Data in Brief 38 (2021) 107398

Contents lists available at ScienceDirect

Data in Brief

journal homepage: www.elsevier.com/locate/dib

Data Article

Dataset of shell commands used by participants of hands-on cybersecurity training

Valdemar Švábenský[a,b,1,*], Jan Vykopal[a,1], Pavel Seda[b,1], Pavel Čeleda[a,1]

[a] Institute of Computer Science, Masaryk University, Šumavská 15, Brno 60200, Czech Republic
[b] Faculty of Informatics, Masaryk University, Botanická 68a, Brno 60200, Czech Republic

ARTICLE INFO

Article history:
Received 26 July 2021
Accepted 17 September 2021
Available online 22 September 2021

Keywords:
Cybersecurity education
Cybersecurity exercise
Linux shell
Metasploit
Command-line history
Host-based data collection
Educational data mining
Learning analytics

ABSTRACT

We present a dataset of 13446 shell commands from 175 participants who attended cybersecurity training and solved assignments in the Linux terminal. Each acquired data record contains a command with its arguments and metadata, such as a timestamp, working directory, and host identification in the emulated training infrastructure. The commands were captured in Bash, ZSH, and Metasploit shells. The data are stored as JSON records, enabling vast possibilities for their further use in research and development. These include educational data mining, learning analytics, student modeling, and evaluating machine learning models for intrusion detection. The data were collected from 27 cybersecurity training sessions using an open-source logging toolset and two open-source interactive learning environments. Researchers and developers may use the dataset or deploy the learning environments with the logging toolset to generate their own data in the same format. Moreover, we provide a set of common analytical queries to facilitate the exploratory analysis of the dataset.

* Corresponding author.
E-mail addresses: svabensky@ics.muni.cz (V. Švábenský), vykopal@ics.muni.cz (J. Vykopal), seda@fi.muni.cz (P. Seda), celeda@ics.muni.cz (P. Čeleda).
[1] https://twitter.com/cybersecmuni

https://doi.org/10.1016/j.dib.2021.107398

Specifications Table

Subject	Computer science applications
Specific subject area	Cybersecurity training with assignments solved via a Linux command-line
Type of data	Command-line histories from Bash, ZSH, and Metasploit shell with associated metadata in JSON format Analytical software for processing the data in Elasticsearch, Logstash, and Kibana
How data were acquired	We used an open-source logging toolset based on the Syslog protocol [1]. The toolset was deployed in a virtual environment (sandbox) consisting of several emulated computer systems and networks [2]. As the trainees solved the training assignments in the sandbox, the logging toolset transparently captured their command-line histories.
Data format	Raw
Parameters for data collection	All commands with their arguments that the trainees submitted in the command-line were automatically captured and formatted as JSON records. The logs were captured exclusively in the training sandbox; therefore, they do not contain any sensitive information about the trainees. The data are completely anonymous.
Description of data collection	Trainees at various proficiency levels (high school, university, and professional learners) attended cybersecurity training sessions hosted at our university or by our collaborators. They solved cybersecurity assignments to practice their skills with command-line tools in Kali, a Linux distribution for penetration testing and digital forensics. The training occurred in virtual sandboxes that emulated realistic computer systems. During the training, the commands submitted by the trainees were collected along with associated metadata.
Data source location	Masaryk University, Brno, Czech Republic
Data accessibility	The material associated with this article can be found at https://zenodo.org/record/5517479 (DOI: https://doi.org/10.5281/zenodo.5517479). It includes the dataset itself, as well as software to facilitate its analysis. Finally, we share a public GitLab repository at https://gitlab.ics.muni.cz/muni-kypo-trainings/datasets/commands that we aim to gradually update with new data in the future.

Value of the Data

- Educational data mining and learning analytics are emerging research fields that analyze data from educational contexts. Such research enables to improve the methods for educating cybersecurity experts. However, it relies on high-quality primary data, and few cybersecurity education datasets exist. We believe this is the first human-generated dataset of shell commands and corresponding metadata from authentic cybersecurity training, which features realistic tools for penetration testing and digital forensics.
- Researchers in computing or education may benefit from these data. The possible use cases include, but are not limited to: training and testing machine learning models (for example, classifiers for skill assessment [3]), evaluating data mining methods, correlating actions from multiple sandboxes, prototyping student models, or detecting security threats.
- The data are normalized and formatted as JSON (JavaScript Object Notation) records. This standard, semi-structured, and easily reusable format enables researchers to directly process the data in a way that suits their needs. The possibilities range from employing analytical tools, such as ELK (Elasticsearch, Logstash, Kibana), to writing dedicated analytical scripts.
- Preparing and developing cybersecurity training sessions requires substantial resources, time, and effort. As a result, instructors and educational researchers often have little time to set up an infrastructure for rigorous data collection and analysis. We contribute to the research community by sharing these original, raw data from cybersecurity training.

- The dataset is freely available and may be used without restrictions, since it does not contain any sensitive or personally identifiable information. Ethical aspects of data collection were adhered to, and the privacy of the trainees who submitted the commands was preserved.

1. Data Description

The dataset features 13446 commands originating from Bash [4], ZSH [5], and Metasploit [6] shell. The commands were submitted by 175 trainees, distributed among 27 training sessions with approximately 6–7 trainees per session on average.

1.1. Data format

The commands submitted in the training sandbox (see Section 3.2) are stored in the files titled `sandbox-id-useractions.json`, where `id` is an arbitrary numerical identifier. The command history files contain JSON entries in the format shown in Listing 1. Each such entry corresponds to a single command submitted by the trainee. In total, the dataset comprises 13446 such records.

The meaning of the individual data fields follows.

- `timestamp_str` represents the time of the command's submission in the ISO 8601 format (up to millisecond or microsecond precision).
- `cmd` is the full command (the tool name and its arguments) submitted by the trainee.
- `cmd_type` is the application used to execute the command: either `bash-command` for the tools executed from Bash/ZSH, or `msf-command` for Metasploit shell.
- `username` is the account name on the sandboxed machine under which the command was executed. The account names are set by the training author, and they never store personal information of the trainee. `username` is stored only for the `bash-command` type.
- `hostname` is the name of the machine in the sandbox on which the command was executed.
- `ip` is the IPv4 address of the corresponding virtual host in the sandbox. The IP addresses do not represent any real machine on the Internet.
- `wd` is the working directory in which the command was executed. Combined with the data fields above, the trainees' command-line prompt looks like this: `username@hostnamewd$`, e.g., `root@attacker/home$` for Listing 1. `wd` is stored only for the `bash-command` type.
- `sandbox_id` is an arbitrary numerical identifier of the trainee's sandbox. It is a duplicate of the `id` in the filename `sandbox-id-useractions.json` for sanity checks.
- `pool_id` is an arbitrary numerical identifier that associates the sandboxes from one training session into a so-called *pool*. All sandboxes from the same session belong to the same pool.

1.2. Data background

The data were collected from multiple different cybersecurity training sessions. In each session, the trainees practiced using Linux command-line tools in a training sandbox of emulated computer systems.

To understand the origin of the data, we briefly explain the training background. Each training is comprised of two components shown in Fig. 1:

- A *sandbox definition* is a text file that describes the training network topology and the host configuration. It defines, for example, which software is installed on the machines in the training sandbox. After the definition is instantiated, a training sandbox is created.
- A *training definition* is a text file that specifies the wording of the assignments that the trainee completes in the training sandbox. Based on the selected training definition, the trainees use different tools to solve the tasks.

When these two components are deployed in an interactive learning environment (see Section 4.2), a *training instance* is created. Each training instance is associated with a specific

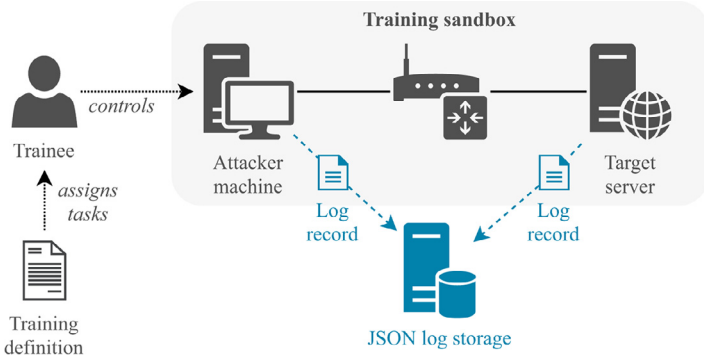

Fig. 1. The training workflow, including the collection of command logs.

```
{
    "timestamp_str" : "2021-02-22T12:51:30.296Z",
    "cmd"           : "nmap -sV 172.18.1.5",
    "cmd_type"      : "bash-command",
    "username"      : "root",
    "hostname"      : "attacker",
    "ip"            : "10.1.135.83",
    "wd"            : "/home",
    "sandbox_id"    : "9",
    "pool_id"       : "1",
}
```

Listing 1. A single log record representing a command submitted by the training participant.

Table 1
Per-trainee statistics of commands submitted by the 175 trainees.

Command type	Min	Max	Median	Avg	Stdev	Total
Bash/ZSH shell	4	266	52	62.6	50.0	10956
Metasploit shell	0	213	0	14.2	25.8	2490
All types of shell	4	358	63	76.8	55.7	13446

date and time and is attended by a certain number of trainees. Thus, it corresponds to a single training session.

1.3. Data properties

The typical training session lasted up to two hours (the median difference between the first and last command was one hour and 13 minutes). Table 1 presents the descriptive statistics of the collected dataset. For each of the 175 command history files, we counted the number of Bash/ZSH and Metasploit commands. Then, we computed their properties both separately and jointly.

The trainees submitted approximately 77 commands on average. Given the training duration, this seems appropriate because the trainees had to read the documentation and contemplate their approach during the training.

Overall, the dataset features 586 unique Bash/ZSH tools and 41 unique Metasploit tools. However, only 107 and 18 were used at least five times, suggesting that most of them were typos. Table 2 shows the top ten most frequently used tools across the whole dataset. The

Table 2
The most used commands. The command `sudo` was ignored for this table since it should not be counted as a tool.

Command (tool)	Type	Usage count	Unique argument combinations
`ls`	Bash/ZSH	2291	140
`cd`	Bash/ZSH	1714	292
`nmap`	Bash/ZSH	779	206
`set`	Metasploit	771	326
`ssh`	Bash/ZSH	715	180
`fcrackzip`	Bash/ZSH	652	164
`scp`	Bash/ZSH	640	331
`cat`	Bash/ZSH	395	199
`use`	Metasploit	280	86
`show`	Metasploit	277	17

commands with these most common tools represent approximately 63.3% of the whole dataset (8514 records out of 13446).

Other notable properties of the dataset are:

- *Attribute values are not guaranteed to be unique.* The data fields described in Listing 1 might not be unique across the whole dataset. For example, there may be two different JSON files with the same `sandbox_id`, but this does not mean that the trainee was the same. Nevertheless, the dataset archived on Zenodo [7] features unique `sandbox_id`s.
- *When processing the data line-by-line, timing is not guaranteed to be preserved.* The lines within a single JSON file might not be ordered chronologically when commands gathered from different machines (hosts) are interleaved. Even though the machines have synchronized time, they may send the commands to the central storage at different times.
  Consider the example in Listing 2, where the two commands have been stored in the given order, but the second one was submitted 30 seconds before the first one. So, the data are not sorted or reordered upon their arrival to the storage.

```
{
    "timestamp_str" : "2020-07-15T08:14:08.184511Z",
    "hostname"      : "webserver",
    "cmd"           : "ifconfig",
    ...
}
{
    "timestamp_str" : "2020-07-15T10:13:38.088960+02:00",
    "hostname"      : "attacker",
    "cmd"           : "nmap 172.18.1.5",
    ...
}
```

Listing 2. An example of command records stored successively but not chronologically.

  Therefore, analysts should always consider the actual value of the `timestamp_str` attribute (including the time zone) instead of relying only on the order of the lines.
- Some log entries are sequenced in rapid succession (for example, 20 records within one second). These are valid entries often indicating that the trainee copied and pasted multi-line strings into the terminal. They can also indicate a brute-force approach.

Table 3
Seven hands-on cybersecurity trainings that comprise the dataset.

Training name	Participants / Commands count	Key command-line tools	Notes / Training background
Junior hacker	18 / 1711	`nmap`, `scp`, `ssh`, `fcrackzip`	The simplest, introductory training
Junior hacker adaptive	58 / 4216	`nmap`, `scp`, `ssh`, `fcrackzip`	The Junior hacker training with tasks experimentally adapting to the trainees' skill level
Kobylka 3302	46 / 3994	`nmap`, Metasploit modules, `john`, `ssh`	Exploitation of CVE-2019-15107 (Webmin)
Secret laboratory	13 / 1262	`nmap`, Metasploit modules, `john`, `ssh`	A variation on the Kobylka 3302 training
Webmin exploit practice	10 / 824	`nmap`, Metasploit modules, `john`, `ssh`	A variation on the Secret laboratory training
House of cards	22 / 1261	`nmap`, Metasploit modules, `ssh`	Exploitation of CVE-2018-10933 (libssh)
SQL injection	8 / 178	`sqlmap`, `ssh`	Features also graphical tools, so there are fewer commands recorded

1.4. Structure of the data repository

The Zenodo data repository [7] is structured into seven folders. Each folder corresponds to one training described in Table 3. Each folder contains JSON files with the raw command-line data captured from that training. In order to provide detailed context to the data, each training includes its sandbox definition, and in some cases, its training definition as well. This way, the training can be further used to generate new data.

2. Methods

This section explains the format and content of the cybersecurity training, the participants' background, and data collection. We also discuss related datasets. Privacy and ethical considerations are featured in a separate section.

2.1. Cybersecurity training format

In each training session, the trainee controls a virtual machine that runs Kali Linux: a penetration testing distribution that provides the necessary command-line tools. The trainee completes a sequence of assignments that mostly involve attacking one or more vulnerable networked hosts, though some assignments feature defensive or analytical actions as well. The hosts in the training sandbox are emulated and isolated from the outside network. Almost all the assignments are solved using command-line tools in Bash, ZSH, or Metasploit shell.

2.2. Interactive learning environments for the training

The virtual machines for the training sessions from which we collected the data were hosted in one of two interactive learning environments: KYPO Cyber Range Platform (CRP) [2,8] or Cyber Sandbox Creator (CSC) [2,9]. KYPO CRP is a cloud-based infrastructure for emulating complex networks, while CSC is a tool for creating lightweight virtual labs hosted locally on the trainees' computers.

Table 4
Abstract operations that can be implemented and executed on the dataset.

Operation name	Definition	Example
`desc_stats(cmds)`	Compute descriptive statistics of commands `cmds`.	The average number of submitted commands per trainee is 76.8 (see also Table 1).
`top_n_tools(cmds, n)`	Show the `n` most used tools among `cmds`, sorted by their usage count.	The most used tool is `ls` with 2291 occurrences (see also Table 2).
`arg_comb(cmds, tool)`	For a given `tool`, show all combinations of its used arguments.	For the `nmap` tool, the following arguments were used: `-sn` (10×), `-p 20` (8×), `--help` (7×).
`time_gap(cmds, x, y)`	Compute the time difference between the `x`-th and the `y`-th command from `cmds`.	The time between the first and the last command in `sandbox-789-useractions.json` is 59 minutes and 21 seconds.
`eq_classes(cmds, gr)`	Categorize `cmds` into equivalence classes based on their syntactical grammar `gr`.	`nmap 1.2.3.4 -sV == nmap -sV 1.2.3.4`, `nmap 1.2.3.4 -p 9 != nmap 1.2.3.4 9 -p`.

The choice of the used learning environment did not affect the training content, and the data collection was equivalent. The deployment of two learning environments demonstrated the flexibility of the data logging toolset. Moreover, both environments are open-source, and anyone can freely use them for their needs [2].

2.3. Training participants

From August 2019 to July 2021, we hosted 27 cybersecurity training sessions for a total of 175 trainees. Each training session usually took two hours to complete. Some of the sessions were held on-site at our university premises; others were remote due to COVID-19 restrictions. The participants included (sorted from the most to the least represented):

- undergraduate and graduate students of computer science from various universities,
- selected high school students, finalists of the national cybersecurity competition, and
- cybersecurity professionals.

All of them attended the training sessions voluntarily because of their interest in cybersecurity and were not incentivized. Although self-selection bias may be present, the sample represents a broad range of cybersecurity students, experts, and enthusiasts.

2.4. Data collection

To acquire the data, we developed an open-source toolset for collecting shell commands [1]. As the trainees solve the training assignments, the toolset automatically acquires their submitted commands and the associated metadata. Then, the data are formatted as JSON records and stored in dedicated storage.

2.5. Queries for exploratory data analysis

To facilitate analyzing the data, Table 4 provides a set of analytical queries, standard operations that can be executed on them. The queries result from our educational data mining and learning analytics research. They can be implemented in a way that suits the analysts' needs; as an example, we provide a basic implementation in ELK along with the data in Zenodo [7]. It enables to import the dataset, process it, and analyze it.

```
(
  (unix OR linux) AND command AND (dataset OR data set) AND collect
)
OR
(
  (cybersecurity OR cyber security) AND (education OR training
    OR exercise) AND (dataset OR data set)
)
```

Listing 3. A query submitted to Google Scholar on July 15, 2021 to search for related papers.

Table 5
An overview of related published datasets sorted from the most recent within each category.

Type	Dataset description	Dataset size	Availability
shell cmds	Shell commands with their textual explanation	12609 commands	public [13]
	Shell commands only, no metadata	750000 commands	public [14]
	Shell commands only, no metadata	303628 commands	on request [15]
cyber training	Traffic and event logs from yearly competitions	(varies each year)	public [16]
	Traffic and event logs from a cyber exercise	275 MB (compressed)	public [17]
	Event logs from a cyber exercise	135 GB (compressed)	on request [18]

2.6. Related work

Linux commands have been collected for research purposes for decades [10], but not in the cybersecurity context. At the same time, various datasets (not only shell commands) from cybersecurity exercises are a crucial source of evidence for educational research [11,12]. However, mostly packet captures and system logs have been previously collected from cybersecurity training sessions, and few datasets remain available today.

To review related work, we searched for publications indexed on Google Scholar using the query in Listing 3. Table 5 lists the few examples we discovered. In comparison with the related work, our dataset intersects and bridges the two domains by collecting shell commands from hands-on cybersecurity training. We incrementally collected the commands over several training sessions with human participants. Therefore, the dataset fills the discovered gap and represents an original contribution for the community of computing or educational researchers. We believe that others will find value in it and use it to foster further research and development.

Ethical Statement

Before conducting the training sessions, we discussed the data collection issues with the institutional review board of our university. We obtained a waiver from the ethical committee since we intentionally do not collect any personally identifiable information that could reveal the trainees' identity. The data are anonymous and cannot be linked to specific individuals. As a result, even if one person attended more training sessions, it is impossible to track him/her throughout several training sessions and compare his/her past performance.

The trainees agreed to the anonymized data collection for research purposes via informed consent before starting the training. We ensured they would not be harmed or negatively affected by the research, and they had the right to stop participating at any time without any restrictions.

After collecting the data, we manually checked them for personal information to avoid any privacy issues. If the trainees typed any personal information in the command-line, we anonymized it, though such occurrences were sporadic. Currently, we have no fully automated solution for this, since trainees can type anything in the command line. Other than that, no changes were made to the raw data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships which have, or could be perceived to have, influenced the work reported in this article.

CRediT Author Statement

Valdemar Švábenský: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Visualization, Project administration; **Jan Vykopal:** Methodology, Software, Validation, Investigation, Resources, Data curation, Writing – review & editing; **Pavel Seda:** Methodology, Software, Validation, Investigation, Resources, Data curation, Writing – review & editing; **Pavel Čeleda:** Methodology, Writing – review & editing, Supervision, Funding acquisition.

Acknowledgments

This research was supported by the ERDF project CyberSecurity, CyberCrime and Critical Information Infrastructures Center of Excellence (No. CZ.02.1.01/0.0/0.0/16_019/0000822).

References

[1] V. Švábenský, J. Vykopal, D. Tovarňák, P. Čeleda, Toolset for Collecting Shell Commands and Its Application in Hands-on Cybersecurity Training [in press], in: Proceedings of the 51st IEEE Frontiers in Education Conference, in: FIE '21, IEEE, New York, NY, USA, 2021, pp. 1–9. https://www.muni.cz/en/research/publications/1783801
[2] J. Vykopal, P. Čeleda, P. Seda, V. Švábenský, D. Tovarňák, Scalable Learning Environments for Teaching Cybersecurity Hands-on [in press], in: Proceedings of the 51st IEEE Frontiers in Education Conference, in: FIE '21, IEEE, New York, NY, USA, 2021, pp. 1–9. https://www.muni.cz/en/research/publications/1783808
[3] Q. Vinlove, J. Mache, R. Weiss, Predicting student success in cybersecurity exercises with a support vector classifier, J. Comput. Sci. Coll. 36 (1) (2020) 26–34, doi:10.5555/3447051.3447055.
[4] C. Ramey, B. Fox, Bash Reference Manual, Version 5.1, 2020, Accessed: 2021-09-20, https://www.gnu.org/savannah-checkouts/gnu/bash/manual.
[5] Robby Russell & Contributors, Oh My Zsh, 2020, Accessed: 2021-09-20, https://ohmyz.sh.
[6] Offensive Security, Metasploit Unleashed, 2021, Accessed: 2021-09-20, https://www.offensive-security.com/metasploit-unleashed/.
[7] V. Švábenský, J. Vykopal, P. Seda, P. Čeleda, Dataset: Shell Commands Used by Participants of Hands-on Cybersecurity Training, 2021, doi:10.5281/zenodo.5517479
[8] Masaryk University, KYPO Cyber Range Platform, 2021a, Accessed: 2021-09-20, https://www.kypo.cz.
[9] Masaryk University, Cyber Sandbox Creator, 2021b, Accessed: 2021-09-20, https://gitlab.ics.muni.cz/muni-kypo-csc/cyber-sandbox-creator.
[10] R.E. Kraut, S.J. Hanson, J.M. Farber, Command Use and Interface Design, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, in: CHI '83, Association for Computing Machinery, New York, NY, USA, 1983, pp. 120–124, doi:10.1145/800045.801594.
[11] K. Maennel, Learning Analytics Perspective: Evidencing Learning from Digital Datasets in Cybersecurity Exercises, in: 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS PW), 2020, pp. 27–36, doi:10.1109/EuroSPW51379.2020.00013.
[12] J. Garae, R.K.L. Ko, J. Kho, S. Suwadi, M.A. Will, M. Apperley, Visualizing the New Zealand Cyber Security Challenge for Attack Behaviors, in: 2017 IEEE Trustcom/BigDataSE/ICESS, 2017, pp. 1123–1130, doi:10.1109/Trustcom/BigDataSE/ICESS.2017.362.
[13] X.V. Lin, C. Wang, L. Zettlemoyer, M.D. Ernst, NL2Bash: A Corpus and Semantic Parser for Natural Language Interface to the Linux Operating System, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation LREC, 2018, pp. 1–12.
[14] M. Schonlau, W. DuMouchel, W.-H. Ju, A.F. Karr, M. Theus, Y. Vardi, Computer intrusion: detecting masquerades, Statistical Science 16 (1) (2001) 58–74. http://www.jstor.org/stable/2676780
[15] S. Greenberg, Using Unix: Collected traces of 168 users(1988). doi:10.11575/PRISM/30806
[16] DEF CON, CTF Archive, 2021, Accessed: 2021-09-20, https://defcon.org/html/links/dc-ctf.html.
[17] D. Tovarňák, S. Špaček, J. Vykopal, Traffic and log data captured during a cyber defense exercise, Data in Brief 31 (2020), doi:10.1016/j.dib.2020.105784.
[18] N. Munaiah, J. Pelletier, S.-H. Su, S. Yang, A. Meneely, A Cybersecurity Dataset Derived from the National Collegiate Penetration Testing Competition, in: Hawaii International Conference on System Sciences, 2019, pp. 1–6.