

**M U N I**  
**F I**

# **Process Mining Analysis of Puzzle-Based Cybersecurity Training**

Martin Macak, **Radek Oslejsek**, Barbora Buhnova

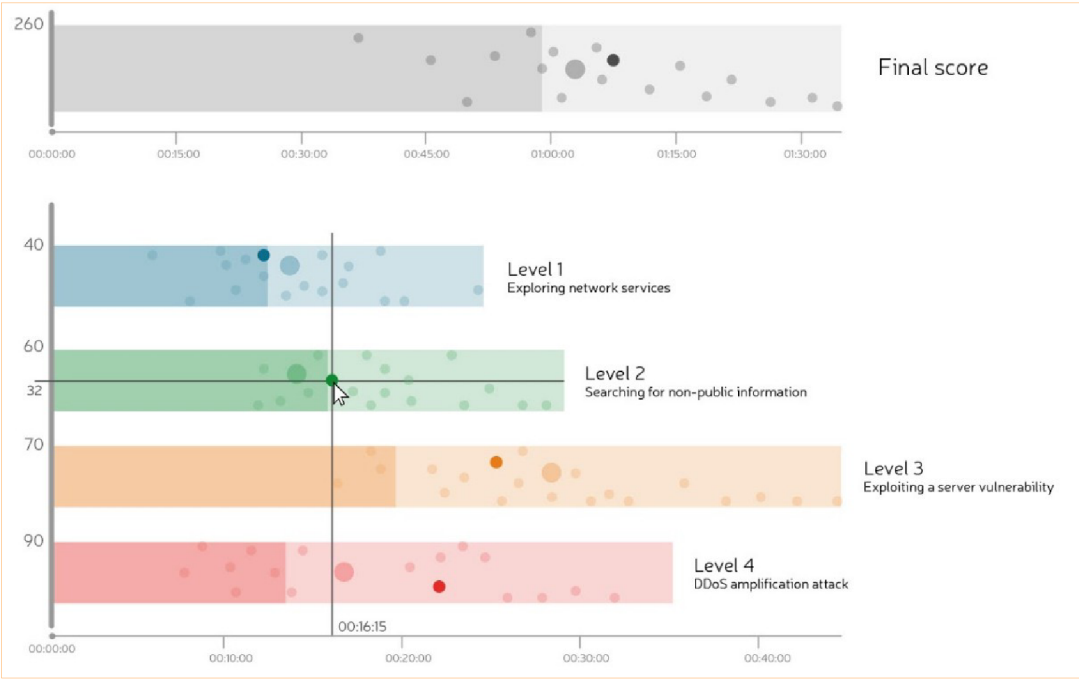
# Problem statement

## Cyber range

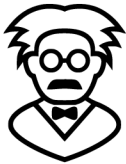
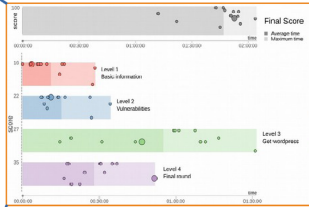


# Problem statement

## Cyber range



## Learning analytics



Instructor

# Problem statement

Cyber range



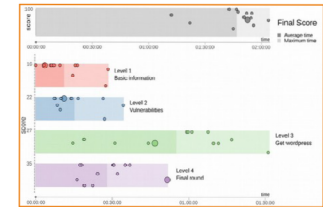
Event logs

**No tangible output, only sparse events:**

```
User1;2.08.2020 10:31:43;use webmin_backdoor
User1;2.08.2020 10:32:44;set RHOST
User1;2.08.2020 10:33:19;set LHOST
User1;2.08.2020 10:34:27;set SSL
User1;2.08.2020 10:34:35;set TARGET
User2;2.08.2020 10:32:17;use webmin_backdoor
User2;2.08.2020 10:32:43;exploit
User2;2.08.2020 10:44:33;set RPORT
User2;2.08.2020 10:45:21;exploit
User2;2.08.2020 10:56:02;set LHOST
User2;2.08.2020 10:56:20;set SSL
User2;2.08.2020 10:58:35;set TARGET
```

...

Learning analytics



Instructor

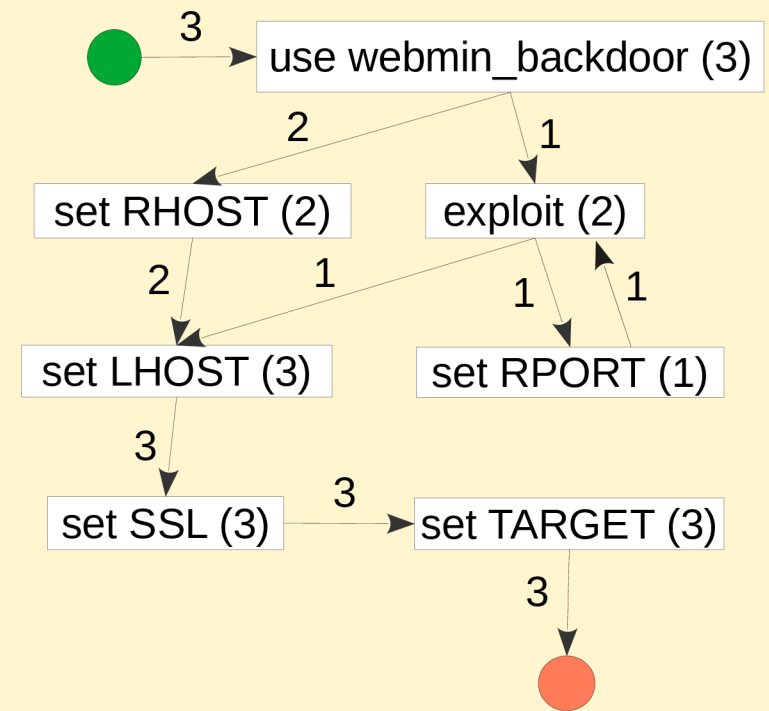
# Idea: Transfer events into process graphs

## Events log

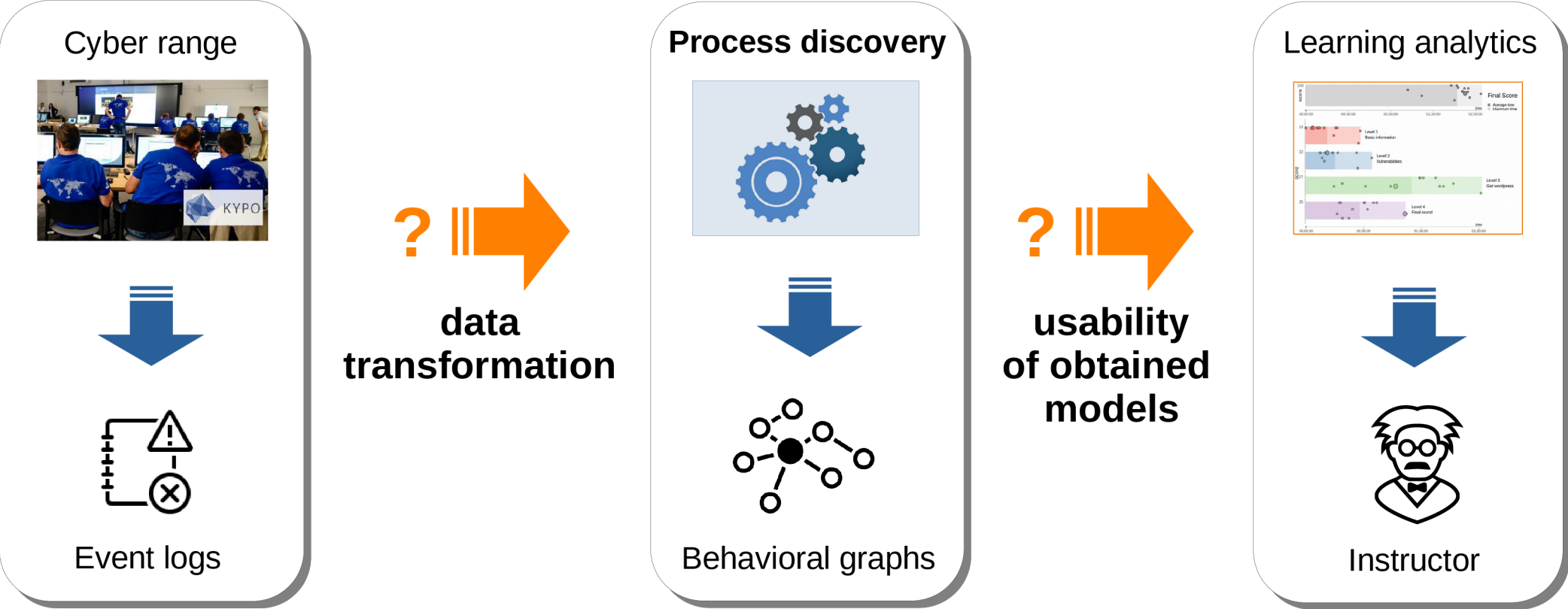
User1;2.08.2020 10:31:43;use webmin\_backdoor  
User1;2.08.2020 10:32:44;set RHOST  
User1;2.08.2020 10:33:19;set LHOST  
User1;2.08.2020 10:34:27;set SSL  
User1;2.08.2020 10:34:35;set TARGET  
User2;2.08.2020 10:32:17;use webmin\_backdoor  
User2;2.08.2020 10:32:43;exploit  
User2;2.08.2020 10:44:33;set RPORT  
User2;2.08.2020 10:45:21;exploit  
User2;2.08.2020 10:56:02;set LHOST  
User2;2.08.2020 10:56:20;set SSL  
User2;2.08.2020 10:58:35;set TARGET  
...



## Process model



# Research Questions



# RQ1: Required training events

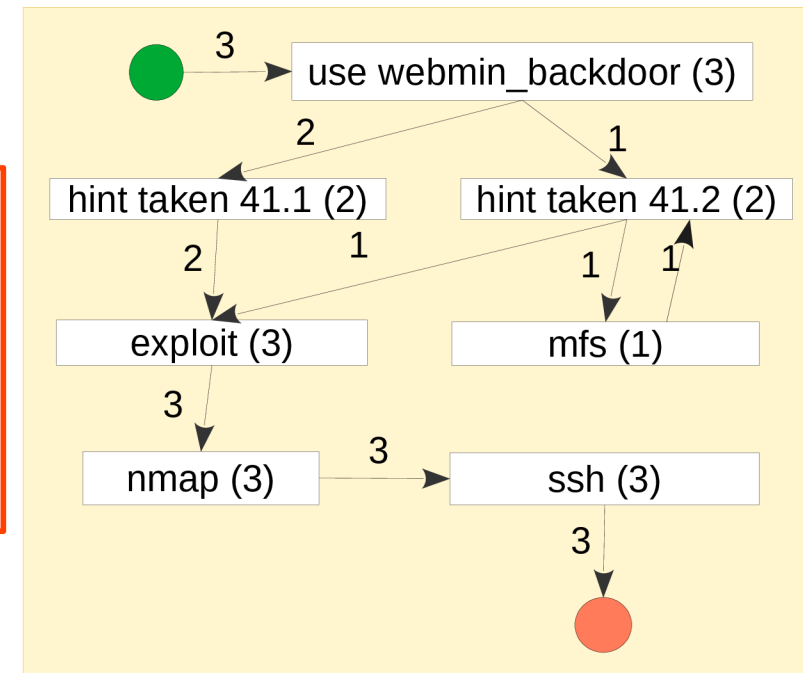
```
"hostname":"attacker","ip":"10.1.135.83","timestamp_str":"2022-05-04T10:12:09",
"sandbox_id":"664","cmd":"ping 172.18.1.5", "pool_id":"73", "wd":"/home/kali",
"cmd_type":"bash-command","username":"kali"
```

- **Problem:** Process mining methods have strict data requirements.
- **RQ:** How to convert cyber training data to the format suitable for process mining?
- **Solution:** We proposed a unified data mapping

*maps to nodes*

*maps to paths*

EVENT TYPE	EVENT	EVENT PARAMETERS	TIMESTAMP	TRAINEE
game	Hint taken 41.1		10:16:11	User 1
game	Hint taken 41.2		10:16:34	User 1
metasploit	exploit	-j	10:18:23	User 2
bash	nmap	-sL 10.1.26.9:5050	10:32:16	User 1



# RQ2: Process discovery

- **Problem:** Data characteristics affect the process of graph generation and then the overall practical usability for educational process mining.
- **RQ:** What are the obstacles in the process discovery phase, and how to overcome them?
- **Approach:** We analyzed data from our cyber range, aiming to identify the most significant features according to the [R. Cristóbal et al.] **classification of significant challenges** that appear when using event logs for educational process mining.
- Analyzed data sets:

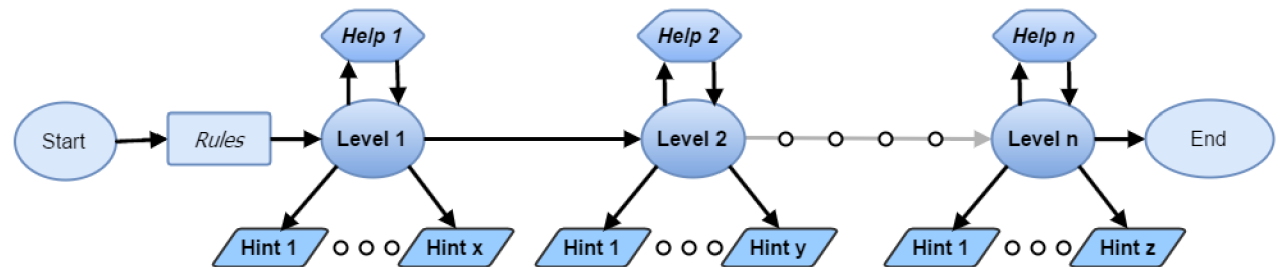
	MIN	MAX	AVG
length (minutes)	65	210	119
# trainees	4	20	10
# events per session	370	3000	1100
# events per trainee	53	150	108

# scenarios	6
# sessions	16



# Noise in the data

- **Description:** Exceptional behavior, which is not representative of typical behavior.
- **Observations:** Many students leave the training unfinished due to a lack of time or a loss of motivation.
- **Discussion:**
  - No problem for **puzzle-based capture-the-flag games** that provide clear milestones. Therefore, it is easy to spot this situation in process graphs and further investigate the reason.
  - Can pose a problem for other loosely conceived types of cyber security training programs, e.g., cyber-defense exercises (CDX)



# Data incompleteness

- **Description:** Incomplete data in a data set can produce biased process models.
- **Observations:** The problem **often appears** because modern cyber ranges are complex, distributed with asynchronous communication, running on underlying virtualization, and then unreliable. Moreover, it is usually very **difficult to notice** from process graphs that there is something wrong with the raw event logs.
- **Discussion:**
  - The problem is tightly connected to the specific measurement infrastructure of the cyber range.
  - Data cleansing and completeness checking have to be done in the preprocessing phase of the analytical workflow, which makes the automation of data processing difficult.

# Timestamps and events ordering

- **Problem description:** Incorrect order of event logs produces biased process models.
- **Observations:** We identified three major problems with log ordering:
  - **Not sufficiently synchronized data sources.** Only a small shift in times can re-order events and then produce significantly different process models.
  - **Trainees start the exercise at different times**, even if they sit in the same classroom. Their actions have to be synchronized relative to the “start of the exercise” to be mutually comparable.
  - **Suspended playing** introduces inactivity spans that make multiple trainees incomparable.
- **Discussion:**
  - **Puzzle-based training** provides milestones, e.g., the *start of the game*, that can be used for synchronization.
  - **Time-limited in-class training** naturally avoids idle time.
  - **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.

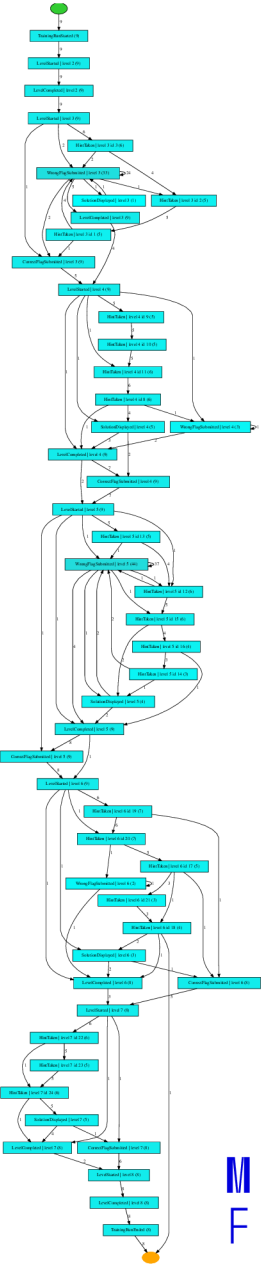
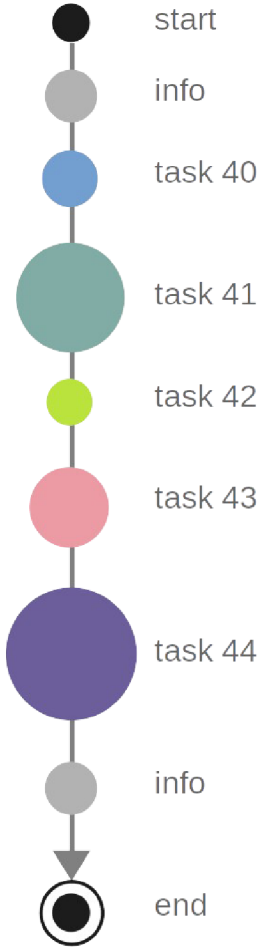
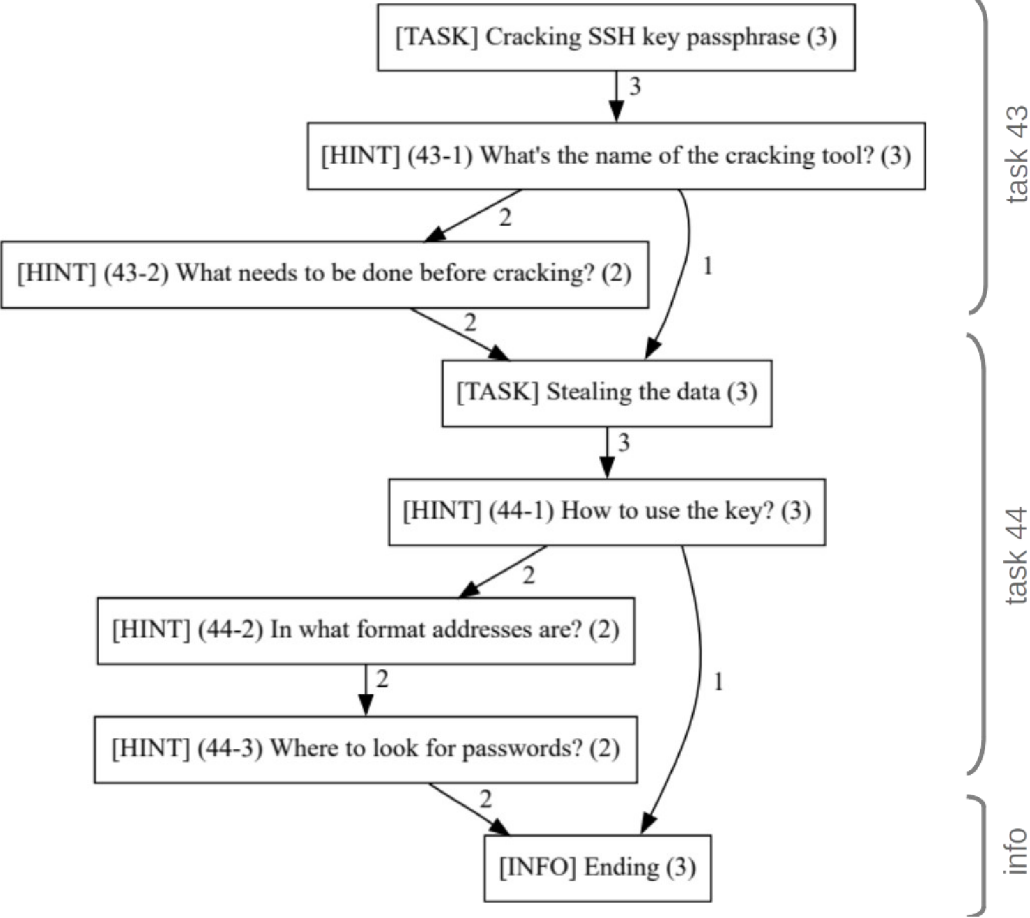
# Data size

- **Description:** 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, the comprehensibility of produced models can quickly decrease with higher amounts of data.
- **Observations:** The speed of PM algorithms is not the problem, but the complexity of process models is.
- **Discussion:**
  - **Fewer trainees** produce fewer data. As **in-class training** sessions limit the number of participants, their process graphs are simpler than graphs from online training programs with an unlimited number of participants.
  - **Simpler training** produces less data because it restricts the number of possible solutions (paths in process graphs). Therefore, CtF games are more suitable than CDX.
  - However, even in-class CtF games can produce too complex graphs.

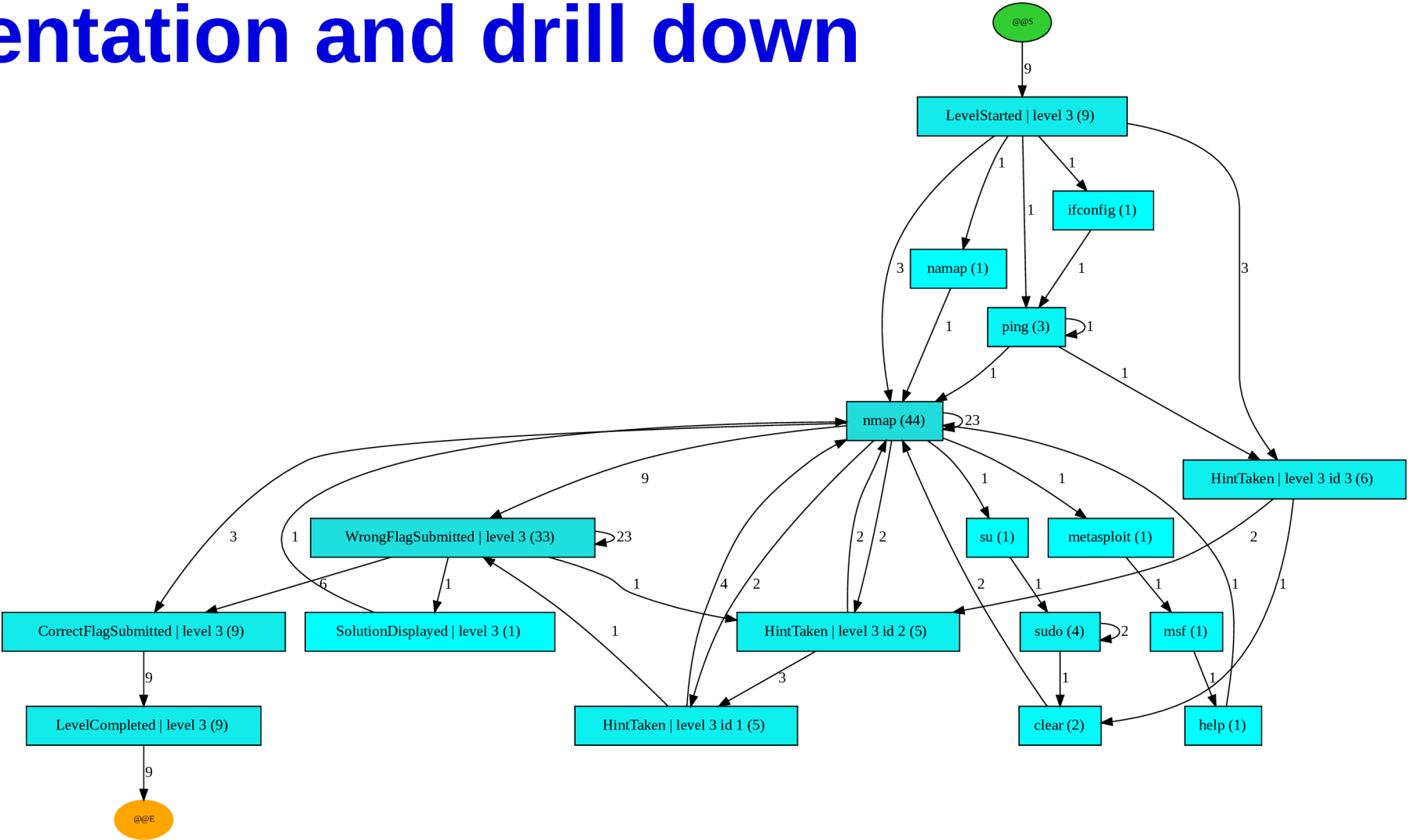
# RQ3: Exploratory analysis

- **Problem:** Even a limited CtF games can produce too complex graphs.
- **RQ:** How to deal with the complexity of process graphs during analysis?
- **Approach 1:** Filtering driven by our unified data abstraction
- **Approach 2:** Puzzle-based fragmentation and drill down

# Fragmentation and drill down



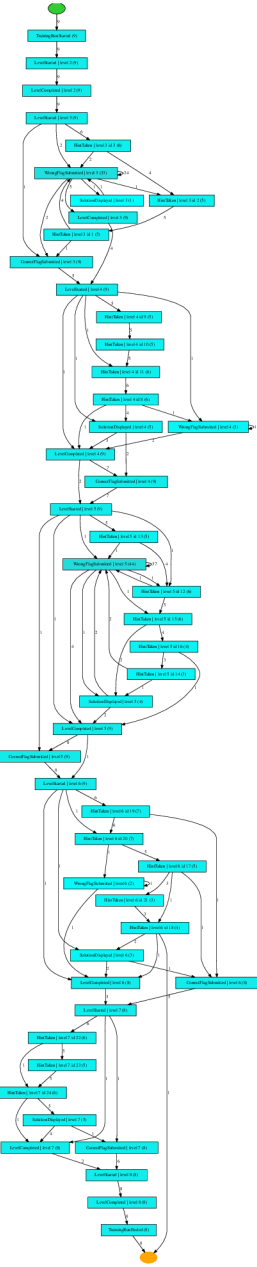
# Fragmentation and drill down



# Filtering by event types

EVENT TYPE	EVENT	EVENT PARAMETERS
game	Hint taken	4.1
game	Hint taken	4.3
bash	ssh	192.168.1.10
bash	ssh	192.168.1.11

- Different event types define different level of abstraction
  - Game events = high-level abstraction
  - Bash command = middle-level abstraction
  - Details of tool usage = low-level abstraction

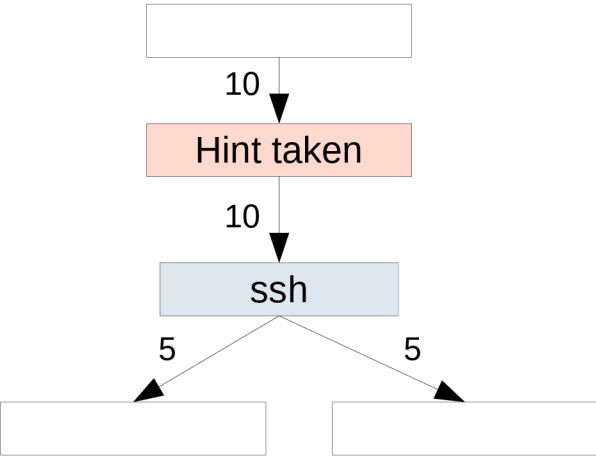
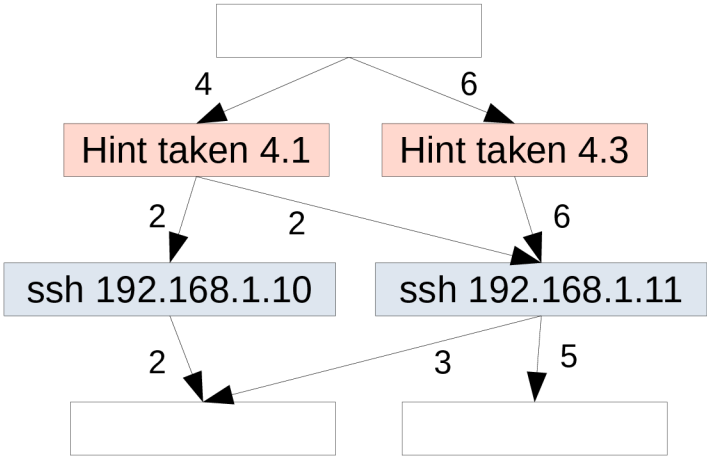




# Filtering by event details

EVENT TYPE	EVENT	EVENT PARAMETERS
game	Hint taken 4.1	
game	Hint taken 4.3	
bash	ssh 192.168.1.10	
bash	ssh 192.168.1.11	

EVENT TYPE	EVENT	EVENT PARAMETERS
game	Hint taken	4.1
game	Hint taken	4.3
bash	ssh	192.168.1.10
bash	ssh	192.168.1.11



# Conclusion and future work

- **Summary:**
  - Process discovery is usable and useful **if we restrict ourselves to in-class puzzle-based training sessions** (i.e., popular cyber security capture-the-flag games).
  - For other types of training programs, the practical usability remains an open question.

- **Future work:**
  - To provide complementary interactive views to traditional graphs

