Managing Learning and Difficulty Satisfaction in Cybersecurity Training Challenges

Lovisa Nyholm[1], Teodor Sommestad[1 [0000−0002−2606−4139]], Henrik Karlzén[1 [0009−0009−4394−7695]], and Hanna Kvist[1]

Swedish Defence Research Agency FOI, Linköping, Sweden {lovisa.nyholm, teodor.sommestad, henrik.karlzen, hanna.kvist}@foi.se
https://www.foi.se

Abstract. This paper describes Skade, a web-based system used to expose technical challenges in a cyber range to trainees, without the involvement of instructors or cyber range engineers. Skade has been evaluated by 45 prospective trainees (e.g., junior threat hunters), each spending 2-5 hours on one of three different threat hunting challenges. The trainees' assessed Skade to be acceptable on the System Usability Scale. Skade is designed to manage difficulty and learning using optional clues, but contrary to our hypotheses, the availability of clues did not affect trainees' satisfaction with difficulty or perceived learning. Meanwhile, many trainees reported feedback concerning the difficulty level and getting stuck on details. It is concluded that managing the difficulty level of technical educational cyber challenges is important, and remains to be solved.

Keywords: cyber threat hunting · cyber defence exercises · training · cyber range.

1 Introduction

Cybersecurity competence remains a scarcity in Europe [10] and the US [27]. The European Union Agency for Cybersecurity (ENISA) stated that "While the demand for people with ICT and cybersecurity skills is rapidly increasing, the cybersecurity skills and talent shortage is growing too" [10]. The field requires a wide range of competencies, among which the role of the cybersecurity analyst is particularly in demand. A report by OECD [26], which analysed the job market in Poland, France and Germany, found that "both France and Germany have seen an increase in the demand for cybersecurity analysts, individuals providing insights to support planning, operations, and maintenance of systems security."

Technical competencies like these require both a particular mindset and technical skills. This is especially true for the activity of cyber threat hunting. Threat hunting has been described as a "highly unstructured task that demands deep technical know-how, data analytics savvy, and out of the box thinking" [24]. A recent survey by SANS [32] found that 61 percent of respondents across the globe

mention workforce limitations as a problem for threat hunting, and 40 percent are increasing their staffing investment for threat hunting.

Proficiency in complex cybersecurity tasks such as threat hunting cannot be developed through theoretical instruction and lectures alone [7,18,19,11]. At the same time, training these skills in operational environments is challenging. When hunting real threats, there is limited opportunity for experimentation, while conducting exercises in production systems may negatively affect business processes. The use of isolated training environments such as cyber ranges is an option, and similar to the separation of development, test and production environments, as recommended by ISO 27001 Annex A 8.31. Nevertheless, designing meaningful technical challenges in cyber range training environments requires considerable effort. First, because threat hunting involves active exploration of systems, training environments must be sufficiently complex and realistic to replicate a meaningful portion of the conditions and options present in real networks. Access to good hands-on labs has been identified as a barrier to learning in the cybersecurity field [2]. Second, because the tasks are complex and varied, adaptation of challenges to trainees' proficiency levels is difficult. Balancing the difficulty level is known to be important [15].

Skade [33] is a system designed to address these challenges. It automates the instantiation of cyber defence challenges in a cyber range, removing manual challenge setup and the need for human instructors. Skade's library of challenges can also be filled with optional clues to trainees. These clues may provide the trainee with threat intelligence, technical tips, or mindset guidance. This allows trainees to dynamically lower the challenge barrier when needed, without breaking the threat hunting workflow. The availability of clues is expected to increase trainee satisfaction with the difficulty and perceived learning, and these factors are expected to be associated with higher usability of Skade.

In this paper, we present Skade and tests of the theories that the system is built upon. More specifically, we empirically test four hypotheses that link satisfaction of the difficulty level, usability, and perceived learning:

H1. Satisfaction with difficulty level is associated with higher usability.
H2. Perceived learning is associated with higher usability.
H3. The availability of investigative clues increases satisfaction with difficulty.
H4. The availability of investigative clues increases perceived learning.

The evaluation target system administrators and novice threat hunters.

Section 2 presents the challenge management system Skade. Section 3 presents related works. Section 4 describes the evaluation method. Section 5 presents the results. Section 6 discusses these results, and Section 7 concludes the paper.

2 The Challenge Management System Skade

The sections below introduce the design principles Skade is built upon, Skade's main components, and three threat hunting challenges used in the test.

2.1 Design principles

Skade was developed based on four ideas, drawn from pedagogical research: ensuring constructive alignment, covering multiple learning dimensions, providing feedback and assessment, and supporting a motivating setting. These ideas were converted to a set of seven features (in the earlier development of Skade [33]).

None of the ideas or features are novel or controversial. For instance, most hands-on training covers multiple learning dimensions (e.g., abstract conceptualization and active experimentation) and it is common to provide feedback and assessments to trainees (e.g., if they were successful).

The focus of this research is the feature *Enabling optional recommendations and hints at all stages*. This feature was derived from the expectation that the availability of clues has a strong link to trainee motivation to learn. This was based on ideas such as the six Cs [35], three of which suggest that educational motivation is influenced by: choice, challenge, and control. Although the framework was initially developed for children, research indicates that it is also relevant and applicable to teaching in higher education [30].

2.2 Main Components

The main components of Skade are a front-end user interface written in Angular, a back-end written in Python, and Json-specifications that detail the challenge and its chosen configurations of the environment and threat actor. Fig. 1 shows the main components and integrations of Skade.

Skade is built on top of the cyber range platform Crate [12]. It uses the Crate API to instantiate the cyber range specified using previous snapshots. It also uses the APIs of Sved [14] and Lore [13] to start simulated users and threat agents in the cyber range. Threats can be simulated with dynamic features that allow trainees to train over and over again on the same challenge but with small changes to the challenge. For Sved this is solved by using a dynamic configuration where, for example, filenames and attacked machines can differ. Lore is ML-based and will always perform slightly different actions each run.

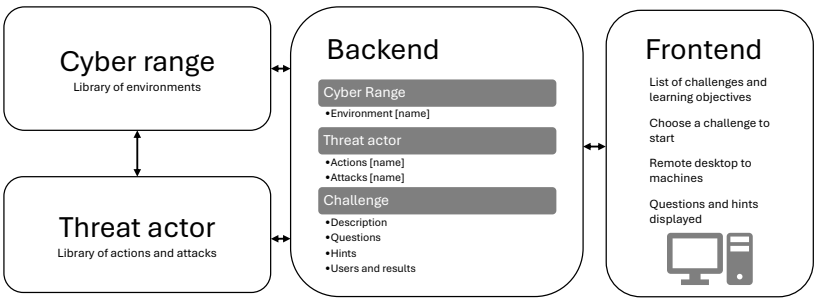


Fig. 1: The main components and integrations of Skade.

An important part of the Json-specifications are the questions used for the challenge. These include response alternatives, details on how to check answers, when the question should be displayed to the trainee, learning objectives, and hints. Most of this is prepared in advance, while parts of are dynamically generated at runtime, e.g., by interacting with the logs of the threat agents to get the correct answer for questions. In the code snippet below, the *segments_affected* and *correct_segments* are references to Python-methods extracting data from a Sved log:

```
{
    "name": "Looking for network traffic",
    "text": "From which segments in Curea can you identify the malicious traffic?",
    "alternatives":{
      "response_type": "MULTIPLE_ANSWER",
      "alternatives": {
        "0": "HMI (10.100.1.0/24)",
        "1": "HQCLIENT (198.51.100.0/24)",
        "2": "SCADA (10.100.7.0/24)",
        "3": "SRV (203.0.113.0/24)"
      }
    },
    "answer":{
      "correct_answer": "segments_affected",
      "correction_method": "correct_segments"
    }
 }
```

2.3 Threat Hunting Challenges

In our tests we cover three pairs (or packages) of threat hunting challenges. The packages were separated into two distinct parts because these parts were manipulated differently in the experiment. During the study, each trainee completed one of the three packages. These are briefly described below, with expected time for a trainee in parentheses:

- Package A: The first challenge (2 hours) focused on baselining of a small military network with 30 machines using OpenSearch; the second challenge (3 hours) had the trainee threat hunt in the network.
- Package B: Both challenges focused on intelligence driven threat hunting in a small company with 40 machines. The first (2 hours) focused on host logs; the second (2 hours) on the use of a SIEM system.
- Package C: The challenges (1 hour + 1 hour) were theoretical and asked the trainees questions about typical situations and proper reasoning during threat hunting.

We here give some examples of how Skade represents the challenges. The examples are drawn from the second part of package A, a challenge inspired by the military cyber threat scenario described in [34].

Trainees get access to challenges after a scheduling stage. During the scheduling stage, each trainee is offered multiple challenges to choose from, and each challenge is displayed with a short description and its learning objectives. Our example challenge uses these learning objectives: *uncover new patterns and TTPs*, and *investigate via tools and techniques*.

Each challenge includes background information and network data, presented to the trainee at the start. In our example, the trainee is provided with a network

map containing 30 machines and is advised to focus on accessing the SIEM server with full log access. The example challenge has a background text of approximately 500 words, detailing assets to protect, the threat environment, and how to safely access the network SIEM system.

Trainees connect to machines in the cyber range with RDP via Skade, running in their browser. When working on the challenge, trainees are presented with investigative questions that guide them toward identifying baseline and anomalous behaviors. Hints, when available, may be requested at any time. One question and associated hints are shown in Fig. 2. The first hint describes baseline behavior and reminds the trainee of results from the first challenge in the package. The second and third hints guide the trainee toward investigating the relevant machines.

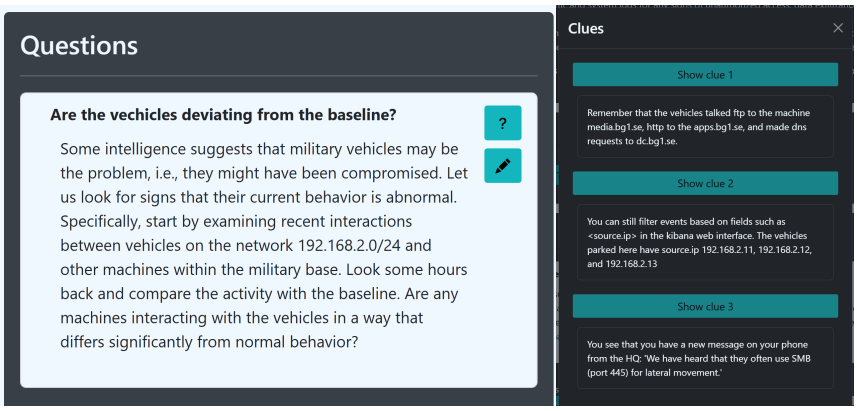


Fig. 2: An example of a question from package A and the hints associated with it.

After completion of the challenge, trainees unlock a reflecting assignment, specifically designed to ask follow-up questions.

3 Related Works

The sections below describe research about how learning depends on training at a satisfactory difficulty level, as well as on how the difficulty level is managed in practice.

3.1 The Importance of the Difficulty Level

The use of simulated cyber environments to support learning is common [6,11]. The importance of managing the difficulty level in such trainings is widely recog-

nized among cybersecurity researchers. It is reasonable to expect that the suitable difficulty level depends on the proficiency of the trainee. Indeed, Vykopal et al. write that "matching learners proficiency and exercise difficulty is a key success factor of the whole exercise" [37]. Additionally, when Pfaller et al. [28] list challenges associated with cyber exercises the first one enumerated is "adaption for different skill levels".

The idea that the training outcomes are improved if the training is adapted to trainees' proficiency levels is by no means unique to cybersecurity. Alessi [1] has formulated a general hypothesis concerning the relationship between simulation fidelity, trainee proficiency level, and learning. The hypothesis suggests that fidelity induces complexity that would become an obstacle for learning at a certain point. This has been made even sharper in research on training of aviators, where it is said that flight simulators can have too much fidelity to be optimal if the trainee is a non-expert [25].

While this is a widespread idea, we are aware of only one quantitative empirical test in the cybersecurity domain supporting the importance of managing the difficulty level [15]. This study asked trainees whether they found the exercise educational (a kind of perceived learning) and if it was challenging. Scoring either very high or very low on the latter was interpreted as a result of the wrong (unbalanced) difficulty level. The study found a medium correlation (r=0.38) between perceived learning and balanced difficulty.

More generally, perceived usability of learning systems has been suggested to influence learners' ability to acquire knowledge, as poor usability may increase extraneous cognitive load and divert attention away from the learning task itself [23]. Accordingly, the System Usability Scale (SUS) is often used in studies of educational technology across diverse contexts and user groups [36]. In our research, we measure usability of Skade using SUS. In addition, we measure trainees' perceived learning and if they are satisfied with the difficulty level.

3.2 Managing the Difficulty Level

Pfaller et al. [28] suggest that exercises should be flexible and adapted to each team or participant. However, much of the research on cyber security training does not explicitly mention any approach to managing difficulty levels. For instance, Miazi et al. [24] describe a university campus threat hunting competition. Two of three challenges were not solved by any participant, indicating the problem of finding a balanced difficulty level. Similarly, ICSTASY [20] is a "scenario-based, interactive, and immersive cybersecurity training platform", but features for adjusting the difficulty level are not mentioned in its description. Additionally, neither the paper about the platform CRACK [31] nor the documentation about the platform TopoMojo [5] mention how to manage or document challenge skill requirements.

There are some exceptions that introduce difficulty levels, although not by trainee choice. Dobson et al. [9] describe a training and exercise model called R-EACTR that manages complexity with variables such as difficulty level of

attacks, obfuscation efforts, deception, and noise. Wei et al. [39] present a university course that uses six threat hunting hands-on assignments on instrumented virtual machines. The assignments were static and categorized as low, medium, or high difficulty level, and designed for different types of students. Beuran et al. [3] propose a lab platform with automated challenge progression that activates certain components in the training based on the progress of the trainee. In other work, Beuran et al. integrate practical hands-on challenges with a learning management system and include the option of hints in systems [4]. Vykopal et al. [38]collect student actions within a virtual environment and responses submitted through a training portal. By analyzing this data, a system determines the most appropriate next task for the student, enabling adaptive learning based on performance. None of these evaluate if hints are appreciated or improve learning.

It is difficult to find details on how difficulty levels are managed in commercial platforms and services for hands-on cybersecurity training. Most offerings are lab-based and provide small, self-contained tasks that run on one or a few virtual machines, or they distribute downloadable artifacts such as logs, disk images, and PCAP files. Examples of such lab-based platforms include CyberDefenders, LetsDefend, and BlueTeamLabs Online. These challenges seem to target different proficiency levels in their design rather than adapting instructions to trainees dynamically.

Platforms such as CyberBit and Cloud Range are similar to Skade in their use of realistic networks instantiated in a cyber range, and they offer training for both individuals and teams at different levels. It is not entirely clear how the training content is adapted to differing proficiency levels. However, the idea of providing clues appears to be present. In particular, CyberBit offers an option to "allow hints on quiz" during exercise planning [8]. Consequently, their approach to difficulty adaptation includes a mechanism similar to the one evaluated in this paper.

4 Evaluation Method

As described in Section 2, Skade rests on multiple ideas concerning how cybersecurity analysts should be trained. Fig. 3 presents an overview of the study design and the measurements used. The following sections give further details on the study design, how measurements were collected, the sample (trainees) in the tests, and how the data were analyzed.

4.1 Study Design

As illustrated in Fig. 3, a within-sample crossover design was used. In this design, each trainee got two challenges in a package: one challenge with clues available and one challenge without clues. Before each trainee started their training it was randomized (50/50) if clues should be available in the trainee's first or second challenge. Bias that may be introduced by the order in which clues are offered are balanced out, because both sequences have (approximately) as many subjects.

When planning the study, we expected effect sizes close to Cohen's d=0.5, and estimated that 34 trainees would be sufficient for the test ($1-\beta$=0.80, α=0.05).

According to the ethical review practices in Sweden, a study with this design is exempt from the requirement of review by an ethical review board. Nevertheless, basic ethical requirements have been upheld by acquiring informed consent from the participants, avoiding putting pressure to participate, minimizing risk, and by following data protection procedures.

4.2 Measurements

As illustrated in Fig. 3, four questionnaires (surveys) were administered.

The first one included information about the study, questions concerning informed consent, and demographic questions (e.g., years of experience in cybersecurity).

The second and third questionnaires were identical and asked questions related to the previous challenge. It had two scales. Three items adapted from the After-Scenario Questionnaire [21] were used to measure trainee satisfaction with the difficulty level, e.g., "Overall, I am satisfied with the ease of completing the tasks in this challenge". The reliability (Cronbach's α) was good in both questionnaires (0.837 and 0.892). Four items were used to measure perceived learning. This was inspired by Bandura's self-efficacy scale but re-phrased in terms of a change in ability, to measure the situation after taking each challenge. After each challenge, we asked trainees to provide their agreement on the following statements: a) "I am more ready to do threat hunting than before", b) "I have changed the way I think about how to investigate threats", c) "I feel better prepared to proactively search for threats", and d) "My threat hunting skills have improved". This way of measuring learning effects is common, see for example [29]. Our variant scale had high reliability (0.966 and 0.906).

The final questionnaire used ten items adapted from the System Usability Scale [22] to measure system usability. An example item was "I thought the system was easy to use." The questionnaire also asked questions concerning potential technical problems, and asked for open ended feedback on the overall experience. All survey questions from the second, third and final questionnaire are presented in Table 4 and Table 5 in the appendix.

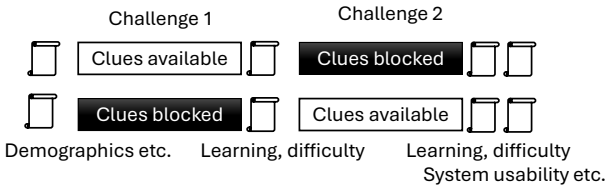


Fig. 3: The challenges in the two study sequences and the four questionnaires administered.

In addition to the four questionnaires, a set of measures were collected from the training platform: various completion timers, how clues were used, and if questions were answered correctly.

4.3 Sample

The target users of the Skade application are trainees who are potential cybersecurity analysts. The challenges included in the present test were designed for training security professionals to become threat hunters (e.g., tier one in a SOC) or for training computer network generalists (e.g., network administrators) in the basics of threat hunting. The sampling frame was set accordingly.

The Swedish Civil Contingencies Agency (MSB) provided support in the recruitment process by inviting persons fitting this sample frame. This was done by distributing a brief description of the training and the study to professional networks that MSB administers. These networks cover cyber-related personnel within organizations considered societally important in Sweden, e.g., specific government agencies and electrical power companies. A similar invite was sent out to a smaller group of threat hunting professionals within CISA, US Department of Homeland Security. The trainees were allowed to select which of the three packages of challenges they wanted to enroll in. All trainees were informed about the study and their personal data rights under the GDPR.

Recruitment was closed after 17 weeks, when 54 persons had chosen to enroll and scheduled to take a package of two challenges. Seven gave up after the first challenge. Two had serious technical issues that made the results incomplete, and were therefore excluded, leaving 45 trainees. The trainees chose their challenge package, with 14 selecting package A (5 hours), 12 selecting package B (4 hours), and 19 selecting package C (2 hours). Participants were not provided with any compensation for their participation.

There was considerable variance among the trainees in their years of work experience in different parts of the field. Most trainees had 0-10 years of work experience in relevant fields. The overall averages and standard deviations (in parentheses) were, per field: cybersecurity (μ=3.98, σ=4.42), incident handling (μ=4.62, σ=6.43), endpoint and host analysis (μ=2.62, σ=4.28), and network event analysis (μ=2.47, σ=3.31). Almost half of trainees had work experience greater than 10 years in system and network administration (μ=11.43, σ=9.14). There were 38 trainees (84%) who knew what threat hunting was before they signed up for the study.

4.4 Analysis Method

The four hypotheses were tested using Jamovi [17]. H1 and H2 were tested with correlational analyses and regression models. Because the crossover design is more or less balanced, H3 and H4 were tested by comparing mean values for the different conditions. This was complemented by a mixed model that controlled for potential effects of the challenge package and whether it was the first or second challenge. In all tests, $p<0.05$ was considered statistically significant.

Post-hoc analyses included additional tests and simple comparisons of mean values. The post-hoc analyses were also supported by the responses to open ended questions in the questionnaires, e.g., because they indicated problems that trainees had with the challenges.

5 Results

Our evaluation focuses on the four hypotheses presented in Section 1: two related to the perceived usability of the system and two related to the utility of optional clues. The sections below describe the tests of these hypotheses.

5.1 Effects on System Usability

Usability was measured with the widely adopted System Usability Scale. This scale can be converted to a value 0-100, for which benchmarks exists. On this scale, Skade received an overall mean of 67.2, the range was 17.5–90.0, and the standard deviation was 16.3. The spread in ratings is illustrated in Fig. 4. H1 says

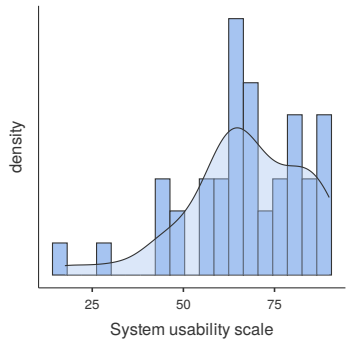


Fig. 4: The trainees' ratings of Skade with the System Usability Scale.

that usability is associated with trainee satisfaction with the difficulty level of the challenges, and H2 says that usability is associated with the perceived learning. Both satisfaction with the difficulty level and the perceived learning were measured twice (after each challenge) and here we use the mean value of each rating. As outlined in Table 1, the Pearson correlations support both hypotheses. The correlation coefficient for system usability and difficulty satisfaction was $r=0.481$ ($p<0.001$) and the correlation coefficient between the system usability and perceived learning was $r=0.334$ ($p=0.025$).

When both difficulty satisfaction and perceived learning are entered into a linear regression model to predict score from the *System Usability Scale*, the coefficient of multiple correlation is moderately high ($R=0.509$). However, only

Table 1: Pearson correlations between the difficulty satisfaction, perceived learning, and the System Usability Scale (SUS).

	Perceived learning	System usability
Difficulty satisfaction	0.372	0.481
Perceived learning	—	0.334

difficulty satisfaction is a significant predictor (β=0.414, p=0.006), while *perceived learning* (β=0.180, p=0.215) is not. On interpretation is that *perceived learning* influences system usability, but the overlap (collinearity) with *difficulty satisfaction* make the relationship weaker when both variables are used as predictor variables.

5.2 Effects from Clue Availability

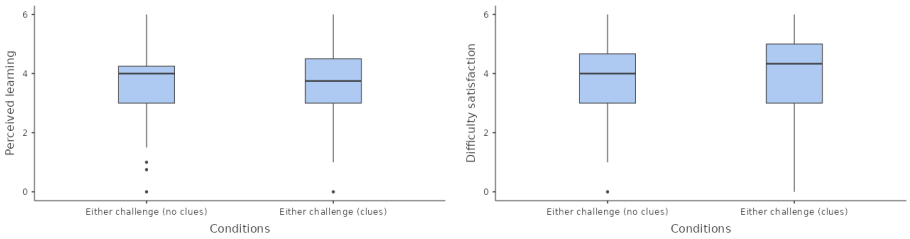
The paired values from measures with or without clues were used to test H3 and H4. The overall mean of these values are at the bottom of Table 2, and are almost the same regardless of condition. Accordingly, no significant difference is present in the pairwise comparisons. For *difficulty satisfaction* the mean difference is d=0.10 (p=0.506); for *perceived learning* the difference is d=0.02 (p=0.915). Thus, there is no clear effect from clue availability on either satisfaction with the difficulty level (H3) or perceived learning (H4). The table also indicates inconsistencies in effects due to when clues were available. For example, trainees performing the first challenge with clues were more satisfied with the difficulty level than trainees performing the first challenge without clues (μ=4.49 vs. μ=4.03), but the opposite is true for the second challenge (μ=3.31 vs. μ=3.35). Figure 5 shows the results for *difficulty satisfaction* and *perceived learning* for conditions with and without clues.

A mixed model was set up to control for the influence of different variables. The model clustered responses based on trainee (two responses each) and used two predictor variables other than clues: the challenge number (first or second) and the challenge package (A, B, or C). In the model, both control variables are statistically significant predictors of *difficulty satisfaction*. Thus, it matters whether the challenge was the first or second (p<0.001), with higher values for the first challenge. This order effect can be due to many reasons, e.g., that the second challenge of a package was too easy or too difficult, or that fatigue set in. The package also has an effect, with a clear difference between package A and C (p=0.005). When controlling for these other variables, the presence of clues has no effect (p=0.231). The same tendencies are found in a mixed model predicting *perceived learning*, but here no variables were statistically significant.

Overall, the fixed effects in the mixed models explain 23 percent of the variance in difficulty satisfaction and 10 percent for perceived learning. Considering also random effects, each model explains 70 percent. Thus, most variance is between trainees, and not between how a specific trainee rated the two challenges.

Table 2: Conditions and their mean values.

Conditions	Difficulty satisfaction (μ)	Perceived learning (μ)
First challenge (clues)	4.49	3.89
First challenge (no clues)	4.03	3.66
Second challenge (clues)	3.31	3.47
Second challenge (no clues)	3.35	3.64
Either challenge (clues)	3.86	3.67
Either challenge (no clues)	3.71	3.65



(a) Perceived learning (b) Difficulty satisfaction

Fig. 5: Trainees' responses on perceived learning and difficulty satisfaction with and without clues.

6 Discussion

The sections below discuss the usability of Skade, why the availability of clues did not help in the tests, and the overall prospect of automating hands-on cybersecurity training.

6.1 The Usability of Skade

In the tests we used the widely adopted System Usability Scale. This scale is associated with benchmarks. The mean rating of Skade (67.2) is on a par with public facing websites (67) and business-to-business software (67.6) [22]. Considering that Skade is a research prototype, and that this test was its first exposure to real trainees, we are satisfied with this mean rating. It suggests that Skade has the basic functions a trainee expects, and successfully integrates cyber range simulations. However, there is room for improvement.

The final questionnaire asked about problems that we anticipated trainees may have. For each potential problem, the 45 trainees answered if they had no problems, some problems, or serious problems. The results show that serious problems were rare, apart from getting needed help, e.g., using the hints (N=8 had serious problems with getting needed help, N=11 had some problems). Trainees experienced (some, or rarely, serious) problems with the purpose of the challenge questions (N=16), access permissions within machines (N=14), unre-

sponsive systems (N=11), interaction within machines (N=9), challenge background information (N=9), and low performance of the platform (N=8).

The written feedback provided by the trainees largely aligns with these numbers. While most of this feedback pointed to problems, some were also clearly positive. For example, "Very good learning experience!" and "Tasks were fun. Hope platform evolves with more threat hunting. Liked it a lot."

6.2 Limitations

Our experiment was executed according to plan. Yet, the two hypotheses concerning the utility of clues, which we based on previous research, were not supported in the data. We performed some post-hoc analysis to try to identify why, and tested several potential explanations.

First, one might suspect that the clues themselves were poor. Our answer is that they were not. We designed the clues to provide a direct investigation path, with explicit scaffolding on how queries are structured in SIEM and related log analytics tools, and with context tied specifically to the observed attack and its correct details (e.g., IP addresses). Trainee feedback was in line with this, e.g., one trainee commenting: "Include example search syntax for OpenSearch, Wazuh etc. to improve experience." Thus, the clues we provided were already focused on the type of information that trainees wanted more of.

Second, the challenges may have been either too difficult or too easy for the clues to have a meaningful effect. This is plausible, as both *perceived learning* and *difficulty satisfaction* exhibit ceiling effects and therefore violate the assumption of normality. This is common in scales like these, were respondents use the end of the scales. To assess whether this violation affects the results, we compared Pearson correlations with non-parametric alternatives (Spearman and Kendall's Tau) and conducted bootstrapped tests with 1000 iterations. Due to the ceiling effects, the Spearman correlation between the two non-normal variables is 0.17 lower than the Pearson correlation we reported. However, all relationships remain robust and retain their statistical significance. This also applies to the non-significant relationship involving clues, even when outliers are removed. For example, the effect does not become statistically significant if we exclude trainees with the lowest ratings (< 2), the highest ratings (> 5), or both.

Third, the outcome measures could have been chosen differently. For instance, the concept of flow is known to relate to learning. We used *perceived learning* and *difficulty satisfaction* because they are the main targets of the intervention. The most established measure of flow is the Flow State Scale [16], which consists of 36 items. Some of these overlap with our measures (e.g., "My abilities matched the high challenge of the situation"), while others are not expected to be influenced by clues (e.g., "I had total concentration"). Thus, although other measures could be tested with clues and used as quality indicators for systems such as Skade, we believe the scales used in this test are reasonable and meaningful.

Fourth, one may suspect that the results were biased because some trainees dropped out after not receiving clues in the first challenge, or because of biased responding (e.g., social desirability). Dropouts are unlikely to bias the results,

since roughly half of those who dropped out had been assigned to each condition: four received clues in the first challenge and three in the second challenge. We see no indications of other response biases. In addition, participation was voluntary and uncompensated, so there is little reason to suspect that participants answered in a particular way because they believed it was expected.

Fourth, one may suspect that trainees missed the availability of clues. Indeed, the results show that trainees do not always use available clues. This could lead to clue availability sometimes lacking an impact on perceived learning or difficulty satisfaction. However, the percent of clues used by trainees is not related to their difficulty satisfaction ($r=-0.079$, $p=0.606$) or perceived learning ($r=-0.014$, $p=0.926$). Thus, any missed clues did not alter the results.

Fifth, the sample and tests are limited to junior threat hunters and system administrators with limited cybersecurity competence. It is possible that clues would have a clearer effect in other sample frames. For instance, it may be that the individuals who needed clues in this study were overall too inexperienced to benefit from them, whereas the availability of clues could be the factor that tips the scale for a more competent audience.

Sixth, it could be that the randomization procedure has produced some bias. We randomized for each trainee, and the result was not perfectly even, as shown in Table 3. In combination with the results of our mixed model, suggesting that the challenge package and order were somewhat important, this could lead to our dependent variables being biased by the randomization procedure. Further research, with other randomization procedures, will be needed to clarify the impact of this bias. However, the effects of clues are insignificant when the mixed model is set to control for the effects of the randomization procedure.

Altogether, we believe it is unlikely that these potential issues have biased the result and hidden a true relationship. On the contrary, we believe that the availability of clues is not an effective way to control the difficulty satisfaction or the perceived learning. This may be, for example, because the trainees get stuck on very different things and need more personalized help than what can be offered with predefined clues.

6.3 The Prospect of Automating Cybersecurity Training

The need to close the competence gap is real. Consequently, there are many efforts on how to provide training and exercises cheaper and more effectively. Some

Table 3: The result of the randomization for each challenge package (pair of challenges).

Package	Clues first challenge	Clues second challenge	Total
A	4	10	14
B	6	6	12
C	11	8	19
Total	21	24	45

of these efforts were described in Section 3, and they share the design principles of Skade. For instance, most of the platforms employ structured learning tasks and clearly defined learning objectives (often related to MITRE ATT&CK). Accordingly, Skade is not the first documented attempt to automate exercise management of challenges in cyber ranges. For instance, the progression management described in [3] is along the same lines.

In our research, Skade received an acceptable rating according to the System Usability Scale, despite being only an initial prototype. However, our main focus was on clues, and their impact on trainee satisfaction with difficulty and their perceived learning. Unfortunately, our implementations of clues did not have much success in this regard. We believe that there are other ways of supporting trainees with clues or similar:

- Provide more dynamic clues by integrating large language models that target what the particular trainee needs help with better.
- Provide an "I give up"-button, which displays a step by step solution to the question and gives access to the next question.
- Differentiate clues of different types, e.g., separate clues on technical details and syntax from clues on mindset or threat intelligence.
- Adapt the difficulty level to the trainee, e.g., by asking for introductory self-assessments or by predicting proficiency based on past performance.

Thus, apart from our static clues, there are others ways that can be explored in balancing the difficulty levels of challenges.

7 Conclusions

Adapting the difficulty to trainees' proficiency levels is critical in hands-on cybersecurity training, such as threat hunting. However, the technical complexity of many threat hunting tasks makes it hard to find and implement the right difficulty levels.

Our tests indicate that the usability of our threat hunting training platform, Skade, is shaped by the trainees' satisfaction with the task difficulty, and by whether they believe that the system improved their threat hunting skills. While many trainees reported that the challenges were too difficult or that they lacked guidance, providing relevant clues did not produce measurable improvements in the satisfaction of the difficulty or in the perceived learning. These findings suggest that difficulty management cannot be solved by adding clues alone.

Acknowledgments. This study was funded by MSB (grant number 2022-1213). This study was also supported by DHS S&T and CISA pursuant to Technical Annex 07 of Project Arrangement 016-2011 between USA and Sweden.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Alessi, S.M.: Fidelity in the design of instructional simulations. Journal of Computer-Based Instruction **15**(2), 40–47 (1988)
2. Beason, R.E., Phelan, M., Devine, S., Aiken, M., Orban, J.: Evaluation of hands-on cybersecurity skills development. Tech. rep., Idaho National Laboratory (INL), Idaho Falls, ID (United States) (2021)
3. Beuran, R., Inoue, T., Tan, Y., Shinoda, Y.: Realistic cybersecurity training via scenario progression management. In: 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). pp. 67–76. IEEE (2019)
4. Beuran, R., Tang, D., Tan, Z., Hasegawa, S., Tan, Y., Shinoda, Y.: Supporting cybersecurity education and training via lms integration: Cylms. Education and Information Technologies **24**(6), 3619–3643 (2019)
5. Carnegie Mellon University.: TopoMojo: A VM Topology Manager (June 2019), `https://www.sei.cmu.edu/library/topomojo-a-vm-topology-manager/`
6. Chowdhury, N., Gkioulos, V.: Cyber security training for critical infrastructure protection: A literature review. Computer Science Review **40**, 100361 (2021)
7. Crumpler, W., Lewis, J.A.: The cybersecurity workforce gap. Report, Center for Strategic and International Studies, Washington, DC (Jan 2019), `https://www.csis.org/analysis/cybersecurity-workforce-gap`
8. Cyberbit: Cyberbit range: Assessing cyber skills to improve soc team performance. Tech. rep., Cyberbit (2020), `https://www.cyberbit.com/wp-content/uploads/2020/11/Assesing-Cyber-Skills-to-Improve-SOC-Team-Performance.pdf`
9. Dobson, G., Podnar, T., Cerini, A., Osterritter, L.: R-eactr: A framework for designing realistic cyber warfare exercises. Tech. Rep. CMU/SEI-2017-TR-005, Carnegie Mellon University, Software Engineering Institute's Digital Library (Sep 2017), `https://insights.sei.cmu.edu/library/r-eactr-a-framework-for-designing-realistic-cyber-warfare-exercises/`, accessed: 2024-Aug-14
10. ENISA: 2024 report on the state of cybersecurity in the union panel series. Tech. rep., ENISA (2024), `https://www.enisa.europa.eu/publications/2024-report-on-the-state-of-the-cybersecurity-in-the-union`
11. Floros, E., Stavrou, E., Smyrlis, M., Nikoloudakis, N., Potamos, G., Apostolidis, A., Bempis, P., Grigoriadis, A., Magkos, K., Merkouris, D., et al.: Towards the design of cyber range training programs for enhanced preparedness: investigating the training needs in critical infrastructures. In: 2025 IEEE Global Engineering Education Conference (EDUCON). pp. 1–10. IEEE (2025)
12. Gustafsson, T., Almroth, J.: Cyber range automation overview with a case study of crate. In: Secure IT Systems: 25th Nordic Conference, NordSec 2020, Virtual Event, November 23–24, 2020, Proceedings. pp. 192–209. Springer (2021)
13. Holm, H.: Lore a red team emulation tool. IEEE Transactions on Dependable and Secure Computing (2022)
14. Holm, H., Sommestad, T.: Sved: Scanning, vulnerabilities, exploits and detection. In: MILCOM 2016-2016 IEEE Military Communications Conference. pp. 976–981. IEEE (2016)
15. Holm, H., Sommestad, T.: Realistic and balanced automated threat emulation. Computers & Security **151**, 104351 (2025)
16. Jackson, S.A., Marsh, H.W.: Development and validation of a scale to measure optimal experience: The flow state scale. Journal of sport and exercise psychology **18**(1), 17–35 (1996)

17. Jamovi: jamovi (version 2.7.12) [computer software] (2025), https://www.jamovi.org
18. Kebande, V.R.: The impact of virtual laboratories on active learning and engagement in cybersecurity distance education. arXiv preprint (2024), https://arxiv.org/abs/2404.04952
19. Lazarov, W., Schafeitel-Tähtinen, T., Squillace, J., Martinasek, Z., Coufalikova, A., Helenius, M., Gallus, P., Fujdiak, R.: Lessons learned from using cyber range to teach cybersecurity at different levels of education. Technology, Knowledge and Learning pp. 1–36 (2025)
20. Lee, D., Kim, D., Lee, C., Ahn, M.K., Lee, W.: Icstasy: An integrated cybersecurity training system for military personnel. IEEE Access **10**, 62232–62246 (2022)
21. Lewis, J.R.: Psychometric evaluation of an after-scenario questionnaire for computer usability studies: the asq. SIGCHI Bull. **23**, 78–81 (1 1991). https://doi.org/10.1145/122672.122692
22. Lewis, J.R.: The system usability scale: Past, present, and future. International Journal of Human-Computer Interaction **34**, 577–590 (7 2018). https://doi.org/10.1080/10447318.2018.1455307
23. Mayes, J.T., Fowler, C.J.: Learning technology and usability: a framework for understanding courseware. Interacting with computers **11**(5), 485–497 (1999)
24. Miazi, M.N.S., Pritom, M.M.A., Shehab, M., Chu, B., Wei, J.: The design of cyber threat hunting games: A case study. In: 2017 26th International Conference on Computer Communication and Networks (ICCCN). pp. 1–6. IEEE (2017)
25. Noble, C.: The relationship between fidelity and learning in aviation training and assessment. Journal of Air Transportation World Wide **7**(2-3) (2002)
26. OECD: Building a Skilled Cyber Security Workforce in Europe. OECD Publishing (2 2024). https://doi.org/10.1787/3673cd60-en, https://www.oecd.org/en/publications/building-a-skilled-cyber-security-workforce-in-europe_3673cd60-en.html
27. Office of the national cyber director: 2024 report on the cybersecurity posture of the united states office of the national cyber director executive office of the president. Tech. rep., The White House (2024), https://bidenwhitehouse.archives.gov/wp-content/uploads/2024/05/2024-Report-on-the-Cybersecurity-Posture-of-the-United-States.pdf
28. Pfaller, T., Skopik, F., Smith, P., Leitner, M.: Towards customized cyber exercises using a process-based lifecycle model. In: Proceedings of the 2024 European Interdisciplinary Cybersecurity Conference. pp. 37–45 (2024)
29. Rovai, A.P., Wighting, M.J., Baker, J.D., Grooms, L.D.: Development of an instrument to measure perceived cognitive, affective, and psychomotor learning in traditional and virtual classroom higher education settings. Internet and Higher Education **12**, 7–13 (2009). https://doi.org/10.1016/j.iheduc.2008.10.002
30. Ruhe, E., Mcnamee, M., Du, G., Malmborg, J.: Tools for improving student motivation used during online teaching in higher education due to covid-19. Pedagogiska inspirationskonferensen-Genombrottet **11** (2021)
31. Russo, E., Costa, G., Armando, A.: Building next generation cyber ranges with crack. Computers & Security **95**, 101837 (2020)
32. SANS Institute: New 2025 SANS Threat Hunting Survey Reveals 61% of Organizations Struggle with Staffing Shortages. Press release, GlobeNewswire (Mar 2025), led by SANS Principal Instructor Josh Lemon; Bethesda, MD; full text available from GlobeNewswire

33. Sommestad, T., Karlzén, H., Kvist, H., Gustafsson, H.: Skade – A Challenge Management System for Cyber Threat Hunting. In: 9th Workshop On The Security Of Industrial Control Systems & Of Cyber-Physical Systems. Haag, Netherlands (2023)
34. Thuv, A., Enemo, G., Mancini, F., Endregard, M., Grov, G., Wurzenberger, M., Hotwagner, W., Landauer, M., Skopik, F., Ivanov, E., Ivanov, V., Rilling, G., Colò, G., Antonioni, E., Antonioni, E., Corsi, A., Zinonos, K., Sommestad, T., Karlzén, H., Tsirigotis, A., Keleris, V., Papadakis, N., Psychas, A., Jaber, A., Eian, M., Jaron, G., Petit, M., Blancher, X., Koufos, I., Christopoulou, M., Xylouris, G., Gardikis, G., Drosos, N., Priovolos, A., Dimos, A., Ferrari, R., Ferrari, R., Kerkdijk, R., Mourousias, N., Sampountzakis, C., Papamartzivanos, D., Veliou, E., Mancini, F., Christopoulou, M., Hotwagner, W.: Ainception scenario and storylines. Tech. rep., Norwegian Defence Research Establishment FFI (10 2025)
35. Turner, J., Paris, S.G.: How literacy tasks influence children's motivation for literacy. The reading teacher **48**(8), 662–673 (1995)
36. Vlachogianni, P., Tselios, N.: Perceived usability evaluation of educational technology using the system usability scale (sus): A systematic review. Journal of Research on Technology in Education **54**(3), 392–409 (2022). https://doi.org/10.1080/15391523.2020.1867938
37. Vykopal, J., Vizvary, M., Oslejsek, R., Celeda, P., Tovarnak, D.: Lessons learned from complex hands-on defence exercises in a cyber range. In: Proceedings - Frontiers in Education Conference, FIE. vol. 2017-Octob, pp. 1–8. Institute of Electrical and Electronics Engineers Inc. (2017). https://doi.org/10.1109/FIE.2017.8190713
38. Vykopal, J., Seda, P., Švábenskỳ, V., Čeleda, P.: Smart environment for adaptive learning of cybersecurity skills. IEEE Transactions on Learning Technologies **16**(3), 443–456 (2022)
39. Wei, J., Chu, B.T., Cranford-Wesley, D., Brown, J.: A laboratory for hands-on cyber threat hunting education. In: Journal of The Colloquium for Information Systems Security Education. vol. 7 (2020)

A Questionnaires

Table 4: The questions in the questionnaires answered after each challenge.

After challenge survey
Part 1 - Overall, I am satisfied with the...
Completely disagree - Completely agree
1 ... ease of completing the tasks in this challenge.
2 ... amount of time it took to complete the tasks in this challenge.
3 ... support information (on-line help, messages, documentation) when completing the tasks.
Part 2 - After this challenge...
Completely disagree - Completely agree
1 ... I am more ready to do threat hunting than before.
2 ... I have changed the way I think about how to investigate threats.
3 ... I feel better prepared to proactively search for threats in our operational computer networks.
4 ... my threat hunting skills have improved.

Table 5: The questions in the final questionnaire.

Evaluation of the app	
Part 1	
Strongly disagree - Strongly agree	
1	I think that I would like to use this system frequently.
2	I found the system unnecessarily complex.
3	I thought the system was easy to use.
4	I think that I would need the support of a technical person to use this.
5	I found the various functions in this system were well integrated.
6	I thought there was too much inconsistency in this system.
7	I would imagine that most people would learn to use this system very quickly.
8	I found the system very cumbersome to use.
9	I felt very confident using the system.
10	I needed to learn a lot of things before I could get going with this system.
Part 2	
No problem at all - Serious problems	
1	Background information, e.g., the purpose of the challenge.
2	Permissions to machines and data, e.g., accessing logs.
3	Purpose of the questions, e.g., the learning objectives that questions are linked to.
4	Getting the help you needed, e.g. using the hints.
5	Interaction with machines, e.g., remote desktops in the browser.
6	Unresponsive systems, e.g., downtime in the cyber range and 404 errors.
7	Low performance, e.g., because the web page is slow.