# On the design of security games: from frustrating to engaging learning

Jan Vykopal
*Masaryk University*
*Brno, Czech Republic*

Miloš Barták
*Masaryk University*
*Brno, Czech Republic*

## Abstract

Hands-on cyber security training is generally accepted as an enjoyable and effective way of developing and practising skills that complement the knowledge gained by traditional education. At the same time, experience from organizing and participating in these events show that there is still room for making a larger impact on the learners, and providing more engaging and beneficial learning. In particular, the area of the game and exercise design is not sufficiently well-developed. There is no comprehensive methodology or best practices that can be used to prepare, test, and carry out events.

We present the concept of a security game and lessons learned from a prototype game played by 260 participants. Based on the lessons, we describe the enhancements to the game design and a user study evaluating new game features. The results of the study show the importance of logging events which describe the course of the game. It also suggests what type of information can be predicted from the game logs and what can be found by other methods such as surveys.

## 1 Introduction

As a response to the current lack of cyber security professionals [5, 1], a wide spectrum of sponsors (states, educational organizations and private companies) has launched or supported numerous educational campaigns and programs. Besides the conventional methods of teaching cyber security such as classroom lectures, lab sessions or home assignments, we have witnessed an outbreak of various hands-on competitions, challenges, and exercises. It is believed that they enable participants to effectively gain or practice diverse cyber security skills in an entertaining way. The most popular are Capture the Flag (CtF) games [6] and Cyber Defence eXercises (CDX) [10]. While CtF games can be focused on attacking, defending or both, CDXs train solely the defence. CtFs which put participants in the role of the attacker support the development of adversarial thinking, which is necessary for anticipating future offensive actions [9].

However, preparing and carrying out hands-on security games and exercises require substantial time, effort and financial investments [3]. The major workload is carried out by organizers and educators, particularly in the preparation phase. But the involvement of the target group should not be overlooked. Learners have to reserve time for intensive sessions spanning from hours to a few days. If they are employed, their employers have to cover costs that may be incurred by their absence. We believe that the all investments are justified if desired learning goals of the event are reached.

Nevertheless, there are some concerns about the actual impact of contemporary hands-on training [11, 12, 4]. First, some learners may participate in these activities prematurely, i. e. before they gain the prerequisite knowledge or skills for a particular training session. This leads to learning goals being missed and the waste of resources dedicated to the preparation of the session. Moreover, it could discourage novice learners from participating in any other hands-on activities. Second, other learners may meet the prerequisites but the absence of an adaptability to the learner's performance during the session lessens the efficiency of the training. For instance, there may be only one path of successfully completing that may be too easy or too difficult for some participants. One way to help the learners is to structure the whole training into the smaller phases, tasks or levels. Another is to offer hints and clues that learners can access, often in exchange for penalty points.

In this paper, we introduce the concept of a security game for teaching and practicing penetration testing. The game is divided into several levels, each level offers a few hints as well as a recommended solution as a last resort. We illustrate the concept on the example of a prototype game. Based on lessons learned from running this game in the past 2 years, we have enhanced the general game design and implemented an improved help system and system for logging of learners' actions. To evaluate the enhancements, we created a new game and conducted a user study to answer two research questions:

- How helpful are the hints and solutions for the learners?
- What can be predicted from the participants' actions?

While the first question is focused on a specific component of the game design, the second one is focused on exploring the informative value of various game-related events that may have a broader impact on the game design. In particular, we will investigate whether the events are in line with the learners' self-assessment.

The paper is divided into six sections. The following section introduces the KYPO Cyber Exercise & Research Platform, a vehicle for educational security games. In Section 4, we describe the conceptual design of the game, the topic and structure of the prototype game, its implementation within the KYPO platform and the lessons learned. Section 5 explains game's enhancements, presents the experiment with real learners and discusses answers to the research questions. Section 6 summarizes the paper and outlines future work.

## 2 The KYPO platform

KYPO [8] is an academic cyber range developed and hosted in the Czech Republic. It is based on computing, storage and networking resources provided by the CERIT Scientific Cloud, Czech national Infrastructure-as-a-Service cloud providing more than 4,800 CPU cores and around 4 PB of storage.

The KYPO platform supports the dynamic instantiation of arbitrary networks of virtual hosts running various operating systems and applications (sandboxes), ranging from single node networks to multiple connected networks. [7] These sandboxes are, by default, completely isolated from each other as well as the outside world. The plaform provides built-in monitoring of the network traffic (packet capture and flow acquisition from selected nodes) and individual hosts (node metrics and system logs). Authorized users interacts with KYPO remotely using a web portal and its modules (portlets) based on Liferay technology (see Figure 1). There are two fundamental modules: the *network topology portlet*, which visualizes the topology of the sandbox and the *VNC portlet*, which enables sharing of graphical desktop of virtual hosts using VNC connection.

One of KYPO use cases [2] covers a diverse type of educational hands-on activities requiring additional features which supports teachers and instructors. Security challenges, competitions, jeopardy-style capture the flag games and other similar activities are usually designed to be held without much direct input of the teacher. Instead, the assignments for the learners (including additional instructions) and evaluation of the submitted solutions are implanted into the platform where the game is deployed. The learners typically choose individual tasks or follow the predefined path of the game. Once they find a solution, they submit requested data to the game platform that immediately provides a response whether the solution is correct or not. If it is, they can proceed further.

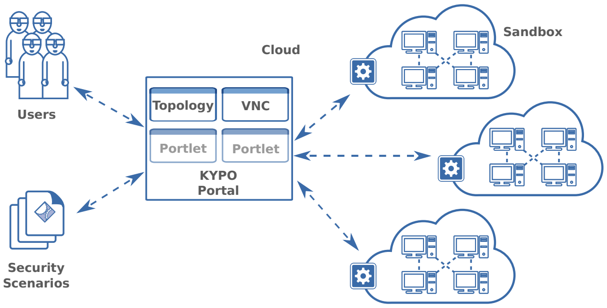


Figure 1: KYPO cyber range architecture

For educators using cyber defence exercises (red vs. blue teams) in their teaching, there is a need for a scoring system for evaluating the teams' (defenders') performance in real-time and a logging infrastructure for detailed post-mortem analysis and overall evaluation of the exercise. However, during the actual exercise, the learners do not interact much with this infrastructure but the teachers do. They can control the flow of the exercise based on automatically mined status information about the defended infrastructure. They can also manually trigger tasks for learners and evaluate learners' actions and submitted reports. The learners may only watch their score.

## 3 KYPO game

To support the KYPO educational use case, we have developed a new KYPO module for creating and running various games and a prototype game focused on penetration testing. The module enables teachers to structure the learners' interaction during the session. It also shortens and simplifies the preparation of the learning session. Learners are provided with a new interface (portlet) that guides them through the entire session.

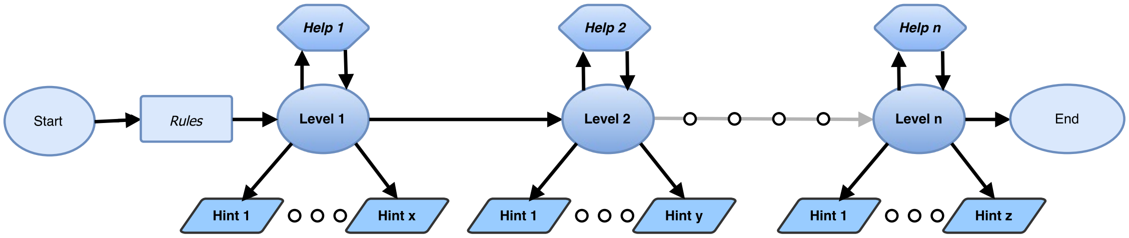


Figure 2: General structure of a KYPO game

### 3.1 Game design

The game is structured into several consecutive *levels* that lead to one *objective*, e. g. data theft from a remote server. Each level is accomplished by finding a correct *flag*, a short string, typically a checksum of some data discovered by the learner. The accomplishment of each level is awarded by a specified number of points that contribute to the learners' total score.

The scheme of the game is shown in Figure 2. At the very beginning, before the start of the game, the learners have access to limited network resources and brief information about the objective and their current resources. Once they start the game and read the rules, they enter the first level. The assignment of the level is presented, e. g., network service discovery, along with the entry field for the flag. If the learners struggle with the level, there are optional hints whose use is penalized with negative points. It is also possible to skip the level and quit the game at any time. The game ends when the last flag is entered or the predefined final check of the system's state is successful.

Each level has a time limit for finding the solution. A countdown in seconds is presented to learners to simulate the real-life constraint of the scarcity of time. For example, they might opt for hints if there is not much time left. The time limit may also give the learners a clue about the difficulty of the level in comparison with other levels. After the time expires, learners can still finish the level.

The game can be played both by the single learner and a team of several players. The former may be suitable for more advanced learners and the latter for beginners or if it is desirable to develop team spirit. The game can be held in multiple instances in parallel to promote friendly competition among players or teams. Last but not least, if the players achieve a high score they enter the hall of fame, listing the top players.

Since the main focus is put on building and developing learners' skills, the teacher may allow or even advise learners to use web search engines, online knowledge bases or their own notes. This setting corresponds to real-life situations where the use of external knowledge is common.

### 3.2 Prototype game

To thoroughly test the game design and the new module, we prepared a prototype game for teaching penetration testing. It guides the learners to the ultimate objective of conducting a DoS amplification attack. The game is split to four consecutive levels focused on the following topics:

1. exploring network services,
2. searching for non-public information,
3. exploiting a server vulnerability,
4. preparing a DDoS amplification attack.

The learners start with access to a single host and no detailed knowledge about the network environment. They can only see the network topology as depicted in Figure 3.

Level 1 is designed to practice network port scanning. The assignment is very straightforward. It mentions a recommended tool that a learner should use to perform the action. The goal of this level is to run the tool with the correct parameters. The requested flag is a list of open network ports. There are two optional hints if the learner struggles to use the recommended tool in the desired way. The first suggests the target of the scanning and some useful features of the tool. The second hint shows the exact command to be executed.

Level 2 is devoted to a SQL injection exercise using Linux command line tools. Again, the recommended tools are stated in the assignment. The flag for this level is a CVE[1] identifier, which can be found in the database content listed using the SQL injection. There are two hints: the specification of a target server script vulnerable to the injection and the main body of a command using the recommended tool.

In Level 3, the learners practice the exploitation of a vulnerability by Metasploit. Their task is getting access

[1] Common Vulnerabilities and Exposures

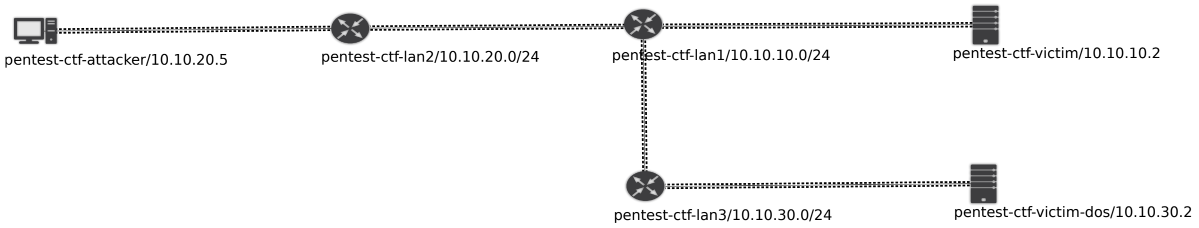

pentest-ctf-attacker/10.10.20.5
pentest-ctf-lan2/10.10.20.0/24
pentest-ctf-lan1/10.10.10.0/24
pentest-ctf-victim/10.10.10.2
pentest-ctf-lan3/10.10.30.0/24
pentest-ctf-victim-dos/10.10.30.2

Figure 3: Network topology of the prototype game as displayed in the KYPO web portal

to the server using CVE previously discovered. As evidence of this compromise, they have to submit the flag located in a local file residing on the server. This level offers four hints. The first says how to start the console of the recommended tool. The second refers to a search engine to find an online tutorial for exploiting this particular vulnerability. The third shows the Metasploit module that should be loaded. Finally, the last hint specifies the name of the searched file containing the flag.

The last level trains the skills of system administration to reach the game's objective: change the current configuration of a NTP service to act as an amplifier of NTP requests and, thus, allow a DDoS amplification attack on another host. In contrast to the previous levels, there is no flag to be submitted. Instead, the learner is instructed to click on the button which checks whether the server is configured for the attack or not. If the check is passed, the game is successfully finished. Two hints point the learner to the location of the service configuration file and the need to restart the service after changing the configuration file.

## 3.3 Game implementation in KYPO

From an implementation perspective, the game is represented by a new module of the KYPO web portal providing two different views for different user roles: the teacher and the learner.

The teacher is authorized to manage games using a new game administration portlet. The teacher can create a new game and modify or delete existing ones. The portlet enables the teacher to edit the game's introduction, rules and add, modify and remove individual levels. Within each level, the teacher can edit level assignments, the time limit, flag, points to be awarded, hints and corresponding point penalties.

Apart from the level assignments and information for the control of the game (flags, hints, points, time limits) covered by the new portlet, the teacher has to provide a network topology and set up all game hosts with an operating system, application and services used in the game.

During the game session, the teacher can access all the game's sandboxes and connect to the game's hosts using the VNC portlet. This feature is particularly useful if the session is held remotely and learners are not present at the same place as the teacher.

The learner interacts with a different portlet, which represents a game interface with all the game-related information and controls described in Section 3.1. The access to the learner's hosts is provided by the network topology portlet and the VNC portlet.

## 3.4 Lessons learned

The prototype game was developed in 2014 and held 18 times with approximately 260 participants in total. It was played at various venues and settings: locally and remotely; at university, national and international events. The players represented very diverse mix in age, work experience, knowledge, skills and backgrounds, which has brought us valuable feedback.

The game was received very positively by the vast majority of learners. They appreciated the hands-on nature of the activity that enlightens the gap between their perceived knowledge and their skills. In this section, we describe the following lessons we learned from all events where the game was played:

- the difficulty of levels is not balanced,
- learners are hesitant whether the hints will help them,
- game-related information provided outside the platform is inconvenient,
- the teacher has no information about the learners' performance and progress in the ongoing event.

Some levels are much more challenging for the learners than other levels. We experienced that many learners needed much more time and teacher's help than we expected for the last two levels. Although the learners discovered the concept of the solution, they had problems carrying out concrete steps to achieve the solution. For example, they had to use only software tools with a command line interface, which was not common for many learners. In other examples, they did not restart a network service after its reconfiguration so the service ran without any change. Even though, it is generally accepted that the difficulty should increase over the intervention, we feel that these nuances may distract learners from the main focus of the game.

The learner has no prior information about the nature and number of hints. At each level, the hints are offered linearly with an increasing number of penalty points. This means the first hint, providing only basic help, is available at the beginning and the number of penalty points is displayed. Once the learner opts for it, the next hint, providing potentially more help, is offered (if available). We noted that some learners decided to spend points for the hint but they got information which they already knew. Other learners managed to successfully accomplish several steps leading to the flag but struggled with the last steps. They did not use any hints since they supposed they would lose too many points for the hints they do not need at that stage of the level.

Providing supplementing information outside the game platform is inconvenient for both teachers and learners. During some sessions, the teacher showed the learners the recommended solution after each level, explained attacker's motivation and referred to related study materials. All this information was projected in the room where the sessions were held. The learners may be distracted by interrupting their game and need to switch their visual focus from their working place to the screen in the room back and forth. This is also impractical when the learners copy some text (e. g., commands) from the screen to their console. Next, the learners need information at various time during the session based on their readiness. In the event of remote sessions, information was provided via mail. If it was before the session, some learners were tempted to look for the solution too quickly. If it was afterwards, the learners have to ask the teacher to provide this information so this brings additional requirements for the teacher.

The only way the teacher know how the learners are progressing in the game is to ask them or watch their consoles on-site or via the KYPO portal. The teacher's game portlet does not provide any overview of the progress of individual learners. The sessions were often attended by groups of learners of various levels of readiness, and including beginner learners. Some of them did not have sufficient prerequisite knowledge and skills for the game. Even though they used the game's hints, they still needed the assistance of the teacher to finish the game. Currently, the teacher had very limited methods to recognize these learners early. As a result, some sessions run out of the time and a few learners were frustrated.

## 4 Enhancement of the game module

Based on the lessons learned, we redesigned the game and added new features to the KYPO game module to better support both students and learners.

### 4.1 Hints and level solutions

We introduce improved hints and embedded level solutions into the game module. In each level, all hints are presented with an indication of their nature. For example, in a level focused on network reconnaissance, there are two hints, one with a label "what tool to use" and one with "how to use the tool". The learners can then decide what type of hint would be more helpful for them and thus spend penalty points more effectively.

If the hints do still not help, and learners cannot proceed further, they can view the solution of the level in the game's portlet. Then, they can take the necessary steps to capture the flag of the level such as a set of commands that have to be typed by the learner to run a successful exploit of a vulnerability in a remote host. The availability of the solution on an on-demand basis supports the individual learning pace of each learner. This is useful in remote sessions where the learners are not present in the same room as a teacher and cannot follow the slides with the solution.

### 4.2 Logging the learner's actions

To monitor the course of the learning activity, we collect timestamps of several events related to the game progress:

- start and end of the game,
- start and end of each level,
- submission of incorrect flags and their content,
- hints used,
- skipping a level,
- displaying a level's solution,
- game ID.

These events capture only the interaction of the learner and the game module of the KYPO platform. These events can be observed during any game of this type

| Level | Topic | # Hints | Time limit |
|---|---|---|---|
| 1 | Reconnaissance | 2 | 10 |
| 2 | Web scan | 2 | 10 |
| 3 | Web exploit | 3 | 25 |
| 4 | Credential steal | 2 | 20 |
| 5 | Privilege escalation | 2 | 20 |
| 6 | Information theft | 2 | 15 |

Table 1: Levels of the new prototype game with a number of hints and time limit in minutes

since the events are not bound to the specific nature of the hosts, application or services of the game network. However, they provide limited information about actions and steps taken by the learner within the sandbox.

# 5 Evaluation of the enhanced module

To test the new enhancements to the game module and evaluate their contribution, we created a new prototype game and prepared an experiment involving real learners.

## 5.1 Prototype game within a new module

The new prototype game follows the structure introduced in Section 3.1. In general, the new game is similar to as described in Section 3.2 but the game network (sandbox) contains more hosts that should be attacked. The final game objective is to conduct information theft. The learner starts with access to a single host and has to pass six levels, as depicted in Table 1.

## 5.2 Methodology

In April 2016, we held two game sessions including a questionnaire investigation with 21 participants in total. The learners were undergraduate and Ph.D. students of the Faculty of Informatics, Masaryk University, Brno and the IT staff of various European universities, who were attending the Masaryk University Staff Training Week focused on cyber security.

Before the learners started the game, they were asked to fill in a pre-game questionnaire about such things as their age, formal education, working experience, experience with Linux, network security in general, familiarity with particular vulnerabilities and tools used for penetration testing. Last but not least, we also asked whether they had ever played a security game before.

Once the learners started the game, we instructed them not to talk with anyone in the room and ask the teacher only questions related to the KYPO platform and its game module, not the game itself. If they needed some help, they had to use hints or view the solution as the last resort. After finishing or skipping each level, we asked them to fill in an in-game questionnaire assessing the levels' difficulty, whether the hints were helpful, adequacy of the time limit for the level and what they had learned.

| Category | Learners |
|---|---|
| User | 13 |
| Administrator | 7 |
| Developer | 1 |

Table 2: Learners' experiences with Linux OS

| Level of experience | Learners |
|---|---|
| 5 | 1 |
| 4 | 3 |
| 3 | 9 |
| 2 | 3 |
| 1 | 5 |

Table 3: Experiences with network security. 1 – none, 5 – practitioner

| Years of exp. | Learners |
|---|---|
| 10 and more | 2 |
| 6–10 | 2 |
| 1–5 | 7 |
| 0 | 10 |

Table 4: Work experiences as a system administrator or a member of security team

By the end of the game, the last questionnaire asked learners about the overall assessment of difficulty of the game, its duration, learners' feelings during the session and whether the game motivated the learners to play another one.

Apart from the questionnaires, we collected all logs from the enhanced KYPO game module as described in Section 4.2. We then joined the logs with the respective questionnaires using game IDs and analyzed the consolidated data describing the learners' background, performance during the game and assessment of individual learners.

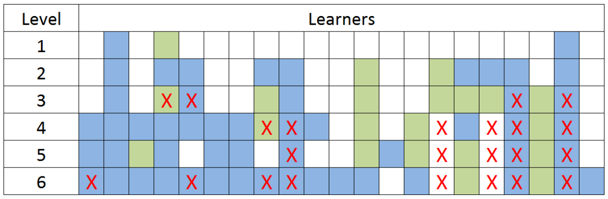
## 5.3 Results and findings

### 5.3.1 Learners background

Tables 2, 3 and 4 summarize the learners' experiences that are the most relevant to the prototype game. The game hosts run a Linux operating system so a familiarity with it is the basic prerequisite for the game.

### 5.3.2 Benefit of hints and solutions

**Hints** Figure 4 depicts the use of hints in each level by each learner. The six levels form rows and 21 individual learners form columns. A white background of a cell means that the learner used no hint in the level. A blue background indicates that all hints in the level were taken linearly, i. e. if the level offered two levels, the learner took Hint 1 first and then Hint 2, or Hint 1 only. A green background of a cell tells us that the learner took another hint than Hint 1 first (i. e. started with Hint 2 or 3). A



| Level | Learners | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 🟦 | | 🟩 | | | | | | | | | | | | | | | | 🟦 | |
| 2 | | 🟦 | | 🟦 | 🟦 | | | 🟦 | 🟦 | | | 🟩 | | | 🟩 | 🟦 | 🟦 | 🟦 | | 🟦 | |
| 3 | | 🟦 | | 🟩 X | 🟦 X | | | 🟩 | 🟦 | | | 🟩 | | | 🟩 | 🟩 | 🟩 | 🟦 X | 🟩 | 🟦 X | |
| 4 | 🟦 | 🟦 | 🟦 | 🟦 | 🟦 | 🟦 | 🟦 | 🟩 X | 🟦 X | 🟦 | | 🟩 | | 🟩 | X | 🟦 | X | 🟦 X | 🟩 | 🟦 X | |
| 5 | 🟦 | 🟦 | 🟩 | 🟦 | | 🟦 | 🟦 | | 🟦 X | | | 🟩 | 🟦 | 🟩 | X | 🟩 | X | 🟦 X | 🟩 | 🟦 X | |
| 6 | 🟦 X | 🟦 | 🟦 | 🟦 | 🟦 X | 🟦 | 🟦 | 🟦 X | 🟦 X | 🟦 | 🟦 | 🟦 | | 🟦 | X | 🟩 | X | 🟦 X | 🟩 | 🟦 X | 🟦 |

Figure 4: The use of hints in each level by the learners.

cell without any sign indicates that the level was accomplished and 'x' that it was skipped. In total, the learners opted for hint(s) in 59 % of all played levels (74 out of 6 × 21). If they took a hint, they decided not to take Hint 1 in 28 % (21 out of 74) of all decisions. This means the learners benefited from the game feature which enables them to do so and not to take hints that might be useless.

To find out whether the hints used actually helped the learners to accomplish the level, we cross-checked the logged game action of the learners with their relevant answers from the questionnaires. In the game logs, we see that 77 % of all levels where learners opted for a hint(s) were then accomplished (57 out of 74). However, there is no evidence that the information provided by the hint(s) actually contributed to the success. The learners' answers to the relevant question after each level do not clarify the contribution of the hints. While some learners answered that the hints helped them, we can see in the game logs that they did not use a hint. Conversely, some other learners reported they did not need any help but game logs show that hints were taken. All in all, we are not able to match the hint-related game events in the logs to the respective answers in any game level.

**Solutions** We observed in the game logs that the level solutions were displayed 27 times in total, by 8 distinct learners. Table 5 shows the distribution of the solution views in all levels. If the learners displayed the solution and then submitted the correct flag, they are counted in *Level accomplished*. If they displayed the solution and then skipped the level or quit the game, they are counted in *Level skipped*.

We expected a higher rate of levels accomplished since the solutions were designed to be self-explanatory, short step-by-step guidelines that bring the learners hands-on experience.

### 5.3.3 The informative value of game logs

In total, we observed 472 game events, i. e. each learner generated about 22 events on average. Figure 5 depicts the distribution of the duration of each level using a box plot. Skipped levels are not counted. The boxes bounds the lower and the upper quartile, the black line marks the median and whiskers the minimal and maximal values observed. The blue line marks the time limit, i. e. the adequate time needed for accomplishing the level, estimated by the designer of the game shown in Table 1. While time limits for levels 1, 2, and 5 were estimated relatively accurately (the median is very close or the same as the limit), other estimates differ significantly, particularly Level 4 and 6.

| Level | Solution displayed | Level accomplished | Level skiped |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 2 | 4 | 4 | 0 |
| 3 | 6 | 3 | 3 |
| 4 | 9 | 5 | 4 |
| 5 | 3 | 2 | 1 |
| 6 | 6 | 3 | 3 |

Table 5: Level solution views and its use

| Level | Average assessment |
|---|---|
| 1 | 1.6 |
| 2 | 2.7 |
| 3 | 3.6 |
| 4 | 4.2 |
| 5 | 3.8 |
| 6 | 4.5 |

Table 6: Difficulty of each level. 1 – easy, 5 – very hard

This data corresponds to the learners' assessment of the difficulty of the level provided in the questionnaires after each level. Averages of the difficulty are depicted in Table 6. The difficulty perceived by the learners grows with the increasing level except for Level 4. We intended to increase the difficulty of the game gradually up to Level 3. The last three levels (4, 5, and 6) were designed to have relatively similar difficulty. They contain more steps which were meant to be easy. As shown in Figure 5, assessment of Level 5 is in line with this intention. However, assessment of levels 4 and 6 suggest to redesign these levels, either by reducing the count of steps within both levels or by prolonging the level time limits.

## 6 Conclusions

We have presented a design of generic security games that can be deployed at the KYPO Cyber Exercise & Research Platform. The games are intended for training penetration testing in a network environment. The learners pass through several levels to accomplish the main learning objective of the game. Each level contains several optional hints for learners who cannot complete the level.

During 18 sessions of the prototype game (with the topic of a DoS amplification attack), we identified four

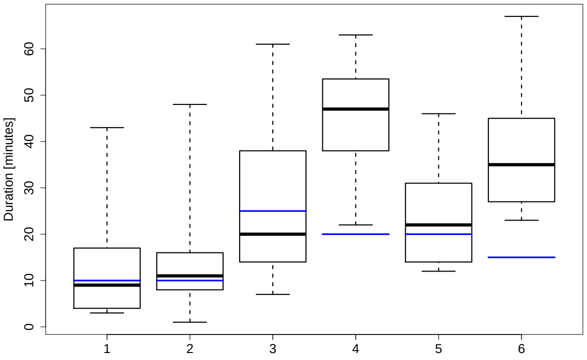


Figure 5: A box plot of the duration of levels with the blue line indicating the set time limit

shortcomings related to the general game design: i) unbalanced difficulty of the levels, ii) rigid system of hints, iii) recommended solution available out of the game platform, and iv) no game status information available for the teachers.

These lessons led us to enhance the game design and to pose two research questions focused on the evaluation of the actual benefits of the hint system and the informative value of game events.

We conducted an experiment involving a diverse mix of users to answer these questions. An analysis of the collected events and the supplemental user survey showed that learners did use the redesigned hint system and recommended solutions. However, the learners' answers in the survey neither confirm nor disconfirm the benefit of the hints and solutions used. For instance, while the learner reported they did not need any help in a given level, the respective game log showed that the hint was taken. Conversely, other games events matched the learners' assessment, namely the level difficulty and duration. Next, we found that logging the game's events is useful for the teacher during the ongoing session. With status information about each learner, the teachers can adapt to the pace of the individual participants or the whole group without any disturbance.

Considering future work, these findings open the question whether user surveys represent reliable tools for designing and evaluating hands-on training. Next, logging of game events provides valuable information that may trigger a redesign of some levels or the whole game. Nonetheless, this approach can be used only *after* testing a game involving real participants and that might be costly. We believe that the design of a new game could be facilitated by a methodology covering various aspects of the game design before the first game testing with users. For example, the game might be structured to the levels with respect to the particular stages of a typical lifecycle of an attack which is the main objective of the game. We suppose that pushing all these ideas forward will bring more effective preparation for the sessions and more benefits for the learners.

## Acknowledgments

This research was supported by the Security Research Programme of the Czech Republic 2015–2020 (BV III/1 – VS) granted by the Ministry of the Interior of the Czech Republic under No. VI20162019014 – Simulation, detection, and mitigation of cyber threats endangering critical infrastructure.

## References

[1] Burning Glass Tech. Job market intelligence: Cybersecurity jobs, report. Online, 2015. `http://burning-glass.com/wp-content/uploads/Cybersecurity_Jobs_Report_2015.pdf`.

[2] Čeleda, P., Čegan, J., Vykopal, J., and Tovarňák, D. KYPO–A Platform for Cyber Defence Exercises. In *STO-MP-MSG-133: M&S Support to Operational Tasks Including War Gaming, Logistics, Cyber Defence* (2015), NATO Science and Technology Organization.

[3] Childers, N., Boe, B., Cavallaro, L., Cavedon, L., Cova, M., Egele, M., and Vigna, G. Organizing large scale hacking competitions. In *Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 2010, pp. 132–152.

[4] Chung, K., and Cohen, J. Learning Obstacles in the Capture The Flag Model. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)* (San Diego, CA, Aug. 2014), USENIX Association.

[5] Cisco Systems. Cisco 2014 annual security report. Online, 2014. `http://www.cisco.com/web/offer/gist_ty2_asset/Cisco_2014_ASR.pdf`.

[6] Davis, A., Leek, T., Zhivich, M., Gwinnup, K., and Leonard, W. The Fun and Future of CTF. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)* (San Diego, CA, Aug. 2014), USENIX Association.

[7] Kouřil, D., Rebok, T., Jirsík, T., Čegan, J., Drašar, M., Vizváry, M., and Vykopal, J. Cloud-based testbed for simulation of cyber attacks. In *2014 IEEE Network Operations and Management Symposium (NOMS)* (May 2014).

[8] Masaryk University. KYPO Cyber Exercise & Research Platform. Web page, 2016. `http://www.kypo.cz/`.

[9] Mirkovic, J., and Peterson, P. A. H. Class Capture-the-Flag Exercises. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)* (San Diego, CA, Aug. 2014), USENIX Association.

[10] NATO Cooperative Cyber Defence Centre of Excellence. Cyber Defence Exercises. Web page, 2016. `https://ccdcoe.org/event/cyber-defence-exercises.html`.

[11] Pusey, P., David Tobey, S., and Soule, R. 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, Aug. 2014), USENIX Association.

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