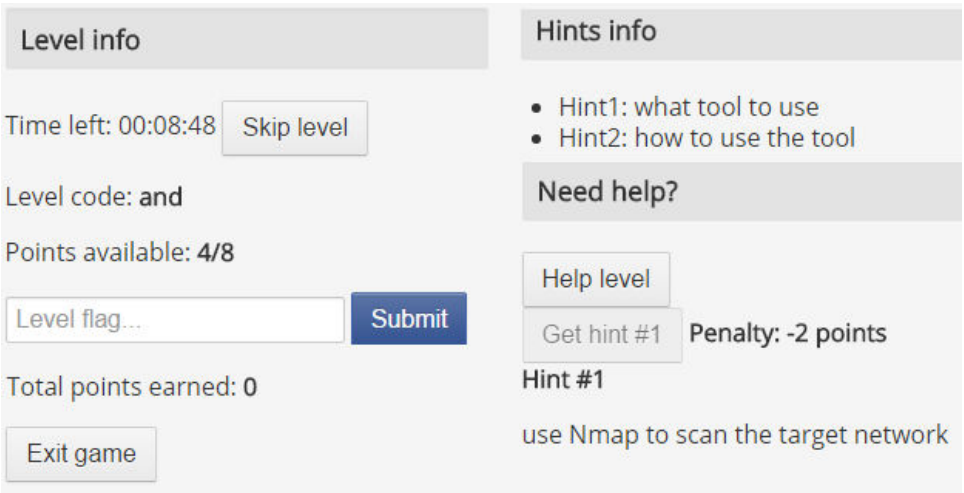

Figure 3.2: KYPO game portlet: time, scoring, and hint panels

effectiveness of learning [40]. It is also possible to skip the level, display the recommended solution (marked as "Help" in Figure 3.1 and "Help level" in Figure 3.2), and quit the game at any time. Prerequisite testing is not implemented yet.

The generic nature of the game format enables collecting generic game events, regardless of the topic of the particular game and technical infrastructure used. The game events carry information about the interaction of the player with the game interface, namely: starting and ending the game, starting and ending each level, submitting incorrect flags and their content, using hints, skipping a level, and displaying a level's solution. Apart from the event type, each event also contains timestamp and player's ID, which is a randomly generated 5-digit pseudonym that matches the actions to the learner. By processing the events, it is possible to derive these metrics (per level or for the whole game):

- the time played,
- the number of hints taken,
- the number of incorrect flags submitted,
- the number of solutions displayed, and
- the number of levels completed.

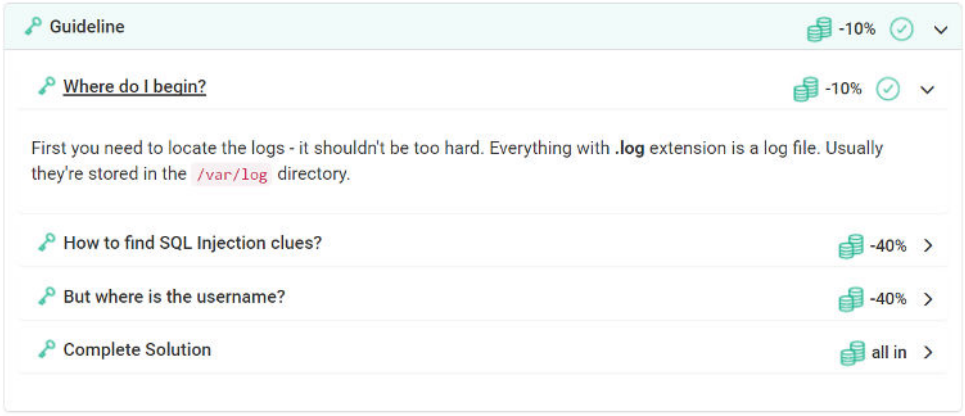

Figure 3.3: Avatao: a panel with hints and scoring penalty

Avatao

Avatao [4] is a commercial online platform containing practical IT security challenges for individuals. Upon logging in, a user can choose one or more learning paths: a series of exercises constructed to teach specific cybersecurity skills. Each learning path is briefly described along with its expected learning outcome.

Every challenge on a given path contains a short description of the task, and usually also a list of recommended reading on related topics. Prerequisite testing is not performed before starting an exercise. The challenge runs in a virtual online environment or is available as a downloadable executable; no setup is needed.

The platform is still evolving, and at the beginning of the year 2017, hints were added to some of the challenges. The hints are typically ordered from a small starting tip to a complete description of the solution. The player receives points for each completed challenge and can track the progress on the learning path. The number of hints taken is inversely related to the number of points received; the player gets zero points if asking for a complete solution (see also Figure 3.3).

General tags, such as Web Security, PHP, or SQL injection are attached to individual challenges. No finer distinction is made, for example, three different challenges on creating a fake public-key certificate, breaking a MAC function algorithm, and attacking a custom implementation

of prime generation in RSA are all tagged simply with a "cryptography" keyword. The tags apparently serve for the website's search engine only, as they are not ingrained within the learning process. The specified learning outcome is not tested upon path completion.

ENISA CEP

The Cyber Exercises Platform (CEP) [22] maintained by European Union Agency for Network and Information Security (ENISA) offers several composite exercises. As a part of Cyber Europe training program, it is available exclusively for IT security teams from EU and EFTA member states. After individual players log in to the platform website, they can choose any available exercise. Each is briefly described; classified into categories such as cryptanalysis, malware analysis, or system forensics; and contains a difficulty estimate on a five-step scale.

When starting an exercise, the player is presented with a detailed description of the task, technical information including helpful tools, and resources, such as recommended reading or required files. The platform is intended for experienced users, and prerequisites are neither described nor tested. No learning outcome is tested upon completing a task.

Each exercise offers four hints, mostly general tips that are not penalized. After completing a challenge, the player can fill in a questionnaire containing questions related to the exercise. Based on the number of correct answers, the player is awarded points and can see the score in comparison with other players who completed the same challenge.

NetWars Continuous

NetWars Continuous [37] by SANS Institute is a complex cybersecurity training program. Upon purchase, the game is distributed to individuals as a virtual machine image of a Linux system. After mounting the image, no other files are required. The player then only needs to open a scoring website [38] with a list of tasks.

The game is divided into five rounds, each containing several challenges. The individual tasks progress gradually: the first few require only basic knowledge of Linux. Prerequisites are neither defined nor tested before playing. Although SANS Institute offers cybersecurity courses, they are not linked as prerequisites within the game.

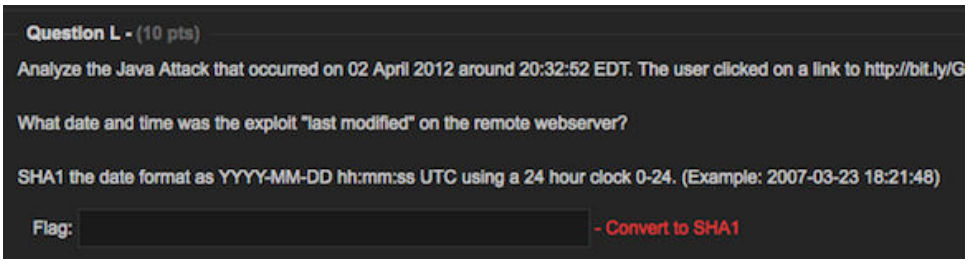

Figure 3.4: NetWars Continuous: a scoring website

After completing a task in the virtual machine, the player inputs a resulting text flag in the scoring website (see Figure 3.4). If the result is correct, the player receives points, and a next task is unlocked. If the result is incorrect, there is no penalty, but a wrong attempt is counted.

Three hints for each question are contained in the scoring website. The first hint is a general keyword connected to the solution, the second hint is the main idea leading to solving the problem, and the third hint is a step-by-step description of the solution. Requesting hints gives no penalty to the player, but the number of hints used is again counted.

The scoring website includes a scoreboard of all active players, which is updated live during the game. The game progress is mostly linear, but the player does not need to finish the last few tasks in each round before moving to the next round.

Insomni'hack Teaser

Insomni'hack Teaser [36] is an online Jeopardy CTF contest regularly organized for teams of at least one player. As opposed to previously mentioned platforms, which are rather complex, Insomni'hack Teaser is one of the many examples of simpler online hacking playgrounds. No prerequisites are defined or tested. Anyone can participate for free.

During the game, no hints are available, although sometimes the task description includes a few tips. The teams are awarded a fixed number of points after submitting a correct flag. The six teams scoring the most points are invited to a follow-up Insomni'hack CTF competition and conference, which are held annually in Geneva.

Platform	Pretest	Hints	Played	Players	Score	Access
KYPO	◌	✔ ⚠	🌐 🖧	👤 👥	✔	🔒 $
Avatao	✖	✔ ⚠	🌐	👤	✔	$ 🔓
CEP	✖	✔	🌐	👤	✔	🔒
NetWars	✖	✔	🌐 🖵	👤	✔	$
Insomni'hack	✖	✖	🌐	👤 👥	✔	🔓

Table 3.1: Summary of platform comparison (explanation in text below)

Summary

Table 3.1 summarizes the properties of all five previously mentioned exercise platforms. The answers to the seven questions posed at the beginning of Section 3.3 follow.

1. *Goal*: The goal of all the platforms is to provide practical training by solving cybersecurity-related tasks.
2. *Pretest*: None of the platforms support prerequisite testing, which provides further motivation for the thesis. The development of pretests for KYPO is in progress (◌); see Section 5.4.
3. *Hints*: The platforms, except Insomni'hack, use a hint system. KYPO and Avatao give a scoring penalty (⚠) for using hints.
4. *Played*: The platforms are available online (🌐). Additionally, KYPO sets up a virtual network (🖧) for the players and NetWars uses a virtual machine (🖵) for the tasks.
5. *Players*: The platforms aim at individual (👤) training. KYPO and Insomni'hack also allow teams (👥).
6. *Score*: Progress in the game is always awarded by points.
7. *Access*: Most of the platforms have restricted access bound to an exclusive invitation (🔒) or payment ($). The only exception is Insomni'hack and selected challenges in Avatao, which are available for free (🔓).

Further examples of active learning in cybersecurity

Numerous different CTFs are listed on the website CTFtime [15]. An Attack-defense CTF competition is regularly held at the hacker conference DEF CON [14], marking its 25th year in 2017. International CTF (iCTF) [80], organized by Vigna, is an extensive mixed academic CTF [12] established in 2003.

Cybersecurity games that are not CTFs include Targeted Attack by Trend Micro [52]. The game resembles a movie with real actors where the player, taking on a role of a major software's company CIO, can influence the plot development by making certain decisions. When the game arrives at an end, successful or not, the player is informed about the decisions made and their consequences.

An example of cybersecurity exercise is a Cyber Defense Exercise (CDX) [18] by the National Security Agency (NSA). CDX is a competition in building, securing, and defending networks from attacks carried out by the organizers. The first CDX was held in 2001 and has continued every year since then. Similarly, Locked Shields exercise [66] focuses exclusively on defending a network from external attackers. The exercise has been held annually since 2010.

4 Game and pretest development

This chapter proposes a procedure for designing cybersecurity games in Section 4.1. Furthermore, Section 4.2 and Section 4.3 define a methodology for creating prerequisite tests.

4.1 Creating new cybersecurity games

Before addressing the topic of creating prerequisite tests, mentioning related aspects of game design is necessary to help clarify the issue. To provide a systematic guide for creating cybersecurity games, I describe methods for specifying player background and learning outcomes of a game, designing levels for game balance, and improving the game.

Specifying player background and learning outcomes

Every game designed for learning or practicing must have a clear educational purpose described as *learning outcomes* [69, chap. 37]. (Nagarajan et al. [61] call them the *training goals*.) Learning outcomes are the answers to the question "What should the player learn from playing the game?" They are concrete, testable statements, such as "learning to apply an exact procedure for exploiting the Heartbleed vulnerability".

Learning outcomes often arise after specifying general *aims*, which tend to be broad (for example, "studying network vulnerabilities") or abstract (for example, "adopting the mindset of an attacker"). These aims are decomposed into concrete learning outcomes by answering the question: "What do the players need to learn before being able to perform complex tasks like this?" Using a standard goal management technique, such as SMART [19], allows defining learning outcomes with desirable properties.

Defining in advance who will play the game helps setting aims and learning outcomes that will suit the players. Intended players (such as high school students, system administrators, or security researchers) should be described regarding their occupation, characteristics (age, education), and assumed knowledge and skills.

Finally, it is important to argue why given learning outcomes were specified for the intended players, that is, how and why playing the

game helps. For example, is it increasing the player's working capabilities, keeping the player up to date with recent exploits, or building a professional community? Answering these questions can motivate the player to participate, and it can aid designers in creating the game by designing tasks that reflect these goals.

The text includes a summary of a general framework for educational game design by Annetta [3], who provides a different point of view. Nagarajan et al. [61] further reference these guidelines in connection to cybersecurity games. The scheme can supplement the ideas above, rather than conflict with them, by defining six "Is" as core principles of game design:

1. Identity: Giving the player a unique identity in the game, such as a security engineer in a major software company, a wanted hacker, or simply a personalized avatar in an online game. The reason is creating an emotional connection to the game.

2. Immersion: Creating a state of presence and flow; using rewards, narratives, and overall strengthening of the player's experience. These motivate the player to continue and, as a consequence, acquire more information or skill.

   Kiili [42] defines three essential aspects of a flow model in educational games: "immediate feedback, clear goals, and challenges that are matched to players' skill levels" [42].

3. Interactivity: Allowing the player to interact with other players or non-player characters, which engages the player even more.

4. Increased Complexity: Keeping the player challenged reasonably but not frustrated; providing game balance. Annetta [3] argues that this might be the most difficult aspect of game design.

5. Informed Teaching: Tracking players' performances, actions, and mistakes, serving as a feedback to both the instructors and the players.

6. Instructional: Delivering the educational content adequately to the player's existing knowledge to maximize learning.

Designing levels of the game for game balance

The player background and learning outcomes provide a baseline for designing individual game levels and are closely tied to achieving game balance. Pusey et al. [71] state that every level in an educational game must consider the player's competence. Beginners can usually solve only straightforward problems asking to apply a certain rule, procedure, or tool. Experts, however, can attempt complex, open problems that require combining different skills in a creative way.

Task duration is a significant factor in level design. As experts gain self-esteem from previous successes, they persist longer [69, p. 47]. On the contrary, less motivated students (typically beginners) can become frustrated by a lengthy task [69, p. 297]. Petty proposes that a sequence of learning activities should start with an easy, brief task [69, p. 49], so that the player experiences success early and is motivated. By gradually increasing the difficulty and the required time, the sequence should finish with a long, complicated task. Note that beginners are often overestimated and assigned tasks are too difficult for them [69, p. 49].

Prerequisites for individual game levels should be formulated, too, to aid in the following design of pretests. Based on my experience with existing games, I suggest that too many prerequisites for a level that are neither covered in previous levels nor assumed in the player background imply that the level is too complex and should be split.

Iteratively improving the game

Figure 4.1 summarizes the process of game design. Since the process is iterative[1], a new game should be played by both the domain experts and the target audience. The gameplay is then reviewed based on predefined evaluation criteria, the findings are transformed into an action plan, and applied to improve the game [69, chap. 49]. Achieving learning goals and other aims is confirmed by post-game tests, questionnaires, or other methods described by Petty [69, chap. 49].

1. Blizzard Entertainment is one of the leading companies in the field of computer games. During the development of Diablo III, balancing the game mechanics involved frequent changes based on trial and error. The game designer stated: "We refined [the game] through playing. It's time-consuming, but it gives high-quality results." [23]

Figure 4.1: The key steps in the process of educational game design

4.2 Building a toolbox for assessments

Diagnostic assessment of a learner before playing a cybersecurity game helps to reach game balance, which is crucial when trying to maximize the game's value for the player. However, diagnostic assessment is not implemented in current cybersecurity games (see Chapter 3). Although Section 3.2 pointed out that some cybersecurity games use player self-assessment, the results are often inaccurate.

The rest of this chapter aims towards proposing a methodology for creating a suitable diagnostic assessment. The process starts by defining a toolbox of components necessary for question design, building upon a method for developing peer instruction questions in cybersecurity education by Johnson et al. [41]. The method is based on the work of Beatty et al. [6]. Both papers extensively describe assessment of students, which served as an important part of learning itself, often replacing traditional lectures. On the other hand, the goal of this work is to quickly assess the players' already existing knowledge before playing a game, not necessarily teach them something new.

Note that the term "question" is also used for tasks, that is, the sentence need not finish with a question mark.

Goal \ Question	Name a tool for network scanning.	As a system administrator, type a command for scanning own network.
Content (What is the question about?)	Awareness (knowledge) of network scanning tools	Usage of network scanning tools
Process (How is the question answered?)	Remembering that `nmap` is a network scanning tool	Applying an `nmap` command
Metacognitive (What notions are reinforced?)	Network scanning is possible	Network scanning has a legitimate purpose

Table 4.1: Three question goals with example questions (tasks)

Question goals

Beatty et al. [6] state that every assessment question should have three goals. The *content goal* defines what knowledge and skills are tested. The *process goal* defines which cognitive processes from the Bloom's revised taxonomy[2] should be used and how. Finally, the *metacognitive goal* defines what beliefs about learning and the topic (in this context, cybersecurity) should be reinforced. Table 4.1 provides two examples.

Question tactics

Along with defining three goals when stating a question, one or more *tactics* can be used to help vary the questions. Since 6 of the 21 tactics suggested by Beatty et al. [6] are only viable for in-class teaching,

2. Bloom's revised taxonomy [60] is a model describing six cognitive processes that can occur when answering a question. The processes are: *remembering*, *understanding*, *applying*, *analyzing*, *evaluating*, and *creating*. In this order, each process can be considered more cognitively demanding than the previous. Still, it is hard to separate the processes; thus the hierarchy is not necessarily cumulative. Note that the taxonomy is also helpful when defining learning outcomes (described in Section 4.1).

I selected and described 15 tactics applicable in pretests for cybersecurity games. The names of the tactics (in *italics*) correspond to the original paper; the description is often adapted for the purpose of this work.

1. *Compare and contrast* multiple objects/processes, also describe a situation and ask about the effect of changing its aspects.
2. *Identify a set or subset* of objects/processes having some property.
3. *Rank variants*: order objects/processes based on some quality.
4. *Remove nonessentials*, that is, features unrelated to the question's goals (not necessarily all extra information).
5. *Include extra information* that is not used.
6. *Omit necessary information* needed for a complete answer.
7. *Use "none of the above"* as an answer choice.
8. *Multiple defensible answers*: offer more than one correct subset of choices[3] depending on the interpretation of the question.
9. *Answer choices reveal likely difficulties* and highlight student's misunderstandings. The choices can include common errors, misconceptions, or inelegant solutions to the problem.
10. *Constrain the solution* by hints to be used when answering.
11. *Extend the context*: present an advanced scenario after a student answers a simple question.
12. *Interpret representations* from the presented features, for example, code understanding, verbally interpreting graphs (such as network usage statistics), or inferring conclusions from logs.
13. *Qualitative questions* ask about concepts and their relations.
14. *Analysis and reasoning questions* require significant decisions.
15. *Strategize only*: identify (and possibly argue) about an approach for solving a problem without actually solving it.

3. To illustrate, consider a multiple choice question with several correct choices. If selecting any single one of the correct options is acceptable (see 1:n later), then the tactic is employed. If on the other hand, the player must select all the correct choices (see n:n later), then there is only one correct subset of choices.

Question types

Regardless of the tactics used, a posed question can usually be classified into one of the following three types: *multiple choice*, *free form*, or *matching*. Their description follows, along with comments on their purpose and usage.

Multiple choice questions can test any skill of the Bloom's revised taxonomy. Two choices are present in true/false questions; three choices are optimal [72], four choices are commonly used. In a 1:1 question, exactly one choice is correct. A 1:n question has several possible correct choices; it suffices to select only one of them. Finally, an n:n question has several possible correct choices, out of which all need to be selected. Offering reasonably sounding choices without giving unintended clues (in both content and appearance) reduces guessing [10, 57].

Free form questions offer no options; instead, the student types the answer. This carries some implementation problems: a precise answer format must be defined (for example, whether to accept "7" or "seven"), and the user input must be checked. As a result, it is easier to use free form questions when asking a closed question with an unambiguous answer; however, this often allows testing of only lower-order skills of Bloom's revised taxonomy [51].

Given two lists L_1 and L_2 of items, *matching* is a task of connecting items in L_1 to the items in L_2 based on some relationship. If L_1 is regarded as "questions" and L_2 as "answers", then *matching questions* can be considered a generalization of *multiple choice questions*.

The number of items in the lists can differ. There must be at least two items in L_1 (to fulfill the definition of matching). Increasing the number of elements in L_2 reduces the chance of guessing; 8 items are recommended [11]. If the player is confident in matching, for example, two questions out of four, it is then easier to match the remaining two questions if only two answers are remaining.

Matching tasks assess higher cognitive levels efficiently and describe student performance more accurately than true/false questions [11]. Their usage includes connecting terms with their definitions, tools with their usage scenarios, or concepts with their properties. Moreover, matching can be used to order steps of a process in a time sequence: L_1 simply contains items "Step 1", "Step 2", "Step 3", and so on; L_2 contains the individual actions.

4.3 Creating prerequisite testing questions

This section provides a guideline for creating prerequisite testing questions using the toolbox from Section 4.2. Related issues are addressed, including assessment validity and reliability (see Section 2.2).

Deciding on the purpose of the pretest

First, the goals of the assessment must be formulated. The pretest can be used formatively. After answering the questions, individual players receive "a medal and a mission feedback" [69, chap. 43], including both recognition for correct answers and a (reference to) relevant training to compensate for incorrect answers. The players can study the provided reference and proceed to the game afterward or start playing immediately, using the reference when needed. The assessment's results can also highlight players that need assistance. On the contrary, summative use of the pretest is categorizing the players, for example, for the purpose of creating balanced teams based on the results.

Rules and practical considerations for developing pretests

Pretest questions, especially their content and process goals, must be relevant to the test's purpose and the game's prerequisites (defined during the game design; see Section 4.1). The content of the questions should test a representative sample of the key knowledge and skills. All questions must have clear, concise and unambiguous wording [69, p. 500], following best practices of item writing [31]. Domain experts should check every question's formulation, consider possible effects on students' perception, and confirm relevance to the prerequisites.

The games require transferring the player's knowledge to practice. Therefore, when considering the process, testing for *understanding* and *applying* (for example, how something works or how it is performed) is more suitable than testing for *remembering* (for example, repeating textbook definitions[4]). Since critical thinking is crucial for security experts, it is even better to test for higher-order skills: *analyzing*, *evaluating*, and *creating*, although developing the questions to test these skills is

4. Even if testing for *remembering*, the questions should at least ask about relevant knowledge; unlike, for instance, the meaning of the abbreviation HTTP.

difficult. Pusey et al. [71] state that player evaluation should measure deep understanding by testing for domain knowledge, skill application, and creative problem-solving.

In the domain of IT, one problem can be solved using more than one tool. Therefore, the tools applied in the game need not be tested in the assessment, unless they belong to the category of "core security tools" (see Appendix B). The reason is that the particular tools can change over time, while the general principles usually stay the same. Moreover, if the pretest focuses only on one tool, an incorrect answer in the test does not imply the inability of finishing the game level. For similar reasons, a pretest should usually not ask about knowledge applicable only under very specific circumstances.

Finally, the pretest should be brief. By generalizing my experience with two KYPO games, which have the play time limit of around two hours, a rough estimate is that the test should take less than 10 minutes. The main motivation is that if a player performs poorly in a prerequisite test and decides not to play the game, considerable time is saved and can be used, for example, for studying. Another motivation is that large surveys might reduce response rate. For this reason, Fan and Yan [24] recommend the length of at most 13 minutes.

Scoring the pretest

If pretests are used summatively, a scoring mechanism must be employed. A scoring mechanism also helps when proving validity and reliability, as qualitative data offer a limited choice of statistical tests. For simplicity, this thesis suggests the dichotomous scoring method: awarding one point for a fully correct answer, and zero points for a partially or entirely incorrect answer per every question.

An overview of more advanced scoring methods is presented by Lesage et al. [47]. Scharf and Baldwin [73] mathematically compare common scoring approaches. Petty [69, chap. 44] offers further methods.

Aiming for validity

The importance of assessment validity was explained in Section 2.2. To check validity, a hypothesis is formulated that a certain score implies certain conclusions [20]. For example, a test designer might assume that

a player scoring zero points in the pretest will be unable to finish the game. Then, relevant data must be collected from multiple sources and analyzed to support or reject the hypothesis.

There are five aspects of validity [20]. *Content* validity is vital for written, objectively scored pretests [20]. It refers to the quality of questions: whether they are written and checked by experts, adequately sampling the domain, and following best practices of question-writing [31]. The questions must be based on the game's prerequisites. Further, if most learning outcomes are connected to *application* or higher cognitive levels, the questions should have corresponding process goals.

Response process validity means there were no errors in the processing of the assessment data, and that the score was meaningfully defined per question and accurately combined into a composite score. The scoring process must be documented, explained, and justified.

Internal structure validity refers to the test's statistical characteristics. Scores of questions measuring the same or related construct should be correlated more than scores of unrelated questions. Usually, a Cronbach's alpha coefficient is computed [21], measuring inter-relatedness of questions that are supposed to measure the same construct on a scale from 0 to 1.

Relationship to other variables examines the correlation between test scores and another measure of a related skill with well-known characteristics, such as in-game achievement or university course mark. In other words, various measures of the same variable should correlate with each other. Evidence showing no (or negative) correlation with a measure of an unrelated skill is also useful.

Consequences refer to the more or less subjective evidence of the impact of conclusions drawn from the assessment. Effects of false positives or false negatives (such as declaring a low-performing student having a high skill or vice versa) should be explored, for instance, by examining "the statistical properties of the passing scores" [20].

Achieving validity is an iterative process, which includes collecting data, analyzing them, and reflecting on their results. The more important is to draw accurate conclusions from the pretest, the more care should be given to proving validity.

If a statistically significant amount of players succeeds in both the pretest and the game or fails in both, then the test can be considered relevant to the game. Secondly, if players fail the test but succeed in the

	Succeeds in the game	Fails the game
Succeeds in the pretest	Pretest is relevant	Pretest is easy
Fails the pretest	Pretest is difficult	Pretest is relevant

Table 4.2: Possible outcomes of prerequisite testing and playing

game, the test can be regarded as invalid for the game and too difficult. Thirdly, if players succeed in the test but cannot complete the game, the test can again be considered invalid for the game and too easy. This is summarized in Table 4.2. The meaning of "succeeding/failing in the test/game" depends on the designer.

Aiming for reliability

Cronbach's alpha is a lower bound of reliability [30, 48] applicable for dichotomously scored tests. Subsequently, the standard error of measurement is calculated for the entire score data as $\sigma \cdot \sqrt{1-\alpha}$, where σ is the standard deviation of the total scores [21]. This error is used to form confidence intervals around the scores. A sufficient number of medium-difficulty questions adhering to the rules for question writing (see Section 4.3) is needed to improve reliability [21].

A recapitulation of essential phases

Figure 4.2 shows the main steps of the process of creating pretests.

Figure 4.2: The process of prerequisite test design

Answer \ Confidence	Pure guess	Unsure	Neutral	Quite sure	Absolutely sure
Correct	+10	+27	+37	+45	+50
Wrong	+5	−4	−16	−32	−60

Table 4.3: Scoring mechanism for pretests with confidence testing [33]

Extending pretests by confidence testing

Confidence testing can be employed to partially compensate for guessing the answers and also improve learning [34]. After responding to a question [34], a student rates a level of certainty in the answer on a five-step scale developed by Hassmén and Hunt [33]. The confidence categories (sometimes slightly renamed) and the logarithmic scoring method is given in Table 4.3. Gardner-Medwin [27] proposes a differently scored method of confidence assessment on a three-step scale. A disadvantage of both approaches is that it complicates measuring reliability, as Cronbach's alpha cannot be employed.

Further suggestions and remarks for improving pretests

One concept can be tested by a series of follow-up questions (using the tactic Extend the context; see Section 4.2) to improve the assessment's reliability. Incorrect answers of players can be logged and then analyzed to discover common misconceptions. Artificial intelligence and machine learning techniques can recognize complex answers in free form questions. A time limit can be set on each question to constrain the pretest and also to hinder players when looking up the answers[5], if this is unwanted. The appropriate time limit can be determined statistically, by logging the times of answers of players who answer a certain question.

5. My experience with solving the tasks in NetWars Continuous (see Section 3.3) is that although I often did not know a certain tool that had to be used, I was still able to google it.

5 Experiment design

This chapter describes an experiment investigating whether prerequisite test and self-assessment can identify learners' readiness before playing a KYPO cybersecurity game. At first, each player completes a pretest and a self-assessment questionnaire. The players then proceed to the game, where their performance is tracked using the generic game events. Finally, the players fill in a post-game feedback questionnaire.

Section 5.1 poses the research questions of the experiment. Section 5.2 provides details about the experiment participants. Section 5.3 describes the chosen cybersecurity game and data collected from it, arguing about their suitability for the experiment. Section 5.4 applies the ideas from Chapter 4 by creating a pretest relevant to the chosen game. Section 5.5 adds self-assessment questionnaire to the pretest. Section 5.6 adds post-game feedback questionnaire. Lastly, Section 5.7 selects relevant statistical methods for processing the acquired data.

5.1 Research questions

We seek to answer three research questions:

1. How accurately can self-assessment or pretest model learners' performance in a cybersecurity game?
2. Does confidence testing increase the model's accuracy?
3. How to construct accurate pretests for cybersecurity games?

Our hypothesized, expected answers to these questions are:

1. An objective prerequisite test is more accurate than subjective self-assessment. Still, the best solution is using a combination of different methods in diagnostic assessment (see Section 3.2).
2. Yes. Confidence testing addresses guessing in the quiz; therefore, it can increase the accuracy of skill modeling.
3. While Chapter 4 proposed the method for creating pretests to the best of my knowledge, this will be the method's first practical application. New insights on the topic will probably yet arise.

5.2 Participants

A total of 35 cybersecurity students and professionals of various levels of expertise, background, and nationality participated, covering a broad spectrum of the game's target audience. The participants voluntarily responded to open invitations from the KYPO team. Their only motivation was their interest, as they did not receive any incentives for taking part in the experiment. They had no previous contact with the organizers (including the author of this thesis), who served only as instructors for the game session. The participants were informed about the intended use of the acquired data solely for the purpose of this experiment. The data was anonymized during the statistical processing.

The 35 learners (28 male, 7 female) were divided into three 2-hour sessions to play the game. The first session included 10 computer science students (9 male, 1 female) of St. Pölten University of Applied Sciences, Austria. The second session consisted of 7 players (4 male, 3 female): employees of CSIRT of Pavol Jozef Šafárik University in Košice, Slovakia. The third session included 18 computer science students (15 male, 3 female) of the Faculty of Informatics, Masaryk University, Brno, Czech Republic. Before each session, the organizers communicated only a general focus of the game (penetration testing) and basic prerequisites such as the operating system used in the game (Linux). During the session, every participant could stop playing and leave anytime.

5.3 KYPO Information theft game

The participants played an Attack-only CTF KYPO game (see also Section 3.3) designed by Barták [5]. The topic of the game is information theft from a database server of a fictitious bank. Each player initially controls a single Linux host in an unknown network. The player must gradually gain and maintain access to other hosts that are a part of the bank's network infrastructure, and, finally, steal confidential information.

This mission is split into six levels, in which the players exercise penetration testing skills. Table 5.1 details the play time estimated by the game's author, the median of play time of 21 players from the game author's experiment [5], maximum possible score, the number of hints available, and a scoring penalty for taking all the hints in each level.

	Estimated time [min]	Measured time (median) [min]	Max. score	Offered hints	Penalty for all hints
Level 1	10	9	8	2	−2, −2
Level 2	10	11	12	2	−2, −3
Level 3	25	20	23	3	−1, −3, −2
Level 4	20	47	20	2	−2, −3
Level 5	20	22	22	2	−3, −4
Level 6	15	35	15	2	−5, −2
Total	100	144	100	13	−34

Table 5.1: Characteristics of the KYPO Information theft game

While playing, the total score and the game events listed in Section 3.3 are tracked for each player. Since the score was set arbitrarily by the game designer without any justification, it does not seem to be a reliable metric. However, according to the time estimates, the scoring method roughly follows the principle of summative assessment that more time-consuming tasks are given more points [69, p. 502]. Except in levels 4 and 6, which were the longest, also the players' times correspond to the proposed scoring method, approximately in the ratio "1 point per 1 minute of the task". As a result, the total score was used as a measure of the player's skill. This metric aggregates the numbers of:

- hints taken (by applying the scoring penalties after the level's correct flag is submitted),
- solutions displayed (zero points are received for the level if its solution is displayed), and
- levels finished (scored with a non-zero number of points).

For comparison, the number of levels finished was also used as another measure of a player's performance, which includes skill and perseverance. Unlike the score, which is a complex, rather obscurely computed metric, the count of completed levels is intuitive and straightforward.

Focusing on the number of hints taken or wrong flags submitted to measure a player's skill causes difficulties in statistical processing. Since not all the players are expected to complete all the levels, the total counts would be misleading; adjusting them to the number of levels completed can introduce bias. Considering the first few levels completed by the majority of participants is also unsuitable. Selecting only the part of the diagnostic assessment corresponding to the selected levels would compromise the validity of the short test. For these reasons, other metrics for measuring skill were disregarded.

Gameplay time can provide yet another point of view, but not on its own. A short time might mean that the player is skilled, or that she skipped through the levels. A long time might mean that the player wants to work out the solution slowly, or that he does not know what to do. There is also a tradeoff between asking for hints soon and completing the level soon or not taking hints and having possibly longer play time. Therefore, time was not considered.

5.4 Developing pretests for selected games

Before focusing on KYPO Information theft game, I developed pretests for three other cybersecurity games. This was done to recognize knowledge and skills required to play different games, test the methods proposed in Chapter 4, and understand the specifics of creating pretests.

First, I chose KYPO DDoS attack game originally designed by Neudert [62] and extended by the KYPO team. To examine other areas of cybersecurity, I also created testing questions for Avatao's learning path focused on applied cryptography, especially on weaknesses of certain ciphers and protocols. Finally, I created questions for ENISA training carried out in the CEP (see Section 3.3).

I focused on one of the three platforms (KYPO, Avatao, and ENISA) at a time and randomly selected several challenges. After completing each challenge, I analyzed the required knowledge, including theoretical concepts needed to be understood, actions needed to be performed, and tools needed to be used. What was important was the process and not the questions themselves; therefore, they are not included in the text. I did not experience any significant findings and concluded that the methods defined in Chapter 4 are relevant and straightforward to use.

General overview of the experimentally used pretest

The pretest for the KYPO Information theft game was created as the first prerequisite test for any KYPO game. The goal was summatively categorizing learners for the purpose of the experiment. When designing the pretest, I listed the key prerequisites and created one question for each of the first five levels of the game, which are specified below. After all the questions are posed, Table 5.2 at the end of this section summarizes goals, tactics, and types of questions used.

Level 1: Network exploration

The goal of the player is scanning a server to discover its open ports. To complete the level, the player should have a basic knowledge of UNIX shell, network services and ports, and a scanning tool, such as Nmap.

The question asks about the execution of a `ping` command to test basic understanding of shell commands and network principles. I chose a simplified task instead of directly asking about Nmap, because in this case, `nmap <IP address>` is almost a full solution to the level.

Question: What is the effect of the command `ping 10.0.0.3`?

- ✔ tests the reachability of a host with an IP address 10.0.0.3
- ✖ scans open ports of the server with an IP address 10.0.0.3
- ✖ error, incorrect syntax of the command
- ✖ measures the number of network hops to a host with an IP address 10.0.0.3

Level 2: Discovering CMS vulnerability

Upon discovering that port 80 is opened on the server and visiting the corresponding website, the player should notice that the HTTP server uses WordPress CMS. Scanning the application using `wpscan` tool finds a vulnerable version of a plugin that allows uploading images but omits to check the file type, creating a file upload vulnerability.

Since the player needs to exploit a file upload vulnerability, the question asks about the protection methods to indirectly test the understanding of this type of security flaw.

Question: Among the following choices, select all the possible methods of prevention against an unwanted file upload.

- ✔ whitelisting file extensions
- ✖ using database triggers
- ✔ limiting maximum file size
- ✖ saving data to an NTFS volume

Level 3: Web exploitation

In this level, the player exploits the previously found vulnerability using Metasploit, allowing access to the HTTP server.

The question asks about the knowledge of the tool since it must be used to finish the level.

Question: Which of these tools can be used to run a malicious code against a remote machine?

- ✖ Hydra
- ✖ Burp Suite
- ✖ Aircrack-ng
- ✔ Metasploit

Level 4: Stealing credentials

After gaining access to the HTTP server, the player should examine an unprotected file `wp-config.php` containing the credentials for the WordPress database. The database stores a login name of a user, whose password can be cracked by performing a dictionary attack (using Medusa, for example). Subsequently, the player can log in to the HTTP server of the bank's employees[1] with this username and password.

The question tests the knowledge of common password attacks using matching to test different concepts at once.

1. In the game, this server is called "the HTTPS server" [5].

Question: Connect the following exemplary situations with the corresponding type of password attack.

a) trying all possible alphanumeric combinations of 8 characters
b) trying common words of English language
c) looking up the value of a hashed password
d) tricking a user into giving away his password by posing as a service administrator

1. dictionary attack
2. social engineering
3. brute force attack
4. rainbow table attack

Level 5: Privilege escalation

In this level, the player must copy a file readable only for superusers. By exploiting the CVE-2015-8660 vulnerability, the player gains root privileges on one of the network hosts. Afterward, the task is to move the file to the HTTP server of the bank's employees and run it.

The question tests that a given command requires root privileges, on a quite specific example in a particular problem setting.

Question: Consider a UNIX system with two regular users: `alice` and `bob`, who are in distinct groups. `alice` owns a file `secret.txt`. `bob` executes a command `chown bob secret.txt`. Assuming the system does not define POSIX capabilities, which of the following happens?

✖ The command execution fails, since the users are not in the same group.
✖ The command execution completes normally.
✔ The command execution fails, since in this setting, `chown` requires `root` permissions.
✖ Insufficient information: it is impossible to decide.

Level 6: Information stealing

The player is informed that the employee server can remotely access a database with sensitive information by using a private key and a certificate. These files are readable by the superuser; thus the player can copy them and access the database. Afterward, a hash from one of the tables must be retrieved and inverted, yielding a secret message. Knowledge for this level was not tested: I assumed that completing the first five levels implies having the prerequisites to finish the game.

Scoring the pretest

As mentioned in Section 4.3, each fully correct answer was awarded one point. Partially or entirely incorrect answers were awarded zero points. For comparison, confidence testing was included as defined in Section 4.3. A sum of the respective scores was considered as an estimate of each learner's total readiness.

Addressing validity and reliability

Content validity was achieved by using well-established rules for question-writing. The thesis' supervisor verified and approved the relevance of the questions to the game. Response process validity was reached by using scoring methods with known properties. Moreover, only a summation was applied to create a composite score. Internal structure validity and reliability were examined using the Cronbach's alpha. Relationship and consequential validity is discussed in Section 6.2.

Implementation details

The players were surveyed using Google Forms. Afterward, a custom Python script connected to Google Drive using `gspread` module, authenticated using `oauth2client` module with a generated private key, and extracted a `pandas` data frame for further processing[2]. All questionnaires required the player's ID to match the answers to the game actions, which were logged in a CSV file with the following structure: `ID, timestamp, logical time, level, event`.

2. The approach is based on the blog Practical Business Python [56].

	Content goal	Process goal	Metacognitive goal	Tactics	Type
1	Knowledge of `ping` and its purpose	Understanding the tool's execution	It is possible to test reachability of network hosts	Interpret representations, Qualitative questions	Multiple choice (1:1)
2	Protection against file upload vulnerability	Understanding the vulnerability, analyzing the preventive measures	The protection against unwanted input is needed	Remove nonessentials, Qualitative questions	Multiple choice (n:n)
3	Tools for exploiting network hosts	Analyzing the problem and selecting a correct tool	It is possible to exploit a vulnerability of a service	Qualitative questions	Multiple choice (1:1)
4	Common attacks on passwords	Understanding the meaning of the given terms	There are different strategies for password attacks	Remove nonessentials, Qualitative questions	Matching
5	Only a superuser can take ownership of a file	Analyzing the given conditions	Some commands require root privileges	Use "none of the above", Qualitative questions, Analysis and reasoning questions, Include extra information	Multiple choice (1:1)

Table 5.2: Summary analysis of the prerequisite testing questions used in the experiment

5.5 Self-assessment questionnaire

Apart from the prerequisite test, each player completed a self-assessment questionnaire before starting the game. The survey asked the players to self-evaluate their expertise with using three tools applied in the game: for port scanning (Nmap), vulnerability exploiting (Metasploit), and password attacks (John the Ripper). For each tool, the player selected one of four levels of competence on the following ordered scale: zero experience, beginner (basic knowledge), intermediate (some practical experience), and expert (professional working experience).

Based on the traditional Stevens' typology [75], the self-assessment data are ordinal. In general, ordinal data should not be aggregated quantitatively by assigning numerical values to them and performing mathematical computations, such as the arithmetic mean. Therefore, the median is used to express a central tendency of each player's self-assessment. Median is a metric recommended for ordinal data, as it is permissible under all circumstances [75].

Finally, values 0, 1, 2, and 3, respectively, were assigned to the steps on the self-assessment rating scale. These values were chosen for simplicity; any order-preserving transformation of the scale (for example, to values 0, 1, 42, and 1024) would be admissible [75].

5.6 Post-game feedback questionnaire

After finishing the game, the participants completed a post-game feedback questionnaire. The goal of the survey was to have each player subjectively assess the game's difficulty on a Likert scale from 1 (trivial) to 5 (impossible) and reflect on if any learning occurred. This reflection helps to determine if the game balance was achieved, and if the player perceived the game as educational. Moreover, the Kolb's cycle of experiential learning states that reflection must follow after a particular activity if learning is to take place [69]. Therefore, the survey not only generates data for the experiment but as a bonus provides a means for the players to contemplate on their experience.

The complete wording of the questionnaire is available in Appendix C. The implementation of both self-assessment and post-game feedback surveys was the same as the pretest's.

Figure 5.1: Phases of the experiment and the most relevant information collected in them

5.7 Statistical processing of the data

Figure 5.1 summarizes the phases of the experiment and the key information acquired in them. The quantitative data analysis aims towards two main directions. One is reporting basic descriptive statistics, contingency tables, and correlation coefficients of the selected variables to look for any trends in players' actions. The other is creating linear regression models describing the relationship between the players' skill and the skill measurement.

Descriptive statistics, contingency tables, correlation

Minimum, maximum, average, and median will be reported for six variables: self-assessment, prerequisite test (both with and without confidence testing), total score, the number of levels completed, and difficulty estimates. All 15 possible pairs of variables will be examined using contingency tables and correlation coefficients.

For contingency tables, Fisher–Freeman–Halton's statistic (shortly Fisher's exact test or simply Fisher's test) will be reported to test for any relationship in the frequency distributions. Unlike the χ^2 test, Fisher's test performs well even with small sample sizes and does not require the data to be normally distributed [50]. Since the tables will always be larger than 2×2, a two-sided p-value will automatically be used.

Both Pearson and Spearman correlation coefficients will be reported for the pairs of variables. Contrary to popular belief, Pearson correlation "is extremely robust with respect to violations of assumptions" [63], eliminating the need to use large sample sizes or assume the normal distribution. Two-sided p-values will be used to provide a more conservative estimate, and due to having no assumptions about the alternative hypothesis [49]. The value of $p < 0.05$ will be regarded as statistically significant.

The quantitative analysis will disregard several variables resulting from the game events. The score reflects the number of hints taken and the number of solutions displayed; the latter is also reflected in the number of levels completed, as displaying a solution implies not completing the level. The number of wrong flags is misleading, as incorrect attempts sometimes arise from misunderstandings of required flag format rather than lack of skill. Gameplay time is omitted because of reasons stated in Section 5.3. Instead, a qualitative examination of individual players' game events will address the omitted variables.

Linear regression model of the players' skill

Let P_a be the prerequisite test's result using the dichotomous scoring, ranging from 0 to 5. P_b is the prerequisite test's result using the confidence testing scoring, ranging from -300 to 250. S denotes player's self-assessment, a median of the three answers assigned a number from 0 to 3 to the ordered scale. T denotes the player's total in-game score, ranging from 0 to 100, and L the number of levels completed by the player, ranging from 0 to 6.

Pretest and self-assessment data are used to create a linear regression model of learners' skill, which is expressed by two metrics: the total game score and the number of levels finished. In general, linear regression models a dependent variable y using k independent variables $x_1, \ldots, x_k$ by an equation $y = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$, where $\beta_i \in \mathbb{R}$ [83]. Here, the dependent variables T and L are modeled using selected combinations of the three predictors: independent variables P_a, P_b and S.

A total of 10 models will be compared. First, T will be modeled using S, P_a, P_b, $S + P_a$, and $S + P_b$. Then, the corresponding models will be constructed analogously to describe L. Constructing per-level models was disregarded due to the small sample size.

6 Experiment results

This chapter lists and discusses the results of the experiment, as well as the study limitations.

6.1 Analysis of collected data

The six reported variables are: self-assessment (S), dichotomously scored pretest (P_a), pretest with confidence testing (P_b), total game score (T), the number of levels completed (L), and difficulty estimate (D). The data were collected from 35 participants. However, the results of P_b and D are available only from 25 players due to a technical error.

Descriptive statistics, contingency tables, correlation

Table 6.1 reports the examined variables and descriptive statistics of the collected data, which are further detailed in Figure 6.1. Table 6.2 lists the results of the tests defined in Section 5.7. Based on the Fisher's test, statistically significant relations emerged between D and all the variables except P_b and between T and L. As expected, there were almost no differences between Pearson and Spearman coefficients. Negative correlations were found between D and all the variables. P_b positively correlated with S, P_a, T, and L. P_a positively correlated with T. Unsurprisingly, T and L were extremely strongly correlated.

Variable		Possible range	Min	Max	Avg	Med
Self-assessment	S	0 to 3	0	2	0.8	1
Pretest (binary)	P_a	0 to 5	0	5	3.9	4
Pretest (conf.)	P_b	−300 to 250	−91	227	141.0	155
Game score	T	0 to 100	0	85	47.1	53
Levels finished	L	0 to 6	0	5	3.4	4
Difficulty	D	1 to 5	2	5	3.3	3

Table 6.1: Examined variables and descriptive statistics of the data

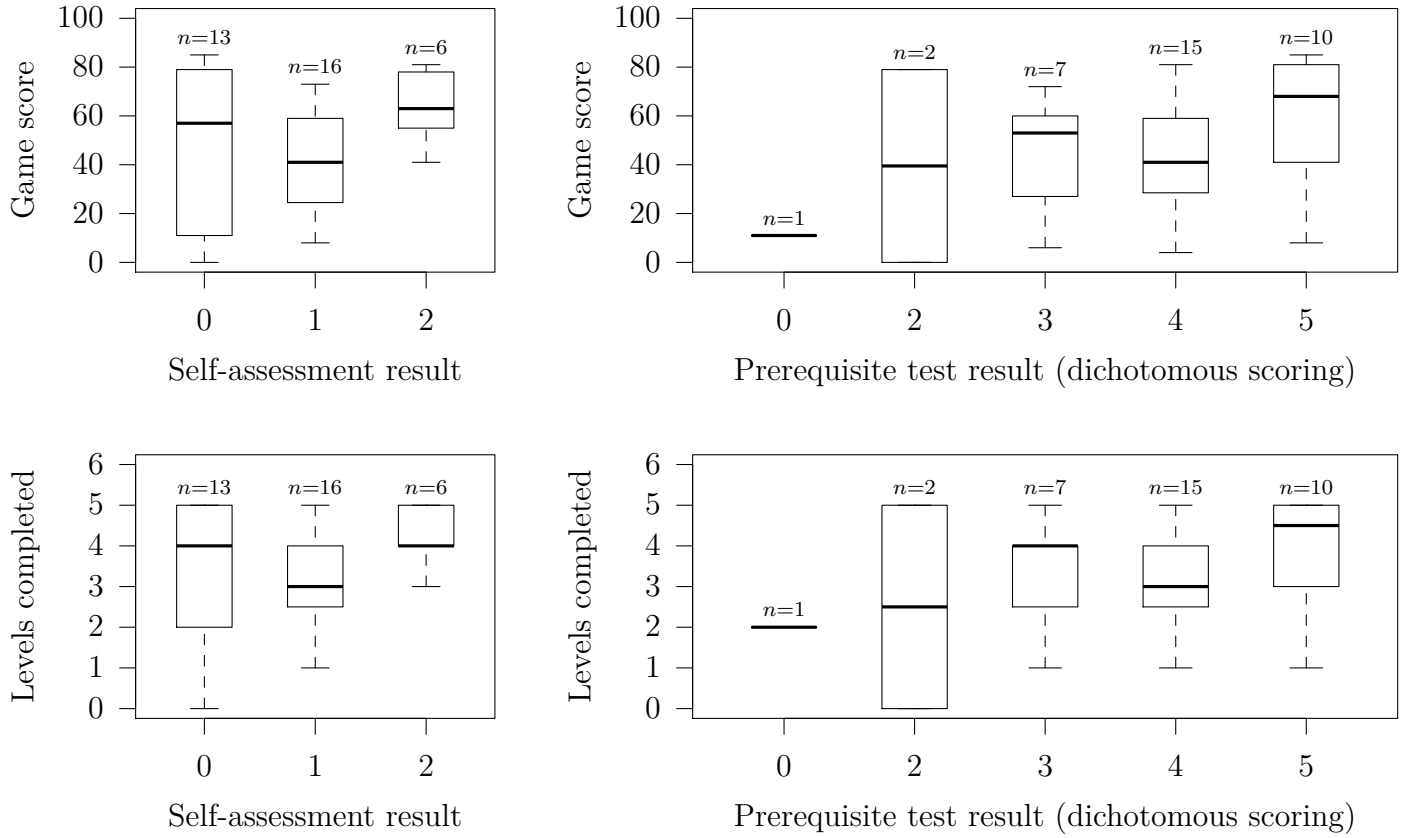

Figure 6.1: The boxplots illustrate relationships between performance predictors (self-assessment and dichotomously scored pretest) and skill descriptors (game score and the number of levels completed); the thick black line expresses the median. No one scored 3 points in the self-assessment or 1 point in the pretest.

Variables	n	r	sig.	ρ	sig.	Fisher sig.
S vs. P_a	35	0.23	0.193	0.14	0.436	0.204
S vs. P_b	25	0.64	< 0.001	0.71	< 0.001	0.136
S vs. T	35	0.16	0.355	0.10	0.560	0.403
S vs. L	35	0.16	0.346	0.08	0.663	0.074
S vs. D	25	-0.55	0.004	-0.59	0.002	0.032
P_a vs. P_b	25	0.87	< 0.001	0.73	< 0.001	0.192
P_a vs. T	35	0.34	0.047	0.33	0.050	0.226
P_a vs. L	35	0.31	0.072	0.28	0.102	0.308
P_a vs. D	25	-0.47	0.025	-0.48	0.014	0.022
P_b vs. T	25	0.46	0.020	0.56	0.004	1.000
P_b vs. L	25	0.39	0.051	0.48	0.015	1.000
P_b vs. D	25	-0.51	0.010	-0.64	< 0.001	0.213
T vs. L	35	0.98	< 0.001	0.98	< 0.001	0.011
T vs. D	25	-0.59	0.002	-0.59	0.002	0.025
L vs. D	25	-0.57	0.003	-0.52	0.007	0.036

Table 6.2: Pearson correlation (r), Spearman correlation (ρ), and the result of Fisher's test for all pairs of examined variables, with the correlated variables and statistically significant two-sided p-values highlighted

Linear regression models

Table 6.3 reports the regression models. Statistically significant fits were computed for the score prediction based on the pretest ($T = 15.56 + 8.18 \cdot P_a$, $R^2 = 0.11$). Even more promising relationship emerged when incorporating confidence testing ($T = 15.39 + 0.17 \cdot P_b$, $R^2 = 0.22$). None of the coefficients in the other models showed statistical significance. The two best fits are shown in Figure 6.2 and detailed in Appendix A.

Model	β_0 sig.	β_1 sig.	β_2 sig.	R^2	Adj. R^2	F-test sig.
$T = 42.40 + 5.90 \cdot S$	< 0.001	0.355	—	0.026	−0.004	0.355
$T = 15.56 + 8.18 \cdot P_a$	0.334	0.047	—	0.114	0.087	0.047
$T = 15.39 + 0.17 \cdot P_b$	0.156	0.020	—	0.215	0.181	0.020
$T = 14.82 + 3.28 \cdot S + 7.69 \cdot P_a$	0.365	0.603	0.071	0.122	0.067	0.125
$T = 16.52 + 10.17 \cdot S + 0.10 \cdot P_b$	0.127	0.227	0.256	0.266	0.200	0.033
$L = 3.12 + 0.32 \cdot S$	< 0.001	0.346	—	0.027	−0.003	0.346
$L = 1.85 + 0.39 \cdot P_a$	0.037	0.072	—	0.094	0.067	0.072
$L = 1.89 + 0.01 \cdot P_b$	0.004	0.051	—	0.155	0.119	0.051
$L = 1.81 + 0.19 \cdot S + 0.36 \cdot P_a$	0.043	0.565	0.107	0.104	0.048	0.173
$L = 1.97 + 0.69 \cdot S + 0.01 \cdot P_b$	0.002	0.143	0.513	0.236	0.166	0.052

Table 6.3: Linear regression models with the best fits and statistically significant two-sided p-values highlighted (for the description of the variables, see the beginning of Section 6.1)

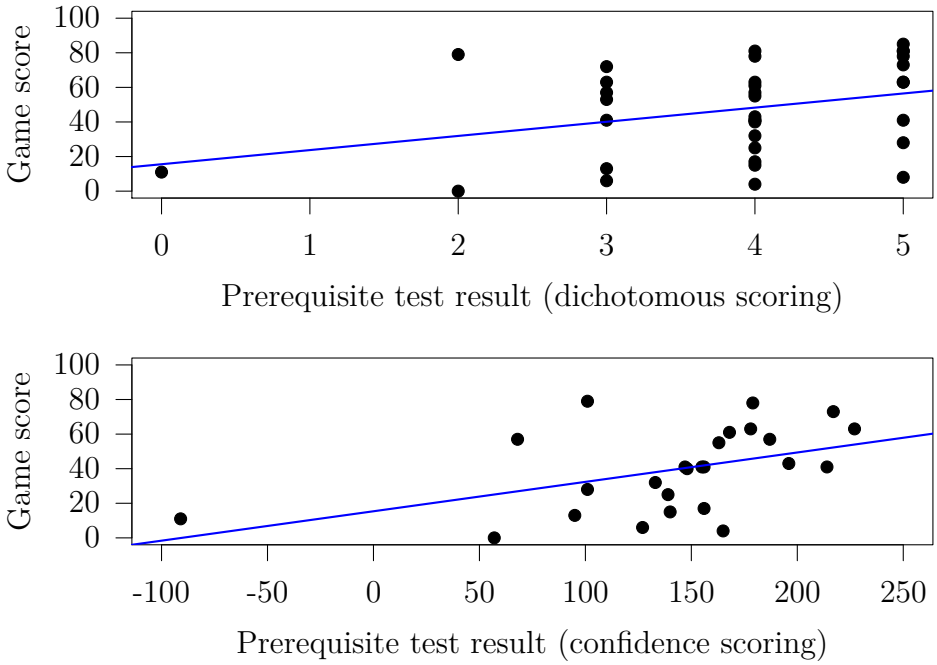

Figure 6.2: Linear regression models describing game score by pretest

Game events

Figure 6.3 details all the participants' game events, using the tool by Uhlár [78]. Each line represents the actions of one player distributed in time. Levels are marked as colored line segments: dark blue displays level 1, light blue level 2, and so on. If the level ends by submitting a correct flag, a big circle with the level's color is shown. If the level is skipped, it is marked by a small black circle. A small red circle refers to submitting a wrong flag. A small gray circle means taking a hint; a small purple circle portrays opening or closing a level's solution (also called "a help level"). Each gameplay ends in one of four ways:

- Requesting a premature exit (big black circle; example: #10).
- Submitting the final flag (big green circle; example: #11). Note that this happened only after displaying the solution.
- Skipping through the levels (big green circle with a light border and dark center; example: #12).

- Stopping playing (example: #2). As this event has no timestamp, the length of the last level's line segment cannot be determined.

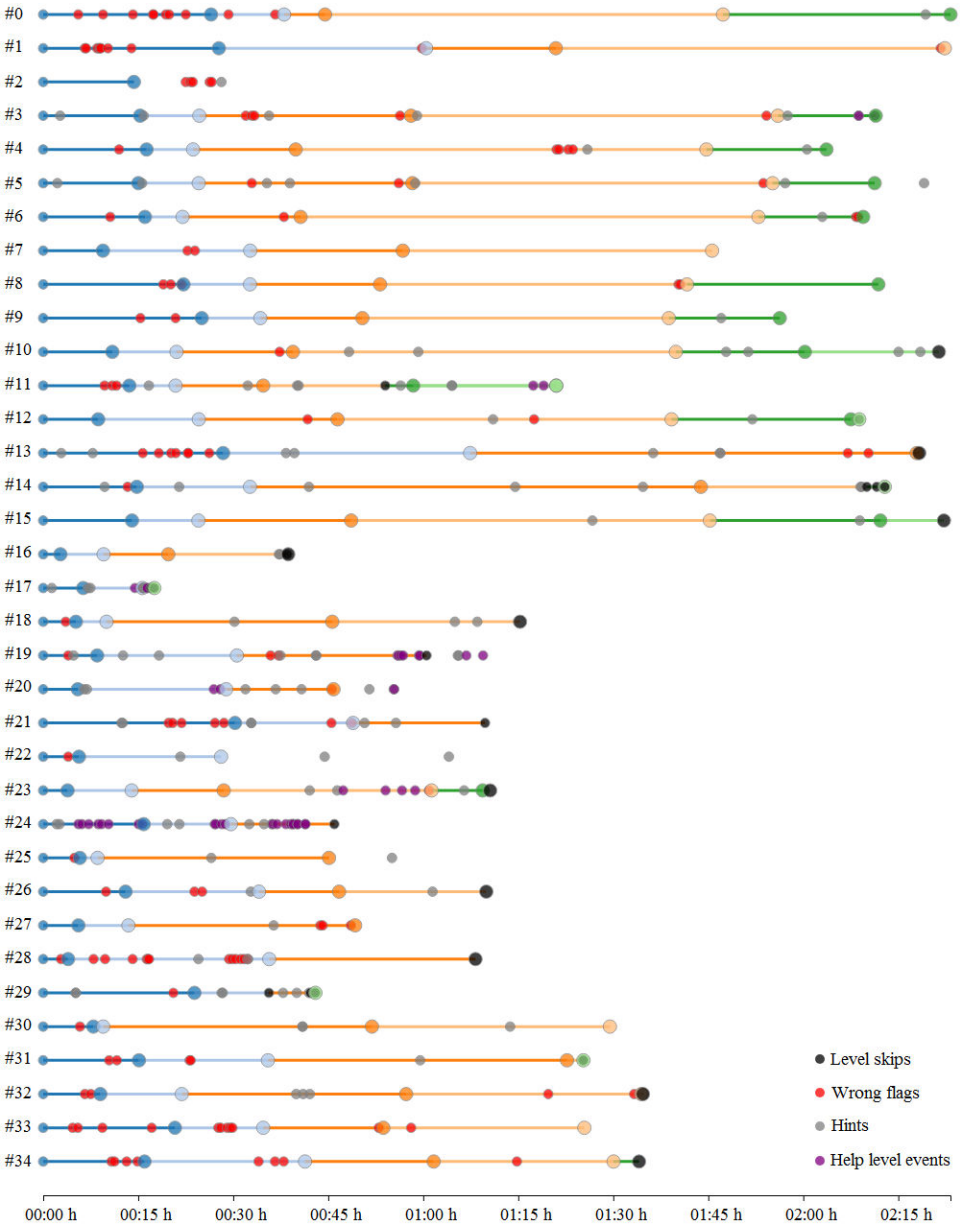

Figure 6.3: Participants' game events distributed in time (see Section 6.1)

Post-game feedback

The overall summary evaluation of post-game feedback yielded the following results: out of 25 players, 0 perceived the game as trivial, 1 as easy, 16 as a medium, balanced challenge, 7 as hard, and 1 as impossible. Regarding their existing knowledge, 0 participants felt overqualified, 6 felt that their current knowledge and skills matched what the game required, 17 felt they were missing some knowledge or skills, and 2 participants did not choose any of the options.

The participants were asked whether they learned something new about attacks, defense, or cybersecurity tools. 22 players reported learning in at least one of these three categories, the most frequent being attacks (20 times), followed by tools (15 times), and, despite the fact the game was Attack-only, also about defense (3 times). The remaining 3 players did not answer, but no one reported learning nothing.

Learning cybersecurity practically was appreciated by 21 participants, and 23 players considered the game a good practice. As for their most significant learning experiences, the participants appreciated learning about new tools (Metasploit was often mentioned), and about how and when to use them. The players often valued the opportunity for hands-on training and applying theory in practice, and they perceived the game as "fun". Some of the players commented that they realized it is relatively easy to hack something using existing tools.

Comments on the gameplay included minor technical difficulties related to the platform ("The game sometimes froze and I had to reset the virtual machine"; "The window in the virtual machine is too small, it'd be nice if it'd be resizable", "Copy-paste would be nice"). Levels 3 and 4 were perceived as rather difficult. Sometimes, frustration about the requirement to fill in pretest survey was mentioned. Nevertheless, the majority of participants appreciated the new experience.

6.2 Discussion

Quantitative view

Returning to the first research question posed in Section 5.1, the models accurately predicted the learners' total game score when employing prerequisite testing, regardless of the scoring method used. Overall, the

results confirmed the intuition that players with a high score from the pretest would often be performing better in the game compared to the others. The reverse was also true.

However, the differences in game performance among the observed self-assessment levels are slight (see medians in Figure 6.1). Judging by the p-values and the values of R^2, using self-assessment in the regression analysis introduced uncertainty, and the models seem unlikely to reach statistical significance even if the sample size would increase. This result is consistent with previous results by Mirkovic et. al [54], who used a similar self-assessment scale (see Section 3.2). A possible explanation is that experience with using in-game tools is not a key factor in deciding learners' readiness. Most of the players had worked with Linux Terminal before the game, thus were able to discover and understand the application of other command-line tools for themselves.

Considering the second research question, confidence testing of the 25 learners revealed guessing: 6 players randomly guessed at least one correct answer. It also showed misconceptions, since 5 players were quite or absolutely sure about at least one incorrect answer. As expected, this information improved the models: the value of R^2 doubled compared to the dichotomous scoring of the pretest.

Several results support the pretest's validity. Regarding relationships to other variables, pretest results (both scoring methods) negatively correlated with how difficult the game was perceived and positively with the game score. Moreover, the confidence testing result positively correlated with the count of levels finished. While the correlations were not extremely strong, Fisher's test further showed dependence between dichotomously scored pretest and the difficulty estimate. Consequential validity is problematic, as players scoring either low or high in the pretest performed either poorly or well in the game. The reason might be that the result of a knowledge quiz was used to model practical skill.

As for the reliability of the dichotomously scored pretest, the reported Cronbach's alpha was $\alpha = 0.44$, and the standard deviation of the pretest result was $\sigma = 1.07$, implying a measurement error of at most ± 0.8 points. Cronbach's alpha when question removed ranged from 0.28 (after removing the question 3) to 0.52 (after removing the question 2). The rather low alpha is not an issue in this context. On the contrary, it supports the expectations, because it results from the small number of questions that measured inherently different constructs.

Qualitative view

It can be argued that applying statistical tests and using regression models on a small sample might bias the quantitative results. Therefore, the actions of individual players were further explored from the time perspective (see Figure 6.3). Several notable anomalies were identified and are addressed below. For simplicity, all the players are referred to as males, even if their gender is unknown.

Player #2, whom we nicknamed "the quitter", reached the full score in the binary pretest but got frustrated as early as in the level 2. Over the course of less than 6 minutes, he attempted 5 wrong flags, took a hint, and stopped playing. As a result, his game score was only 8 points. However, if I removed his data and recomputed the model of the game score based on the binary pretest, the regression's accuracy would improve: p-value would drop to 0.02 and R^2 would rise to 0.16. Moreover, the model predicting the number of levels finished would reach statistical significance ($L = 1.61 + 0.48 \cdot P_a$, $R^2 = 0.15$, $p = 0.02$).

Player #16 ("the achiever") followed the same pattern as "the quitter": at the beginning, he seemed like a competent learner but got frustrated with the game. "The achiever" scored 4 points in the binary pretest and solved the first three levels surprisingly quickly: in 3, 7, and 10 minutes, respectively (compare the times with Table 5.1). Shortly before the time limit for the fourth level ran out, he quickly took both available hints and then prematurely exited the game. As a result, he scored a below-average 43 game points. However, he later reported not knowing that the time limit was only informative and had no impact on the game. Instead, the player thought that if the time runs up, he cannot play anymore, which annoyed him and caused him to quit the game. Based on the interaction with "the achiever" during the session, we felt that he was quite competitive, aiming to reach a high score in the game, but got demotivated while playing.

We hypothesize that "the achiever" and "the quitter" had possessed the necessary prerequisites for finishing the game. However, they were thwarted by ambiguous game mechanics or design, by the insufficient attention paid to the rules, or by some other reason. Due to unanticipated situations like these, the dataset includes players scoring well in the pretest but poorly in terms of game score or levels completed. This might have introduced noise in the regression models.

Player #21 ("the underdog") was the only one scoring 0 points in the pretest, which was rather easy for most of the other players. Moreover, he reported zero experience with all the tools in the self-assessment. He requested both hints in the first level, but it took him 5 incorrect attempts and 30 minutes to finish the level: the longest time from the sample. His course of action was similar in the second and the third level, where he gave up. Unlike in the previous two examples, it seems his reason for quitting was the lack of skill, and the resulting score of 11 game points corresponds to the pretest. Still, the participant persisted in playing the game for 70 minutes.

Another interesting case is player #12 ("the determined one"), who scored 0 in the self-assessment and 2 points in the pretest, in which only two other players had the same or worse result. Still, he completed 5 levels and scored 79 points in the game: one of the best results in the sample. The player used only 2 hints and attempted only 2 incorrect flags in total, all in the later phases of the game. The time spent in the five levels was 9, 16, 22, 53, and 28 minutes, respectively. The rather long time is reflected in the post-game feedback, where he rated the game as hard (4). Although the player did not possess theoretical knowledge from the pretest, his determination allowed him to perform very well.

Finally, the player #5 ("the practitioner") scored 0 in the self-assessment and 3 points in the pretest. However, by taking some hints, using the means of trial and error, and given enough time he was able to complete 5 levels and score 72 game points, which is a good result. This player, like "the determined one", might not have had the theoretical background, but was still able to solve the practical tasks.

These case studies show that many different, unanticipated aspects influence players' performance. An arising challenge is recognizing and deeply understanding all factors contributing to a successful game play. We believe that solving this challenge is essential for designing a useful diagnostic assessment and, by extension, the whole game.

6.3 Study limitations and lessons learned

One of the main difficulties in working on this thesis was finding participants for the experiment. Since the learners participated voluntarily, this could have caused self-selection bias. Furthermore, the vast majority

of participants shared common characteristics. The average participant was 20 to 30 years old, male, computer science student at the university level. However, this limitation is difficult to address, as the KYPO game is rather time-consuming and has narrowly focused target audience.

During the first session, the 10 players were strongly frustrated by the game's difficulty. As a result, they were allowed to play in pairs, if they wanted, knowing that it could bias the experiment data. However, their learning experience was a priority for the organizers. During the second and the third session, the game was played individually. As a result, the data involving confidence assessment and difficulty estimate, which were collected only from the remaining 25 participants, are considered unbiased. Apart from this deviation, all the players participated in the experiment under the same conditions.

When using questionnaires in any study, some participants ignore the instructions [68], skip through the questions, or even select answers randomly. Nevertheless, the collected data are assumed to be proper, since filling in the surveys took from 4 to 7 minutes on the self-assessment and pretest, and from 2 to 5 minutes on the post-game feedback.

Since only a few participants received a low score in the pretest, regardless of the scoring method, the regression models might be biased when predicting the low-performing players (see Figure 6.2). Increasing the difficulty of the questions can address this limitation. While the values of R^2 in the models are relatively low, they can be attributed to the unanticipated factors influencing a successful gameplay, which were discussed during the qualitative exploration of in-game events.

Despite using a well-established framework for question-writing and following best practices of assessment design, two major challenges of prerequisite testing arose. One is calibrating the test to predict the possession of skills most relevant to the game. While the players often performed well in the quiz, no one finished the last level of the game. It seems that theoretical knowledge might not be enough for succeeding in practical tasks. Considering the qualitative analysis, we believe that knowledge and experience are not the only factors determining success in a cybersecurity game. Personal characteristics that were not tested might also be relevant. For instance, some players refused to take hints in their pursuit of scoring as many points as possible.

The other challenge was a limited time frame for participant assessment. It is impractical and discouraging to perform a lengthy examina-

tion when the learners are eager to play the game. Both the test and self-assessment combined were designed to take at most 8 minutes, yet were perceived by some of the players as an inconvenience. At the same time, it is hard to achieve validity and reliability with a small number of items.

Ultimately, successful diagnostic assessment largely depends on the quality of game design. This thesis attempted to prove the validity of the proposed prerequisite test based, to some extent, on its relationship to the game. However, if the game scoring mechanism or individual levels are poorly designed, this can invalidate the pretest. Therefore, we underline the need for careful consideration of educational game design.

While the experiment proved a link between prerequisite testing and players' performance in the KYPO game, the generalizability of the results might be questioned because of the dependence on the particular game and its scoring method. This might change when designing and implementing pretests, and, by extension, cybersecurity games differently. Inspired by the results of Lee at al. [46], who report positive effects of assessments in educational games, the third research question is addressed by proposing two main improvements.

One is dissolving the assessments into the story of the game. Compared to using questionnaires, which distract the players and shift them into a "testing mode", in-game tests are more engaging [46]. They also allow using more assessment questions, which, in turn, brings more validity and reliability in the results.

This approach of employing in-game testing necessitates another improvement: in the design of the game itself. Individual levels can be created such that each has only one particular learning outcome. Appropriate prerequisites can be tested before or during that level. Alternatively, if the levels require similar skills that build on each other, the testing need not be performed during the whole game, but only for the first few levels. As a result, the need for predicting readiness for the entire game during a single test before the start of the gameplay is eliminated.

7 Conclusions

Creating this thesis required combining knowledge from different fields, including cybersecurity, serious games, educational assessment, statistics, and data analysis. The major contribution of this pioneering attempt is providing new insights into the area that is, to the best of our knowledge, not widely researched. While investigating prerequisite testing of cybersecurity skills is by no means at an end, this work provides a stepping stone for the scientific community to explore the area further.

My research started by creating a general methodology for developing cybersecurity games and prerequisite tests. The method resulted from a thorough literature review and exploration of game elements in state-of-the-art cybersecurity platforms. The methodology was applied in practice to create the first prerequisite test for a KYPO cybersecurity game. The pretest, along with self-assessment, were used in an experimental study investigating their predictive value for identification of learners' readiness before playing the KYPO game.

The analysis of game events and information provided by players showed that when describing the players' performance, the knowledge quiz exhibited larger accuracy than self-assessment. The linear regression model predicting game score based on the pretest was statistically significant, and one of the key results was the improvement of the fit after incorporating confidence assessment. Moreover, the exploratory analysis of game events indicated that including other components of readiness might further increase the models' accuracy. Statistically significant positive correlations were reported between score and pretest (regardless of the scoring method), and between levels completed and pretest with confidence testing. Finally, there was strong evidence that both performance predictors and skill descriptors negatively correlate with how difficult the game is perceived.

The lessons learned from the experimental study are vast, the most notable being the implications on game design. KYPO games were previously constructed using an intuitive, unstructured approach, which was time-consuming and sometimes departing from instructional design guidelines in literature. The proposed methodology and the results of the study can aid designers in creating future games of higher quality, with thorough consideration for learning outcomes and players' engagement.

The post-game feedback from the players, who rated the enhanced KYPO game as educational, practical, and interesting supports the belief that active learning in cybersecurity is worthy of both security practitioners' and educators' attention.

7.1 Future work

This work motivated the development of an open-source tool for visualization of the game events (see Figure 6.3). The tool allowed discovering important patterns that would otherwise stay hidden in the global statistics. The visualizer can be employed not only by researchers in a post-mortem analysis but also by tutors during a game session. Moreover, this work implied new theses topics, as the created pretest is being integrated into the KYPO using the LimeSurvey tool.

Since a player's achievement is influenced by many factors, one direction of future work is research on whether any personal characteristics are highly relevant to the success in a game. A different direction is integrating the means of diagnostic assessment into the game story, which would eliminate the requirement to minimize the quiz's length. Moreover, a game directly reflecting the assessment results, for example, by awarding more points or time for correct answers, can motivate the player and provide more detailed assessment data. To engage the players even further, they could place bets based on the level of confidence in their answers and subsequently gain or lose an advantage in the game.

Petty [69, chap. 44] suggests methods other than an objective test to measure a learner's practical skill. One is having an external examiner confirm a checklist of criteria, such as "the student can discover the open ports on a server with given IP address". Knowledge map of cybersecurity skills, similar to the one by Khan Academy [1], can aid in designing both skill checklist and assessment questions. Another approach is using a short cybersecurity game itself as a prerequisite test, constructed to match the skills required by the main game that would follow.

An in-depth exploration of available data can outweigh the disadvantage of the small sample. Analysis of command-line history, including pattern recognition, can be employed to measure in-game performance. An individual summary of the player's actions can be displayed afterward. Furthermore, a dynamic analysis can individualize learning by providing hints and other scaffolding, or even tasks, adaptively.

Bibliography

[1] Khan Academy. Knowledge Map. https://www.khanacademy.org/exercisedashboard [Online; accessed 2017-05-12].

[2] Jim Allen and Rolf Van Der Velden. The role of self-assessment in measuring skills. In *Transition in Youth Workshop, Valencia, Spain*, 2005.

[3] Leonard A. Annetta. The "I's" Have It: A Framework for Serious Educational Game Design. *Review of General Psychology*, 14(2):105–112, 2010.

[4] Avatao. Learn to build secure software. https://www.avatao.com [Online; accessed 2017-05-12].

[5] Miloš Barták. Security games for various skill levels. Master's thesis, Masaryk University, Faculty of Informatics, Brno, 2016. http://is.muni.cz/th/396568/fi_m/ [Online; accessed 2017-05-12].

[6] Ian D. Beatty, William J. Gerace, William J. Leonard, and Robert J. Dufresne. Designing effective questions for classroom response system teaching. *American Journal of Physics*, 74(1):31–39, 2006.

[7] Sylvia Beyer. Gender differences in the accuracy of self-evaluations of performance. *Journal of personality and social psychology*, 59(5):960, 1990.

[8] Paul Black and Dylan Wiliam. Developing the theory of formative assessment. *Educational Assessment, Evaluation and Accountability [formerly: Journal of Personnel Evaluation in Education]*, 21(1):5–31, 2009.

[9] Alicia Bolívar-Cruz, Domingo Verano-Tacoronte, and Sara M. González-Betancor. Is university students' self-assessment accurate? In *Sustainable Learning in Higher Education*, pages 21–35. Springer, 2015.

[10] Richard F. Burton. Multiple-choice and true/false tests: myths and misapprehensions. *Assessment & Evaluation in Higher Education*, 30(1):65–72, 2005.

[11] Madawa Chandratilake, Margery Davis, and Gominda Ponnamperuma. Assessment of medical knowledge: The pros and cons of using true/false multiple choice questions. *The National Medical Journal of India*, 24(4), 2011.

[12] Nicholas Childers, Bryce Boe, Lorenzo Cavallaro, Ludovico Cavedon, Marco Cova, Manuel Egele, and Giovanni Vigna. Organizing Large Scale Hacking Competitions. In *Proceedings of the 7th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, DIMVA'10, pages 132–152, Berlin, Heidelberg, 2010. Springer-Verlag.

[13] Tom Chothia and Chris Novakovic. An Offline Capture The Flag-Style Virtual Machine and an Assessment of Its Value for Cybersecurity Education. In *2015 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 15)*, Washington, D.C., 2015. USENIX Association.

[14] DEF CON. DEF CON® Hacking Conference. https://www.defcon.org/ [Online; accessed 2017-05-12].

[15] CTFtime. CTFtime.org / All about CTF (Capture The Flag). https://ctftime.org [Online; accessed 2017-05-12].

[16] Cybrary. Free and Open Source Cyber Security Learning. https://www.cybrary.it/ [Online; accessed 2017-05-12].

[17] Andy Davis, Tim Leek, Michael Zhivich, Kyle Gwinnup, and William Leonard. The Fun and Future of CTF. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*, San Diego, CA, 2014. USENIX Association.

[18] Information Assurance Directorate. Cyber Defense Exercise (CDX). https://www.iad.gov/iad/programs/cyber-defense-exercise/ [Online; accessed 2017-05-12].

[19] George T. Doran. There's a SMART way to write management's goals and objectives. *Management review*, 70(11):35–36, 1981.

[20] Steven M. Downing. Validity: on the meaningful interpretation of assessment data. *Medical education*, 37(9):830–837, 2003.

[21] Steven M. Downing. Reliability: on the reproducibility of assessment data. *Medical education*, 38(9):1006–1012, 2004.

[22] ENISA. Cyber Exercises Platform. https://www.enisa.europa.eu/topics/cyber-exercises/cyber-exercises-platform [Online; accessed 2017-05-12].

[23] Blizzard Entertainment. Diablo III Collector's Edition: Behind the Scenes. [Video documentary], 2012.

[24] Weimiao Fan and Zheng Yan. Factors affecting response rates of the web survey: A systematic review. *Computers in human behavior*, 26(2):132–139, 2010.

[25] Robert L. Fanelli and Terrence J. O'Connor. Experiences with Practice-focused Undergraduate Security Education. In *Proceedings of the 3rd International Conference on Cyber Security Experimentation and Test*, CSET'10, pages 1–8, Berkeley, CA, USA, 2010. USENIX Association.

[26] National Initiative for Cybersecurity Careers and Studies (NICCS). Cybersecurity Workforce Framework. https://niccs.us-cert.gov/workforce-development/cyber-security-workforce-framework [Online; accessed 2017-05-12].

[27] A.R. Gardner-Medwin. Confidence assessment in the teaching of basic science. *Research in Learning Technology*, 3(1), 1995.

[28] Mark Gondree, Zachary Peterson, and Portia Pusey. Talking about talking about cybersecurity games. *USENIX ;login:*, 41(1):36–39, 2016.

[29] Thavamalar Govindasamy. Successful implementation of e-learning: Pedagogical considerations. *The internet and higher education*, 4(3):287–299, 2001.

[30] James M. Graham. Congeneric and (essentially) tau-equivalent estimates of score reliability: What they are and how to use them. *Educational and psychological measurement*, 66(6):930–944, 2006.

[31] Thomas M. Haladyna, Steven M. Downing, and Michael C. Rodriguez. A review of multiple-choice item-writing guidelines for classroom assessment. *Applied measurement in education*, 15(3):309–333, 2002.

[32] Aman Hardikar. Penetration testing practice lab – Vulnerable apps/systems. http://www.amanhardikar.com/mindmaps/Practice.html [Online; accessed 2017-05-12].

[33] Peter Hassmén and Darwin P. Hunt. Human self-assessment in multiple-choice testing. *Journal of Educational Measurement*, 31(2):149–160, 1994.

[34] Darwin P. Hunt. The concept of knowledge and how to measure it. *Journal of intellectual capital*, 4(1):100–113, 2003.

[35] International Computing Education Research (ICER). ICER 2017. https://icer.hosting.acm.org/ [Online; accessed 2017-05-12].

[36] Insomni'hack. Insomni'hack: Swiss security conference and ethical hacking contest. https://insomnihack.ch/ [Online; accessed 2017-05-12].

[37] SANS Institute. NetWars: Core Continuous. https://www.sans.org/netwars/continuous [Online; accessed 2017-05-12].

[38] SANS Institute. Netwars: DFIR Tournament. https://www.sans.org/netwars/dfir-tournament [Online; accessed 2017-05-12].

[39] ISO. Societal security – Guidelines for exercises. ISO 22398:2013, International Organization for Standardization, Geneva, Switzerland, 2013.

[40] Jincheul Jang, Jason J. Y. Park, and Mun Y. Yi. *Gamification of Online Learning*, pages 646–649. Springer International Publishing, 2015.

[41] William E. Johnson, Allison Luzader, Irfan Ahmed, Vassil Roussev, Golden G. Richard III, and Cynthia B. Lee. Development of Peer Instruction Questions for Cybersecurity Education. In *2016 USENIX Workshop on Advances in Security Education (ASE 16)*, Austin, TX, 2016. USENIX Association.

[42] Kristian Kiili. Digital game-based learning: Towards an experiential gaming model. *The Internet and higher education*, 8(1):13–24, 2005.

[43] Richard Kissel, Richard Kissel, Rebecca Blank, and Acting Secretary. Glossary of key information security terms. In *NIST Interagency Reports NIST IR 7298 Revision 1, National Institute of Standards and Technology*, 2011.

[44] Piet Kommers. *Cognitive support for learning: imagining the unknown*. IOS Press, 2004.

[45] Fedwa Laamarti, Mohamad Eid, and Abdulmotaleb El Saddik. An Overview of Serious Games. *International Journal of Computer Games Technology*, 2014.

[46] Michael J. Lee, Andrew J. Ko, and Irwin Kwan. In-game assessments increase novice programmers' engagement and level completion speed. In *Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research*, ICER '13, pages 153–160, New York, NY, USA, 2013. ACM.

[47] Ellen Lesage, Martin Valcke, and Elien Sabbe. Scoring methods for multiple choice assessment in higher education – Is it still a matter of number right scoring or negative marking? *Studies in Educational Evaluation*, 39(3):188–193, 2013.

[48] Joseph F. Lucke. "Rassling the hog": The influence of correlated item error on internal consistency, classical reliability, and congeneric reliability. *Applied psychological measurement*, 29(2):106–125, 2005.

[49] John Ludbrook. Should we use one-sided or two-sided p values in tests of significance? *Clinical and Experimental Pharmacology and Physiology*, 40(6):357–361, 2013.

[50] Stian Lydersen, Vivek Pradhan, Pralay Senchaudhuri, and Petter Laake. Choice of test for association in small sample unordered $r \times c$ tables. *Statistics in medicine*, 26(23):4328–4343, 2007.

[51] William F. McComas and Linda Abraham. Asking More Effecive Questions. *Rossier School of Education*, 2004.

[52] Trend Micro. Targeted Attack: The Game – Defend your data. Choose wisely. Succeed or fail. http://targetedattacks.trendmicro.com/index.html [Online; accessed 2017-05-12].

[53] Martin Mink and Rainer Greifeneder. Evaluation of the Offensive Approach in Information Security Education. In *Security and Privacy – Silver Linings in the Cloud*, volume 330 of *IFIP Advances in Information and Communication Technology*, pages 203–214. Springer, 2010.

[54] Jelena Mirkovic and Peter A. H. Peterson. Class Capture-the-Flag Exercises. In *2014 USENIX Summit on Gaming, Games, and*

Gamification in Security Education (3GSE 14), San Diego, CA, 2014. USENIX Association.

[55] Jelena Mirkovic, Aimee Tabor, Simon Woo, and Portia Pusey. Engaging Novices in Cybersecurity Competitions: A Vision and Lessons Learned at ACM Tapia 2015. In *2015 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 15)*, Washington, D.C., 2015. USENIX Association.

[56] Chris Moffitt. Collecting Data with Google Forms and Pandas. http://pbpython.com/pandas-google-forms-part1.html [Online; accessed 2017-05-12], 2015.

[57] Rafael Moreno, Rafael J. Martínez, and José Muñiz. New guidelines for developing multiple-choice items. *Methodology*, 2(2):65–72, 2006.

[58] Deanna L. Morgan. Best Practices for Setting Placement Cut Scores in Postsecondary Education. An NCPR Working Paper. *National Center for Postsecondary Research*, 2010.

[59] Gary R. Morrison, Steven M. Ross, Jerrold E. Kemp, and Howard Kalman. *Designing effective instruction*. John Wiley & Sons, 2010.

[60] Cecelia Munzenmaier and Nancy Rubin. Bloom's Taxonomy: What's Old Is New Again. *The Elearning Guild. Santa Rosa*, 2013.

[61] Ajay Nagarajan, Jan M. Allbeck, Arun Sood, and Terry L. Janssen. Exploring game design for cybersecurity training. In *IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 256–262, 2012.

[62] Lukáš Neudert. Capture the Flag contests. Master's thesis, Masaryk University, Faculty of Informatics, Brno, 2014. http://is.muni.cz/th/359981/fi_m/ [Online; accessed 2017-05-12].

[63] Geoff Norman. Likert scales, levels of measurement and the "laws" of statistics. *Advances in health sciences education*, 15(5):625–632, 2010.

[64] Trail of Bits. CTF Field Guide. https://trailofbits.github.io/ctf/ [Online; accessed 2017-05-12].

[65] NYU Tandon School of Engineering. Cyber Security Awareness Week. https://csaw.engineering.nyu.edu/ [Online; accessed 2017-05-12].

[66] NATO Cooperative Cyber Defence Centre of Excellence. Locked Shields 2017. https://ccdcoe.org/locked-shields-2017.html [Online; accessed 2017-05-12].

[67] Special Interest Group on Computer Science Education (SIGCSE). SIGCSE 2017. https://sigcse2017.sigcse.org/ [Online; accessed 2017-05-12].

[68] Gordon Pennycook, James Allan Cheyne, Nathaniel Barr, Derek J. Koehler, and Jonathan A. Fugelsang. On the reception and detection of pseudo-profound bullshit. *Judgment and Decision Making*, 10(6):549, 2015.

[69] Geoffrey Petty. *Teaching Today: A Practical Guide*. Nelson Thornes, 2009.

[70] Psifertex. Practice CTF List. http://captf.com/practice-ctf/ [Online; accessed 2017-05-12].

[71] Portia Pusey, Sr. David Tobey, and Ralph Soule. An Argument for Game Balance: Improving Student Engagement by Matching Difficulty Level with Learner Readiness. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*, San Diego, CA, 2014. USENIX Association.

[72] Michael C. Rodriguez. Three options are optimal for multiple-choice items: A meta-analysis of 80 years of research. *Educational Measurement: Issues and Practice*, 24(2):3–13, 2005.

[73] Eric M. Scharf and Lynne P. Baldwin. Assessing multiple choice question (MCQ) tests – a mathematical perspective. *Active Learning in Higher Education*, 8(1):31–47, 2007.

[74] Offensive Security. Kali linux tools listing. http://tools.kali.org/tools-listing [Online; accessed 2017-05-12], 2017.

[75] Stanley Smith Stevens. On the theory of scales of measurement. *Science*, 1946.

[76] David H. Tobey, Portia Pusey, and Diana L. Burley. Engaging Learners in Cybersecurity Careers: Lessons from the Launch of the National Cyber League. *ACM Inroads*, 5(1):53–56, 2014.

[77] Open Security Training. Welcome. http://www.opensecuritytraining.info [Online; accessed 2017-05-12].

[78] Juraj Uhlár. Visualization of a run of a security game. Bachelor's thesis, Masaryk University, Faculty of Informatics, Brno, 2017. http://is.muni.cz/th/422160/fi_b/ [To be yet submitted].

[79] USENIX. ASE '16. https://www.usenix.org/conference/ase16/call-for-papers [Online; accessed 2017-05-12].

[80] Giovanni Vigna. The UC Santa Barbara iCTF Competition. https://ictf.cs.ucsb.edu/ [Online; accessed 2017-05-12].

[81] Giovanni Vigna. Red Team/Blue Team, Capture the Flag, and Treasure Hunt: Teaching Network Security Through Live Exercises. In *In World Conference on Information Security Education*, 2003.

[82] Noreen M. Webb, Richard J. Shavelson, and Edward H. Haertel. 4 reliability coefficients and generalizability theory. *Handbook of statistics*, 26:81–124, 2006.

[83] Sanford Weisberg. *Applied linear regression*. John Wiley & Sons, 2005.

[84] Joseph Werther, Michael Zhivich, Tim Leek, and Nickolai Zeldovich. Experiences in Cyber Security Education: The MIT Lincoln Laboratory Capture-the-flag Exercise. In *Proceedings of the 4th Conference on Cyber Security Experimentation and Test*, CSET'11, pages 12–12, Berkeley, CA, USA, 2011. USENIX Association.

[85] Pieter Wouters, Christof Van Nimwegen, Herre Van Oostendorp, and Erik D. Van Der Spek. A meta-analysis of the cognitive and motivational effects of serious games. *Journal of Educational Psychology*, 105(2):249, 2013.

[86] Pavel Čeleda, Jakub Čegan, Jan Vykopal, and Daniel Tovarňák. KYPO – A Platform for Cyber Defence Exercises. In *STO-MP-MSG-133: M&S Support to Operational Tasks Including War Gaming, Logistics, Cyber Defence*, page 12, Munich (Germany), 2015. NATO Science and Technology Organization.

A Linear regression diagnostic plots

This appendix reports four diagnostic plots for the two best regression models, which predicted the game score (T) based on the pretest using both scoring methods (P_a and P_b). For more information, see Section 6.1.

Figure A.1 reports the residuals (the vertical distance from a point to the regression line) versus the fitted values. The relatively flat red curve close to the horizontal gray line in the center is a mark of homoscedasticity [83], which is one of the assumptions for linear models.

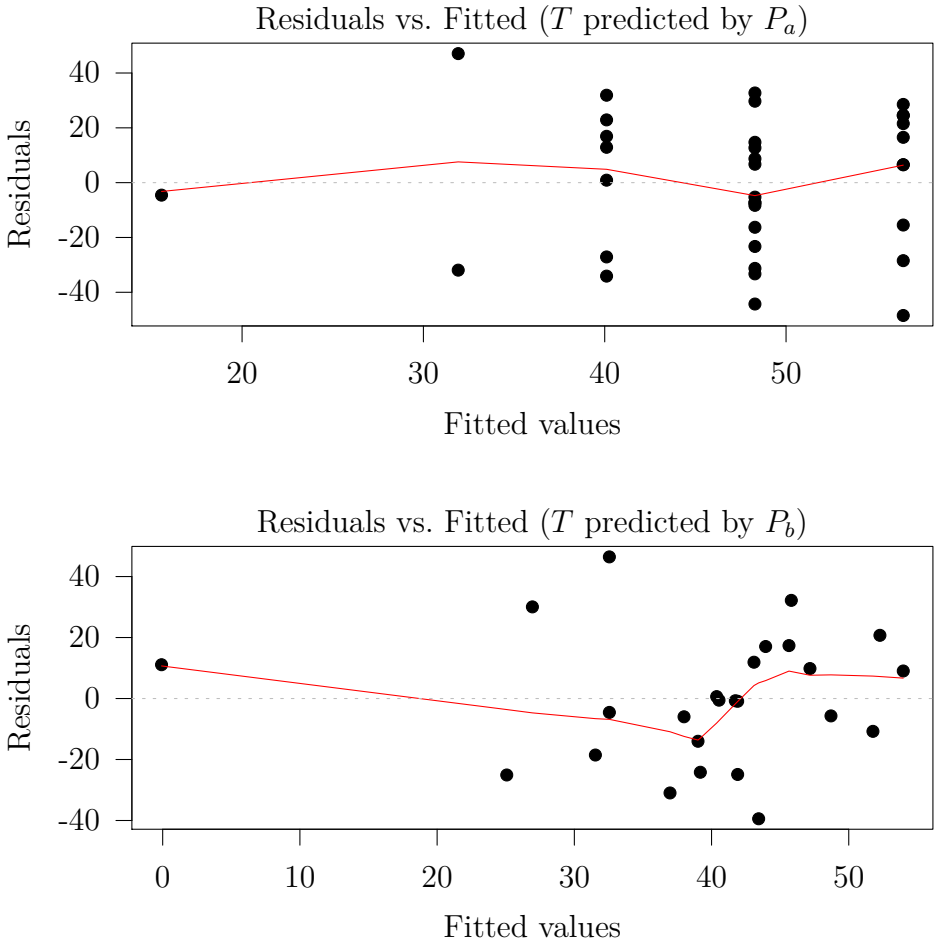

Figure A.1: Residuals versus the fitted values for the models predicting game score (T) based on the pretest (both scoring methods, P_a and P_b)

Figure A.2 reports the square roots of the standardized residuals versus the fitted values. Again, the graphs help to verify the assumption of homoscedasticity. Therefore, the red curve is expected to be relatively flat. This holds, except for the far left end, where it is sloped towards one data point.

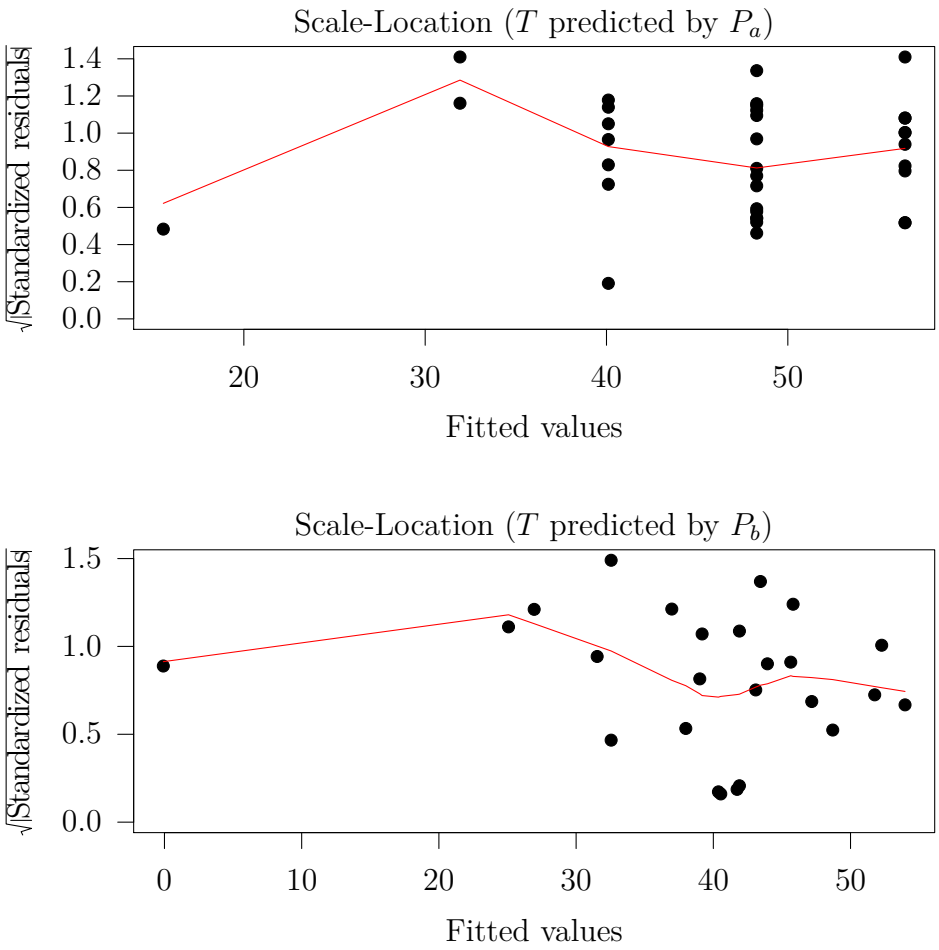

Figure A.2: Square roots of standardized residuals versus the fitted values for the models predicting game score (T) based on the pretest (both scoring methods, P_a and P_b)

Another assumption of linear regression models is that the residuals are normally distributed [83]. Figure A.3 reports the observed quantiles of the residuals. In an ideal case, the points would lie exactly on the diagonal gray line in the center. Note that both plots exhibit almost no skewness and show a good fit.

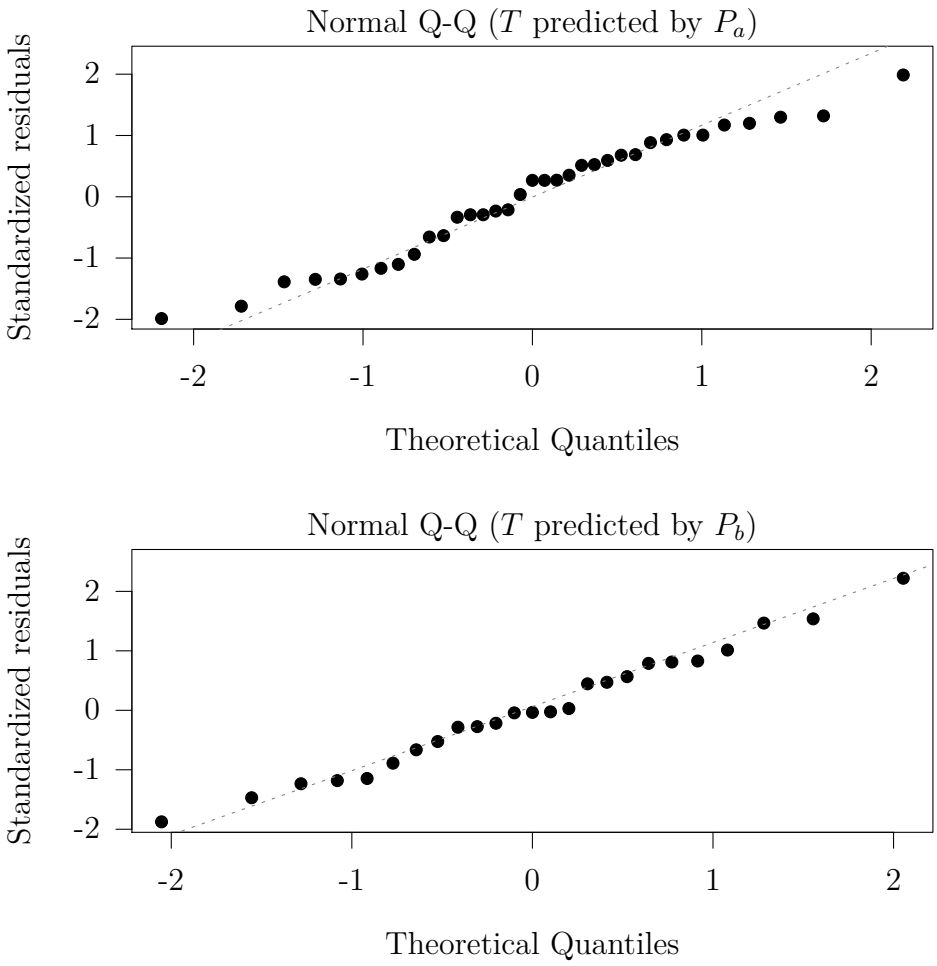

Figure A.3: Q-Q plots for the models predicting game score (T) based on the pretest (both scoring methods, P_a and P_b)

Finally, Figure A.4 shows the standardized residuals against leverage of the data points (the red curve), and also borders of Cook's distance (the dashed red curves). Since the residual-leverage plot stays close to the horizontal gray line in the center and no data point has Cook's distance greater than 0.5, no data point is significantly distorting the model. Still, one point on the second graph shows a borderline case.

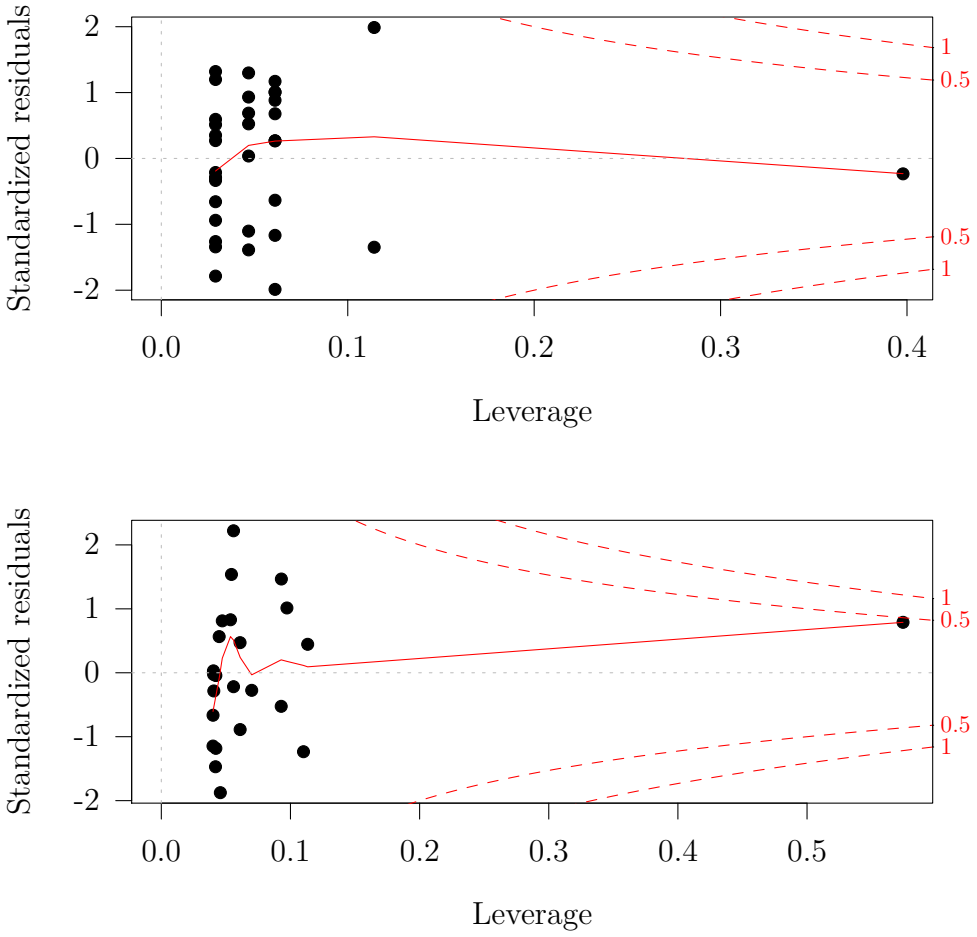

Figure A.4: Standardized residuals versus leverage for the models predicting game score (T) based on the pretest (both scoring methods, P_a and P_b)

B Core cybersecurity tools

With the help of the KYPO team, I defined the following set of "core cybersecurity tools" for penetration testing. The starting point was a list of Kali Linux tools [74]. The tools are grouped by category, listed alphabetically within a category, and supplemented with usage areas description.

The tools are reported in Table B.1. The following categories are defined (with an abbreviation in the parentheses to be used later):

- Exploitation tools (ET)
- Information gathering (IG)
- Network utilities (NU)
- Password attacks (PA)
- Sniffing and Spoofing (SS)
- Vulnerability analysis (VA)
- Web applications (WA)

Tool	Category	Usage
Arachni	WA	Security testing
Burp Suite	WA, PA, SS	Security testing
sqlmap	WA, VA, ET	Taking over database servers
Web browser	WA	Examining source code
WPScan	WA	WordPress vulnerabilities
zaproxy	WA, PA, SS	Security testing
Hydra	PA	Password cracking
John the Ripper	PA	Password cracking
Medusa	PA	Password cracking
Armitage	ET	Metasploit collaboration tool
Metasploit	ET	Exploiting remote targets
Ettercap	SS	Man-in-the-middle attacks
sslstrip	SS, IG	HTTP hijacking
Nmap	IG, VA	Network discovery
openvas	VA	Vulnerability scanning
tcpdump	IG	Packet analysis
Wireshark	IG	Packet analysis
dig	NU	DNS server querying
host	NU	DNS server querying
nc	NU	Reading/writing network data
SSH/telnet	NU	Terminal emulation
whois	NU	WHOIS service

Table B.1: Core security tools for penetration testing

C Complete wording of the post-game feedback questionnaire

1. How easy or difficult was the game for you?
   (a) Trivial: Super easy
   (b) Easy: A minor challenge, but nothing surprising
   (c) Medium: Balanced, reasonable challenge; not too easy nor too hard
   (d) Hard: I was lost and/or needed help often
   (e) Impossible: I was frustrated and/or needed constant help
2. Do you see any educational value in the game? Check all that applies to you.
   - I felt I was missing some knowledge/skills needed for playing the game.
   - My existing knowledge/skills before playing matched what the game required.
   - I felt I was overqualified, my knowledge is already far greater compared to what the game required.
   - I learned about new attacks and exploits.
   - I learned about new defensive measures.
   - I learned about new security tools.
   - I like learning cybersecurity practically (e.g., by playing games like this).
   - I consider the game a good practice (for using the tools, working under time pressure, etc.).
   - I didn't learn anything, and the game wasn't useful as practice.
   - Other (please fill in).
3. What are the most important learning experiences you take from playing the game?
4. Any other comments or remarks?

D Content of the thesis archive

The thesis archive available at https://is.muni.cz/th/395868/fi_m/ includes experiment data and source code organized in the following folder structure:

- `data`
  - `data-binary.csv`: Data of examined variables collected for all 35 participants
  - `data-confidence.csv`: Data including confidence testing and difficulty estimates collected only for 25 participants
  - `user-events-log.csv`: Log file with a total of 488 game events of all 35 players
- `src`
  - `events-analyze.py`: A Python 3 script for processing the game events to generate the total score and the number of levels finished
  - `questionnaire.py`: A Python 3 script for downloading[1] and pre-processing the results of the prerequisite test with confidence scoring for the KYPO Information theft game
  - `statistical-tests.R`: An R script for statistical analysis of the collected data
- `readme.txt`: A file explaining the folder structure above

An MIT license is used for the source code.

1. Downloading the original data requires a private authentication key to access the Google Drive storage. The key is not published for security reasons. Therefore, the script is not immediately executable.