Process Mining Analysis of Puzzle-Based Cybersecurity Training

Martin Macak
Masaryk University
Brno, Czech Republic
macak@mail.muni.cz

Radek Oslejsek
Masaryk University
Brno, Czech Republic
oslejsek@mail.muni.cz

Barbora Buhnova
Masaryk University
Brno, Czech Republic
buhnova@mail.muni.cz

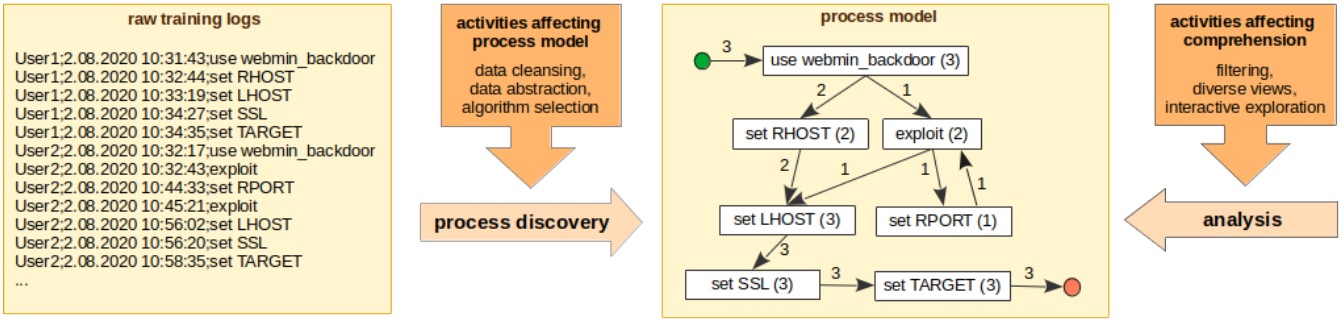

Figure 1: Principles of process discovery. Although the event log (left) captures a simple use of the Metasploit tool by only three users, it is very difficult to analyze. Process models (right) provide better cognitive features simplifying comprehension. However, the practical usability depends on many factors affecting the process discovery and process model analysis.

ABSTRACT

The hands-on cybersecurity training quality is crucial to mitigate cyber threats and attacks effectively. However, practical cybersecurity training is strongly process-oriented, making the post-training analysis very difficult. This paper presents process-mining methods applied to the learning analytics workflow. We introduce a unified approach to reconstructing behavioral graphs from sparse event logs of cyber ranges. Furthermore, we discuss significant data features that affect their practical usability for educational process mining. Based on that, methods of dealing with the complexity of process graphs are presented, taking advantage of the puzzle-based gamification of in-class training sessions.

CCS CONCEPTS

• **Security and privacy**; • **Social and professional topics** → *Computing education*; • **Human-centered computing** → *Information visualization*; • **Applied computing** → Interactive learning environments;

KEYWORDS

cybersecurity training, CTF game, process mining, data analysis

ACM Reference Format:
Martin Macak, Radek Oslejsek, and Barbora Buhnova. 2022. Process Mining Analysis of Puzzle-Based Cybersecurity Training. In *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education Vol 1 (ITiCSE 2022), July 8–13, 2022, Dublin, Ireland.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3502718.3524819

ITiCSE 2022, July 8–13, 2022, Dublin, Ireland.

1 INTRODUCTION

In many learning areas, hands-on training produces a tangible output, e.g., a code that can be checked, analyzed, and evaluated. However, this is not the case in cybersecurity, where learning goals consist of process-oriented tasks related to attack or defense skills, e.g., scanning the network for vulnerable servers or protecting the server by a firewall. Modern cyber ranges, in which practical cybersecurity training is often organized [7, 12, 28, 35], provide only limited data about these tasks. Produced event logs are not directly usable for answering questions like "Where did the trainees get in trouble and why?" Some analytical tools are needed to aggregate a more abstract insight into trainees' behavior from this low-level data.

Process mining (PM) methods have great potential in answering analytical questions in process-oriented application domains [29]. However, the quality and practical usability of models produced by existing PM algorithms are influenced by many factors, as schematically depicted in Figure 1. This paper addresses open questions that affect the usability of existing PM techniques for scalable post-training analysis of hands-on cybersecurity exercises, aiming to provide guidelines for their usage. Specifically, we formulated the following research questions.

RQ1: Data Abstraction – how to convert cyber training data to the format suitable for process mining? The PM workflow is used to find a descriptive model of the process. The idea lies in transferring event logs captured during training sessions into behavioral graphs that would provide better cognitive features for understanding users' behavior than the raw data. As the data collected from hands-on training environments are highly variable, the suitable

mapping to the input of PM algorithms is unclear. Our goal is to propose a data abstraction that can serve as a generic approach to data preprocessing in different cyber ranges with various data sources.

RQ2: Process Discovery – what are the key obstacles in the process discovery phase, and how to overcome them? Having the data in the format suitable for process discovery algorithms does not guarantee that generated process models are reasonable. Many input data characteristics affect the process discovery and its practical usability for educational process mining. We aim to identify key features of cyber training data and discuss the limits and obstacles of the process discovery phase.

RQ3: Exploratory Analysis – how to deal with the complexity of process graphs during analysis? Even a relevant process graph obtained by process discovery can become too big and complex for practical learning analytics. Their cognitive features can decrease and become similar to searching for answers in the raw data. Therefore, we research exploratory tactics that can help to tackle the graph complexity problem. We combine specific features of the cybersecurity training data with well-established visual analysis approaches to discuss and demonstrate their usability for interactive learning analytics.

In this paper, we restrict ourselves to the in-class (i.e., supervised) *Capture The Flag* (CTF) cybersecurity games [9, 20, 27, 30, 34]. They follow puzzle-based gamification principles, where puzzles are used as a metaphor for getting students to think about how to frame and solve unstructured problems [17]. This training type is very popular in the education of beginners, e.g., students of cybersecurity courses. Focusing on supervised CTF games helps us to tackle the scalability of process models.

2 RELATED WORK

In the context of cybersecurity games, we can see the related work of process mining utilization in cybersecurity [13], education [5], and game analysis [14]. However, this section will focus on the education domain, as it is the most mature one of those [10].

The term educational process mining (EPM) is often used for the application of PM to raw educational data [24]. Bogarin et al. [5] provide a comprehensive overview of EPM techniques and challenges. We use their classification of event log challenges to clarify features of cybersecurity CTF games and then to address key hurdles, especially the data complexity.

PM has already been applied in numerous specific educational situations. Macak et al. [15] used process mining to understand student coding-behavior patterns from the Git log of their projects. Mukala et al. [19] use process mining to obtain and analyze students' learning habits from Coursera. It is also used to detect students' learning difficulties by Bannert et al. [2]. Multiple other approaches use process mining to gain a better understanding of the educational process from Moodle logs [6, 24]. Our work shares some ideas and principles with these approaches. Still, it addresses a different application domain — puzzle-based cybersecurity training, aiming to utilize the specific data properties to deal with process mining challenges.

Some papers also directly address the utilization of process mining for the analysis of hands-on cybersecurity exercises [1, 32, 33, 37]. These approaches demonstrate that directed graphs carefully constructed from command history can provide useful information about trainees' behavior. The graphs are built on restricted data samples to deal with the complexity — only commands identified as significant are selected. Our research is more generic, covering a wider variety of training events and enabling to extend process modeling with new data types. Another difference is in the conceptual approach. The previous approaches are based on conformance checking, i.e., monitoring and checking whether the real execution of the process corresponds to the reference process model. Our solution focuses on exploratory learning analytics based on general process discovery methods where no process model of expected behavior is assumed.

Mirkovic et al. [18] use terminal histories and exercise milestones to enable automated assessment of hands-on cybersecurity training. In contrast, our work aims to introduce a human into the analytical loop [21] when process models are reconstructed from logs automatically but then interactively analyzed by domain experts so that they are able to reveal hidden relationships in the data.

3 RQ1: DATA ABSTRACTION

In this section, we classify cybersecurity training data and provide their unified mapping into the input of process discovery algorithms to address the research question *RQ1*. The proposed solution builds on our long-term experience in developing *KYPO Cyber Range*[1] [31], which we have been operating since 2013 and which serves as a platform for regular practical training of students of our university.

3.1 Principles of CFT games

Cybersecurity CTF games consist of well-described cybersecurity goals divided into consecutive tasks (puzzles). Completing each task yields a text string called the *flag* that must be inserted into the system to proceed to the subsequent task. Moreover, trainees can take hints or skip the entire task. Points are awarded or deducted for these actions so that the final scores of individual trainees are mutually comparable and can be used for their basic evaluation.

Trainees perform cybersecurity tasks on remote hosts located inside isolated computer networks. Modern cyber ranges provide a virtualized implementation of such networks, where each trainee has its own copy of the network and is able to access hosts via remote command lines or desktops, likewise in the physical world.

Our data sets were captured from games focusing on network-oriented attack vectors like server exploitation, privilege escalation, and cracking stolen passwords. However, other cybersecurity tasks, e.g., binary exploitation or reverse engineering, can be considered if the relevant user activities are logged.

3.2 Data Types

Events are captured from multiple sources and can provide different levels of granularity [36]. This paper discusses three distinct data categories, but other data types or levels can be included if available.

Game events are produced by the gaming interface of KYPO Cyber Range. Its goal is to provide instructions and guide trainees through the whole training session. User interaction with the interface produces events that capture the gameplay state in the training scenario. Events summarized in Table 1 reflect the puzzle-based principles, and they are typical for all games regardless of the content.

[1] KYPO is a Czech acronym for Cybersecurity Polygon.

Table 1: Game events and their meaning.

Event	Description: The trainee ...
TrainingRunStarted	... started the training.
TaskCompleted	... submitting a correct flag.
WrongFlagSubmitted	... submitted a wrong flag.
HintTaken	... took a hint.
SolutionDisplayed	... viewed the task solution.

Bash commands are produced inside computer networks in which the cybersecurity tasks are solved. Currently, commands executed on the UNIX command line (shell) are available.

Metasploit tool capture the usage of the Metasploit framework – a popular command-line application used for penetration testing. These events are also captured at hosts of the computer network, likewise the bash history, but represent even a finer-grained type of data.

Each event, regardless of its type or granularity, is extended with additional pieces of information, such as the trainee's identifier, timestamp, and the task (puzzle) in which the event appeared. Moreover, individual event types can have specific mandatory or optional data. For example, the submission of the flag always includes also the flag value, and commands may include their parameters.

3.3 Unified Data Mapping

All process mining techniques require the presence of data with specific semantics [29]: (a) Each event in the log file needs to refer to a single process instance, named *case*, (b) each event needs to refer to a step in the process, named *activity*, and (c) events within the case have to be ordered, either sequentially or by including *timestamp*. These minimal requirements ensure that each case represented by the sequence of activities can be treated as a *trace* of user actions, enabling the process discovery algorithms to produce graph models capturing all the traces compactly.

Unfortunately, CFT data are highly variable. They can differ in abstraction and semantics, as shown by the three aforementioned data types — game events, bash commands, and the Metasploit tool.

To deal with variability, we introduce a generic data abstraction layer that makes the mapping smooth and transparent regardless of the specific training content. The proposed classification scheme serves as a mediator between heterogeneous event logs of the cyber range and the data format required by process mining techniques. Table 2 provides a mapping example, where a snippet of CTF data (four events) is mapped into the abstraction layer.

Event type: A rough classification defining different types of events. We can distinguish player actions in the game from reactions of the system on the player's actions, or assessment events, for instance [25]. The exact classification used for process mining depends on available data and analytical goals. Usually, each event type has its specific structure (required or optional data), and they affect how the pieces of information are spread across the other elements of the data abstraction. This paper deals with three aforementioned event types: `game` events capturing the players' progress, `bash` commands used on network hosts, and `msf` for Metasploit commands also used on hosts. Additional types can be easily defined.

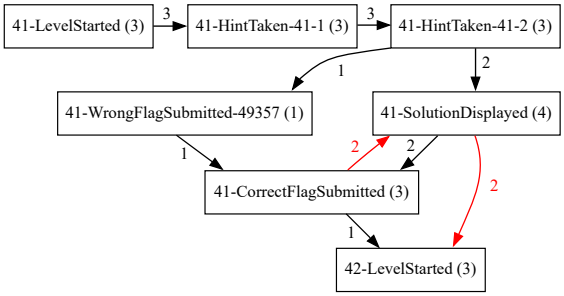

Figure 2: Heuristic net of *game events* of a single game task (puzzle). Nodes represent *activities*, numbers denote the number of passes through. The red arrows highlight a flow in the gameplay caused by a bug in platform implementation.

Event: Finer classification of event types. Events represent a primary subject of behavioral analysis. They should capture significant steps in the development of training sessions. Therefore, they represent the *activities* of the process models. In CTF games, events are either game events defined in Table 1 or any shell or Metasploit commands.

Event parameters: Optional data associated with events. Additional information that extends events and enables the analyst to distinguish finely between them. For example, shell or Metasploit commands can have additional arguments, or the *HintTaken* game event can be equipped with a short hint description.

Timestamp: A timestamp of the event. This is required because of the aggregation of multiple event types.

Trainee: An anonymized unique identifier of the trainee who produced the event. Using the trainee identifier as the *caseID* ensures that the process discovery reconstructs the walkthroughs of individual trainees. The walkthrough perspective presents a primary subject of learning analytics. It enables analysts to compare trainees' behavior mutually as well as to analyze the expected versus anomalous behavior with respect to the training definition.

3.4 Usability

We conducted practical experiments with PM4Py [3] library to check the utilization of CTF data mapping for process discovery. The experiments proved that the proposed workflow is able to generate meaningful process models from our raw data at runtime.

Our initial research revealed that heuristic nets especially fit the analytical goals of educators the best. Even though the goal of these experiments was not to study the impact of obtained models on learning analytics, we were able to notice several interesting facts from process graphs. For example, the game task in Figure 2 can be considered tough for the trainees because they all took two hints, then two gave up (they looped at the solution with the correct flag). One trainee found and submitted the correct flag on the second try. Using the same model, we also have discovered a flaw in the implementation of KYPO Cyber Range. If trainees submitted the correct flag after displaying a task solution, they were redirected to the solution page instead of being moved to the next puzzle. More examples of process models can be found in our experience report [16].

Table 2: Mapping of raw data onto the unified CTF data abstraction and process mining inputs.

Process mining input:		*activity*		*time/ordering*	*caseID*
Data abstraction:	EVENT TYPE	EVENT	EVENT PARAMETERS	TIMESTAMP	TRAINEE
Raw data (snippet):	game	HintTaken 41-1		2020-05-14 10:16:11	user 1
	game	HintTaken 41-2		2020-05-14 10:16:34	user 1
	msf	exploit	-j	2020-05-14 10:18:23	user 2
	bash	nmap	-sL 10.1.26.9:5050	2020-05-14 10:32:16	user 1

4 RQ2: PROCESS DISCOVERY

The proposed data abstraction enables us to use cyber-training data transparently in process mining algorithms. However, the practical usability of process discovery is affected by many factors.

The most significant challenges that appear when using event logs for educational process mining are addressed in the previous research [5, 24]. Using their classification, we analyzed raw data of multiple training sessions to identify the most significant features that affect the utilization of cyber training logs in post-training process discovery. Fifteen training sessions of six different CTF games were analyzed for statistical properties like training duration or the number of log events of different types.

In what follows, we summarize our observations and lessons learned. Each data characteristic is introduced by the problem statement followed by our findings.

4.1 Data Size

The number of cases or events in event logs may become so high that they exceed the time or memory requirements of process discovery algorithms. Moreover, as we aim to provide interactive data exploration, the speed of PM generation should be close to real-time.

The real amount of collected data depends on the number of participants, the difficulty of training (i.e., the amount of potentially recorded activities), and training duration. CTF games are intended primarily for beginners and then relatively small – consisting of 4-6 cybersecurity tasks (puzzles) solvable in roughly 120 minutes (observed minimum was 65, maximum 210, average 119). Moreover, in-class training sessions considered in this paper restrict the data even more because also the number of trainees is limited. In our datasets, the number of trainees varied between 4 and 20, 10 on average.

The number of logged events (and then potential nodes of the process graphs) in our datasets varied between 370–3000 per the whole training session (average 1100, median 814) and between 53–150 per participant (average 108, median 111).

The data amount does not pose any problems to current process discovery algorithms that can treat such an amount of data very quickly [11]. On the other hand, our experiments revealed that the problem could be with the comprehensibility of produced models, even for limited in-class CTF data. Obtained graphs consisting of tens or hundreds of nodes are usually too complex to be cognitively treated by analysts. Employment of data filtering and interactive exploration is, therefore, necessary.

4.2 Distribution of Data Sources

Data for educational process mining may be distributed over a variety of sources, e.g., theory and practice classroom or online learning environments. These sources provide event logs with different structures and meanings, which makes their unification and aggregation for process discovery challenging.

We focus on only the data collected from cyber ranges. Considering other supporting sources of information is out of the scope of this paper. However, the problem with data distribution occurs as well because modern cyber ranges produce data from at least two data sources: (a) the gaming environment responsible for the training tasks and learning milestones and (b) sandboxes (computer networks) where the tasks are performed. Nevertheless, the data abstraction discussed in Section 3 solves this problem by unifying the way the data from multiple data sources is handled and used transparently for process discovery.

4.3 Granularity

Events in event logs may have a different level of detail. The data presented in Section 3 demonstrates that this problem also occurs in the cyber training logs.

Thanks to the unified data abstraction, the granularity poses no technical problem for the process discovery of cyber training logs because the data can be mapped transparently. Moreover, this unified approach brings significant advantages when data with different granularity are put together. Multiple levels of granularity can define multiple levels of abstraction that can drive data exploration strategies, as discussed in Section 5.

4.4 Noise in the Data

Noise is defined as exceptional behavior which is not representative of the typical behavior of the process.

We observed that many students leave the training unfinished due to a lack of time or a loss of motivation. Typically, this kind of noise could pose troubles for educational PM analysis. However, the puzzle-based structure of CTF games provides clear milestones — correct flags that explicitly delimit the borders of training phases. Therefore, it is easy to spot this situation in process graphs and further investigate the reason. In general, the puzzle-based structure can be considered a template of expected behavior, and any difference revealed by PM models can indicate flows in the game design or the training organization worth further investigation.

4.5 Incompleteness

Possible data incompleteness is tightly connected to the measurement infrastructure. Although the individual cyber ranges can differ in these aspects, they are often complex, distributed with asynchronous communication, running on underlying virtualization, and then unreliable. Our long-term experience with organizing training

sessions of many types reveals that many things can go wrong due to failures in low-level virtualization services, network connectivity, or improper usage. These failures then cause missing data.

The experiments have shown that keeping this incomplete data in the dataset can produce biased process models. Unfortunately, it is usually very difficult to notice from process graphs that there is something wrong with the raw event logs. Data cleansing and completeness checking have to be usually done in the preprocessing phase of the analytical workflow.

4.6 Timestamps

Distributed environments like cyber ranges can produce timestamps that are not sufficiently synchronized. Only a small shift in times can re-order events and then produce significantly different process models. Therefore, precise synchronization at the level of the underlying infrastructure is required.

Trainees can start the exercise at different times, even if they sit in the same classroom. This aspect becomes even more significant if the training is not organized as a fixed-time group session, but the training content is available online at any time. Fortunately, the puzzle-based structure of CTF games enables us to identify the exact start of the gameplay of individual trainees and then compute relative times instead of using absolute times, obtaining meaningful models.

An even worse situation can appear if the trainees can "pause the training". It does not happen on session-based training courses with a tight schedule. However, loosely conceived training programs would enable participants to stop playing for a while and continue with tasks later, even the next day. Therefore, datasets from the loosely organized training events require much more attention and expertise to be paid by the analyst, who has to take care of time corrections and interpretation of obtained models.

5 RQ3: EXPLORATORY ANALYSIS

Despite the limited size of event logs produced by in-class CTF games, the experiments turned out that obtained process graphs can be too complex and incomprehensible for effective analysis. As the complexity of process graphs can pose a critical aspect for practical usability, we discuss possible strategies for tackling this problem in this section.

5.1 Filtering Driven by Data Abstraction

The granularity of training logs discussed in Section 4 can be used to control the level of detail and then the size of obtained process graphs. This kind of semantic classification can be used for efficient data filtering and implementing the well-known Shneiderman's visual information-seeking principle: Overview first, zoom and filter, then details-on-demand [26].

The granularity is encoded in the *Event types* data abstraction parameter that defines different semantic views of the data. CTF *game events* delimit boundaries of individual puzzles in which other events appear. On the other hand, *bash commands* represent a detailed view of solving tasks within a puzzle. *Metasploit commands* also provide a similar view but at an even more fine-grained level of detail – the usage of a specific tool. Based on this observation, filtering of the process model to only a specific EVENT TYPE could provide the desired level of detail and reduce information complexity. Figure 1 depicts the model limited to the Metasploit only, while in Figure 2, only game events are selected, entirely omitting bash and Metasploit commands.

Another graph reduction technique utilizes the distribution of the raw data between *event* and *event parameters*. Consider the situation when a trainee takes a hint 41-1 and then a hint 41-2. If these events are mapped into the data abstraction like in Table 2, then the process discovery algorithm distinguishes between hints 41-1 and 41-2, creates separate nodes for them, and produces a model like that in Figure 2. On the contrary, if we change the mapping, so that EVENT = "*HintTaken*" and hint numbers 41-1 and 41-2 are provided only as EVENT PARAMETERS, then a simplified model is produced with only a single joint "41-HintTaken" node covering all hints taken in the task.

This filtering principle is even more important for Bash and Metasploit commands than game events because trainees have big freedom of what to type on the command line. Mapping only command names without parameters to EVENTS seems to be a reasonable strategy for initial analysis. On the other hand, an `ssh` command, for instance, says nothing about the remote connection that has been made. In this case, the analyst should rather map the connection argument to the EVENT and then produce separate nodes like "`ssh root@172.18.1.5`" and "`ssh admin@172.18.1.5`" in the process graph. Only then is the analyst able to explicitly see different attempts (traces) and evaluate their correctness. Therefore, the mapping has to be used carefully and iteratively during the analytical process to balance information hiding with graph complexity.

5.2 Puzzle-based Fragmentation and Drill Down

While the unified data abstraction can serve as a fine-grained filtering mechanism of the entire process model across multiple puzzles, the puzzle-based structure of the training content provides a vertical fragmentation of the data usable for drill-down exploration. Process graphs can be logically split into loosely coupled coherent parts that correspond to individual puzzles, as shown in Figure 3, where the puzzles of tasks 43, 44, and an info puzzle are visually recognizable.

Based on this observation, we can tackle the complexity of multi-puzzle graphs by allocating puzzles' boundaries and using obtained graph fragments for coarse-grained filtering and data aggregation. Statistical data of each fragment (puzzle) can be distilled into an overview of the whole training session, while smaller detail process graphs of individual puzzles can be used for drill-down exploration.

Figure 3 illustrates this multi-layer approach. An overview part consists of a series of circles whose size and color can encode interesting metrics. For example, spheres' size can be calculated from the number of activities – nodes of the puzzle's process graph. In this case, the size of task 44 would indicate that it is the most complicated part of the training with a lot of recorded activities (note that Bash and Metasploit commands are omitted from the graph view in Figure 3, but they can be calculated in the complexity metric and reflected in the sphere's size). Then the analyst can interactively drill down into selected puzzles to analyze the reason that could be either the task complexity (it requires many commands to be used) or difficulty (trainees struggled with the task completion and then generated many events).

The number of activities discussed in the previous example is not the only possible metric. Alternate metrics can provide a different

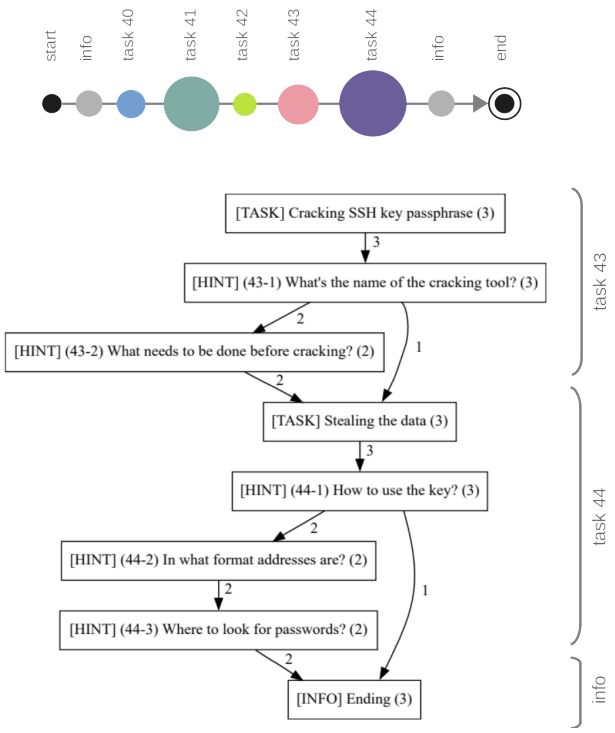

Figure 3: The approximate visualization of tasks difficulty (up), where a bigger node means the more complex task, and the corresponding process graph (bottom). Only *game events* are included in the process graph. Only tasks 43, 44, and the info puzzle are shown in this example to save space.

perspective on the training results. For example, the number of displayed solutions can indicate how many trainees gave up the puzzle due to the task's difficulty, demotivation, or lack of time.

6 DISCUSSION AND FUTURE WORK

This section summarizes our results, putting emphasis on the simplifications that we put on the training data.

6.1 Limitations

The results of this study are limited by two key constraints put on hands-on training: puzzle-based gamification and in-class education.

Considering only well-structured puzzle-based CTF games enables us to better classify data and fragment complex process models for drill-down exploration. We omit other training concepts, e.g., complex Cyber Defense Exercises [22] that are intended for experienced professionals. They use wide network topologies, provide freedom in the exercise scenarios to simulate reality, and participants collaborate in teams, which may produce higher amounts of less-structured data and then violate the prerequisites of our observations.

In-class training is limited to time and number of participants, which helps us keep data size within reasonable limits. However, cybersecurity training programs can also have the form of online courses accessible at any time by an unlimited number of participants. These courses can produce a significantly larger amount of data and pose some troubles with time dependencies that are crucial for correct process discovery, as discussed in Section 4.6. Therefore, extending our approach beyond in-class teaching requires further research.

We are also aware that data sets used in this paper were collected from CTF games organized by a single team in a single cyber range, which could affect our observations and limit generalization. On the other hand, to the best of our knowledge, other modern cyber ranges supporting the CTF training style, e.g., Cyris, CyTrONE, or Ares [4, 8, 23], share the same concepts and principles that are discussed in this paper. Therefore, the training content and collected data may differ in detail, but the key aspects like size or types of events are very similar.

6.2 Implications for Teaching Practice

It is tough to identify flows in training design or analyze trainees' behavior without transforming events into models with better cognitive features. The proposed unified data abstraction can be directly used to map the data from cyber ranges into the input of exiting process mining tools and algorithms.

On the other hand, the practical usability of these generic tools can be limited. Their usability depends on the support of discussed data filtering and exploration techniques that are often domain-specific. Therefore, we are currently working on integrating these techniques into the analytical interface of KYPO Cyber Range so that the process mining analysis becomes an integral part of the training life cycle.

7 CONCLUSION

This paper explores the practical usability of existing process mining algorithms to analyze cybersecurity training sessions and provides observations that support this direction.

Despite the variability of data collected from cyber ranges, we proposed a unified data abstraction for using the data as the input of process mining algorithms. We tested the usability with data captured in a cyber range that we operate. The practical experiments proved the usefulness of our approach for answering questions related to learning analytics and evaluating corresponding hypotheses but also revealed limits caused by concrete features of the raw data.

We analyzed data from 15 training sessions to reveal significant features that affect the practical usability of process discovery algorithms. The main problem we faced was the complexity of the obtained graphs. Therefore, we introduced several strategies of data filtering and interactive data exploration that are built on specific features of the puzzle-based in-class form of exercises. Time-limited in-class education reduces problems with time unification and interrupted gameplay. Puzzle-based structure enables us to employ a drill-down approach to data exploration. The practical usability in training programs that are not restricted, e.g., complex cyber defense exercises, remains an open question for future work.

ACKNOWLEDGMENTS

This research was supported by the Security Research Programme of the Czech Republic 2015–2022 (BV III/1–VS) granted by the Ministry of the Interior of the Czech Republic under No. VI20202022158 – Research of New Technologies to Increase the Capabilities of Cybersecurity Experts.

REFERENCES

[1] Mauro Andreolini, Vincenzo Giuseppe Colacino, Michele Colajanni, and Mirco Marchetti. 2020. A Framework for the Evaluation of Trainee Performance in Cyber Range Exercises. *Mobile Networks and Applications* 25, 1 (2020), 236–247.

[2] Maria Bannert, Peter Reimann, and Christoph Sonnenberg. 2014. Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and learning* 9, 2 (2014), 161–185.

[3] Alessandro Berti, Sebastiaan J van Zelst, and Wil van der Aalst. 2019. Process mining for Python (PM4Py): bridging the gap between process-and data science. *arXiv preprint arXiv:1905.06169* (2019).

[4] Razvan Beuran, Dat Tang, Cuong Pham, Ken-ichi Chinen, Yasuo Tan, and Yoichi Shinoda. 2018. Integrated framework for hands-on cybersecurity training: CyTrONE. *Computers & Security* 78 (2018), 43–59.

[5] Alejandro Bogarín, Rebeca Cerezo, and Cristóbal Romero. 2018. A survey on educational process mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8, 1 (2018), e1230.

[6] Alejandro Bogarín, Cristóbal Romero, Rebeca Cerezo, and Miguel Sánchez-Santillán. 2014. Clustering for Improving Educational Process Mining. In *Proceedings of the Fourth International Conference on Learning Analytics And Knowledge* (Indianapolis, Indiana, USA) *(LAK '14)*. Association for Computing Machinery, New York, NY, USA, 11–15. https://doi.org/10.1145/2567574.2567604

[7] Nestoras Chouliaras, George Kittes, Ioanna Kantzavelou, Leandros Maglaras, Grammati Pantziou, and Mohamed Amine Ferrag. 2021. Cyber Ranges and TestBeds for Education, Training, and Research. *Applied Sciences* 11, 4 (2021).

[8] Circadence. 2021. *Ares Project*. https://projectares.academy/

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

[10] Cleiton dos Santos Garcia, Alex Meincheim, Elio Ribeiro Faria Junior, Marcelo Rosano Dallagassa, Denise Maria Vecino Sato, Deborah Ribeiro Carvalho, et al. 2019. Process mining techniques and applications - A systematic mapping study. *Expert Systems with Applications* 133 (2019), 260 – 295. https://doi.org/10.1016/j.eswa.2019.05.003

[11] Sergio Hernández, Joaquín Ezpeleta, S.J. van Zelst, and Wil M.P. van der Aalst. 2015. Assessing Process Discovery Scalability in Data Intensive Environments. In *2015 IEEE/ACM 2nd International Symposium on Big Data Computing (BDC)*. 99–104. https://doi.org/10.1109/BDC.2015.31

[12] Marcus Knüpfer, Tore Bierwirth, Lars Stiemert, Matthias Schopp, Sebastian Seeber, Daniela Pöhn, and Peter Hillmann. 2020. Cyber Taxi: A Taxonomy of Interactive Cyber Training and Education Systems. In *Model-driven Simulation and Training Environments for Cybersecurity*, George Hatzivasilis and Sotiris Ioannidis (Eds.). Springer International Publishing, Cham, 3–21.

[13] Martin Macak, Lukas Daubner, Mohammadreza Fani Sani, and Barbora Buhnova. 2022. Process mining usage in cybersecurity and software reliability analysis: A systematic literature review. *Array* 13 (2022), 100120. https://doi.org/10.1016/j.array.2021.100120

[14] Martin Macak, Lukas Daubner, Julia Jamnicka, and Barbora Buhnova. 2022. Game Achievement Analysis: Process Mining Approach. In *Advanced Data Mining and Applications*, Bohan Li, Lin Yue, Jing Jiang, Weitong Chen, Xue Li, Guodong Long, Fei Fang, and Han Yu (Eds.). Springer International Publishing, Cham, 68–82.

[15] Martin Macak, Daniela Kruzelova, Stanislav Chren, and Barbora Buhnova. 2021. Using process mining for Git log analysis of projects in a software development course. *Education and Information Technologies* (2021), 1–31.

[16] Martin Macák, Radek Ošlejšek, and Barbora Bühnová. 2022. Applying Process Discovery to Cybersecurity Training: An Experience Report. In *2022 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*.

[17] Zbigniew Michalewicz and Matthew Michalewicz. 2008. *Puzzle-based learning*. Hybrid Publishers, Ormond, Australia.

[18] Jelena Mirkovic, Aashray Aggarwal, David Weinman, Paul Lepe, Jens Mache, and Richard Weiss. 2020. Using Terminal Histories to Monitor Student Progress on Hands-on Exercises. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. 866–872.

[19] P Mukala, J Buijs, M Leemans, and W van der Aalst. 2015. Learning analytics on coursera event data: a process mining approach. In *5th International Symposium on Data-Driven Process Discovery and Analysis (SIMPDA 2015*. CEUR-WS. org, 18–32.

[20] Radek Ošlejšek, Vít Rusňák, Karolína Burská, Valdemar Švábenský, and Jan Vykopal. 2019. Visual Feedback for Players of Multi-Level Capture the Flag Games: Field Usability Study. In *2019 IEEE Symposium on Visualization for Cyber Security (VizSec)*. IEEE, 1–11. https://doi.org/10.1109/VizSec48167.2019.9161386

[21] Radek Ošlejšek, Jan Vykopal, Karolína Burská, and Vít Rusňák. 2018. Evaluation of Cyber Defense Exercises Using Visual Analytics Process. In *Proceedings of the 48th IEEE Frontiers in Education Conference (FIE'18)* (San Jose, California, USA). IEEE, San Jose, California, USA, 1–9. https://doi.org/10.1109/FIE.2018.8659299

[22] Victor-Valeriu Patriciu and Adrian Constantin Furtuna. 2009. Guide for designing cyber security exercises. In *Proceedings of the 8th WSEAS International Conference on E-Activities and information security and privacy*. World Scientific and Engineering Academy and Society (WSEAS), 172–177.

[23] Cuong Pham, Dat Tang, Ken-ichi Chinen, and Razvan Beuran. 2016. CyRIS: A Cyber Range Instantiation System for Facilitating Security Training. In *Proceedings of the Seventh Symposium on Information and Communication Technology* (Ho Chi Minh City, Vietnam) *(SoICT '16)*. ACM, New York, NY, USA, 251–258. https://doi.org/10.1145/3011077.3011087

[24] Cristóbal Romero, Rebeca Cerezo, Alejandro Bogarín, and Miguel Sánchez-Santillán. 2016. Educational process mining: a tutorial and case study using Moodle data sets. *Data mining and learning analytics: Applications in educational research* 1 (2016).

[25] Katie Salen, Katie Salen Tekinbaş, and Eric Zimmerman. 2004. *Rules of play: Game design fundamentals*. MIT press.

[26] Ben Shneiderman. 1996. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE symposium on visual languages*. IEEE, 336–343.

[27] Valdemar Švábenský, Jan Vykopal, Milan Cermak, and Martin Laštovička. 2018. Enhancing Cybersecurity Skills by Creating Serious Games. In *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. ACM, ACM, New York, NY, USA, 194–199.

[28] Elochukwu Ukwandu, Mohamed Amine Ben Farah, Hanan Hindy, David Brosset, Dimitris Kavallieros, Robert Atkinson, Christos Tachtatzis, Miroslav Bures, Ivan Andonovic, and Xavier Bellekens. 2020. A Review of Cyber-Ranges and Test-Beds: Current and Future Trends. *Sensors* 20, 24 (2020).

[29] Wil van der Aalst. 2016. *Process Mining: Data Science in Action* (2nd ed.). Springer Publishing Company, Incorporated.

[30] Giovanni Vigna, Kevin Borgolte, Jacopo Corbetta, Adam Doupé, Yanick Fratantonio, Luca Invernizzi, Dhilung Kirat, and Yan Shoshitaishvili. 2014. Ten Years of iCTF: The Good, The Bad, and The Ugly. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education*. USENIX Association, San Diego, CA.

[31] Jan Vykopal, Radek Ošlejšek, Pavel Čeleda, Martin Vizváry, and Daniel Tovarňák. 2017. KYPO Cyber Range: Design and Use Cases. In *Proceedings of the 12th International Conference on Software Technologies - Volume 1: ICSOFT* (Madrid, Spain). SciTePress, Madrid, Spain, 310–321. https://doi.org/10.5220/0006428203100321

[32] Richard Weiss, Michael E Locasto, and Jens Mache. 2016. A reflective approach to assessing student performance in cybersecurity exercises. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*. 597–602.

[33] Richard Weiss, Franklyn Turbak, Jens Mache, and Michael E Locasto. 2017. Cybersecurity education and assessment in EDURange. *IEEE Security & Privacy* 3 (2017), 90–95.

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

[35] Muhammad Mudassar Yamin, Basel Katt, and Vasileios Gkioulos. 2020. Cyber ranges and security testbeds: Scenarios, functions, tools and architecture. *Computers & Security* 88 (2020), 101636. https://doi.org/10.1016/j.cose.2019.101636

[36] Valdemar Švábenský, Jan Vykopal, Daniel Tovarňák, and Pavel Čeleda. 2021. Toolset for Collecting Shell Commands and Its Application in Hands-on Cybersecurity Training. In *2021 IEEE Frontiers in Education Conference (FIE)* (New York, NY, USA). IEEE, New York, NY, USA, 1–9. https://doi.org/10.1109/FIE49875.2021.9637052

[37] Valdemar Švábenský, Richard Weiss, Jack Cook, Jan Vykopal, Pavel Čeleda, Jens Mache, Radoslav Chudovský, and Ankur Chattopadhyay. 2022. Evaluating Two Approaches to Assessing Student Progress in Cybersecurity Exercises. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education (SIGCSE '22)* (New York, NY, USA). ACM, New York, NY, USA, 787–793. https://doi.org/10.1145/3478431.3499414