Masaryk University

Faculty of Informatics

# Teaching Cybersecurity Hands-on

Habilitation Thesis

**Jan Vykopal**

Brno, Fall 2022

## Acknowledgements

I am indebted to co-authors of the included publications and developers of interactive learning environments I use in my research. In particular, members of CSIRT-MU and Cybersecurity Laboratory. Working in these teams is fun and enriching. I very much appreciate the help of Pavel Čeleda and Valdemar Švábenský, who proofread this thesis. Many thanks also to my students and participants in research studies. I also gratefully acknowledge various grants and taxpayers that have supported the work I present here. Lastly, this thesis would not have been possible without my family. Thank you for your patience and unwavering support.

# Abstract

The lack of a cybersecurity workforce is a global and persisting challenge. To address this issue, higher education institutions have started offering cybersecurity programs based on the guidelines of computing societies and professional associations. These communities agreed on topics that must be taught (what) but did not provide detailed guidance on teaching methods (how). Our work fills this gap and provides innovative methods for teaching cybersecurity skills as well as technical environments for applying these methods in the teaching practice.

This thesis presents our achievements in cybersecurity education research and development. It is organized as a collection of 14 previously published papers with a commentary. At first, we present state of the art in cybersecurity hands-on education, including our contributions to its systematization. Then we introduce three original interactive training environments, including the KYPO Cyber Range Platform. Finally, we summarize our methods for instructing students interacting with the training environment. Both the training environments and instructional methods were deployed and evaluated in teaching practice not only at our institution but also at various institutions in Europe, United States and Asia.

# Shrnutí

Nedostatek kyberbezpečnostních expertů je celosvětovým a přetrvávajícím problémem. Instituce poskytující terciární vzdělávání proto začaly nabízet výukové programy, které vycházejí z doporučení profesních sdružení. Tato sdružení se sice shodla na tématech, která je třeba vyučovat („co učit"), ale už nedoporučila konkrétní výukové metody („jak učit"). Tato práce tuto mezeru zaplňuje a přináší inovativní metody výuky dovedností v oblasti kyberbezpečnosti a technická prostředí pro použití těchto metod ve výukové praxi.

Habilitační práce představuje naše výsledky v oblasti vzdělávacího výzkumu a vývoje v kybernetické bezpečnosti. Je uspořádána jako soubor 14 dříve publikovaných prací s komentářem. Nejprve popisujeme současný stav praktického vzdělávání v oblasti kyberbezpečnosti, včetně našich příspěvků k jeho systemizaci. Poté představujeme tři původní interaktivní výuková prostředí, včetně KYPO Cyber Range Platform. Nakonec popisujeme výukové metody, které jsou navrženy pro použití v interaktivních výukových prostředích. Výuková prostředí i metody byly nasazeny a ověřeny ve výukové praxi nejen na Masarykově univerzitě, ale také na různých institucích v Evropě, USA i Asii.

# Keywords

cybersecurity education, serious game, interactive learning environment, virtual lab, feedback, assessment, adaptive learning

# Contents

# PART I

# COMMENTARY

# 1 Introduction

Cybersecurity is a computing-based discipline involving technology, people, information, and processes to enable assured operations. It involves the creation, operation, analysis, and testing of secure computer systems [1].

The lack of a workforce proficient in cybersecurity is a global and persisting challenge. It was first discussed a decade ago, and still, no significant progress toward its solution has been made to this date [2, 3]. Although millions of workers are active in the cybersecurity domain, other millions are still missing to supply the demand [4].

Computing societies and professional associations such as ACM or IEEE have recently revisited their curricula and now recognize cybersecurity as a distinctive computing discipline [5, 1]. Higher education institutions are now adopting the curricula and offering study programs covering cybersecurity knowledge areas and units [6].

Cybersecurity has also become a part of national strategies and policies. The majority of countries have a national cybersecurity strategy helping them tackle cybersecurity risks endangering the country's economics and society [7, 8, 9]. The strategies often address cybersecurity education, training, and workforce development.

Some countries have been working on frameworks systematically covering the cybersecurity domain for needs of educational institutions and their students as well as job market. The frameworks such as NIST NICE [10] or CYBOK [11] define cybersecurity knowledge, skills, abilities, tasks, and work roles.

Although the curricula and frameworks define topics and competencies that should be taught (*what* and *why*), they do not cover instructional and assessment methods (*how*). Educational organizations use traditional methods and formats used in other computing disciplines (such as lectures, seminars, tutorials, lab sessions, homeworks, tests) and methods unique to cybersecurity, namely cybersecurity hands-on games and exercises. While the hands-on methods are generally considered suitable for teaching cybersecurity, there are no established guidelines or recommendations for applying particular methods to a concrete teaching context.

## 1.1 Focus of the Thesis

This thesis addresses three research questions (RQs) related to the practical teaching of cybersecurity. First, we focus on an environment where hands-on education takes place. We research and develop virtual, controlled, and monitored environments emulating cyber systems and networks. Second, we study methods for learning cybersecurity skills that use the environments. Finally, we research innovative instructional methods based on data analysis about learners' interactions.

### *RQ1: How to create and adopt realistic cybersecurity training environment?*

Huge efforts and investments have been spent worldwide on designing, developing, and deploying various hardware and software environments for cybersecurity experimentation and training. As a result, classified or private enterprise-grade cyber ranges and cybersecurity training platforms are available, but only for a limited number of users. Academia and cybersecurity enthusiasts have developed training environments which are provided as open-source software. However, they are often early-stage prototypes with limited portability to a different teaching context. We research scalable training environments that enable seamless transition between local and cloud deployment based on the instructor's need and available resources.

### *RQ2: What instructional methods are suitable for teaching cybersecurity hands-on?*

The proliferation of training environments has enabled carrying one-off or regular training sessions using various technologies in various educational contexts. Learners can participate in individual training tutorials, complex team exercises simulating real crises, competitions focused on offensive or defensive skills or their mix. Although many educational formats of cybersecurity training are being delivered, their impact on a learner is unclear. We focus on and innovate laborious and costly cybersecurity defense exercises and cybersecurity serious games, both run in environments based on virtual machines and networks.

***RQ3: How to make cybersecurity training more efficient?***

Existing cybersecurity training formats are based on a static set of tasks, which do not adapt to the proficiency of an individual learner. As a consequence, low-performing students are overwhelmed by too difficult tasks, and high performers are bored by too simple assignments. To cater to both groups, we analyze data about learners' interactions in the training environment and provide them with the most suitable task during ongoing training or targeted feedback after the training session. We collect and process logs from the virtual machines and events from a learner's interface.

## 1.2 Thesis Structure

This thesis is organized into two parts. The first part summarizes authors' contribution and effort in answering the presented research questions. Chapter 2 presents an overview of state of the art in cybersecurity hands-on education. Chapter 3 introduces interactive training environments for cybersecurity hands-on education. Chapter 4 discusses methods for the instruction of students interacting with the training environment. Chapter 5 presents conclusions and outlines future research directions. The second part of the thesis is a collection of 14 selected research papers.

# 2 State of the Art

This chapter presents an overview of state of the art in cybersecurity hands-on education. Section 2.1 introduces existing methods for teaching cybersecurity skills used in practice. Section 2.2 introduces state of the art in cybersecurity education research. Section 2.3 presents my own contributions to the systematization of the state of the art.

## 2.1 Instructional Methods

Cybersecurity skills are taught using a wide range of methods, tools, and environments. The following text briefly overviews the most common or distinct instructional methods.

### Hands-on Labs

A lab is a set of practical tasks that learners complete in a dedicated learning environment. Labs are used for training cybersecurity, networking, and operating system skills. For example, the labs can train essential skills using step-by-step instruction, such as changing password of a user account in an operating system, as well as advanced skills such as malware analysis. The learning environment simulates or emulates authentic computer systems or applications. The tasks are provided as text assignments by instructors or embedded in the learning environment, which may also assess task completion. The time required to complete labs varies from tens of minutes to a few hours.

Labs are commonly used at high schools, higher education institutions, and by professional training providers. Labs can be in-person in a computer room or remotely using learners' own computers. Labs are usually created, adopted, and maintained by instructors of a particular institution or in small, closed communities. They are rarely shared with others or released as open educational resources.

SEED Labs [12, 13] or Labtainers [14, 15] are examples of popular and free hands-on labs on various security topics. VulnHub [16] is a catalog of free vulnerable virtual machines created by enthusiasts.

SANS Institute [17] and Cybrary [18] are commercial providers of numerous courses based on hands-on labs.

## Competitions and Games

Cybersecurity competitions and games (such as Capture the Flag, CTF) engage small teams of participants who exercise their skills by solving various tasks in an online environment. The tasks (called *challenges*) feature diverse assignments such as exploiting websites, cracking passwords, or breaching unsecured networks. A successful solution to a challenge yields a text string called a flag that is submitted online to prove reaching the solution [19].

In competitions, participants try to solve as many tasks as possible in the shortest time to achieve the highest score and win the game. The challenges usually have an intentionally unclear or incomplete assignment. Besides their progress, the participants can also see the progress of other players on a common scoreboard. Competitions last from several hours to a few days. Prominent competitions have two stages: qualifications open to the public and the finals only for the selected top-scoring teams. The top teams may receive monetary prizes of thousands of US dollars or a job offer from the host organization. DEF CON CTF [20] and iCTF [21, 22] are one of the world's largest competitions running for two decades.

In contrast to the prominent competitions, educational (serious) games aim to teach the participants something new rather than assess their current proficiency. These games should include one or more elements of educational game design [23]: *identity* (playing one's character, such as attacker or defender), *immersion* (having a sense of presence through individual identity), or *increased complexity* (such as structuring game into levels). Some games provide hints, which may cost penalty points. PicoCTF [24] and Cyber Security Awareness Week CTF (CSAW CTF) [25] are competitions that aim at beginners. In recent years, they attracted tens of thousands of high-school and undergraduate students. HackTheBox [26] and TryHackMe [27] represent online platforms that provide gamified hands-on labs.

When the game is over, participants may summarize their steps and approach into a *writeup*. Some participants publish the writeups to demonstrate they solved the tasks and to share their knowledge with others.

## Defense and Offense Exercises

Cyber defense exercises (CDXs) and cyber offense exercises are training sessions for teams of learners who are developing and strengthening their defensive or offensive skills. Both types of exercises are held in a complex emulated environment and last a few days.

### Defense Exercises

CDXs have been traditionally organized by military and governmental agencies, which require to exercise not only technical skills but also decision-making processes, standard operational procedures, or communication channels in a local and international context. As a result, CDXs are centered around a background story resembling a recent real crisis or attack campaign. Although CDXs and cybersecurity games share some common characteristics (such as the live scoreboard, vague assignments, or time constraints), CDXs are less structured and more realistically mimic the operational environment of a real organization highly dependent on ICT infrastructure. CDXs last from a few days to a week.

Locked Shields [28, 29] is the largest global defense exercise organized by the NATO Cooperative Cyber Defence Centre of Excellence (CCDCOE) in Tallinn, Estonia, since 2010. It exercises technical and strategic decision-making skills in an emulated environment involving business IT, critical infrastructure, and military systems. Cyber Flag [30] and Cyber Shield [31, 32] are examples of U.S. military regular exercises. National Collegiate Cyber Defense Competition [33] has been organized by the University of Texas at San Antonio since 2005 for post-secondary level students who defend a small company.

### Offense Exercises

Offense exercises have evolved as CDXs complements. Crossed Swords [34] is an annual exercise for penetration testers, digital forensics, and situational awareness experts organized by NATO CCDCOE since

2016. This exercise trains members of the Red Team playing the adversary in the Locked Shields CDXs. Collegiate Penetration Testing Competition (CPTC) [35] focuses on mimicking the activities performed during a real-world penetration testing engagement conducted by companies and internal security departments. CPTC is a global exercise organized by the Rochester Institute of Technology since 2016.

## Unplugged Activities

These activities do not use information technologies directly. Card and board games develop awareness about common cybersecurity concepts. Tabletop exercises focus on communication and processes in a particular organization, community, sector, state, or alliance. Other activities, such as unplugged Capture the Flag games [36, 37] focus on students with little or no technical knowledge.

### Card and Board Games

[d0x3d!] [38] is a collaborative board game for up to four players, ethical hackers tasked to retrieve a set of valuable digital assets held by an adversarial network. Control-Alt-Hack [39, 40] is a card game for three to six players who perform security audits and provide consultation services. Deploy or Die [41] is a card game for two to four web developers, which includes cards for attacking and defending.

### Tabletop Exercises

Tabletop or paper-based exercises (TTX) are discussions centered around a hypothetical scenario with a reduced focus on technical or technological matters [42]. These exercises usually last a few hours or a day. TTXs facilitate understanding of the cyber incident and emergency processes, encourage communication and collaboration, and promote the development of hands-on skills for incident response teams (CSIRTs) [43]. In contrast to defense and offense exercises, TTXs are cost-effective [44]. They are developed and organized by governments or provided by commercial companies as a service. For example, U.S. Cybersecurity and Infrastructure Security Agency offers Tabletop Exercise Packages, a comprehensive set of resources designed to assist stakeholders in conducting their own exercises. Each package is customizable and includes template exercise objectives, scenarios, and discussion questions [45].

## 2.2 Education Research

Research in education is a disciplined attempt to address questions or solve problems through the collection and analysis of primary data for the purpose of description, explanation, generalization, and prediction [46]. This habilitation thesis contributes to research in cybersecurity education, which is a part of computing education research.

### Computing Education Research

Computing education research (CER) seeks to build deep understanding of the complex phenomena and processes involved in teaching and learning computing [47]. It started in the 1970s by studying the efficiency of education of future programmers and investigating whether computing is a suitable tool for thinking and problem-solving. Through five decades, the CER community has attempted to answer research questions related to developing a mental model of program execution by a computer, the form of a programming language, and representation of program execution (algorithm animations) [48]. Recent research is still dominated by introductory programming. Other significant topics are software engineering, evaluation, and assessment [49].

### Cybersecurity Education Research

To the best of our knowledge, there is no established definition of cybersecurity education research. We use this term for computing education research focused on teaching and learning the creation, operation, analysis, and testing of secure computer systems. Our work presented in this thesis is an example of cybersecurity education research.

## 2.3 My Contributions

Here we introduce our research contributions, which map the landscape of cybersecurity education research and practice. We surveyed *i*) academic publications on cybersecurity education research in general, *ii*) academic publications on educational data mining and learning analytics in cybersecurity training, and *iii*) topics covered by cybersecurity competitions.

### A Systematic Review of Cybersecurity Education Papers

Cybersecurity education research has been gaining momentum since the 2010s, but it is still fragmented. We examined 1,748 papers published at conferences organized by ACM Special Interest Group on Computer Science Education (SIGCSE) from 2010 to 2019 [50]. In total, 71 of them focus on cybersecurity education. These papers discuss courses, tools, exercises, and teaching approaches, most often in the context of a North American university. In particular, the papers cover technical topic areas (mainly secure programming, network security, and offensive security) as well as human aspects (such as privacy and social engineering). The distribution of knowledge areas is depicted in Figure 2.1.

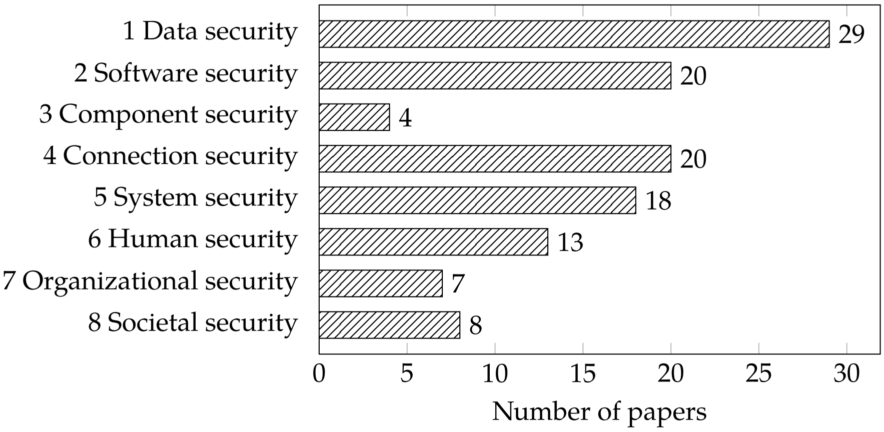

Figure 2.1: The distribution of Cybersecurity Curricular Guideline Knowledge Areas [1] discussed in 71 cybersecurity education papers.

The papers typically reported research conducted with a population of several dozens undergraduate students participating in a one-semester course. Most papers used questionnaires or tests for evaluation. Only eight papers analyzed artifacts produced by students, such as assignment submissions or logs from using a tool. In some papers, the description of the methods was unclear or incomplete. Less than a third of the papers provide supplementary materials for other educators (such as tools or teaching materials), and none of the authors published their dataset.

### Educational Data Mining and Learning Analytics in Cybersecurity Training

Our review of cybersecurity education papers showed the analysis of artifacts produced by students is not yet widely used. We focused on this area and systematically reviewed research works analyzing student data from hands-on cybersecurity training sessions. During these sessions, students interact with the training environment and connect to machines, type commands, communicate over the network, or submit answers. Out of 3,021 papers we identified by a search of relevant keywords in the Scopus database, only 35 papers dealt with analyzing student data from cybersecurity training. We reviewed them in detail and investigated the use of student data and the application of research in practice [51].

The published papers were applied in practice most often in university courses (13), then in cyber defense exercises (5) and Capture the Flag games (5). In total, 18 papers focused on teaching offensive security skills (penetration testing, exploitation, network attacks, cryptographic attacks, and reverse engineering), and 22 papers focused on defensive skills (network security was the most prominent). The most frequent knowledge areas are Connection security and System security (in 26 and 23 papers, respectively). The goal of the vast majority of papers (31 out of 35) was student assessment and evaluation. The student data were collected most often (24 papers) in a physical or virtual environment with one or more hosts with a standard operating system. The types of collected data were largely heterogeneous, see Figure 2.2. The most prominent data types were timestamps (T) of actions, such as command submissions or event triggers, or the derived data, such as duration. The number of participants from which the data were collected ranged from one to 9,873, with a median of 43. Most commonly, the data were collected during several days (14 papers from 1 to 14 days) or hours (ten papers from 1 to 13 hours). Most papers performed a post-hoc analysis of the collected data, which does not allow providing situational awareness for instructors during the training session, nor adapting ongoing training using the collected data. Finally, only eight papers were complemented with supplementary materials for other instructors, such as open-source code of a platform, a tool, virtual machines, and exercises.

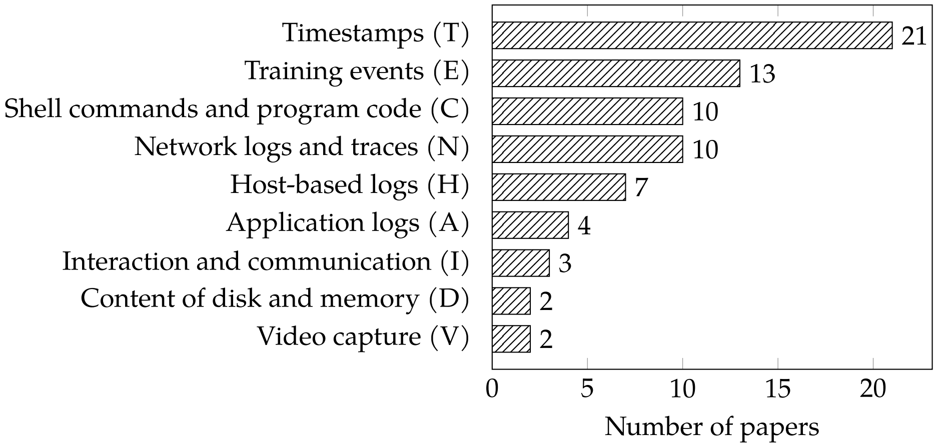

Figure 2.2: The distribution of occurrence of data types in the 35 papers.

### Knowledge and Skills Exercised in Capture the Flag Games

Capture the Flag game is a popular form of state-of-the-art hands-on cybersecurity education (see Section 2.1). Students solve diverse game tasks in an informal setting, most often competitions, which is different from traditional teaching methods delivered by schools. We mapped CTF tasks to formal cybersecurity curricular guidelines to understand how the skills practiced by the tasks match curricula defined by experts and educators. We analyzed 15,963 written solutions of CTF tasks published by players of 969 games held since 2012 to 2020 [19]. In particular, we extracted cybersecurity keywords from the CSEC2017 guidelines [1] and searched for them in the solutions of CTF tasks.

The most prominent knowledge area covered by the tasks were *Data security*, *Connection security*, and *System security*, see Figure 2.3. These three areas were also the most prominent in published academic papers (see Figure 2.1). The most prominent knowledge units were *Cryptography*, *Component design*, *Implementation*, and *System control*. The results show room for covering other areas in the CTF games, such as human aspects of cybersecurity. For instance, although phishing is a ubiquitous cyber threat, our analysis showed CTF tasks do not address this topic. Also, knowing the most frequent knowledge areas and units can help prepare beginners for CTF games.

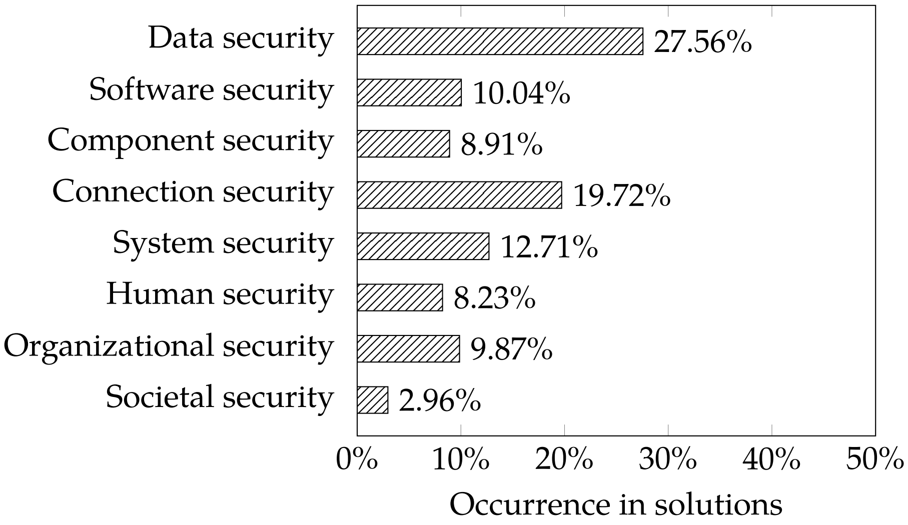

Figure 2.3: The distribution of Cybersecurity Curricular Guideline Knowledge Areas [1] in the solutions of CTF tasks.

## Articles in Collection

The second part of this thesis contains the following papers of the author, which are related to this chapter. The papers are ordered by the publication date.

1. V. Švábenský, **J. Vykopal**, and P. Čeleda. "What Are Cybersecurity Education Papers About? A Systematic Literature Review of SIGCSE and ITiCSE Conferences". In: *Proceedings of the*

*51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, March 2020, pp. 2–8. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366816

- Main track (Computing Education Research)
- CORE conference rank: **A**
- Contribution: **30%**
- CRediT author statement [52]: Conceptualization, Methodology, Validation, Formal Analysis, Investigation, Data Curation, Writing – Original Draft
- **Best Paper Award** in the Computing Education Research track

2. V. Švábenský, P. Čeleda, **J. Vykopal**, and S. Brišáková. "Cybersecurity Knowledge and Skills Taught in Capture the Flag Challenges". In: *Elsevier Computers & Security* 102.102154 (March 2021). ISSN: 0167-4048. DOI: 10.1016/j.cose.2020.102154
   - IF: **5.105** (in the year 2021)
   - Rank: **Q2** (Q1 based on the Journal Citation Indicator)
   - Contribution: **20%**
   - CRediT: Validation, Writing – Review & Editing, Supervision

3. V. Švábenský, **J. Vykopal**, P. Čeleda, and L. Kraus. "Applications of Educational Data Mining and Learning Analytics on Data From Cybersecurity Training". In: *Springer Education and Information Technologies* 1360.2357 (2022). ISSN: 1573-7608. DOI: 10.1007/s10639-022-11093-6
   - IF: **3.666** (in the year 2021)
   - Rank: **Q1** (Q1 based on the Journal Citation Indicator)
   - Contribution: **30%**
   - CRediT: Conceptualization, Methodology, Validation, Formal Analysis, Investigation, Data Curation, Writing – Original Draft, Supervision

# 3 Training Environments

Cybersecurity training environments enable students to complete tasks involving computer systems, hosts, or networks prepared by instructors. Some environments provide various features for analyzing students' progress and achievements during or after the training. Figure 3.1 shows essential components of a training environment.

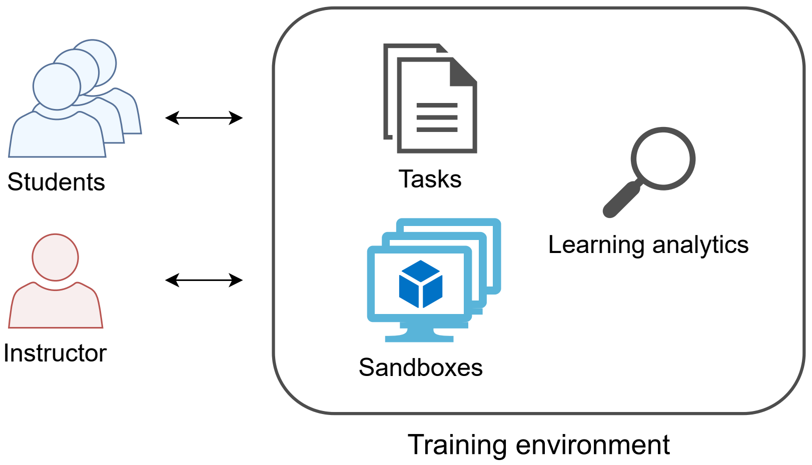

Figure 3.1: A generic cybersecurity training environment.

Training takes place in various technical environments. Some instructors still use dedicated labs with bare-metal hosts and network devices; others prefer virtual labs (hosted locally or in the private or public cloud [53]). Virtual labs span different technologies: from lightweight containers [54] or a single virtual machine [13] to full-fledged cyber ranges, dedicated testbeds, and infrastructures that help realize and execute the training scenarios [55].

We research and develop scalable environments enabling running the same training session in both local and cloud deployment (see Section 3.1, and [56] for details) and an environment for training featuring industrial control systems (see Section 3.2, and [57] for details).

## 3.1 Scalable Virtual Learning Environments

These environments can be deployed locally, either at school-owned or student-owned hosts, or in a cloud. This flexibility enables many students to learn in a small environment or fewer students to learn in an extensive or complex environment. In both cases, students can practice from their school, workplace, home, or other places connected to the Internet. The environment can be repeatedly created for different classes on a massive scale or for each student on-demand.

The environments emulate real-world systems, applications, and infrastructures using virtual networks with full-fledged operating systems, devices, and applications used in authentic workplace settings. Learners' interaction with the emulated systems is driven by a serious game or a step-by-step tutorial facilitated by the learning environment with or without a human instructor's assistance. Regardless of the session's goal, the environments support collecting data from the emulated systems for further analysis, such as learning analytics.

From an instructor's perspective, the environments are described as code using open and standard formats, definitions of individual hosts and their networking, configuration of the hosts, and tasks that the students solve. Consequently, these components can be reused in other learning technologies.

### Building Blocks

Each environment consists of three essential blocks: *sandbox*, *class delivery method*, and *learning analytics*.

### Sandbox

Sandbox is an isolated environment for practicing cybersecurity skills. It is the essential component within the interactive learning environment. In our work, we use the term *sandbox* for either a single *virtual machine* (VM), or a network of VMs. Users can run the sandboxes on their own hosts or access them remotely if deployed in a cloud. Once all the VMs are booted, networked, and running, i.e., they constitute a sandbox, users can interact with them using console or remote access.

### Class Delivery Method

The class delivery method is a computer-assisted instruction that employs the sandbox. It is provided as a part of the learning environment. It can use teacher- or student-centered approach. For instance, instruction can be provided using static (rich) text *assignments* or *serious games*. Some instructional methods require the assistance of an instructor, who also assesses students' progress or outcome. These methods include labs integrated with lectures, project-based learning, and problem-based learning [58]. Other methods fully rely on a learning environment that presents assignments, provides feedback, or assesses students' outcomes without a human instructor's assistance. These methods include self-directed learning or automated tutoring systems [58].

### Learning Analytics

We refer to components analyzing students' interactions in both the sandbox and the class delivery as *learning analytics* (LA). It can be used during and after the class with different goals. During the class, instructors can monitor students' progress and provide formative assessment to students. It can also be used for summative assessment, i.e., student testing and grading. After the class, LA enables students to reflect on the class and plan further learning. Instructors can use LA to improve the class for future runs.

### Approach

Since preparing a hands-on class is a laborious and complex task, instructors strive to use the building blocks repetitively. However, the manual deployment of the sandbox and the class delivery is time-consuming, and it does not scale well for big classes.

We therefore introduce reusable components of the building blocks, which make delivering cybersecurity hands-on classes scalable.

- *Topology definition* and *provisioning definition* are two parts of *sandbox definition*, which specify the internal structure of the sandboxes (networks and hosts) and the configuration of the hosts.

- *Training definition* specifies the tasks and questions for students in the class.
- *Learning analytics stack* is a key component for providing both formative and summative assessments to students.

These components are cornerstones of the two learning environments we have developed.

### Implementation

KYPO Cyber Range Platform (KYPO CRP) [59] is a cloud-based platform designed for running multiple classes in parallel or classes requiring sandboxes with many hosts. Cyber Sandbox Creator (CSC) [60] is a lightweight, distributed lab environment using a commercial off-the-shelf computer in the lab or students' own desktop or laptop.

Both environments use the same formats for topology, provisioning, and training definitions and the same formats of events processed by the learning analytics stack. The key difference is the virtualization technology used for sandbox instantiation. This is most evident for base boxes, which are almost identical except for the features and limits bound to the different underlying virtualization technologies. Another important difference lies in user roles and access control to sandbox and training instances. Both environments have been released as open-source software with documentation and examples of training.

### Cloud-based Learning Environment

The components and usage of KYPO CRP is shown in Figure 3.2. First, the instructor checks that estimated resources for the training session are available in the cloud. The estimated resources are based on the number of students in the class and the number of hosts and networks in the sandbox. After that, the instructor logs in to the web interface of KYPO CRP and allocates a pool of sandboxes for the class. The pool size is usually set to the number of students plus a few more for a reserve. Further, the instructor provides a Gitlab repository with sandbox and provisioning definitions. The repository may enable auxiliary services of the learning analytics stack. KYPO CRP then builds all sandboxes in the allocation pool in the cloud using sandbox and provisioning definitions.

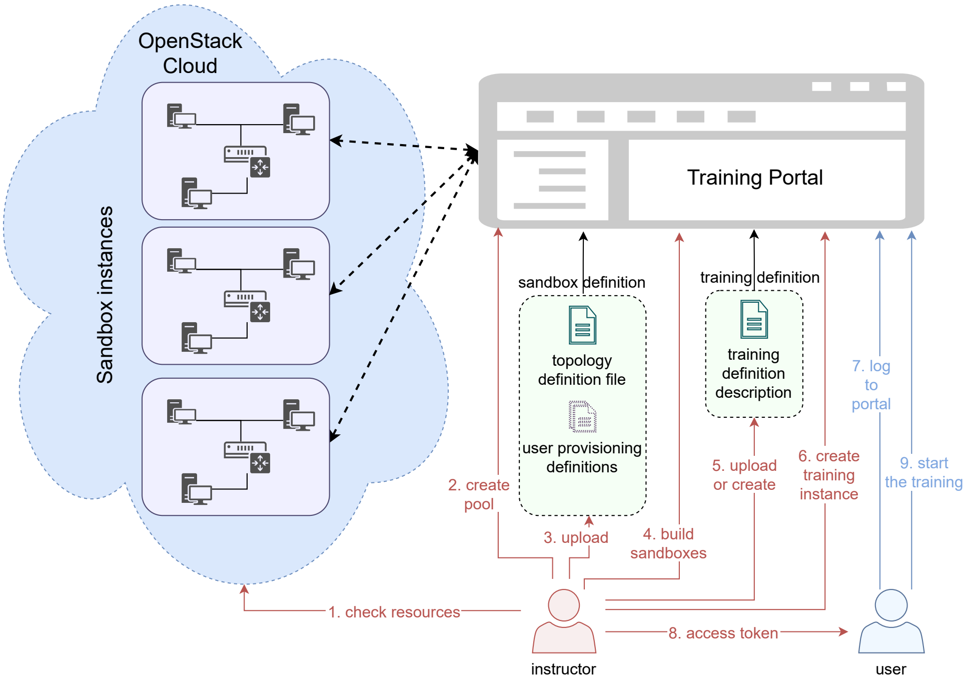

Figure 3.2: The use of the KYPO CRP learning environment.

Next, the instructor creates or imports a training definition describing the task assignments. As the last step, the instructor creates a training instance for a particular class. The instance defines the start and end times of the training session and a respective pool of sandboxes. KYPO CRP thus enables instructors to run multiple (different) classes in parallel.

The student logs into the web interface of KYPO CRP (training portal) and starts the training by entering an *access token* provided by the instructor. If the entered access token is correct, an available sandbox instance from the pool is assigned to the particular student. The student then starts solving the tasks defined in the training definition by interacting with the sandbox instance hosted in the cloud. KYPO CRP provides access to the sandbox host using a web browser, presents tasks, provides predefined on-demand hints, checks submitted answers, and collects events from its web interface and allocated sandboxes. The instructor can monitor the progress of all students in the class during the training on a dashboard. When the training is over, analyses of student progress are available both to the student and instructor.

### Lightweight Learning Environment

The components and usage of CSC are shown in Figure 3.3. First, an instructor (superuser) writes or reuses a sandbox definition, from which an intermediate definition is generated. This can be further extended by provisioning definition, which specifies the software and configuration of the machines in the sandbox.

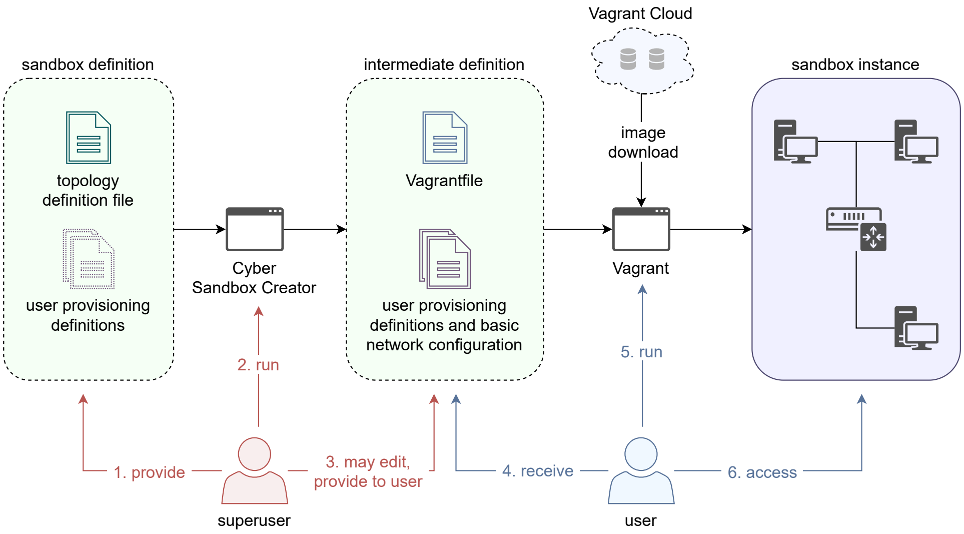

Figure 3.3: The use of the CSC learning environment.

The student (user) receives this intermediate definition and builds the sandbox locally using a single command. This is achieved by using Vagrant, which downloads images of operating systems for the VMs (base boxes), and provisions them according to the instructor's configuration. As a result, the sandbox instance is ready on the student's computer.

The student performs the hands-on tasks in the sandbox instance. The task assignments and scaffolding are delivered via separate software, such as the KYPO CRP training portal. Moreover, the instructor can enable auxiliary services in the sandbox definition, such as command logging from the learning analytics stack. As a result, the commands submitted by the student in the sandbox are forwarded to a central storage, where they can be processed further or viewed by an instructor.

### Use Cases

Since both environments are based on common components, they enable the following use cases.

**Running the same training in either environment:** The common formats of training and sandbox definitions enable running training sessions in either KYPO CRP or CSC without additional instructor's effort. The instructor can consider which features of the particular environment better fit the needs of a particular class and choose a more suitable environment. An example training demonstrating this feature is publicly available at [61].

**Sharing of created training between organizations:** The human-readable formats of training and sandbox definitions enable their use in other organizations in the KYPO CRP and CSC learning environments. Alternatively, it simplifies the adoption in different environments if they are based on the same or similar building blocks. For instance, training definitions could be easily transformed into a learning environment that is based on different technology for sandbox provisioning. Or, a sandbox definition of a typical small enterprise network could be used for different training sessions.

### Teaching Experience

We have been developing and using KYPO CRP in practice since 2013. To complement its capabilities, we started the development of CSC in 2019. Here we report our teaching experience since 2019 when we have an opportunity to choose the most suitable environment for the particular training session.

In total, about 650 students used our learning environments in 38 training sessions. The students were undergraduates and graduates, professional learners, and selected high school students attending a cybersecurity competition. While most students received the training, some were involved in creating the training and cybersecurity games as described in [62]. They were able to create sandbox and training definitions using the learning environment CSC and run the training for their peers in either the learning environment KYPO CRP or CSC.

Some sessions were held by other instructors at other institutions in several European countries. Both students and instructors were able to run the training in the environment CSC without our in-class assistance.

The learning environment KYPO CRP was successfully deployed in a private cloud at our university, Brno University of Technology, Czech Republic, Swedish Defence Research Agency (FOI), and the National University of Singapore.

#### Classes Aimed at Formative Assessment

When teaching practical lab sessions at university courses, our goal is that students gain hands-on experience with using various cybersecurity tools. In this low-stakes context, we do not care if the students see the sandbox definition, so we usually decide to use CSC. Regardless of the number of students, everyone deploys the sandbox locally on their own computer. This way, we have taught classes from 10 up to 200 students.

However, the teachers need to allocate time for preparing detailed setup instructions, and be ready to troubleshoot the setup. The students use a wide range of host operating systems (e.g., Windows, macOS, and different Linux distributions), and the hardware configuration of their devices vary widely. As a result, some students may experience early difficulties before the environment is ready to run on their machines. In particular, a few students encountered issues with the installation of VirtualBox.

Alternatively, we may opt-in for the KYPO CRP. The setup is simpler for students, as the instructors prepare sandboxes in the cloud, and students access them remotely using a web browser or SSH. However, the cloud's resources limit the number of sandboxes we can host. We usually teach classes from 10 to 30 students with this setup. An additional risk is that if the cloud space is shared, external users running other computational tasks in the cloud may jeopardize the stability of the environment and user experience. This risk is not associated with the CSC when students use dedicated hardware.

Regardless of the environment used, we can collect command histories of students solving the tasks [63] and provide them with formative feedback. This includes explaining what they did well and what they can improve, for example, how to address frequently occurring mistakes.

### Classes Aimed at Summative Assessment

When we need to perform summative assessment, such as during a final exam or a competition, we need to hide the sandbox definitions from students. Therefore, we only use the KYPO CRP environment for this use case. We can control the visibility of hosts in the sandbox topology so that students are initially aware only of a limited number of machines. They also cannot directly see what applications are running there, so the setup mimics more realistic situations.

## 3.2 Cyber-physical Learning Environments

Industrial Control Systems (ICS) are used to control processes such as manufacturing, product handling, production, and distribution [64]. Since they are crucial for vital services, such as electricity, water treatment, and transportation, they are attractive targets of cyber attacks [65, 66].

Cybersecurity courses are falling short in training ICS security [67] since they focus on exploiting and defending IT assets. To teach ICS security, a training facility (testbed) is needed to model a real-world ICS system [68] and to provide hands-on experience. However, building and operating a realistic cyber-physical testbed using standard industrial equipment is expensive. It incorporates equipment such as programmable logic controllers (PLC), input/output modules, sensors, actuators, and other devices. As a result, high-fidelity testbeds are rare, and most testbeds use simulations, scaled-down models, and individual components [67].

Our objective is to create an ICS testbed for beginner and intermediate computer science students to learn cybersecurity in a simulated industrial environment with real hardware components such as sensors and actuators. The testbed is built using open-source hardware and software and provides reconfigurable modules of industrial control systems.

### ICS Training Facility

The ICS testbed is placed in a training facility involving six large movable tables, each with three seats, three desktop PCs, and ICS hardware devices as depicted in Figure 3.4.

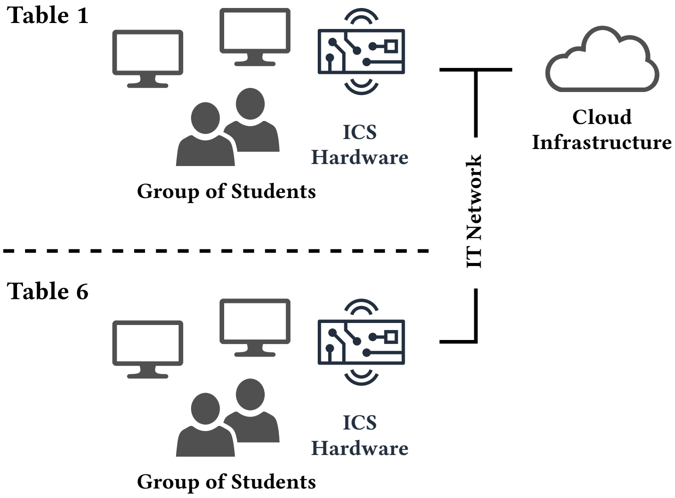

Figure 3.4: ICS training facility setup.

### Hardware Components

Figure 3.5 shows a physical hardware setup of the ICS testbed. Each ICS device contains a linear motor, a high-power LED (heater), large-area LEDs, temperature and light sensors, a key switch, buttons, input/output modules, programmable logic controllers, a touchscreen, a seven-segment and e-paper display, and a communication gateway to the IT network as depicted in Figure 3.6.

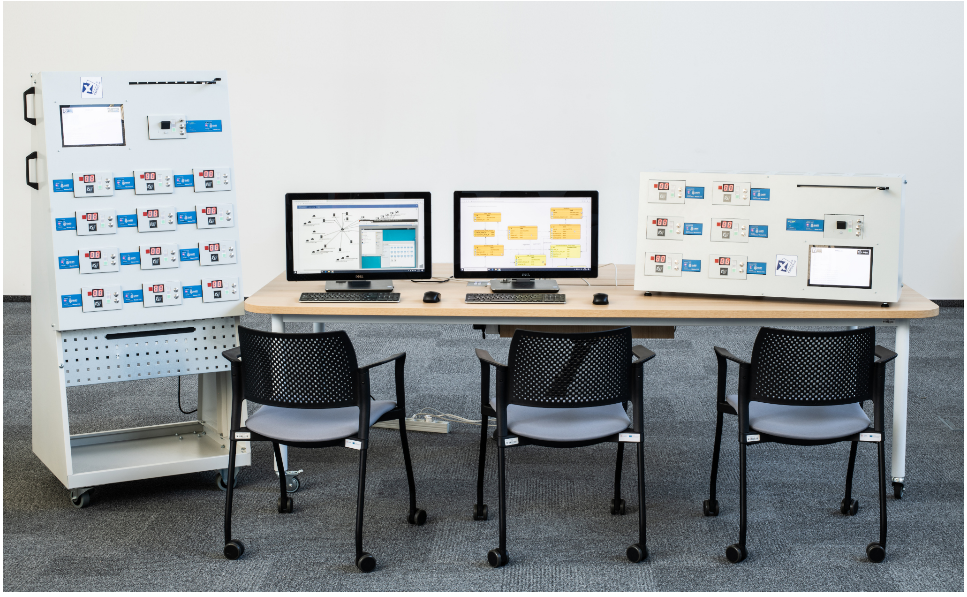

Figure 3.5: Physical hardware setup of the ICS testbed.

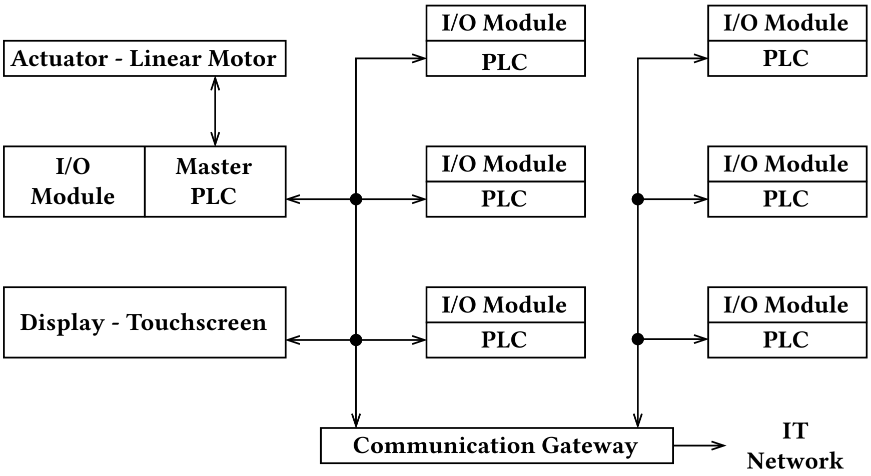

Figure 3.6: Control panel block diagram.

### Software Components

The software stack of the ICS testbed includes Linux OS (Debian optimized for PLC devices), Docker containers [69], and on-premise OpenStack [70] cloud environment.

Table 3.1: The schedule and the structure of the course.

| | **Class content** | **Student homework task** |
|---|---|---|
| 1 | Motivation, real attacks, legal issues | Prepare a presentation about an ICS attack |
| 2 | Student presentations of chosen attacks | Read this paper and some of the references |
| 3 | Hands-on labs on ICS testbed familiarization | Write an ICS security threat landscape report |
| 4 | Threat discussion, demo on ICS testbed | Write a short survey of CTF games in ICS |
| 5 | Merge surveys, introduce game concepts | Select threats for your game |
| 6 | Threat modeling, storyline, consultation | Write a game draft |
| 7 | Preparing ICS part, educational objectives | Add learning outcomes and prerequisites |
| 8 | Preparing ICS and IT part | Prepare an alpha version of the game |
| 9 | Dry run of the games with peers | Improve the game, submit bug reports |
| 10 | Bug presentations, game improvement | Improve the game |
| 11 | Documentation, automation, deployment | Submit the game for presentation |
| 12 | Public run of the games | Write a reflection from the public run |
| 13 | Final reflections | Fix any issues that emerged in the public run |

### ICS Cybersecurity Training with the Testbed

We used the testbed in practice in a course providing undergraduate students with an awareness of threats within the ICS domain via hands-on experience. The course was organized in 13 weeks and taught as flipped classroom [71] with 2-hour lab sessions, homework assignments, and a hands-on project. The project's outcome was a training game for exercising attacks at and defense of an industrial process. Table 3.1 summarizes the course syllabus and student deliverables. The course is divided into three parts: basics of ICS, development of an ICS training game, and its presentation and submission.

The first run of the course in 2019 showed students could learn ICS essentials and created the game featuring the ICS testbed. The games were played by other students who enjoyed them and also learned about ICS. Two students enhanced their games and the testbed in their bachelor and master theses.

The first run also revealed two limitations. First, some hardware components, such as storage cards and sensors, have worn out and must be replaced. In contrast to a training environment featuring solely virtual devices, replacing physical components consumes more time and effort and brings additional costs. Second, students need a lot of guidance on creating games with the ICS component and vulnerabilities. Otherwise, they develop games featuring traditional IT and vulnerabilities and do not fully exploit the capabilities of the ICS testbed.

## 3.3 My Contributions

Since 2013, I have been designing, testing, and evaluating the presented open-source training environments. KYPO Cyber Range Platform and Cyber Sandbox Creator have been deployed and used in practice by several institutions in the Czech Republic and abroad. Both environments[1] serve as vehicles for further research presented in the following chapters. In addition, KYPO CRP received national and international awards (see below).

### Key Results

I co-authored five regular conference papers, two complex software projects, and one data article related to training environments. The list of the results is ordered by the publication date.

1. **J. Vykopal**, R. Oslejsek, P. Celeda, M. Vizvary, and D. Tovarnak. "KYPO Cyber Range: Design and Use Cases". In: *Proceedings of the 12th International Conference on Software Technologies – Volume 1: ICSOFT*. INSTICC. SciTePress, 2017, pp. 310–321. ISBN: 978-989-758-262-2. DOI: 10.5220/0006428203100321
   - Main track
   - CORE conference rank: **B**
   - Contribution: **20%**
   - CRediT: Conceptualization, Methodology, Software, Investigation, Writing – Original Draft

2. P. Čeleda, **J. Vykopal**, V. Švábenský, and K. Slavíček. "KYPO4INDUSTRY: A Testbed for Teaching Cybersecurity of Industrial Control Systems". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland,

[1] Presented together at https://kypo.muni.cz/technologies.

OR, USA: Association for Computing Machinery, March 2020, pp. 1026–1032. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366908

- Main track (Experience Reports and Tools)
- CORE conference rank: **A**
- Contribution: **30%**
- CRediT: Conceptualization, Methodology, Investigation, Writing – Original Draft

3. V. Švábenský, **J. Vykopal**, P. Seda, and P. Čeleda. "Dataset of Shell Commands Used by Participants of Hands-on Cybersecurity Training". In: *Elsevier Data in Brief* 38 (September 2021). ISSN: 2352-3409. DOI: 10.1016/j.dib.2021.107398

   - IF: **N/A** (in the year 2021)
   - Rank: **Q3** (based only on the Journal Citation Indicator)
   - Contribution: **30%**
   - CRediT: Methodology, Software, Validation, Investigation, Resources, Data curation, Writing – review & editing

4. **J. Vykopal**, P. Čeleda, P. Seda, V. Švábenský, and D. Tovarňák. "Scalable Learning Environments for Teaching Cybersecurity Hands-on". In: *Proceedings of the 51st IEEE Frontiers in Education Conference*. FIE '21. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637180

   - Main track (Innovative Practice)
   - CORE conference rank: **C** (**B at the time of paper submission**)
   - Contribution: **35%**
   - CRediT: Conceptualization, Methodology, Software, Investigation, Resources, Writing – Original Draft, Writing – Review & Editing, Visualization, Supervision, Project administration, Funding acquisition

5. V. Švábenský, **J. Vykopal**, D. Tovarňák, and P. Čeleda. "Toolset for Collecting Shell Commands and Its Application in Hands-on Cybersecurity Training". In: *Proceedings of the 51st IEEE Frontiers in Education Conference*. FIE '21. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637052

- Main track (Innovative Practice)
- CORE conference rank: **C** (**B at the time of paper submission**)
- Contribution: **30%**
- CRediT: Conceptualization, Methodology, Software, Validation, Investigation, Resources, Data Curation, Writing – Original Draft, Supervision

6. KYPO Cyber Range Platform. 2013–2022. Open-source software available at https://gitlab.ics.muni.cz/muni-kypo-crp.
   - Deployed and used by various institutions and companies in the Czech Republic and abroad.
   - The winner of the Disruptive Tech category of the European Commission's 2021 Innovation Radar Prize[2].
   - The winner of the 2016 Award of the Czech Minister of the Interior for Security Research[3].
7. Cyber Sandbox Creator. 2019–2022. Open-source software available at https://gitlab.ics.muni.cz/muni-kypo-csc/cyber-sandbox-creator.
   - Used by various institutions in the Czech Republic and abroad.

[2] See https://www.innoradar.eu/innoradarprize.

[3] See https://www.mvcr.cz/clanek/cenu-ministra-vnitra-za-mimoradne-vysledky-v-oblasti-bezpecnostniho-vyzkumu-ziskal-tym-ustavu-vypocetni-techniky-masarykovy-univerzity.aspx (in Czech only).

# 4 Instructional Methods

This chapter covers various methods of delivering cybersecurity training to individual learners or teams. First, we report our research on complex cybersecurity defense exercises (Section 4.1). Then, we discuss methods involving cybersecurity serious games (Section 4.2). Finally, we focus on efficient learning driven by learning analytics (Section 4.3).

## 4.1 Cybersecurity Exercises

We determine the life cycle of a complex cyber defense exercise and challenges related to the exercise's design, development, execution and repeatability [74]. This contribution is based on our experience gained by developing and delivering six runs of a cyber defense exercise scenario with about 120 national and international learners between 2015 and 2017. The exercises have been carried out in the KYPO Cyber Range Platform (Section 3.1).

The exercises follow a Red vs. Blue team format. This implies that exercise participants are divided into teams according to their roles and responsibilities, see Figure 4.1:

- *Red team* – plays the role of attackers and consists of cybersecurity professionals.
- *Blue team* – learners responsible for securing compromised networks and dealing with the Red team's attacks.
- *Green team* – a group of operators responsible for the exercise infrastructure.
- *White team* – exercise managers, referees, organizers, and instructors.

### Exercise Life Cycle

The exercises last several hours or days, but their preparation typically takes several person-months involving experts from various fields

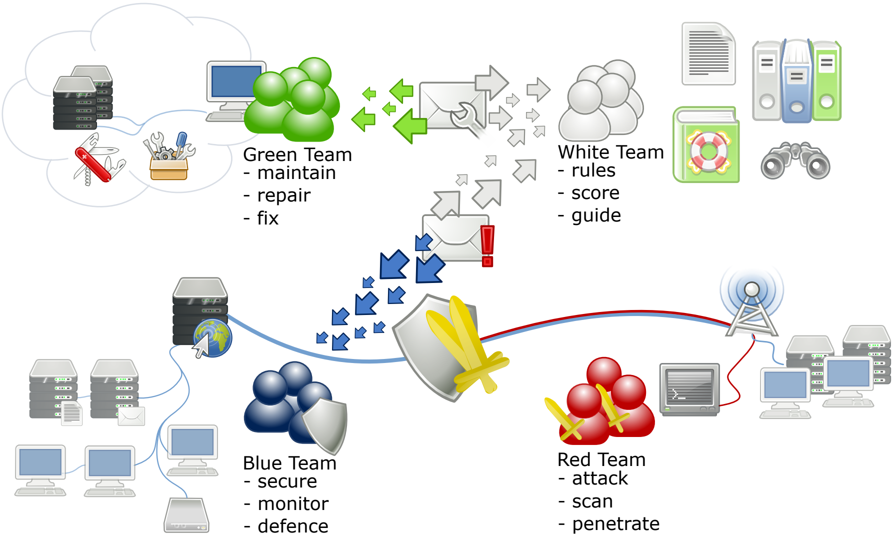

Figure 4.1: Cyber defense exercise participants, their interactions and tasks.

(such as IT administrators, cybersecurity specialists, and legal experts). We introduce the exercise life cycle consisting of four phases that can be mapped to a well-known Plan–Do–Check–Adjust (PDCA) cycle:

1. Preparation.
2. Dry run.
3. Execution.
4. Evaluation.

Carefully planning and considering the relationship of all phases saves a significant amount of invested effort and costs. Figure 4.2 shows the involvement of all teams and effort spent through the cyber exercise life cycle.

### Lessons Learned

The preparation phase consumes the majority of work effort and time. In our experience, the most challenging tasks in this phase are:

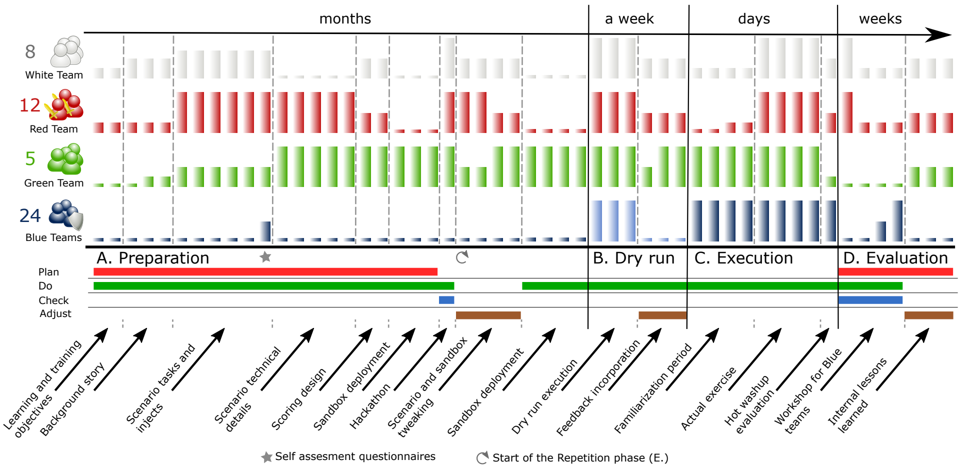

Figure 4.2: Cyber exercise life cycle in time. Coloured bars show relative effort spent by members of White, Red, Green, and Blue teams in respective phases of the life cycle. The four numbers on the left express the size of a particular team in the exercise. The mapping to the PDCA cycle is depicted by coloured lines below the life cycle phases.

- *Setting learning objectives with respect to expected readiness of prospective learners* – We strongly recommend considering a profile of the prospective learners to balance learning objectives and learners' proficiency. The self-assessment questionnaires may provide useful information.

- *Creating balanced teams* – We advise creating teams of learners who possess the necessary skills. For instance, distribute experts in one field to all teams equally.

- *Sandbox configuration documents* – Continually updated specification of used systems, network configurations, and vulnerabilities is crucial for the successful and smooth preparation of the sandbox. We recommend following the Infrastructure as Code (IaC) principle.

The following exercise phase is the dry run, a complete test of the proposed cyber exercise to get diverse feedback. We learned that adjusting exercise scoring based on the dry run might be misleading if the expertise and size of the testing Blue teams participating in the dry run are not similar to target learners.

The execution phase is where the actual exercise takes place for the target learners. We identified five challenges related to it:

- *The level of guidance by organizers* – We advocate providing some hints by the White team to keep the learners in the exercise flow and not to be frustrated because they stuck at one point. However, guidance should be provided to all teams equally to preserve fair play.
- *Exercise situational awareness for learners* – Displaying a real-time total score of all teams on a shared scoreboard is the appropriate indication of learners' performance. It also fuels participants with stress as well as a competitive mood.
- *Exercise situational awareness for organizers* – Monitoring of exercise infrastructure deployed by the Green team enables the White team to provide hints for the Blue teams to distinguish between misconfigurations done by them and exercise infrastructure outages.
- *Service access to exercise infrastructure* – To recognize exercise infrastructure failure from scenario progression (e. g., Red team's attack or Blue team's misconfiguration), the Green team needs service access to all sandbox components.
- *Automation of the attacks and injects* – Since the exercise scenario is fixed and rigid, Red and White teams may benefit from semiautomated routines that execute the predefined attacks and injects.

The exercise life cycle ends with an evaluation. It consists of an assessment of team actions and performance during the exercise, a feedback survey and debriefing, and gathering lessons learned by the organizers. We point out that the learning also happens in the evaluation phase, particularly for novices and learners who rated the exercise as difficult. The debriefing shows the exercise scenario and timeline from the perspective of the Red team and White team. It is the only opportunity when the learners can authoritatively learn about attacks used by the Red team. They can discuss their approach in particular situations and phases. So far, they can only see the results of their experimentation during the exercise without the explanation *why* something happened. The evaluation is usually finished by an after-action report. This document highlights key conclusions from a laborious manual analysis of heterogeneous data acquired during the exercise (survey, written communication, scoring and monitoring logs, and checks). An in-depth analysis of learners' actions requires considerable human effort, which results in days or weeks of delay of the delivery of the report.

### Timely Feedback to Exercise Participants

We researched how to provide valuable feedback to learners *right after* the exercise without any unnecessary delay. In particular, whether learners benefit from simple, but individualized feedback provided just after the end of a two-day intensive exercise [75].

Based on the CDX scoring system, we have developed a new feedback tool that presents an interactive, personalized timeline of exercise events. We deployed this tool during an international CDX, where we monitored participants' interactions and gathered their reflections. Each team was provided with an interactive timeline of its score development during the exercise, with important events emphasized. The timeline was generated automatically from data stored by an existing scoring system. Interactions of exercise participants (mouse clicks and movements) were logged with the scoring timeline. After that, participants were asked to fill out short evaluation questionnaire.

The data and answers we obtained show that learners valued the feedback, even though they still lack more details about particular events. The analysis of learners' interactions with the scoring timeline shows that all teams were using it intensively, regardless of what reflection they provided in the scoring timeline survey. All teams explored all the penalties depicted in their timeline. They also gave us an evaluation of the majority of displayed tasks. The top performing team rated the scoring timeline as less useful than other teams. We believe the feedback may not have been so interesting for the team because it might have already known about its failures. However, even this team was interested in the timely feedback because it explored the scoring timeline for the longest. Regarding the more details about the events, three out of five teams would appreciate knowing “how it happened” in addition to ”what happened”.

### Network Traffic Traces and Logs from CDX

Finally, we collected, normalized, and published network traffic flows and event logs (Linux and Windows) from a two-day cyber defense exercise we have organized [76]. This dataset is useful for cybersecurity experts and researchers that rely on primary security data in their work, e. g., in intrusion detection, traffic analysis, threat identification, and education and training. The lack of datasets coming from a realistic network environment leads to the inefficiency of newly designed methods that are not useful in practice. Since the exercise network was designed as a full-fledged digital twin of a fictitious organization, the data are equal to data generated in real enterprise networks. At the same time, indicators of multiple cybersecurity attacks can be found in the data, spanning a relatively short time interval. To the best of our knowledge, this is the first dataset providing network traffic traces and corresponding event logs from a complex cyber defense exercise where human operators deal with a number of attacks featuring recent vulnerabilities, applications, and systems.

## 4.2 Serious Games

### Learning by Teaching

In alignment with state-of-the-art curricular guidance, we developed two innovative undergraduate courses that apply the learning by teaching approach [77]. Students design serious games with the topic of cyber attack or defense. These games are offered to play by other students at the Open Day event at the end of the semester. The game creators have to cope with numerous interdisciplinary tasks throughout the semester while exercising a broad spectrum of technical and soft skills: system administration, penetration testing, game design, teamwork, project planning, communication, and presentation.

### Course Format

The courses consist of 12 weeks of 2-hour sessions and homework assignments. Table 4.1 lists the learning outcomes and schedule of both courses.

The first course focuses on the basics of offensive cybersecurity. It provides theoretical and practical experience to students who elaborate on a game project on penetration testing in teams of two or three people. The course is intended for at most 24 undergraduates (sophomores and juniors) who passed prerequisite courses on privacy and computer networks and systems, can read technical papers, and write in English.

The second (follow-up) course is offered for at most six students who passed the Introductory course. This number of students enables the instructor to advise student projects thoroughly. The students learn how to secure a particular network service or application by designing a gamified tutorial on that topic. The tutorial consists of step-by-step instructions that enable the learner to secure the service or application running on a host in the cyber range.

While working on their game, students receive formative feedback in three settings: presentations of project milestones to the class, consultation sessions with the course tutors, and a test run of the game with security experts.

At the end of the semester, the students present their projects to a broad audience after incorporating formative feedback from classmates, tutors, and experts. Finally, the teachers review and mark the final revision of each project.

### Learning Experience

After three runs of the courses, we surveyed participants of the Open Day on their teaching experience.

In total, 41 game plays in teams of one to three people occurred. All of the players provided feedback on the game. The educational value was rated as Medium (9), High (27), and Huge (5). The overall quality was rated as Sufficient (1), Good (10), Very good (23), and Excellent (7). The play time ranged from 5 to 70 minutes (median 40 minutes). We attribute the large variance of the time to the fact

Table 4.1: The learning outcomes and schedule of the Introductory and the Follow-up course.

| | **Cyber Attack Simulation (Introductory)** | **Cyber Defense Tutorial (Follow-up)** |
|---|---|---|
| **Knowledge** | Describe the stages of a cyber attack | |
| | Understand system and application security threats and vulnerabilities (e. g., authentication attacks, DoS attacks, MitM attack, OWASP Top 10 vulnerabilities) | |
| | Name cyber defense and vulnerability assessment tools and their capabilities | |
| | Explain laws, regulations, policies, and ethics related to security and privacy | |
| **Skills** | Perform penetration testing focused on a particular threat or vulnerability | Secure a particular network service or application (e. g., Apache or Wordpress) |
| | Use a cyber range both as a learner and as a designer of games running in it | Perform penetration testing of the service or application |
| **Experience** | Give a presentation explaining the vulnerability selected for the game | |
| | Practical work in small teams (Introductory) or individual (Follow-up) including setting up and maintaining systems, assessing their vulnerabilities, and developing a new serious game or a gamified training tutorial | |
| | Give two presentations of the final project (Test run and Open day) and instruct learners who use it | |

| **Week** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Introductory** | Exemplary game | Network security basics, hands-on labs, homework | | | | | Game design tutorial, topic choice | Presentations, consultations | | | Test run | Open day | Final result |
| **Follow-up** | Topic choice | Concept consultations | | | Concept finalization | Technical consultations | | | | Presentation | | | |

that some players experienced technical difficulties and had to wait, to different skill levels of the attendees, and to their different game strategies (some just skipped through the game, others wanted to finish it without asking for any hint).

The self-reported learning experiences included mostly working with Linux terminal, using offensive security tools in Kali Linux distribution, and game-specific learning outcomes, such as packet analysis in Wireshark, securing Apache server or understanding particular vulnerabilities. Of the 41 player teams, 24 included optional comments, which were overwhelmingly positive.

### Teaching Experience

We highlight list six successes and five challenges we experienced over the three semesters of teaching and continuous innovation of our courses. These lessons were distilled based on our observations and the feedback of 46 enrolled students gathered through online surveys and informal discussions.

Successes:

1. The final presentation has a motivating impact.
2. Constraining student efforts is helpful.
3. A test run of the scenarios is helpful.
4. Regular checkpoints and in-class presentations helped identify and correct students' efforts.
5. The Open Day builds cybersecurity awareness.
6. The practical contributions of the courses include developing new serious games applicable in future teaching.

Challenges:

1. Preparing and employing the cyber range poses an additional burden for both instructors and students.
2. Running hands-on cybersecurity courses introduces a lot of extra work for instructors.

3. Finding a suitable and replicable vulnerability for the games is challenging.
4. The students face the challenge of self-managing a small team.
5. Students underestimate the complexity of the project.

Finally, since our experience showed this course format was successful, we used it when designing a similar course for training cybersecurity of industrial control systems described in Section 3.2.

### Assessment

### CTF games as homework assignments

Since CTF games are widely used for cybersecurity competitions and awareness events, we studied using jeopardy CTF games as homework assignments in an introductory undergraduate course. We believe that CTF games are a better assessment method of skills acquired during the semester, especially for large classes. Gamification features bring students a more enjoyable learning experience. Instructors should benefit from the automatic scoring of students' submissions and spend time consumed by the manual marking of students' submissions more efficiently.

We were interested in how students apply taught skills and knowledge in CTF games and what are the advantages and drawbacks of using such gamification in the context of tertiary education. We, therefore, prepared two jeopardy CTF games as homework assignments in a computer security course taught at the National University of Singapore in 2018. We collected and analyzed game events generated by students using the CTF portal, answers from two surveys, and students' marks from other forms of summative assessment of the course.

The first game (assignment) consisted of eight challenges covering topics taught in the first part of the semester: substitution ciphers, hashing, symmetric and asymmetric cryptography, RSA, and cryptanalysis. The second game consisted of 15 challenges on topics of the second part of the semester: network traffic analysis, port knocking, access control, buffer overflow, command injection, format string attack, and SQL injection. The difficulty of challenges varied largely – from a simple execution of one command to a multi-step solution involving binary debugging and writing a helper exploit script.

Out of 120 students enrolled in the course, 37 students agreed to participate in the study. The median age of the participants was 23. Only two participants had played any CTF game before. No participant was a member of any CTF team. Six were employed in a part-time IT-related job.

The detailed results and discussion can be found in our paper [78]. We conclude that replacing traditional homework assignments by CTF games is generally favorable for both instructors and students. The instructors can save time spent on marking the students' submissions and enable students learning practical skills in an interactive and enjoyable way. In the following text, we highlight several recommendations for instructors to overcome pitfalls we experienced.

Usefulness of hints:

1. **Indicate what a hint is about** – The information about hint cost (if any) is not enough for students to decide whether they will benefit from displaying the hint. We recommend adding a short description of what they can expect, such as "what tool to use", particularly in challenges offering two or more hints. Otherwise, students may display a hint which tells them what they already figured out.

2. **Test challenge assignments and hints before the game** – Ask teaching assistants or peer instructors to test the challenge descriptions and hints to balance what should be placed in the challenge assignments and what can be left for one or more hints. While the challenge assignments can be a bit fuzzy, hints should be clear and straightforward.

3. **Prepare backup hints** – Although hints have been tested, students may still struggle with some (advanced) challenges. Monitor the ongoing game (submissions, wrong flags, and hint usage) and be ready to add a new hint if needed.

Flag sharing:

1. **Set rules for students' collaboration during the game in advance** – Decide what will constitute plagiarism in your class. Is any discussion about challenges among students forbidden, or do you allow non-detailed discussions about challenge principles or techniques that can be used? Communicate these rules clearly and explicitly to students.

2. **Inform students about how you will check suspicious submissions in advance** – Describe a procedure that will be applied if instructors spot suspicious behavior. For instance, the instructor may (randomly) select several students for in-person demonstration of how they solved particular challenges.

3. **Structure related problems to challenge chains** – Challenge chains help not only with revealing plagiarism but also explicitly guide students on what must be solved first.

### Cheating Prevention and Detection

Sharing of correct answers among the students in the homework CTF games motivated our follow-up work [79]. Students can share answers because they all receive the same assignment with the same answer. While this is efficient for learning and developing skills, it also allows cheating when used in a summative assessment such as graded homework, a mid-term test, or an exam.

To prevent cheating, we use *Automatic Problem Generation* (APG) and apply it in the context of hands-on tasks completed in a computer lab environment. APG enables instructors to create modified versions of the problems (tasks). Each student is provided with one instance of the same problem. APG can thus mitigate the threat of copied or leaked answers [80]. Although APG has already been applied in computing disciplines, it is not commonly used in hands-on assignments involving a lab environment. We, therefore, developed an open-source software toolset for generating and submitting personalized tasks [81] and conducted a case study. The toolset consists of two core components, the *environment generator* and the *submission server*, see Figure 4.3.

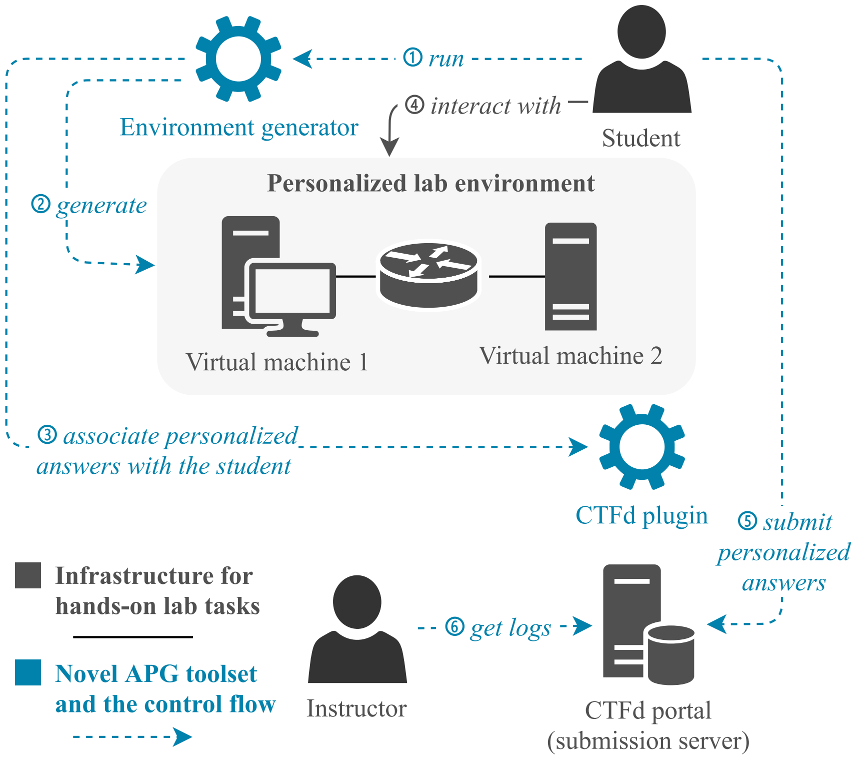
① run
④ interact with
Student
Environment generator
② generate
Personalized lab environment
Virtual machine 1
Virtual machine 2
③ associate personalized answers with the student
CTFd plugin
⑤ submit personalized answers
Infrastructure for hands-on lab tasks
Novel APG toolset and the control flow
⑥ get logs
Instructor
CTFd portal (submission server)

Figure 4.3: The APG toolset design. ① The student starts the environment generator with a unique seed. ② The generator creates a personalized lab environment. ③ Answers specific to that environment are stored in the submission database. ④ The student solves the tasks. ⑤ The submitted answers are checked against the generated personalized answers. ⑥ Instructor examines the submission logs for cheating.

The toolset was used for creating and grading a homework assignment in an introductory security course enrolled by 207 students. The assignment enhanced the skills students learned in the lab session before homework. In particular, it covered network attacks on authentication of Telnet and SSH servers, securing an SSH server, and capturing and analyzing SSH traffic.

To detect suspicious students' submissions, which may indicate cheating, we proposed three methods.

1. **Someone else's answers** – The most reliable detection method is tracking incorrect submissions of correct answers belonging to other students. This method assumes that some students shared their correct answers with other students who unthinkingly submitted someone else's answers.

2. **Task chains** – Another method benefits from locked tasks in a chain. Since a task in the chain is unlocked only after the previous task is successfully solved, this method computes the solve time for consecutive tasks. Then, the student's solve time is compared to the *minimal possible solve time* of a human who immediately performed all actions required to solve the tasks without any mistakes and time for thinking about the steps. Any student's solve time close to or lower than the minimal possible solve time may indicate that the student obtained step-by-step instructions from another student.

3. **Submission proximity** – The least conclusive method lies in searching for *time proximity* or *location proximity* of two or more submissions. Any of these proximities may indicate that students were working together and submitted the answers (correct or incorrect) at the same time or place. We consider submissions to be in the location proximity if they originated from the same IP address, which multiple hosts might share in some networks.

These methods revealed totally seven cases of suspicious submissions. The first method discovered three cases and the second and third method two cases each. The optional survey after the assignment was answered by 45 students. Forty students (89%) reported they would prefer the provided format of completing assignments in security courses. Only one student would prefer the traditional homework assignment, and the remaining four were not sure.

To conclude, we showed that prevention and detection of cheating in hands-on assignments involving the lab environment is possible in large and remote classes. What is more, our approach is lightweight and privacy-preserving. Even though students were not under surveillance when solving their homework, we discovered suspicious submissions only from minimal data collected (submitted answers, timestamps, IP addresses).

## 4.3 Efficient Learning

To make learning more efficient, we employ data about the interactions of learners with the learning environment. We *i*) propose training, which adapts to the proficiency and performance of the learner during the ongoing training, *ii*) research and apply visual analytics in the context of cybersecurity training, and *iii*) research methods for providing feedback to learners after the training or its phase.

### Adaptive Learning

Research of intelligent tutoring systems (ITS) and adaptive learning environment is well-established [82, 83]. There are examples of successful tutoring systems for various fields of computer science, such as SQL-Tutor [84] or ProTuS [85], or systems created by various authoring tools [86], even by non-programmers [87]. However, to the best of our knowledge, there are no ITS for hands-on cybersecurity training in a networked lab environment. We, therefore, proposed a new training format and a tutor model, which we integrated into the KYPO Cyber Range Platform and evaluated using several training sessions attended by 114 students [88].

### Generic Format of an Adaptive Training

The first cornerstone is a generic structure for adaptive cybersecurity training. In general, the training can contain an arbitrary number of phases and tasks. Each phase represents a learning activity. Each task in the phase exercises the same skills but varies in difficulty. Figure 4.4 shows an example of such structure with five phases: three with two tasks and two with three tasks of various difficulty.

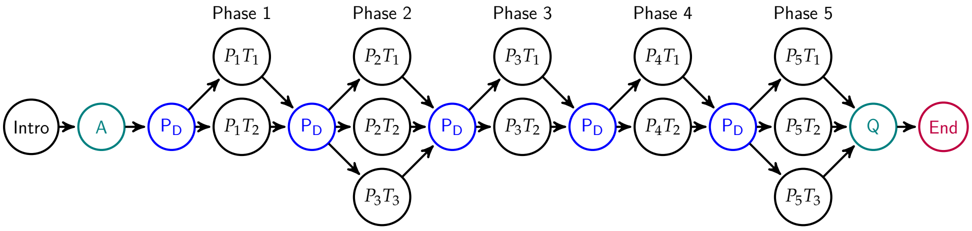

Figure 4.4: Graph structure of adaptive cybersecurity training.

The training consists of several components: the introduction (Intro), the pre-training assessment (A), training phases ($P_x$) including variant tasks ($T_y$), decision components ($\mathrm{P_D}$), and post-training questionnaire (Q).

First, the introduction (Intro) familiarizes the student with the training and communicates necessary information before the training starts.

The pre-training assessment (A) is the first component of collecting data about students' knowledge and skills. The questions asked in the pre-training assessment are grouped into *question groups* based on their relation to specific training phases. Each question can be assigned into several question groups since they can be relevant to more phases. For each training phase, we set the *minimal ratio* of knowledge to determine whether the student's knowledge or self-reported skills are sufficient or not. For example, the minimal ratio can be set to 100%, which would mean the students need to know answers to all the questions or self-report a defined level of skills for a particular phase. In particular, pre-training assessment should mostly include knowledge quizzes, as students' self-assessment can be inaccurate [89, 90].

The training phases contain tasks ($T_y$) that vary in difficulty but all aim at practicing the same topic. The decision component assigns exactly one task from the given phase. This assignment is based on the student performance in previous phases and on the results of the pre-training assessment. Students interact with their dedicated emulated environment, typically by entering shell commands, to find an answer: proof they completed the task. The student performance is measured by time, used commands, submitted answers, and a solution displayed in the phase. These performance indicators were selected based on the capabilities of the KYPO CRP platform and aligned with the review of metrics in cybersecurity exercises [91]. The tasks are denoted as $T_1, T_2, \ldots, T_n$, where $T_1$ represents the most difficult task in the phase and $T_n$ the easiest task in the same phase. We refer to $T_1$ as the *base task* and $T_2, \ldots, T_n$ as *variant tasks*. Further, the decision component ($\mathrm{P_D}$) processes the students' performance and knowledge to assign a suitable task from the training phase.

Finally, the post-training questionnaire (Q) is an optional part of training, which enables instructors to collect immediate feedback from the students. Depending on the training objectives, the post-training questionnaire can be the same or different as the pre-training questionnaire.

### Data Collection

The tutor model, which selects the most suitable task for each student, requires to collect various types of data about student interactions. In particular, the model relies on answers from the pre-training assessment, training actions, and shell commands from the learning environment.

The training actions include answers submitted by the student in all phases, the action of revealing the task solution, and the action of correct/wrong answer to complete the task. All these data are timestamped and saved to the central storage.

When students interact with the emulated environment, they enter commands in shells such as BASH or Metasploit Console. These commands are captured at hosts in the environment in real-time and forwarded using the Syslog Protocol [92] to the central storage using Elastic Stack [93]. The commands are stored in JSON and timestamped with microsecond precision.

All hosts in the emulated environment use clock synchronization via the Network Time Protocol (NTP).This setting is a key requirement for time-correlating the captured commands with training actions and other data. The architecture for collecting shell commands is detailed in [94].

### Tutor Model

The proposed tutor model processes the collected student data and computes the most suitable task in a particular phase for each student. The model was developed with the aim to reinforce the cybersecurity training with respect to the commonly used performance metrics [91]. Nevertheless, it can be applied in any domain collecting such data.

Let us denote the variables $\boldsymbol{p}$, $\boldsymbol{k}$, $\boldsymbol{a}$, $\boldsymbol{t}$, and $\boldsymbol{s}$, which are the binary vectors on the correctness or incorrectness of prerequisites for a particular training phase. Vector $\boldsymbol{p}$ is defined as follows: $\boldsymbol{p} = \begin{pmatrix} p_1 & p_2 & \dots & p_m \end{pmatrix}$, where $m$ is the number of training phases. The other vectors use the analogous notation.

- ***p*** represents the (in)correctness of answers from the pre-training assessment,
- ***k*** indicates if the student used the expected key commands in the command line within the given task,
- ***a*** denotes whether the student submitted the expected answers to the task,
- ***t*** contains the information if the task was completed in a predefined time, and
- ***s*** contains the information whether the student asked to reveal the solution for the task.

The model is defined by the Equations (4.1) to (4.3). By Equation (4.1), we get the *decision matrix* $\boldsymbol{W}$ with weights for the individual phases' metrics. It is specific for each training phase. The weights represent the relationships between phases and their metrics. The value of the weight determines the importance of the metric to the phase. For instance, consider training with six phases where the third phase deepens the topic exercised in the first phase. In this case, we set the weights in the third matrix so that the selected weights for the metrics from the first phase are non-zero. The other performance metrics with weights set to zero are ignored.

The weights have to be manually set by the instructor since each training is unique. The number of decision matrices is equal to the number of training phases. The symbols $\pi, \kappa, \alpha, \theta, \sigma$ denote the columns in the decision matrices and the $i = 1, \ldots, m$ are the rows in the decision matrices.

By Equation (4.2) we get the *student's performance* based on the defined metrics and their weights for completed phases. The value of the performance is in the interval of $[0, 1]$. In Equation (4.2), $s$ is multiplied by $a$, $k$, and $t$ to distinguish between students who satisfy $a$, $k$, and $t$ metrics without using a solution and solved the task on their own.

By Equation (4.3) we get *the number of the most suitable task $y$* in phase $x$ for a particular student (1 is $T_1$, 2 is $T_2$, and so on).

$$\boldsymbol{W}^{(x)} = \left(w_{ij}^{(x)}\right), i = 1, \ldots, m, \quad j = \pi, \kappa, \alpha, \theta, \sigma \tag{4.1}$$

$$f(x) = \frac{\sum_{i=1}^{x} \left[ p_i w_{i\pi}^{(x)} + s_i \left( k_i w_{i\kappa}^{(x)} + a_i w_{i\alpha}^{(x)} + t_i w_{i\theta}^{(x)} + w_{i\sigma}^{(x)} \right) \right]}{\sum_{i=1}^{x} \left( w_{i\pi}^{(x)} + w_{i\kappa}^{(x)} + w_{i\alpha}^{(x)} + w_{i\theta}^{(x)} + w_{i\sigma}^{(x)} \right)} \tag{4.2}$$

$$T_y = \begin{cases} n_x, & \text{if } f(x) \text{ is equal to } 0 \\ \text{trunc}(n_x[1 - f(x)]) + 1, & \text{otherwise} \end{cases} \tag{4.3}$$

where:

$$\begin{aligned}
x &= \text{the phase a student is entering,} \\
y &= \text{the order of the task in a phase,} \\
T_y &= \text{the most suitable task of the phase } x \text{ for the student,} \\
n_x &= \text{the number of variant tasks in the phase } x, \\
p_i &= \begin{cases} 1, \text{ if question group } i \text{ from A is correctly answered} \\ 0, \text{ otherwise,} \end{cases} \\
k_i &= \text{commands corresponding to the phase } i \text{ were used,} \\
e_i &= \text{expected time to complete of the phase } i, \\
o_i &= \text{student's completion time in the phase } i, \\
t_i &= \begin{cases} 1, \text{ if } o_i < e_i \text{ in phase } i \\ 0, \text{ otherwise,} \end{cases} \\
s_i &= \begin{cases} 1, \text{ if the solution of the phase } i \text{ is } \textit{not} \text{ displayed} \\ 0, \text{ otherwise,} \end{cases} \\
a_i &= \text{answers corresponding to the phase } i \text{ were submitted.}
\end{aligned}$$

The model is used by the $\text{P}_\text{D}$ component (see Figure 4.4) of the adaptive training format. The component provides the collected data to the model and assign the most suitable task in the next phase based on the model output.

### Case Study Setup

To evaluate the proposed training format and tutor model, we conducted a case study. The objective was to investigate *i*) how efficiently were individual learners distributed to tasks of various difficulty and *ii*) students' and instructors' experience of using KYPO enhanced by adaptive learning capabilities. In the case of students, we are interested whether the lab eases their learning. In particular, whether low-performing students are provided with easier tasks, which enables them to complete the training in expected time. In the case of instructors, we analyze how much time and effort is saved compared to a manual assignment of training tasks to each student by instructors. The study framework is depicted in Figure 4.5.

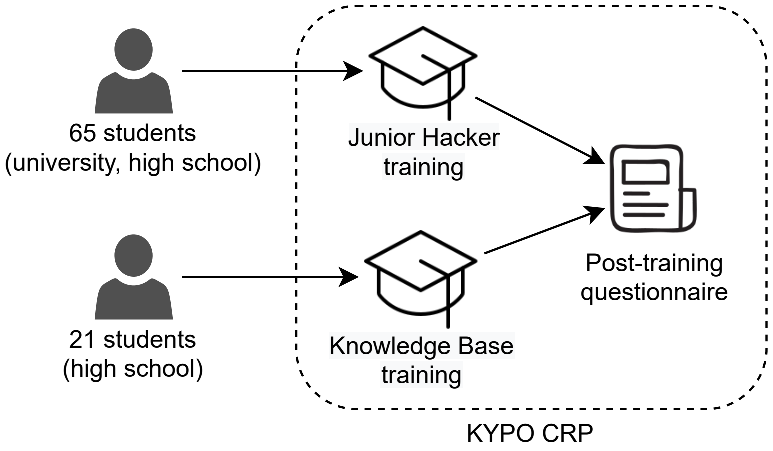

Figure 4.5: Study design: 114 students completed one of two adaptive trainings deployed in KYPO and answered a post-training questionnaire. Most training sessions (86 students) were facilitated by the instructor, but some (28 students) were not.

### Case Study Results

The first training (Junior Hacker) was finished by all 65 participants. Figure 4.6 shows the transitions of all participants between tasks ($P_x T_y$) in all training phases of this training. The diversity of transitions shows that the adaptive training enabled all participants to finish the training, yet by completing less difficult tasks.

The second training (Knowledge Base) was finished by 18 out of 21 (86%) participants. Figure 4.7 shows the transitions of 21 participants between tasks ($P_x T_y$) in the phases of Knowledge Base training. This training session was attended by senior high school students and undergraduates who were finalists of the Czech national cybersecurity competition. Although we expected better performance of this group, Figure 4.7 shows that students also solved easier variants of the tasks in all phases except Phase 1. This shows that adaptive training was beneficial even in this target group.

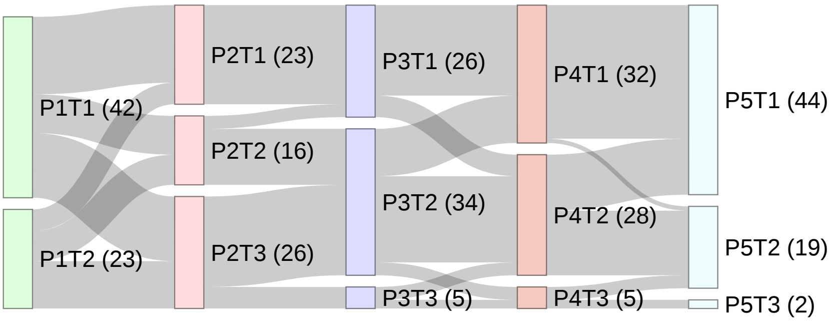

Figure 4.6: Transitions of 65 students between particular tasks in Junior Hacker training. $P_xT_y$ denotes task $T_y$ in the phase $P_x$. The number of students solving the task is in brackets.

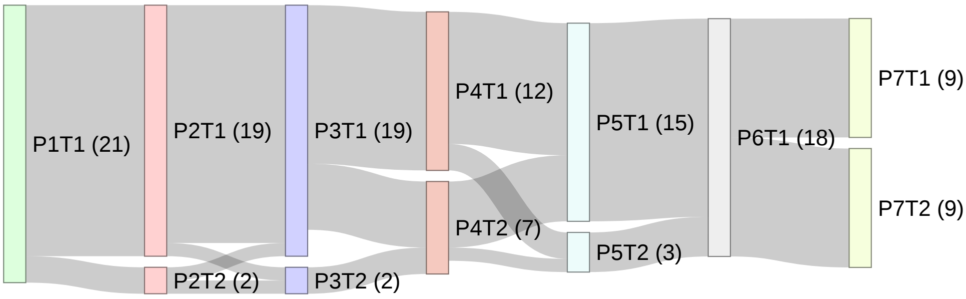

Figure 4.7: Transitions of 21 students between particular tasks in Knowledge Base training. Two students quit the training in $P_3$.

In a post-training questionnaire, students reported that tasks of both trainings were appropriately designed so that they have successfully completed the training in time.

The majority of participants of both trainings (70% in Junior Hacker, 68% in Knowledge Base) did not get stuck *Much* nor *Very much* during the training.The participants of both trainings enjoyed the learning experience. The majority of participants (51% in Junior Hacker, 63% in Knowledge Base) felt the trainings should be only *Slightly* or *Not at all* more difficult, which indicates the provided tasks are not overwhelming yet keep the participants appropriately motivated. Next, the participants engaged in both trainings and would like to continue if possible. Finally, the participants of both trainings would like to join another similar training. This was unequivocal for those who participated in Junior Hacker training. Opinions of participants of Knowledge Base training were mixed, though still mainly positive.

To conclude, we see KYPO with adaptive training features caters to students with various proficiency. Otherwise, these students would likely not have completed the training using other state-of-the-art cybersecurity training platforms.

Regarding the instructors' experience, we compared the effort required to run adaptive trainings manually and using the platform. We collected training events and typed commands during two training sessions with totally 86 students. Each participant performed 36 actions and typed 131 commands on average during one session lasting about two hours. In addition, they also filled in the pre-training assessment comprising eight questions. The total amount of data is so vast that it is infeasible to process manually, thus necessitating automation.

Before each training phase, the instructor would need to analyze captured shell commands (searching for keywords, counting the commands) and training actions of each participant (counting the number of wrong answers, searching whether a solution was taken). This analysis may take tens of seconds, perhaps a minute or more in training events with tens of participants or more. Finally, the instructor would need to combine all these results to compute the suitable task for each participant using the tutor model. While the instructor is extremely busy and overwhelmed at that time, the student is only waiting to be assigned the next task. In summary, the instructor can handle only a few students using this manual approach. Adaptive training of medium to large classes needs to be supported by a learning environment. Furthermore, automated task assignments by the environment enable instructors to focus on providing additional help to struggling students.

### Visual Analytics

To make the design, execution, and evaluation of training sessions more efficient, we apply principles of visual analytics. We proposed a conceptual model that supports instructors and students in various phases of the training life cycle. The model emerged from our long-term experience in designing and organizing diverse training sessions. It provides a classification of visualizations and can be used as a framework for developing novel visualization tools supporting phases of the training life-cycle.

We identified six key visualization tasks $V_1$–$V_6$ that are summarized below and detailed in our paper [95]. They differ in the roles involved in visual analysis, analytical goals, and other aspects.

**Insight of Trainees ($V_1$):** These visualizations support trainees in keeping track of what is happening at the moment and understanding the training content. The view on the data should be strictly person-centered and adapted to the history and performance of each particular trainee so that they can concentrate on the development during the training session from their perspective.

**Insight of Organizing Participants ($V_2$):** These visualizations support mainly supervisors, and operators in gaining insight into the state and progress of training sessions. Views are usually shared across all participants of the same role, providing them a view of the training progression, score, solved tasks, and other milestones and assessment data related to planning and timing. However, the views have to be adapted to each organizing role.

**Personal Feedback to Participants ($V_3$):** It has a significant positive impact on the learning process. A good post-training visual feedback should explain the pros and cons of the chosen approach and indicate the areas for further improvement. Effective person-centered feedback should occur as soon as possible, during or right after the *execution* phase when the trainees remember details of their behavior, decisions, and conducted actions. Deploying such immediate visual feedback requires automated data processing and automatically generated personalized views for individual trainees.

**Quality of Training Exercise ($V_4$):** It reflects the usefulness of training sessions for trainees. The main motivation is to improve future training programs by reviewing collected data by training designers. The quality can be measured and compared by various qualitative attributes that capture individual features of training sessions (such as *Correctness* or *Difficulty*).

**Behavior Analysis ($V_5$):** It can help in discovering relevant facts about trainees, their skills, or behavioral patterns under stress. The observations can either reveal issues or inconsistencies in training design or identify general patterns applicable in practical cyber defense. For instance, visualization of users' actions can reveal patterns of successful cooperation or successful attack/defense strategies.

**Infrastructure Analysis ($V_6$):** It represents another essential activity that can affect the results and impact of cybersecurity training. Any technical difficulties or malfunctions can negatively influence trainees. Related visualizations should support operators and designers in exploring *training definitions* and their requirements on the infrastructure and provide them with a "backstage" view of the operational data captured in the *execution* phase.

As opposed to the *infrastructure management* (a part of $V_2$), this category relates to the feasibility of the underlying infrastructure to serve according to the prescription of the *training definitions*. For example, if a heavily used server is allocated on a shared virtual node in the cyber range, its response time can be prohibitively slow. This can hinder trainees in fulfilling the tasks.

### Providing Feedback

Another approach to efficient learning is providing feedback based on interaction data captured during the hands-on training. We proposed and evaluated four different methods for modeling and visualizing student progress.

### Trainee and Milestone graphs

We model student progress by two types of graphs. We aim to quickly and accurately identify high or low-performing students in the class. In particular, we seek to identify successes and struggles of a specific student and across the class. Individual students can be then provided with their graphs to reflect on their approach used in training.

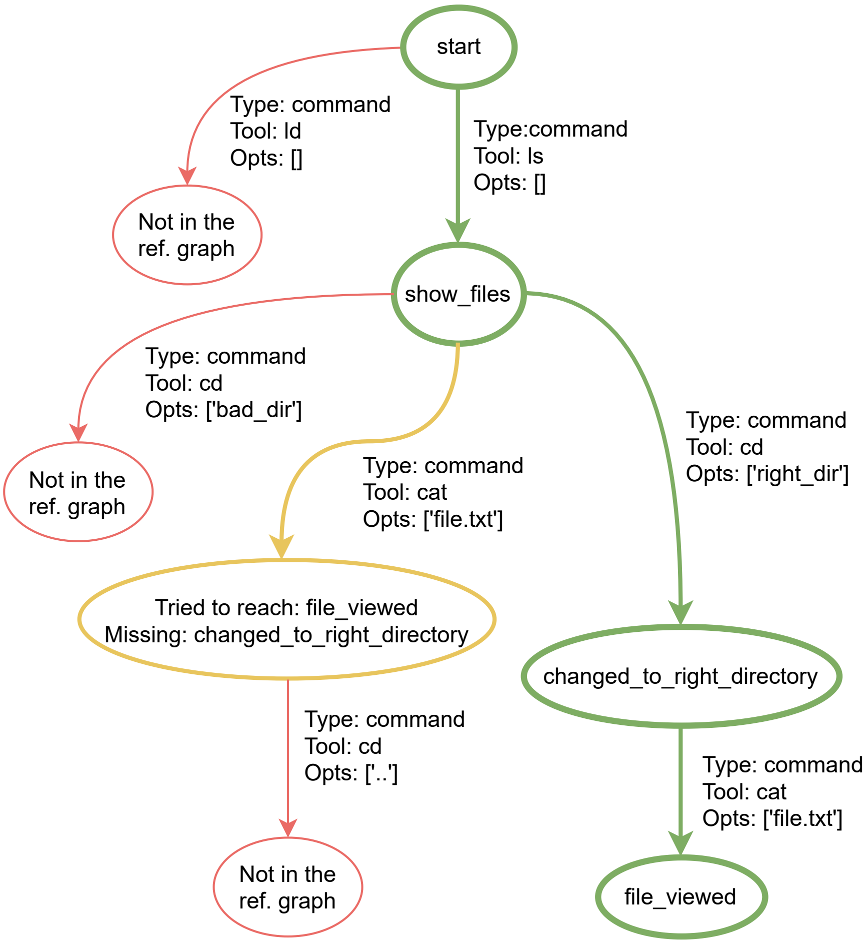

Figure 4.8: A simplified trainee graph.

The trainee graph is based on a *reference graph* (an example solution), created manually by the training designers before the training. The student actions are then automatically compared with the example solution and visualized, as shown in Figure 4.8. The green states and edges represent successful steps mapped to the reference graph. The red states and edges show actions that were likely erroneous or unnecessary. The yellow state and edge show an action with possibly missing prerequisites.

The milestone graph is based on *milestones*, discrete tasks, each identified by a set of unique expected commands. Each student command is compared against these milestones to find matches. In an example depicted in Figure 4.9, the student did not attempt the first of the three tasks, incorrectly attempted the second task but did not complete it, and completed the third task on their first try.

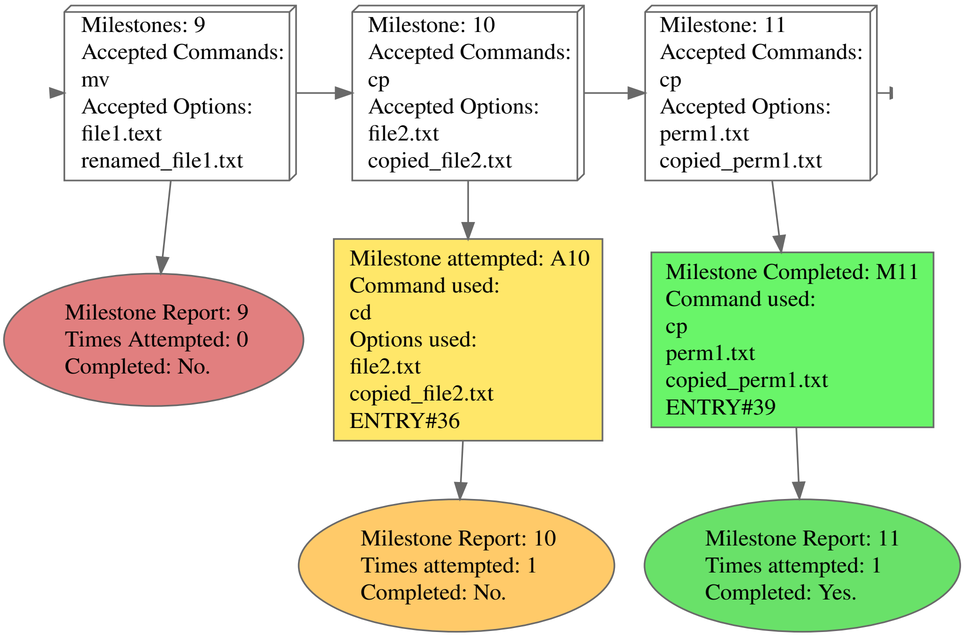

Figure 4.9: An example of the milestone graph.

To evaluate both graphs, we conducted a case study involving two different exercises, 20 students from the Czech Republic, 26 students from the USA, and 22 cybersecurity instructors from both countries. The study framework is depicted in Figure 4.10. The results show that most instructors interpreted each graph effectively and identified strengths, weaknesses, and use cases for each graph. While the trainee graphs carry in-depth information about the student, the milestone

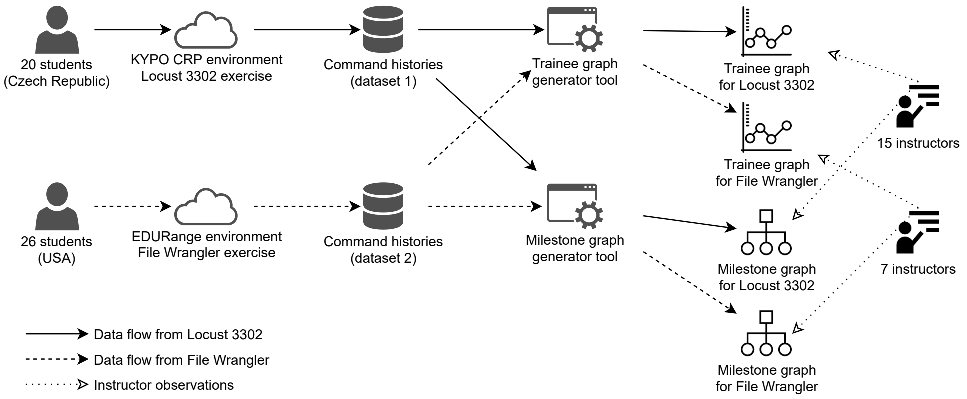

Figure 4.10: Overview of the design of the study.

graphs provide a quick overview of how the student did and how far they got. For more details and findings, see our paper [96].

### Pattern Mining and Clustering

Pattern mining techniques, such as association rule mining and sequential pattern mining, can reveal interesting relationships in datasets [97]. Clustering, on the other hand, forms groups of data based on their similar characteristics [98]. Evaluating these two techniques in the context of cybersecurity training data represents an original contribution to cybersecurity education and beyond [99].

Our research was framed by two research questions: *What insights can we gather from command histories using pattern mining* (RQ1) *and clustering* (RQ2)*?* By *insights*, we mean the following findings:

- trainees' approaches and strategies,
- common mistakes, misconceptions, and problematic tools,
- distinct types of trainees based on their actions, and
- issues in the training design and execution.

To answer these questions, we collected command logs from students and provided them as input for pattern mining and clustering methods, as shown in Figure 4.11. We mined 8,834 commands from several-hours-long training sessions with small groups of computing students. We found out that:

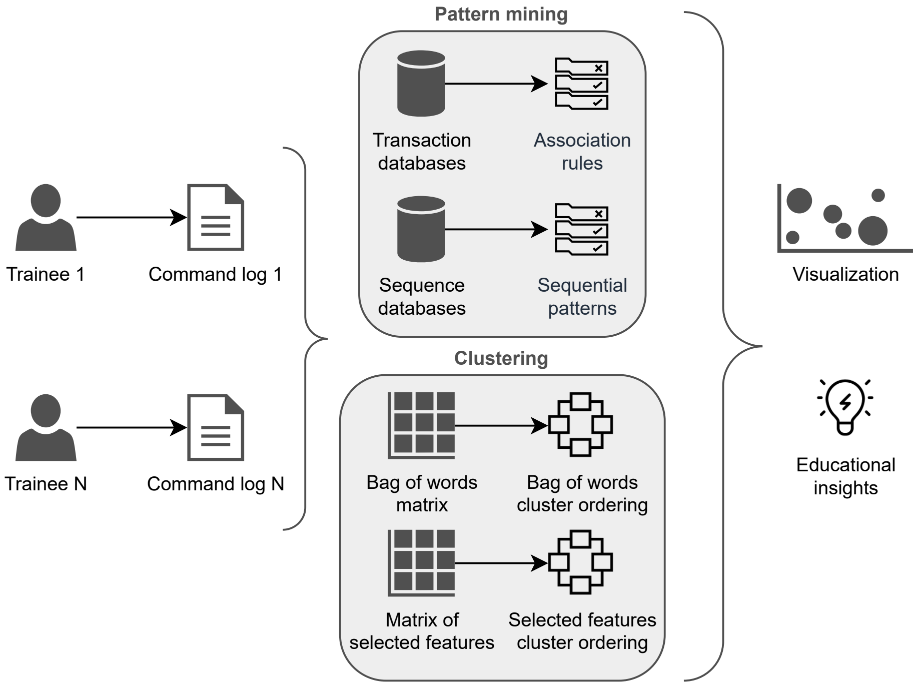

Figure 4.11: The command logs collected from students act as input for pattern mining and clustering.

1. pattern mining is suitable for revealing solution approaches of students, their misconceptions, and difficult training tasks,
2. clustering highlights similarities and differences between approaches of students, grouping them based on their behavioral patterns.

Other educators can use these findings to improve cybersecurity training in their context or adapt them to training in other domains. Pattern mining and clustering are suitable for any problem-solving assignments that yield interaction data.

## 4.4 My Contributions

I defined the life cycle of a cyber defense exercise, summarized lessons learned from carrying out these exercises, and published a dataset from one of the exercises.

Regarding serious games, I applied the format of Capture the Flag (CTF) games in formal education in three different contexts. I shared experience from using the learning by teaching approach and CTF as a homework assignment. Next, I proposed methods for cheating prevention and detection when the game is a part of the course assessment.

My biggest contribution to efficient cybersecurity learning is in researching and developing adaptive learning of cybersecurity skills. The implementation of the proposed training format and tutor model has already been integrated into open-source KYPO CRP. I also defined visualization tasks useful for both trainees and training designers. Finally, I was involved in evaluating methods for providing feedback based on commands typed by trainees.

Three papers were written with international collaborators from Singapore and USA and reported research that involved students from their institutions.

### Key Results

I co-authored 12 regular conference papers, three journal articles, and one data article related to instructional methods. Two conference papers received Best Paper Award in the Experience Reports and Tools track of the ACM SIGCSE Technical Symposium Conference (CORE A). I was also involved in designing, developing, or evaluating several software artifacts provided as supplementary materials to the published papers. The list of the results is ordered by the publication date.

1. **J. Vykopal**, M. Vizvary, R. Oslejsek, P. Celeda, and D. Tovarnak. "Lessons Learned From Complex Hands-on Defence Exercises in a Cyber Range". In: *2017 IEEE Frontiers in Education Conference (FIE)*. 2017, pp. 1–8. isbn: 978-1-5090-5920-1. doi: 10.1109/FIE.2017.8190713

   - Main track (Innovative Practice)
   - CORE conference rank: **B**
   - Contribution: **40%**
   - CRediT: Conceptualization, Methodology, Software, Validation, Investigation, Resources, Writing – Original Draft, Visualization

2. V. Švábenský and **J. Vykopal**. “Challenges Arising from Prerequisite Testing in Cybersecurity Games”. In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE ’18. Baltimore, MD, USA: Association for Computing Machinery, February 2018, pp. 56–61. ISBN: 978-1-4503-5103-4. DOI: 10.1145/3159450.3159454

   - Main track (Computing Education Research)
   - CORE conference rank: **A**
   - Contribution: **40%**
   - CRediT: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Resources, Data Curation, Writing – Original Draft, Visualization, Supervision

3. **J. Vykopal**, R. Ošlejšek, K. Burská, and K. Zákopčanová. “Timely Feedback in Unstructured Cybersecurity Exercises”. In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE ’18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, pp. 173–178. ISBN: 978-1-4503-5103-4. DOI: 10.1145/3159450.3159561

   - Main track (Experience Reports and Tools)
   - CORE conference rank: **A**
   - Contribution: **40%**
   - CRediT: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Resources, Writing – Original Draft, Supervision, Project administration

4. V. Švábenský, **J. Vykopal**, M. Cermak, and M. Laštovička. “Enhancing Cybersecurity Skills by Creating Serious Games”. In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE ’18. Larnaca, Cyprus: Association for Computing Machinery, July 2018, pp. 194–199. ISBN: 978-1-4503-5707-4. DOI: 10.1145/3197091.3197123

   - Main track
   - CORE conference rank: **A**
   - Contribution: **35%**
   - CRediT: Conceptualization, Methodology, Software, Validation, Investigation, Resources, Writing – Original Draft, Supervision, Project administration

5. R. Ošlejšek, **J. Vykopal**, K. Burská, and V. Rusňák. "Evaluation of Cyber Defense Exercises Using Visual Analytics Process". In: *2018 IEEE Frontiers in Education Conference (FIE)*. 2018, pp. 1–9. ISBN: 978-1-5386-1174-6. DOI: 10.1109/FIE.2018.8659299
   - Main track (Innovative Practice)
   - CORE conference rank: **B**
   - Contribution: **20%**
   - CRediT: Methodology, Software, Validation, Investigation, Resources, Writing – Original Draft

6. R. Ošlejšek, V. Rusňák, K. Burská, V. Švábenský, and **J. Vykopal**. "Visual Feedback for Players of Multi-Level Capture the Flag Games: Field Usability Study". In: *Proceedings of the 16th IEEE Symposium on Visualization for Cyber Security*. VizSec '19. Vancouver, Canada: IEEE, October 2019, pp. 1–11. ISBN: 978-1-7281-3877-0. DOI: 10.1109/VizSec48167.2019.9161386
   - Main track
   - CORE conference rank: **C**
   - Contribution: **10%**
   - CRediT: Resources, Writing – Original Draft, Writing – Review & Editing

7. R. Ošlejšek, V. Rusňák, K. Burská, V. Švábenský, **J. Vykopal**, and J. Čegan. "Conceptual Model of Visual Analytics for Hands-on Cybersecurity Training". In: *IEEE Transactions on Visualization and Computer Graphics* 27.8 (February 2020). ISSN: 1077-2626. DOI: 10.1109/TVCG.2020.2977336
   - IF: **4.579** (in the year 2020)
   - Rank: **Q1** (D1 based on the Journal Citation Indicator)
   - Contribution: **10%**
   - CRediT: Investigation, Resources, Writing – Original Draft, Writing – Review & Editing

8. D. Tovarňák, S. Špaček, and **J. Vykopal**. "Traffic and log data captured during a cyber defense exercise". In: *Data in Brief* 31 (2020), p. 105784. ISSN: 2352-3409. DOI: 10.1016/j.dib.2020.105784
   - IF: **N/A** (in the year 2020)
   - Rank: **Q2** (based only on the Journal Citation Indicator)

- Contribution: **15%**
- CRediT: Validation, Investigation, Resources, Data curation, Writing – Original Draft, Writing – review & editing

9. **J. Vykopal**, V. Švábenský, and E.-C. Chang. "Benefits and Pitfalls of Using Capture the Flag Games in University Courses". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, March 2020, pp. 752–758. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366893
    - Main track (Experience Reports and Tools)
    - CORE conference rank: **A**
    - Contribution: **75%**
    - CRediT: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Data curation, Writing – Original Draft, Writing – Review & Editing, Visualization, Project administration

10. P. Seda, **J. Vykopal**, V. Švábenský, and P. Čeleda. "Reinforcing Cybersecurity Hands-on Training With Adaptive Learning". In: *2021 IEEE Frontiers in Education Conference (FIE)*. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637252
    - Main track (Research to Practice)
    - CORE conference rank: **C** (**B at the time of paper submission**)
    - Contribution: **33%**
    - CRediT: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Resources, Data Curation, Writing – Original Draft, Visualization, Supervision, Project administration

11. **J. Vykopal**, V. Švábenský, P. Seda, and P. Čeleda. "Preventing Cheating in Hands-on Lab Assignments". In: *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*. SIGCSE '22. Providence, RI, USA: Association for Computing Machinery, March 2022. ISBN: 978-1-4503-9070-5. DOI: 10.1145/3478431.3499420
    - Main track (Experience Reports and Tools)
    - CORE conference rank: **A**
    - Contribution: **60%**

- CRediT: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Resources, Data Curation, Writing – Original Draft, Visualization, Supervision, Project administration
- **Best Paper Award** in the Experience Reports and Tools track

12. V. Švábenský, R. Weiss, J. Cook, **J. Vykopal**, P. Čeleda, J. Mache, R. Chudovský, and A. Chattopadhyay. "Evaluating Two Approaches to Assessing Student Progress in Cybersecurity Exercises". In: *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*. SIGCSE '22. Providence, RI, USA: Association for Computing Machinery, March 2022. ISBN: 978-1-4503-9070-5. DOI: 10.1145/3478431.3499414

    - Main track (Experience Reports and Tools)
    - CORE conference rank: **A**
    - Contribution: **10%**
    - CRediT: Investigation, Writing – Review & Editing

13. V. Švábenský, **J. Vykopal**, P. Čeleda, K. Tkáčik, and D. Popovič. "Student Assessment in Cybersecurity Training Automated by Pattern Mining and Clustering". In: *Education and Information Technologies* (March 2022). ISSN: 1573-7608. DOI: 10.1007/s10639-022-10954-4

    - IF: **3.666** (in the year 2021)
    - Rank: **Q1** (D1 based on the Journal Citation Indicator)
    - Contribution: **15%**
    - CRediT: Investigation, Resources, Writing – Review & Editing.

14. P. Seda, **J. Vykopal**, P. Čeleda, and I. Ignác. "Designing Adaptive Cybersecurity Hands-on Training". In: *2022 IEEE Frontiers in Education Conference (FIE)*. Uppsala, Sweden: IEEE, October 2022, pp. 1–9

    - Main track (Research to Practice)
    - CORE conference rank: **C**
    - Contribution: **10%**
    - CRediT: Investigation, Resources, Writing – Review & Editing, Supervision.

15. Sufatrio, **J. Vykopal**, and E.-C. Chang. "Collaborative Paradigm of Teaching Penetration Testing Using Real-World University

Applications". In: *Australasian Computing Education Conference*. ACE '22. Virtual Event, Australia: Association for Computing Machinery, 2022, pp. 114–122. ISBN: 978-1-4503-9643-1. DOI: 10.1145/3511861.3511874

- Main track
- CORE conference rank: **Australasian B**
- Contribution: **25%**
- CRediT: Conceptualization, Investigation, Writing – Original Draft

16. **J. Vykopal**, P. Seda, V. Švábenský, and P. Čeleda. "Smart Environment for Adaptive Learning of Cybersecurity Skills". In: *IEEE Transactions on Learning Technologies* (2022). In press. ISSN: 1939-1382. DOI: 10.1109/TLT.2022.3216345

    - IF: **4.433** (in the year 2021)
    - Rank: **Q2** (Q1 based on the Journal Citation Indicator)
    - Contribution: **40%**
    - CRediT: CRediT: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Resources, Data Curation, Writing – Original Draft, Writing – Review & Editing, Visualization, Supervision, Project administration

# 5 Conclusions

Efficient training environments and instructional methods are essential for proficiently teaching cybersecurity hands-on. However, creating efficient environments and methods represents a substantial research challenge since cybersecurity is a complex domain encompassing diverse technical knowledge and skills.

In our work, we created interactive training environments that are successfully used in teaching practice (RQ1). We focused on the flexibility and reusability of the environments in different use cases and deployment contexts.

We also researched various formats and methods for teaching cybersecurity skills (RQ2). First, we eased the laborious preparation and evaluation of complex defense exercises. Then we applied the concept of serious games to cybersecurity hands-on education. Our work enables engaging students in complex topics using gamification elements.

Finally, we proposed methods for providing feedback and scaffolding to students during the training (RQ3). In particular, we research adaptive training, which keeps students motivated to complete the training. Consequently, more students can learn cybersecurity skills and thus fill the global cybersecurity workforce gap.

Our results include open-source software, exemplary trainings, and data we produced to evaluate our research. The most notable example is the KYPO Cyber Range Platform, which we have developed and enhanced by methods we researched. As a result, our research is available to others not only in academic publications but also as open-source software.

### Future Work

The tools and methods we have researched and developed are useful educational resources. They can be used in the teaching practice if complemented with instructional content. However, preparing the content is a complex, manual, and time-consuming task. We see the automation of this task as one direction for future work. The preparation could be sped up by using resources that are already publicly available. For instance, creating an intentionally vulnerable environment featuring recent software vulnerabilities can benefit from vulnerability databases (such as NVD), source code repositories with target applications and proof-of-concept implementations of vulnerability exploits (such as those hosted at Github), or categorizations of attacking techniques (such as MITRE ATT&CK). Integrating all these resources together will save instructors time and effort.

Another direction is making the training environment as much realistic as possible. So far, we have focused on attackers and defenders, but cyberspace also comprises legitimate users and other actors influencing the interaction between attackers and defenders (such as network operators). Our ongoing work focuses on the automated emulation of behavior and interaction of various cyberspace users. We are excited to answer two research questions: *i*) How can a behavior profile of an emulated user be described to reflect human information needs and behavior? and *ii*) How to achieve a user behavior profile within emulation platforms feasibly and simply?

Next, since cybersecurity involves not only technology but also people, information, and processes, cybersecurity training should cover communication and decision-making skills, for instance, in the incident response process. I believe that tools and principles used for training technical skills can be adopted in this context.

Last but not least, any cybersecurity training should be cost-effective and relevant to the practice. That means it responsibly uses allocated computational resources by leveraging available technologies, and the instructional content involves state-of-the-art technologies. Educational researchers should supply practitioners with learning tools and resources featuring modern real-world environments. Unfortunately, we can still see trainings on obsolete topics requiring resource-intensive and costly infrastructure, which is not utilized efficiently.

# Bibliography

[1] Joint Task Force on Cybersecurity Education. *Cybersecurity Curricular Guideline*. Online, accessed February 1, 2022. 2017. URL: http://cybered.acm.org (cit. on pp. 2, 10, 12, 13).

[2] J. Oltsik and B. Lundell. *The Life and Times of Cybersecurity Professionals 2021*. Tech. rep. 2021. URL: https://www.esg-global.com/hubfs/ESG-ISSA-Research-Report-Life-of-Cybersecurity-Professionals-Jul-2021.pdf (cit. on p. 2).

[3] ISACA. *State of Cybersecurity 2020: Part 1: Global Update on Workforce Efforts and Resources*. Tech. rep. 2020. URL: https://www.isaca.org/go/state-of-cybersecurity-2020 (cit. on p. 2).

[4] (ISC)$^2$. *Cybersecurity Workforce Study*. Tech. rep. 2021. URL: https://www.isc2.org/Research/Workforce-Study (cit. on p. 2).

[5] CC2020 Task Force. *Computing Curricula 2020: Paradigms for Global Computing Education*. New York, NY, USA: Association for Computing Machinery, 2020. ISBN: 978-1-4503-9059-0. DOI: 10.1145/3467967 (cit. on p. 2).

[6] N. Dragoni, A. Lluch Lafuente, F. Massacci, and A. Schlichtkrull. "Are We Preparing Students to Build Security In? A Survey of European Cybersecurity in Higher Education Programs [Education]". In: *IEEE Security Privacy* 19.1 (2021), pp. 81–88. DOI: 10.1109/MSEC.2020.3037446 (cit. on p. 2).

[7] ITU Global Cybersecurity Index project. *ITU Member States with National Cybersecurity Strategy*. 2022. URL: https://www.itu.int/en/ITU-D/Cybersecurity/Pages/National-Strategies-repository.aspx (cit. on p. 2).

[8] NATO Cooperative Cyber Defence Centre of Excellence. *Strategy and Governance*. 2022. URL: https://ccdcoe.org/library/strategy-and-governance/ (cit. on p. 2).

[9] ENISA. *National Cyber Security Strategies – Interactive Map*. 2022. URL: https://www.enisa.europa.eu/topics/national-cyber-security-strategies/ncss-map/national-cyber-security-strategies-interactive-map (cit. on p. 2).

[10] National Institute of Standards and Technology (NIST). *National Initiative for Cybersecurity Education (NICE)*. Online, accessed February 15, 2022. 2021. URL: https://www.nist.

gov/itl/applied-cybersecurity/nice/nice-framework-resource-center (cit. on p. 2).

[11] Awais Rashid, Howard Chivers, Emil Lupu, Andrew Martin, Steve Schneider. *CyBOK – The Cyber Security Body of Knowledge, version 1.1.0*. Tech. rep. 2021. URL: https://www.cybok.org/media/downloads/CyBOK_v1.1.0.pdf (cit. on p. 2).

[12] Wenliang Du. *Hands-on Labs for Security Education*. Online, accessed March 15, 2022. URL: https://seedsecuritylabs.org/ (cit. on p. 5).

[13] W. Du. "SEED: Hands-On Lab Exercises for Computer Security Education". In: *IEEE Security & Privacy* 9.5 (2011), pp. 70–73. ISSN: 1558-4046. DOI: 10.1109/MSP.2011.139 (cit. on pp. 5, 15).

[14] Naval Postgraduate School, Center for Cybersecurity and Cyber Operations. *Labtainers*. Online, accessed March 15, 2022. URL: https://nps.edu/web/c3o/labtainers (cit. on p. 5).

[15] M. F. Thompson and C. E. Irvine. "Individualizing Cybersecurity Lab Exercises with Labtainers". In: *IEEE Security & Privacy* 16.2 (2018), pp. 91–95. DOI: 10.1109/MSP.2018.1870862 (cit. on p. 5).

[16] VulnHub. *Vulnerable By Design*. 2022. URL: https://www.vulnhub.com (cit. on p. 5).

[17] SANS Institute. *Cyber Security Skills Roadmap*. Online, accessed February 28, 2022. URL: https://www.sans.org/cyber-security-skills-roadmap/ (cit. on p. 6).

[18] Cybrary. *Cybrary*. Online, accessed March 22, 2022. URL: https://www.cybrary.it (cit. on p. 6).

[19] V. Švábenský, P. Čeleda, **J. Vykopal**, and S. Brišáková. "Cybersecurity Knowledge and Skills Taught in Capture the Flag Challenges". In: *Elsevier Computers & Security* 102.102154 (March 2021). ISSN: 0167-4048. DOI: 10.1016/j.cose.2020.102154 (cit. on pp. 6, 12, 14, 80).

[20] DEF CON. *CTF Archive*. Online, accessed March 15, 2022. 2022. URL: https://www.defcon.org/html/links/dc-ctf.html (cit. on p. 6).

[21] G. Vigna, K. Borgolte, J. Corbetta, A. Doupé, Y. Fratantonio, L. Invernizzi, D. Kirat, and Y. Shoshitaishvili. "Ten Years of iCTF: The Good, The Bad, and The Ugly". In: *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education*

(*3GSE 14*). San Diego, CA: USENIX Association, 2014. URL: https://www.usenix.org/conference/3gse14/summit-program/presentation/vigna (cit. on p. 6).

[22] Shellphish. *iCTF: the International Capture The Flag Competition*. Online, accessed March 15, 2022. 2022. URL: https://shellphish.net/ictf/ (cit. on p. 6).

[23] L. A. Annetta. "The "I's" Have It: A Framework for Serious Educational Game Design". In: *Review of General Psychology* 14.2 (2010), pp. 105–113. DOI: 10.1037/a0018985 (cit. on p. 6).

[24] P. Chapman, J. Burket, and D. Brumley. "PicoCTF: A Game-Based Computer Security Competition for High School Students". In: *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*. San Diego, CA: USENIX Association, 2014. URL: https://www.usenix.org/conference/3gse14/summit-program/presentation/chapman (cit. on p. 6).

[25] New York University, Center for Cybersecurity. *CSAW CTF*. Online, accessed March 15, 2022. 2022. URL: https://www.csaw.io/ctf (cit. on p. 6).

[26] Hack The Box. *Hack The Box*. Hack The Box. 2021. URL: https://www.hackthebox.com/ (cit. on p. 6).

[27] TryHackMe. *TryHackMe*. TryHackMe. 2021. URL: https://www.tryhackme.com/ (cit. on p. 6).

[28] NATO Cooperative Cyber Defence Centre of Excellence. *Locked Shields*. URL: https://ccdcoe.org/exercises/locked-shields (cit. on p. 7).

[29] M. Granåsen, G. Huskaj, and S. Varga. *Data Collection and Research in CDXs-Command and Control, Cyber Situational Awareness and Intelligence Perspectives on Cyber Defense*. Tech. rep. 2019 (cit. on p. 7).

[30] U.S. Cyber Command. *Media Advisory: Cyber Flag 21-2 winner announcement*. June 2021. URL: https://www.cybercom.mil/Media/News/Article/2671401/media-advisory-cyber-flag-21-2-winner-announcement/ (cit. on p. 7).

[31] U. S. National Guard. *National Guard units across the nation complete Cyber Shield*. July 2021. URL: https://www.nationalguard.mil/News/Article/2706870/national-guard-units-across-the-nation-complete-cyber-shield/ (cit. on p. 7).

[32] D. S. Henshel, G. M. Deckard, B. Lufkin, N. Buchler, B. Hoffman, P. Rajivan, and S. Collman. "Predicting proficiency in cyber defense team exercises". In: *MILCOM 2016 - 2016 IEEE Military Communications Conference*. 2016, pp. 776–781. DOI: 10.1109/MILCOM.2016.7795423 (cit. on p. 7).

[33] National Collegiate Cyber Defense Competition. *National Collegiate Cyber Defense Competition Website*. URL: https://www.nationalccdc.org/ (cit. on p. 7).

[34] NATO Cooperative Cyber Defence Centre of Excellence. *Crossed Swords*. URL: https://ccdcoe.org/exercises/crossed-swords (cit. on p. 7).

[35] RIT Global Cybersecurity Institute. *Collegiate Penetration Testing Competition*. URL: https://cp.tc (cit. on p. 8).

[36] V. Ford, A. Siraj, A. Haynes, and E. Brown. "Capture the Flag Unplugged: An Offline Cyber Competition". In: *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. SIGCSE '17. Seattle, USA: Association for Computing Machinery, 2017, pp. 225–230. ISBN: 9781450346986. DOI: 10.1145/3017680.3017783 (cit. on p. 8).

[37] G. Costa, M. Lualdi, M. Ribaudo, and A. Valenza. "A NERD DOGMA: Introducing CTF to Non-Expert Audience". In: *Proceedings of the 21st Annual Conference on Information Technology Education*. SIGITE '20. Virtual Event, USA: Association for Computing Machinery, 2020, pp. 413–418. ISBN: 9781450370455. DOI: 10.1145/3368308.3415405 (cit. on p. 8).

[38] M. Gondree and Z. N. Peterson. "Valuing Security by Getting [d0x3d!]: Experiences with a Network Security Board Game". In: *6th Workshop on Cyber Security Experimentation and Test (CSET 13)*. Washington, D.C.: USENIX Association, August 2013. URL: https://www.usenix.org/conference/cset13/workshop-program/presentation/gondree (cit. on p. 8).

[39] T. Denning, A. Lerner, A. Shostack, and T. Kohno. "Control-Alt-Hack: The Design and Evaluation of a Card Game for Computer Security Awareness and Education". In: *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*. CCS '13. Berlin, Germany: Association for Comput-

ing Machinery, 2013, pp. 915–928. ISBN: 9781450324779. DOI: 10.1145/2508859.2516753 (cit. on p. 8).

[40] T. Denning, A. Shostack, and T. Kohno. "Practical Lessons from Creating the Control-Alt-Hack Card Game and Research Challenges for Games In Education and Research". In: *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education* (*3GSE 14*). San Diego, CA: USENIX Association, August 2014. URL: https://www.usenix.org/conference/3gse14/summit-program/presentation/denning (cit. on p. 8).

[41] Kreativní Laboratoř. *Deploy or Die*. Retrieved April 12, 2022 from https://www.deployordie.com (cit. on p. 8).

[42] R. S. Dewar and A. Wenger. *Cybersecurity and Cyberdefense Exercises*. 2018. URL: https://css.ethz.ch/content/dam/ethz/special-interest/gess/cis/center-for-securities-studies/pdfs/Cyber-Reports-2018-10-Cyber_Exercises.pdf (cit. on p. 8).

[43] G. N. Angafor, I. Yevseyeva, and Y. He. "Game-based learning: A review of tabletop exercises for cybersecurity incident response training". In: *Security and Privacy* 3 (6 2020). ISSN: 2475-6725. DOI: 10.1002/spy2.126 (cit. on p. 8).

[44] M. Bartnes and N. B. Moe. "Challenges in IT security preparedness exercises: A case study". In: *Computers & Security* 67 (2017), pp. 280–290. ISSN: 0167-4048. DOI: 10.1016/j.cose.2016.11.017 (cit. on p. 8).

[45] Cybersecurity and Infrastructure Security Agency. *CISA Tabletop Exercise Packages*. URL: https://www.cisa.gov/cisa-tabletop-exercises-packages (cit. on p. 9).

[46] G. Anderson and N. Arsenault. *Fundamentals of educational research*. Routledge, 2005 (cit. on p. 9).

[47] L. Malmi, J. Sheard, P. Kinnunen, Simon, and J. Sinclair. "Computing Education Theories: What Are They and How Are They Used?" In: *Proceedings of the 2019 ACM Conference on International Computing Education Research*. ICER '19. Toronto ON, Canada: Association for Computing Machinery, 2019, pp. 187–197. ISBN: 9781450361859. DOI: 10.1145/3291279.3339409 (cit. on p. 9).

[48] M. Guzdial and B. du Boulay. "History of Computing Education Research". In: *The Cambridge Handbook of Computing Education Research*. Ed. by S. A. Fincher and A. V. Robins. Cambridge, United Kingdom: Cambridge University Press, 2019. Chap. 1, pp. 11–39. ISBN: 978-1-108-72189-9. DOI: 10.1017/9781108654555.011 (cit. on p. 9).

[49] Z. Papamitsiou, M. Giannakos, Simon, and A. Luxton-Reilly. "Computing Education Research Landscape through an Analysis of Keywords". In: *Proceedings of the 2020 ACM Conference on International Computing Education Research*. ICER '20. Virtual Event, New Zealand: Association for Computing Machinery, 2020, pp. 102–112. ISBN: 9781450370929. DOI: 10.1145/3372782.3406276 (cit. on p. 9).

[50] V. Švábenský, **J. Vykopal**, and P. Čeleda. "What Are Cybersecurity Education Papers About? A Systematic Literature Review of SIGCSE and ITiCSE Conferences". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, March 2020, pp. 2–8. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366816 (cit. on pp. 10, 13, 80).

[51] V. Švábenský, **J. Vykopal**, P. Čeleda, and L. Kraus. "Applications of Educational Data Mining and Learning Analytics on Data From Cybersecurity Training". In: *Springer Education and Information Technologies* 1360.2357 (2022). ISSN: 1573-7608. DOI: 10.1007/s10639-022-11093-6 (cit. on pp. 11, 14, 80).

[52] L. Allen, A. O'Connell, and V. Kiermer. "How can we ensure visibility and diversity in research contributions? How the Contributor Role Taxonomy (CRediT) is helping the shift from authorship to contributorship". In: *Learned Publishing* 32.1 (2019), pp. 71–74. DOI: 10.1002/leap.1210 (cit. on p. 14).

[53] L. Topham, K. Kifayat, Y. Younis, Q. Shi, and B. Askwith. "Cyber Security Teaching and Learning Laboratories: A Survey". In: *Information & Security: An International Journal* 35 (2016). DOI: 10.11610/isij.3503 (cit. on p. 15).

[54] S. Karagiannis, E. Magkos, C. Ntantogian, and L. L. Ribeiro. *Sandboxing the Cyberspace for Cybersecurity Education and Learning*. 2020. DOI: 10.1007/978-3-030-66504-3_11 (cit. on p. 15).

[55] M. M. Yamin, B. Katt, and V. Gkioulos. *Cyber ranges and security testbeds: Scenarios, functions, tools and architecture*. January 2020. DOI: 10.1016/j.cose.2019.101636 (cit. on p. 15).

[56] **J. Vykopal**, P. Čeleda, P. Seda, V. Švábenský, and D. Tovarňák. "Scalable Learning Environments for Teaching Cybersecurity Hands-on". In: *Proceedings of the 51st IEEE Frontiers in Education Conference*. FIE '21. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637180 (cit. on pp. 15, 28, 81).

[57] P. Čeleda, **J. Vykopal**, V. Švábenský, and K. Slavíček. "KYPO4-INDUSTRY: A Testbed for Teaching Cybersecurity of Industrial Control Systems". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, March 2020, pp. 1026–1032. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366908 (cit. on pp. 15, 27, 80).

[58] K. Sanders, J. Boustedt, A. Eckerdal, R. McCartney, and C. Zander. "Folk Pedagogy: Nobody Doesn't Like Active Learning". In: *Proceedings of the 2017 ACM Conference on International Computing Education Research*. ICER '17. Tacoma, Washington, USA: Association for Computing Machinery, 2017, pp. 145–154. ISBN: 9781450349680. DOI: 10.1145/3105726.3106192 (cit. on p. 17).

[59] Masaryk University. *KYPO Cyber Range Platform*. 2021. URL: https://crp.kypo.muni.cz (cit. on p. 18).

[60] Masaryk University. *Cyber Sandbox Creator*. 2021. URL: https://gitlab.ics.muni.cz/muni-kypo-csc/cyber-sandbox-creator (cit. on p. 18).

[61] Masaryk University. *Junior hacker training*. 2021. URL: https://gitlab.ics.muni.cz/muni-kypo-trainings/games/junior-hacker (cit. on p. 21).

[62] V. Švábenský, **J. Vykopal**, M. Cermak, and M. Laštovička. "Enhancing Cybersecurity Skills by Creating Serious Games". In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE 2018. Larnaca, Cyprus: ACM, 2018, pp. 194–199. ISBN: 978-1-4503-5707-4. DOI: 10.1145/3197091.3197123 (cit. on p. 22).

[63] V. Švábenský, **J. Vykopal**, D. Tovarňák, and P. Čeleda. "Toolset for Collecting Shell Commands and Its Application in Hands-

on Cybersecurity Training". In: *Proceedings of the 51st IEEE Frontiers in Education Conference*. FIE '21. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637052 (cit. on pp. 23, 28).

[64] National Institute of Standards and Technology (NIST). *Glossary*. 2022. URL: https://csrc.nist.gov/glossary/term/industrial_control_system (cit. on p. 23).

[65] B. Zhu, A. Joseph, and S. Sastry. "A Taxonomy of Cyber Attacks on SCADA Systems". In: *Proceedings of the 2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*. ITHINGSCPSCOM '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 380–388. ISBN: 978-0-7695-4580-6. DOI: 10.1109/iThings/CPSCom.2011.34 (cit. on p. 23).

[66] A. Humayed, J. Lin, F. Li, and B. Luo. "Cyber-Physical Systems Security—A Survey". In: *IEEE Internet of Things Journal* 4.6 (2017), pp. 1802–1831. DOI: 10.1109/JIOT.2017.2703172 (cit. on p. 23).

[67] J. Butts and M. Glover. "How Industrial Control System Security Training is Falling Short". In: *Critical Infrastructure Protection IX*. Ed. by M. Rice and S. Shenoi. Cham: Springer International Publishing, 2015, pp. 135–149. ISBN: 978-3-319-26567-4. DOI: 10.1007/978-3-319-26567-4_9 (cit. on p. 23).

[68] H. Holm, M. Karresand, A. Vidström, and E. Westring. "A Survey of Industrial Control System Testbeds". In: *Secure IT Systems*. Ed. by S. Buchegger and M. Dam. Cham: Springer International Publishing, 2015, pp. 11–26. ISBN: 978-3-319-26502-5. DOI: 10.1007/978-3-319-26502-5_2 (cit. on p. 23).

[69] D. Inc. *Docker: Enterprise Container Platform*. 2019. URL: https://www.docker.com (cit. on p. 25).

[70] O. Foundation. *OpenStack: Open-source software platform for cloud computing*. 2019. URL: https://www.openstack.org (cit. on p. 25).

[71] J. L. Bishop, M. A. Verleger, et al. "The flipped classroom: A survey of the research". In: *ASEE national conference proceedings, Atlanta, GA*. 2013, pp. 1–18 (cit. on p. 26).

[72] **J. Vykopal**, R. Oslejsek, P. Celeda, M. Vizvary, and D. Tovarnak. "KYPO Cyber Range: Design and Use Cases". In: *Proceedings of the 12th International Conference on Software Technologies – Volume 1: ICSOFT*. INSTICC. SciTePress, 2017, pp. 310–321. ISBN: 978-989-758-262-2. DOI: 10.5220/0006428203100321 (cit. on pp. 27, 80).

[73] V. Švábenský, **J. Vykopal**, P. Seda, and P. Čeleda. "Dataset of Shell Commands Used by Participants of Hands-on Cybersecurity Training". In: *Elsevier Data in Brief* 38 (September 2021). ISSN: 2352-3409. DOI: 10.1016/j.dib.2021.107398 (cit. on p. 28).

[74] **J. Vykopal**, M. Vizvary, R. Oslejsek, P. Celeda, and D. Tovarnak. "Lessons Learned From Complex Hands-on Defence Exercises in a Cyber Range". In: *2017 IEEE Frontiers in Education Conference (FIE)*. 2017, pp. 1–8. ISBN: 978-1-5090-5920-1. DOI: 10.1109/FIE.2017.8190713 (cit. on pp. 30, 58, 81).

[75] **J. Vykopal**, R. Ošlejšek, K. Burská, and K. Zákopčanová. "Timely Feedback in Unstructured Cybersecurity Exercises". In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE '18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, pp. 173–178. ISBN: 978-1-4503-5103-4. DOI: 10.1145/3159450.3159561 (cit. on pp. 34, 59, 81).

[76] D. Tovarňák, S. Špaček, and **J. Vykopal**. "Traffic and log data captured during a cyber defense exercise". In: *Data in Brief* 31 (2020), p. 105784. ISSN: 2352-3409. DOI: 10.1016/j.dib.2020.105784 (cit. on pp. 35, 60).

[77] V. Švábenský, **J. Vykopal**, M. Cermak, and M. Laštovička. "Enhancing Cybersecurity Skills by Creating Serious Games". In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '18. Larnaca, Cyprus: Association for Computing Machinery, July 2018, pp. 194–199. ISBN: 978-1-4503-5707-4. DOI: 10.1145/3197091.3197123 (cit. on pp. 35, 59, 81).

[78] **J. Vykopal**, V. Švábenský, and E.-C. Chang. "Benefits and Pitfalls of Using Capture the Flag Games in University Courses". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, March 2020, pp. 752–758. ISBN: 978-

1-4503-6793-6. DOI: 10.1145/3328778.3366893 (cit. on pp. 40, 61, 81).

[79] **J. Vykopal**, V. Švábenský, P. Seda, and P. Čeleda. "Preventing Cheating in Hands-on Lab Assignments". In: *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*. SIGCSE '22. Providence, RI, USA: Association for Computing Machinery, March 2022. ISBN: 978-1-4503-9070-5. DOI: 10.1145/3478431.3499420 (cit. on pp. 41, 61, 82).

[80] J. Burket, P. Chapman, T. Becker, C. Ganas, and D. Brumley. "Automatic Problem Generation for Capture-the-Flag Competitions". In: *2015 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 15)*. Washington, D.C.: USENIX Association, August 2015. URL: https://www.usenix.org/conference/3gse15/summit-program/presentation/burket (cit. on p. 41).

[81] D. Košč and **J. Vykopal**. *A toolset for automatic problem generation (APG) for hands-on labs*. 2021. URL: https://gitlab.fi.muni.cz/cybersec/apg (cit. on p. 41).

[82] E. Mousavinasab, N. Zarifsanaiey, S. R. Niakan Kalhori, M. Rakhshan, L. Keikha, and M. Ghazi Saeedi. "Intelligent tutoring systems: a systematic review of characteristics, applications, and evaluation methods". In: *Interactive Learning Environments* 29.1 (2021), pp. 142–163. DOI: 10.1080/10494820.2018.1558257 (cit. on p. 44).

[83] V. Aleven, E. A. McLaughlin, R. A. Glenn, and K. R. Koedinger. "Instruction Based on Adaptive Learning Technologies". In: *Handbook of Research on Learning and Instruction*. Ed. by R. E. Mayer and P. A. Alexander. New York: Routledge, 2016, pp. 522–560. ISBN: 9781315736419. DOI: 10.4324/9781315736419 (cit. on p. 44).

[84] A. Mitrovic and S. Ohlsson. "Implementing CBM: SQL-Tutor After Fifteen Years". In: *International Journal of Artificial Intelligence in Education* 26.1 (March 2016), pp. 150–159. ISSN: 1560-4306. DOI: 10.1007/s40593-015-0049-9 (cit. on p. 44).

[85] B. Vesin, K. Mangaroska, and M. Giannakos. "Learning in smart environments: user-centered design and analytics of an adaptive learning system". In: *Smart Learning Environments* 5.1 (2018), p. 24. DOI: 10.1186/s40561-018-0071-0 (cit. on p. 44).

[86] D. Dermeval, R. Paiva, I. I. Bittencourt, J. Vassileva, and D. Borges. "Authoring Tools for Designing Intelligent Tutoring Systems: a Systematic Review of the Literature". In: *International Journal of Artificial Intelligence in Education* 28.3 (2018), pp. 336–384. DOI: 10.1007/s40593-017-0157-9 (cit. on p. 44).

[87] V. Aleven, B. M. McLaren, J. Sewall, M. van Velsen, O. Popescu, S. Demi, M. Ringenberg, and K. R. Koedinger. "Example-Tracing Tutors: Intelligent Tutor Development for Non-programmers". In: *International Journal of Artificial Intelligence in Education* 26.1 (2016), pp. 224–269. DOI: 10.1007/s40593-015-0088-2 (cit. on p. 44).

[88] **J. Vykopal**, P. Seda, V. Švábenský, and P. Čeleda. "Smart Environment for Adaptive Learning of Cybersecurity Skills". In: *IEEE Transactions on Learning Technologies* (2022). In press. ISSN: 1939-1382. DOI: 10.1109/TLT.2022.3216345 (cit. on pp. 44, 63, 82).

[89] V. Švábenský and **J. Vykopal**. "Challenges Arising from Prerequisite Testing in Cybersecurity Games". In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE '18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, pp. 56–61. ISBN: 978-1-4503-5103-4. DOI: 10.1145/3159450.3159454 (cit. on p. 45).

[90] J. Mirkovic and P. A. Peterson. "Class Capture-the-Flag Exercises". In: *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*. 2014. URL: https://www.usenix.org/system/files/conference/3gse14/3gse14-mirkovic.pdf (cit. on p. 45).

[91] K. Maennel. "Learning Analytics Perspective: Evidencing Learning from Digital Datasets in Cybersecurity Exercises". In: *2020 IEEE European Symposium on Security and Privacy Workshops (EuroSPW)*. 2020, pp. 27–36. DOI: 10.1109/EuroSPW51379.2020.00013 (cit. on pp. 45, 46).

[92] R. Gerhards. *The Syslog Protocol*. RFC 5424. March 2009. DOI: 10.17487/RFC5424 (cit. on p. 46).

[93] Elastic NV. *The Elastic Stack*. 2021. URL: https://www.elastic.co/elastic-stack/ (cit. on p. 46).

[94] V. Švábenský, **J. Vykopal**, D. Tovarňák, and P. Čeleda. "Toolset for Collecting Shell Commands and Its Application in Hands-

on Cybersecurity Training". In: *2021 IEEE Frontiers in Education Conference (FIE)*. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637052 (cit. on p. 46).

[95] R. Ošlejšek, V. Rusňák, K. Burská, V. Švábenský, **J. Vykopal**, and J. Čegan. "Conceptual Model of Visual Analytics for Hands-on Cybersecurity Training". In: *IEEE Transactions on Visualization and Computer Graphics* 27.8 (February 2020). ISSN: 1077-2626. DOI: 10.1109/TVCG.2020.2977336 (cit. on pp. 52, 60, 81).

[96] V. Švábenský, R. Weiss, J. Cook, **J. Vykopal**, P. Čeleda, J. Mache, R. Chudovský, and A. Chattopadhyay. "Evaluating Two Approaches to Assessing Student Progress in Cybersecurity Exercises". In: *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*. SIGCSE '22. Providence, RI, USA: Association for Computing Machinery, March 2022. ISBN: 978-1-4503-9070-5. DOI: 10.1145/3478431.3499414 (cit. on pp. 56, 62).

[97] P. Fournier-Viger. *An introduction to frequent pattern mining*. Retrieved February 9, 2022 from http://data-mining.philippe-fournier-viger.com/introduction-frequent-pattern-mining/. October 2013 (cit. on p. 56).

[98] C. Romero, S. Ventura, M. Pechenizkiy, and R. S. Baker, eds. *Handbook of educational data mining*. Boca Raton, FL, USA: CRC Press, 2010. ISBN: 978-1-4398-0458-2. DOI: 10.1201/b10274 (cit. on p. 56).

[99] V. Švábenský, **J. Vykopal**, P. Čeleda, K. Tkáčik, and D. Popovič. "Student Assessment in Cybersecurity Training Automated by Pattern Mining and Clustering". In: *Education and Information Technologies* (March 2022). ISSN: 1573-7608. DOI: 10.1007/s10639-022-10954-4 (cit. on pp. 56, 62, 82).

[100] V. Švábenský and **J. Vykopal**. "Challenges Arising from Prerequisite Testing in Cybersecurity Games". In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE '18. Baltimore, MD, USA: Association for Computing Machinery, February 2018, pp. 56–61. ISBN: 978-1-4503-5103-4. DOI: 10.1145/3159450.3159454 (cit. on p. 59).

[101] R. Ošlejšek, **J. Vykopal**, K. Burská, and V. Rusňák. "Evaluation of Cyber Defense Exercises Using Visual Analytics Process". In:

*2018 IEEE Frontiers in Education Conference (FIE)*. 2018, pp. 1–9. ISBN: 978-1-5386-1174-6. DOI: 10.1109/FIE.2018.8659299 (cit. on p. 60).

[102] R. Ošlejšek, V. Rusňák, K. Burská, V. Švábenský, and **J. Vykopal**. "Visual Feedback for Players of Multi-Level Capture the Flag Games: Field Usability Study". In: *Proceedings of the 16th IEEE Symposium on Visualization for Cyber Security*. VizSec '19. Vancouver, Canada: IEEE, October 2019, pp. 1–11. ISBN: 978-1-7281-3877-0. DOI: 10.1109/VizSec48167.2019.9161386 (cit. on p. 60).

[103] P. Seda, **J. Vykopal**, V. Švábenský, and P. Čeleda. "Reinforcing Cybersecurity Hands-on Training With Adaptive Learning". In: *2021 IEEE Frontiers in Education Conference (FIE)*. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637252 (cit. on p. 61).

[104] P. Seda, **J. Vykopal**, P. Čeleda, and I. Ignác. "Designing Adaptive Cybersecurity Hands-on Training". In: *2022 IEEE Frontiers in Education Conference (FIE)*. Uppsala, Sweden: IEEE, October 2022, pp. 1–9 (cit. on p. 62).

[105] Sufatrio, **J. Vykopal**, and E.-C. Chang. "Collaborative Paradigm of Teaching Penetration Testing Using Real-World University Applications". In: *Australasian Computing Education Conference*. ACE '22. Virtual Event, Australia: Association for Computing Machinery, 2022, pp. 114–122. ISBN: 978-1-4503-9643-1. DOI: 10.1145/3511861.3511874 (cit. on p. 62).

# PART II

# COLLECTION OF WORKS

# List of Selected Papers

Out of 26 key results, I selected five journal and nine conference papers. Papers A–C analyze existing academic literature and topics covered by cybersecurity competitions. Papers D–F describe interactive learning environments we have proposed and developed. Finally, papers G–N deal with topics related to instructional methods.

(A) V. Švábenský, **J. Vykopal**, and P. Čeleda. "What Are Cybersecurity Education Papers About? A Systematic Literature Review of SIGCSE and ITiCSE Conferences". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, March 2020, pp. 2–8. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366816

(B) V. Švábenský, P. Čeleda, **J. Vykopal**, and S. Brišáková. "Cybersecurity Knowledge and Skills Taught in Capture the Flag Challenges". In: *Elsevier Computers & Security* 102.102154 (March 2021). ISSN: 0167-4048. DOI: 10.1016/j.cose.2020.102154

(C) V. Švábenský, **J. Vykopal**, P. Čeleda, and L. Kraus. "Applications of Educational Data Mining and Learning Analytics on Data From Cybersecurity Training". In: *Springer Education and Information Technologies* 1360.2357 (2022). ISSN: 1573-7608. DOI: 10.1007/s10639-022-11093-6

(D) **J. Vykopal**, R. Oslejsek, P. Celeda, M. Vizvary, and D. Tovarnak. "KYPO Cyber Range: Design and Use Cases". In: *Proceedings of the 12th International Conference on Software Technologies – Volume 1: ICSOFT*. INSTICC. SciTePress, 2017, pp. 310–321. ISBN: 978-989-758-262-2. DOI: 10.5220/0006428203100321

(E) P. Čeleda, **J. Vykopal**, V. Švábenský, and K. Slavíček. "KYPO4INDUSTRY: A Testbed for Teaching Cybersecurity of Industrial Control Systems". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, March 2020,

pp. 1026–1032. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366908

(F) **J. Vykopal**, P. Čeleda, P. Seda, V. Švábenský, and D. Tovarňák. "Scalable Learning Environments for Teaching Cybersecurity Hands-on". In: *Proceedings of the 51st IEEE Frontiers in Education Conference*. FIE '21. Lincoln, Nebraska, USA: IEEE, October 2021, pp. 1–9. ISBN: 978-1-6654-3851-3. DOI: 10.1109/FIE49875.2021.9637180

(G) **J. Vykopal**, M. Vizvary, R. Oslejsek, P. Celeda, and D. Tovarnak. "Lessons Learned From Complex Hands-on Defence Exercises in a Cyber Range". In: *2017 IEEE Frontiers in Education Conference (FIE)*. 2017, pp. 1–8. ISBN: 978-1-5090-5920-1. DOI: 10.1109/FIE.2017.8190713

(H) **J. Vykopal**, R. Ošlejšek, K. Burská, and K. Zákopčanová. "Timely Feedback in Unstructured Cybersecurity Exercises". In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE '18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, pp. 173–178. ISBN: 978-1-4503-5103-4. DOI: 10.1145/3159450.3159561

(I) V. Švábenský, **J. Vykopal**, M. Cermak, and M. Laštovička. "Enhancing Cybersecurity Skills by Creating Serious Games". In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '18. Larnaca, Cyprus: Association for Computing Machinery, July 2018, pp. 194–199. ISBN: 978-1-4503-5707-4. DOI: 10.1145/3197091.3197123

(J) R. Ošlejšek, V. Rusňák, K. Burská, V. Švábenský, **J. Vykopal**, and J. Čegan. "Conceptual Model of Visual Analytics for Hands-on Cybersecurity Training". In: *IEEE Transactions on Visualization and Computer Graphics* 27.8 (February 2020). ISSN: 1077-2626. DOI: 10.1109/TVCG.2020.2977336

(K) **J. Vykopal**, V. Švábenský, and E.-C. Chang. "Benefits and Pitfalls of Using Capture the Flag Games in University Courses". In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association

for Computing Machinery, March 2020, pp. 752–758. ISBN: 978-1-4503-6793-6. DOI: 10.1145/3328778.3366893

(L) **J. Vykopal**, V. Švábenský, P. Seda, and P. Čeleda. "Preventing Cheating in Hands-on Lab Assignments". In: *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*. SIGCSE '22. Providence, RI, USA: Association for Computing Machinery, March 2022. ISBN: 978-1-4503-9070-5. DOI: 10.1145/3478431.3499420

(M) V. Švábenský, **J. Vykopal**, P. Čeleda, K. Tkáčik, and D. Popovič. "Student Assessment in Cybersecurity Training Automated by Pattern Mining and Clustering". In: *Education and Information Technologies* (March 2022). ISSN: 1573-7608. DOI: 10.1007/s10639-022-10954-4

(N) **J. Vykopal**, P. Seda, V. Švábenský, and P. Čeleda. "Smart Environment for Adaptive Learning of Cybersecurity Skills". In: *IEEE Transactions on Learning Technologies* (2022). In press. ISSN: 1939-1382. DOI: 10.1109/TLT.2022.3216345