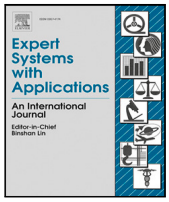
Expert Systems With Applications 254 (2024) 124436

Contents lists available at ScienceDirect

Expert Systems With Applications

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



All flags are not created equal: A deep look into CTF Scoring Algorithms

Abdullah Zafar[1], Muhammad Mudassar Yamin [*,1], Basel Katt, Espen Torseth

Norwegian University of Science and Technology, Teknologivegen 22, 2815 Gjøvik, Norway

ARTICLE INFO

Keywords:
Capture the flag
Attack defense scoring
Jeopardy scoring

ABSTRACT

Capture the Flag (CTF) competitions are popular in the cybersecurity field to train and evaluate the skills of students and professionals alike. Each CTF competition has a scoring system that is fundamental in evaluating a participant's skills by awarding scores for correct behavior and penalizing for incorrect behavior. Even though this topic gets discussed in the CTF community, it has mostly been ignored in previously published research material. The purpose of this research is: (1) to evaluate and understand how scoring algorithms affect the outcome of the two most commonly used CTF formats, i.e., Jeopardy and Attack-Defense. (2) To identify the desired requirements and properties of a CTF scoring algorithm by following a three-step process consisting of conducting a survey targeting experts from the European Cybersecurity Challenge (ECSC), identifying the currently available CTF algorithms using a literature review, and then simulating the identified scoring algorithms using data obtained from real CTFs. Finally, (3) scoring algorithms for both CTF formats are proposed based on the findings of the literature review, survey, and simulation results that fulfill the identified requirements.

1. Introduction

CTF exercises, as a format for training and conducting competitions, are one of the most popular and effective ways of teaching and testing cybersecurity skills. CTFs target a wide range of audiences, whether students, cybersecurity professionals, or people who want to try cybersecurity for the first time. CTFtime[2] has some 124 CTF events listed for 2023 at the time of this writing with still many that are organized for a limited audience (e.g., students of a university or employees of an organization) and do not make it to CTFtime. This is a testament to the popularity and the perceived benefits of CTF events for cybersecurity education in the eyes of the cybersecurity community (CTF Events, 2022).

One component of CTFs that is relatively ignored in academic research is their scoring algorithms. Due to the lack of any study addressing this topic, CTF participants (and even the organizers) can often be left unsure of what the scoring algorithm is evaluating. This study will help fill this gap by serving as a comprehensive reference for CTF scoring methodologies that can be used by both CTF organizers (to select scoring algorithms) and CTF players (to identify the strategies that are incentivized by a particular algorithm).

The study's contributions are multifold. The study aims to present fair CTF scoring methods for two of the most popular CTF formats, i.e., attack-defense and jeopardy CTFs. To achieve this goal, three key tasks are performed. First, the study leverages the expertise of subject matter experts, particularly those accessible through the European Cyber Security Challenge (ECSC), by conducting surveys to gather perspectives on effective scoring mechanisms. Secondly, a comprehensive literature review is conducted to compile and analyze the existing CTF scoring methods, providing insights into their strengths and weaknesses. Thirdly, the research will employ simulation techniques to evaluate the outcomes of different scoring algorithms.

In this way, the paper consists of three separate and complete contributions, i.e., a survey regarding CTF scoring, a literature review of available CTF scoring methods, and a simulator for evaluating scoring algorithms. These results are then utilized to accomplish the fourth objective, i.e., suggesting a fair scoring algorithm. This makes it a first-of-its-kind research work specifically addressing the topic of scoring methods in cybersecurity CTF competitions.

The rest of the paper is organized as follows: Section 2 explains the research methodology, Section 3 provides an overview of the related work, Section 4 presents the results of the survey conducted about CTF scoring, Section 5 discusses the different types of scoring algorithms identified during the research, Section 6 presents the results

* Corresponding author.
E-mail addresses: abdullaz@stud.ntnu.no (A. Zafar), muhammad.m.yamin@ntnu.no (M.M. Yamin), basel.katt@ntnu.no (B. Katt), espen.torseth@ntnu.no (E. Torseth).

[1] equal contribution.

[2] CTFtime (CTFtime.org, 2023) is an online platform for tracking CTF events.

https://doi.org/10.1016/j.eswa.2024.124436
Received 23 April 2024; Received in revised form 17 May 2024; Accepted 5 June 2024
Available online 7 June 2024

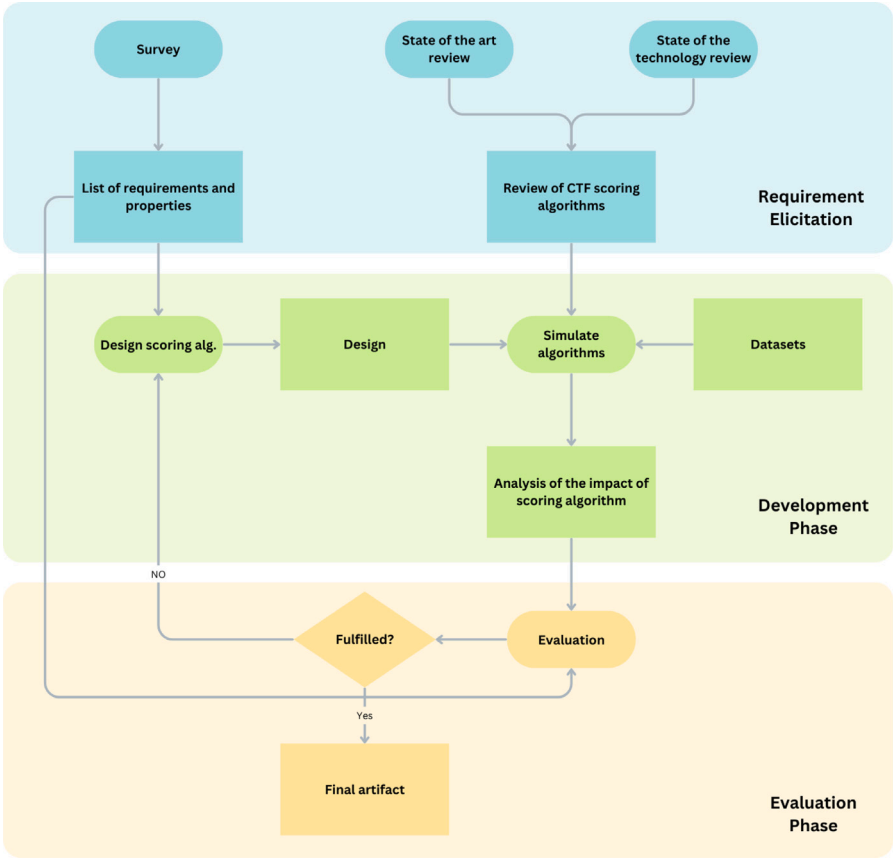


Fig. 1. Research methodology flow diagram.

of simulations, and finally, Section 7 presents the suggested algorithm followed by Section 8 with the conclusion and final remarks.

2. Research methodology

The research was carried out in a mixed-method way using qualitative and quantitative data (Creswell, 1999). The research was based on a survey, literature review, and experimentation of different scoring schemes. The complete process can be divided into three phases, i.e., Requirements elicitation, Development Phase, and Evaluation phase. The chain of events for conducting the research is presented in the flow diagram in Fig. 1.

In the first step, i.e., the **requirement elicitation phase**, a survey was performed to identify the requirements and properties desired by experts in the CTF community. A literature review of the state-of-the-art accompanied this to compile the available scoring algorithms and their properties, including strengths and weaknesses.

This was followed by the **development phase**. The development phase was marked by the development of simulators for Jeopardy and attack defense CTF scoring algorithms. The simulator served two purposes: first, to study the effect of scoring algorithms on a CTF outcome, and second, to analyze the newly designed algorithms.

Lastly, in the **evaluation phase**, the designed algorithm was analyzed according to the list of requirements from the survey. In the case of a failure, the design was sent back to the development phase to make the required adjustments. In case of compliance with the requirements, expert feedback was sought about the proposed algorithm, and the comments were incorporated into the final design.

In the following subsections, we will describe different steps of the research :

2.1. Literature review

A search was carried out to identify the available information on CTF scoring. The search phase was carried out in October 2022. Hence, the results are based on the information available at that time. The search period was divided into three phases:

1. To find different kinds of CTF competitions
2. To find scoring algorithms related to attack-defense CTFs
3. To find different kinds of scoring algorithms used in jeopardy CTFs

Given the popularity of CTFs, it was surprising that the research on CTF scoring is minimal. Initially, the search focused on identifying relevant studies on Google Scholar. Different keywords and search operator combinations were used to maximize the hits while also reducing the false positives, and as a result, the following search string was developed:

```
"ctf" "scoring algorithm"|"scoring technique"|"scoring mechanism"
"attack" "defense"|"defence"
```

Later, the same search string was used to identify CTF platforms on Google Search. We also discovered that many CTF organizers do not maintain archives from old events. This led to the use of Wayback[3] Machine (Internet Archive, 2022) to go around this problem. However, sometimes, the Wayback Machine also did not yield any results.

CTFtime is a website that keeps track of all CTF-related activities (CTFtime.org, 2022). It also has archives for old CTFs dating back to 2011 that helped to identify different CTF events. Lastly,

[3] https://archive.org/web/.

we searched on GitHub for different attack-defense CTF platforms. Here, we were also able to locate some unique results that were not previously identified in the search process.

2.1.1. Search criteria

The search criteria were designed to ensure that only studies meeting the requirements were considered while safeguarding against the omission of useful information related to the research. In the following sections, we will separately explain the inclusion and exclusion criteria employed in this study.

2.1.1.1. Inclusion criteria. To accomplish the research aims, we devised metrics to be considered while selecting a work to be added to the research. Various sources were considered, including research articles, GitHub repositories, online blogs, and CTF event websites that mention CTF competitions and their scoring. The inclusion criteria were based on the following factors:

1. Research articles must be written in English and should mention CTF scoring.
2. Old and upcoming Attack-defense style CTF competitions available on CTFtime.
3. GitHub repositories related to attack-defense platforms or scoring engines were also included in the research.
4. CTF platforms with documentation in languages other than English were accepted and translated using Google Translate if needed.
5. Websites of Jeopardy CTF competitions mentioning scoring or points.
6. Reverse citations of papers identified in search based on title and citation context.

2.1.1.2. Exclusion criteria. Since there were multiple sources of information, we had to employ different exclusion criteria for refining and filtering the results depending on the source. The exclusion criteria were based on the following points:

1. Articles about certain aspects of jeopardy and attack-defense CTFs that do not consider scoring.
2. GitHub repositories that have poor or no documentation.

2.1.2. Literature screening

The final search string on Google Scholar resulted in 37 unique articles. These results identified nine papers as relevant based on the search criteria. These were stored in an Excel sheet with comments, e.g., a summary, year, and link. The next step was to check their citations and include relevant ones. As a result, the total number of relevant papers increased to 19. Next, these papers were scrutinized, and 12 were found to be fulfilling the criteria.

The screening criteria for CTFtime were relatively straightforward. The aim was to identify as many Jeopardy and Attack Defense (AD) CTFs as possible to find the many variations in CTF scoring. Some old CTF events that were discontinued had dead links. They were ignored if these links did not yield relevant results on the Wayback machine. Some of the platforms found on CTFtime can be considered duplicates because they were related to previously identified research papers. But they were treated separately since research papers, depending on their year, may not cover the variations of algorithms used in several years of the gameplay.

2.2. Survey

A survey was conducted to understand and identify the desired requirements and properties for CTF scoring. The survey served as a way to solicit feedback from subject matter experts regarding CTF scoring. Representatives from 28 participating countries in the ECSC were selected as the target audience due to their status as a closed group known to the researchers to comprise experts in the field. Questions relating to CTF scoring were asked, of which 11 responses were received. 14 questions were asked covering topics related to jeopardy and attack-defense scoring, as well as the respondents' background and experience with CTF events. The results from the survey are presented in Section 4.

2.3. Simulation

Simulations were used to aid in understanding the effect of different scoring methods on a CTF outcome. Separate simulators were developed for attack defense and Jeopardy. The jeopardy simulator was developed in C++, whereas the attack defense simulator was developed in Python. In addition, scripts were developed to convert data from different sources into a format acceptable for the simulators. Simulations were carried out on logs from real CTFs made available by the CTF organizers. Section 6 further details the simulators, datasets, and simulation results.

2.4. Designing scoring algorithm

The scoring algorithm design was based on the requirements and properties identified in the survey and the literature review. It involved creating a scoring system that satisfied the desired requirements and properties obtained in the survey and comparing them to the algorithms found in the current scoring systems. The design process was also supported by simulation to explain the effects of scoring on the outcome.

2.5. Evaluation

The evaluation step aimed to ensure that the proposed algorithm fulfilled the requirements identified in the survey and the literature review. The designed algorithms were analyzed based on the requirements identified in the previous steps. Simulations were also used to understand the impact that the algorithms had on the outcome of a CTF. Lastly, the designed algorithm was presented to experts for feedback, and adjustments were made to meet the intended users' requirements.

3. Related work & research background

CTFs have been around for some time. The first known CTF event was held in 1996 at Defcon 4 (DEF, 2022). Despite that, the authors found no other study explicitly addressing the algorithms used for scoring in CTFs. However, given its importance, it could be seen occasionally in CTF-related studies, but only as a secondary topic and not as the primary focus. Researchers have started discussing cybersecurity attack and defense exercises and their efficacy in improving skills since the early 2000s. The authors (Hoffman, Rosenberg, Dodge, & Ragsdale, 2005) in 2005 and Childers et al. (2010) in 2010 talked about the importance of a scoring system in a CTF exercise in their studies. But the first published work where a scoring algorithm for an attack-defense CTF was specified was in 2011 by Werther, Zhivich, Leek, and Zeldovich (2011).

The authors in Werther et al. (2011) presented their experiences conducting an attack-defense style CTF competition in 2011. Different aspects of the CTF exercise are covered, including the scoring algorithm used to award points to the teams participating in the CTF. The scoring is based on the teams' ability to defend their services from attack and attack other teams' services. It was based on a generalized approach where the defense score was not only limited to confidentiality. Instead, it also included the availability and integrity of the team's services. In subsequent CTFs, these two properties were combined into a third metric, i.e., service level agreement (SLA). Attack, confidentiality, and integrity each were multiplied by pre-assigned weights and summed up

to get the team's score for a round. The weights could be adjusted to give more importance to a particular game aspect.

The authors (Davis, Leek, Zhivich, Gwinnup, & Leonard, 2014) in 2014 presented the details of CTF organized in that year at MIT. Here, they also mentioned the scoring algorithm that was used in the exercise. In the scoring formula, the total score was a summation of scores of all services in all rounds. The score of a service is the product of the service's availability, integrity, and flags deposited in the service (the paper uses the term "challenge" instead of "service").

The work (Price, Zhivich, Thompson, & Eagle, 2018) in 2018 is about designing a scoring algorithm for a cybersecurity exercise. In this paper, the authors presented general scoring guidelines like collusion resistance, real-world relevance, and automated evaluation, which are also relevant to any CTF exercise. However, the scoring algorithm is specific to the use case of the CyberGrandChallenge, which is different in format from an attack-defense CTF. As a result, it also tests for other metrics that do not apply to an attack-defense CTF.

The authors in Swann, Rose, Bendiab, Shiaeles, and Li (2021) 2021 have reviewed several CTF platforms based on different features that they have identified. The authors cover both attack-defense and Jeopardy platforms. The paper also talks about the scoring system used in those platforms. However, it does not go into the details of the scoring algorithm implemented. Consequently, it did not identify the advantages and disadvantages of these scoring algorithms. Diakoumakos, Chaskos, Kolokotronis, and Lepouras (2021)'s work also in 2021 has a scoring model for cyber ranges. Here, a generic scoring model for combining different scoring metrics is presented. The weights and optimum values of these metrics can vary depending on the challenges. The metrics used in the paper are the number of steps, time taken, and hints used while solving a challenge. Some of these metrics apply to Jeopardy-style CTFs. However, they are less relevant to other CTF formats like attack-defense.

Order of the overflow (OOO, 2023) organized the Defcon CTF for four years from 2018–2021. They have developed a 'scoring-playground' (o-o-overflow, 2023) that is used to test how different scoring formulas affect the results of a CTF. They have also provided the logs containing flag submission information from Defcon qualification rounds 2019–2021. The tool takes in the dataset and the scoring configuration as inputs and prints out the final scoreboard. In addition, it also prints a summary of challenges. This tool has an interesting feature that allows users to ask for a scoring formula to maximize the ranking of a specific team. The downside is that this tool is only limited to Jeopardy-style CTFs. Moreover, the tool works only in a dynamic configuration where the points are based on the total number of solutions.

In our analysis of scoring algorithms, we considered all papers presenting a scoring formula. Since a limited number of different algorithms were found in research studies, we also had to gather scoring algorithms from CTF events and platforms to cover a broad spectrum of scoring algorithms being used in practice. The scoring algorithms were analyzed to identify different traits or qualities of those scoring algorithms. In our paper, we present a guide comprising all identified techniques employed in a CTF scoring mechanism. In addition, we present the pros and cons of these techniques so it becomes clear how using a particular scoring algorithm affects the overall CTF exercise.

4. Survey results

This section covers the results of the survey conducted among the representatives of participating countries of the ECSC. The purpose of the survey was to obtain input from subject matter experts about the desired requirements and properties of a scoring formula. ECSC is a yearly event in which national teams from the participating European countries compete against each other in a cybersecurity competition. The complete event extends beyond CTFs, but the key highlights are the Jeopardy and attack defense CTF competitions.

Representatives from 28 ECSC member countries were asked to fill out the questionnaire. 11 responses were received. The participants comprised members, support staff, and coaches representing their national teams. Most participants' experiences conducting or participating in CTF events were 3–7 years. Questions were asked about the scoring mechanisms for attack and defense CTFs. The results are summarized in the following subsections.

4.1. Survey significance

The survey aimed to gather opinions from subject matter experts about the requirements for a CTF scoring algorithm. The insights acquired through the survey were valuable in the design of the new scoring algorithm. The survey was answered by 11 respondents, each representing a member country of ECSC. Out of these 11 respondents, 6 were coaches, 4 were support staff, and 1 was a member of the respective national team.

In this way, the survey is used as a systematic way of gathering expert opinions about CTF scoring from qualified people to provide informed answers. The fact that these people have been training their national CTF teams gives more credibility to the result than a general population of CTF players who might not be as familiar with the nuances of CTF scoring. This is even more relevant for attack-defense CTF scoring because attack-defense CTFs are relatively less common among the masses, and the survey results might yield misleading information. That is why this survey targeted only the experts having experience in both attack-defense and jeopardy CTFs.

4.2. Survey limitations

Purposive sampling was used in the survey to target experts who had experience with CTF exercises. Also, the experts represented only countries in the EU. These points affect the generalizability of the survey. This could be improved by including CTF players in the survey participants and extending the targeted audience to include people from geographically distinct locations. However, the question of including CTF players in the survey is debatable as many CTF participants may not understand the fine details of scoring mechanisms.

The survey was conducted in parallel with the literature review. Another approach could be to use the knowledge from the literature review in preparing the questionnaire so that specific questions could be asked about the different scoring techniques used in the past. However, this limitation is covered in this survey by the open-ended questions, where the participants could add anything from their experience to the answers.

4.3. Jeopardy

The questionnaire consisted of two parts. The participants were asked structured questions about the desired jeopardy scoring algorithm and its properties in the first section. This was followed by an unstructured section with open-ended questions where the participants could give their opinions more freely.

Four-fifths of the participants favored dynamic scoring over static scoring (as shown in Fig. 2). In another question asked about setting the initial value of challenges in the case of dynamic scoring, two-thirds voted in favor of having the same value for all challenges. In contrast, one-third voted for setting initial values based on the perceived difficulty (shown in Fig. 3). The participants also mentioned transparency and fairness as the most critical factors a scoring algorithm should fulfill. They also commented on using the suitable score decay functions in a Jeopardy CTF. In summary, the participants suggested the following requirements for a Jeopardy CTF scoring:

1. Kind of scoring — The participants suggested using **dynamic scoring** for jeopardy scoring. In dynamic scoring, the score

Which scoring mechanism do you prefer in Jeopardy style CTF?

Number of submissions: 11

Submissions	Count	% of submissions	
Static (has the same value all the time for all participants)	2	18.2%	18.2%
Dynamic (starts with a value and changes with the time)	9	81.8%	81.8%

Fig. 2. Survey results about preferred jeopardy scoring type.

In dynamic scoring, how do you prefer to set the initial value for challenges?

Number of submissions: 11

Submissions	Count	% of submissions	
The value depends on the difficulty level of the challenge (higher value for more difficult challenges)	4	36.4%	36.4%
The same value for all challenges	7	63.6%	63.6%

Fig. 3. Survey results about setting initial value in dynamic scoring.

awarded for any challenge depends on the total number of participants who solved that challenge.
2. Initial Value — The participants suggested that the initial value of challenges should be the same for all challenges irrespective of their difficulty.

Furthermore, the participants expressed the scoring mechanism in general to fulfill the following properties:

1. Clarity/Transparency — To clearly explain the scoring formula used in the CTF. It can also include public sharing of the scoreboard so anyone can verify the calculations.
2. Fairness — Fairness can have different meanings for different people. For some, it means incentivizing the weaker teams to make them interested in the game. For others, it means awarding scores purely based on the challenge difficulty.
3. Prevent flag hoarding — Flag hoarding is when players wait until the very last minute to submit the flags they have captured.
4. Score decay — In dynamic scoring, the score drop is adjusted using a score decay formula. The characteristics of this formula are also important for fair gameplay, as it will be shown in Section 6.1.1.

4.4. Attack-defense

Like Jeopardy, the attack defense section consisted of two parts: the first part had structured questions, and the second consisted of open-ended questions.

The participants were asked about the importance of different scoring metrics, i.e., attack, defense, and SLA. As shown in Fig. 4, close to one-third of the participants voted for the attack to be the most important factor, followed by SLA and first blood points, with one-fourth favoring each, and defense was voted to be the least significant with less than 10 percent votes in favor of defense to be the most important. The participants also valued **transparency** and **fairness** for the attack-defense scoring mechanism in their feedback. In summary, the following requirements were suggested in the survey for attack-defense scoring:

1. First blood — one participant suggested using first blood to give more points to the first team that has exploited a service.
2. Kind of scoring — The participants suggested using dynamic scoring for attack-defense CTFs.
3. Preferred aspect of the game — Overall, the participants favored giving more weightage to attack points as described above. However, some disagreed and suggested giving more value to service availability to discourage shutting down services.

The suggested properties for attack-defense scoring are summarized below:

1. Transparency/clarity — Transparency and clarity are just as crucial for AD as jeopardy CTFs. However, implementing them here can be more complicated as an AD CTF is far more complex than a Jeopardy CTF.
2. Score decay — Choosing a decay function with properties that ensure fair gameplay is essential.
3. Support for weaker teams — The weaker teams can be given more points to attack the stronger teams and vice versa.

5. Identified algorithms

Various research studies, CTF events, and platforms were analyzed during the literature for compiling a list of the currently used scoring mechanisms in CTFs. Fig. 5 has a timeline showing the development of different types of scoring algorithms in CTFs. It starts with the earliest Defcon CTFs in 1996 till CTFs in 2022, showing how scoring algorithms evolved with time. The identified algorithms are classified based on the CTF format and added under the headings in the following text:

5.1. Jeopardy

The algorithms used in Jeopardy-style CTFs can be divided into two major categories, i.e., static and dynamic. Manual scoring is another scoring type that is based on jury scoring. We have three configurations for dynamic scoring based on the total number of solutions, the number of previous solutions, or the time taken to solve the challenge (CTF Events, 2022). A summary of different scoring algorithms found in some well-known Jeopardy-style CTFs is shown in Table 1. The CTF scoring properties indicated with 'Y' in the table signify that the property is included in the scoring algorithms of the respective CTFs. 'AL' is a special case in the Trend Micro Qualification round, standing for 'Account Lockout', indicating that incorrect submissions will lead

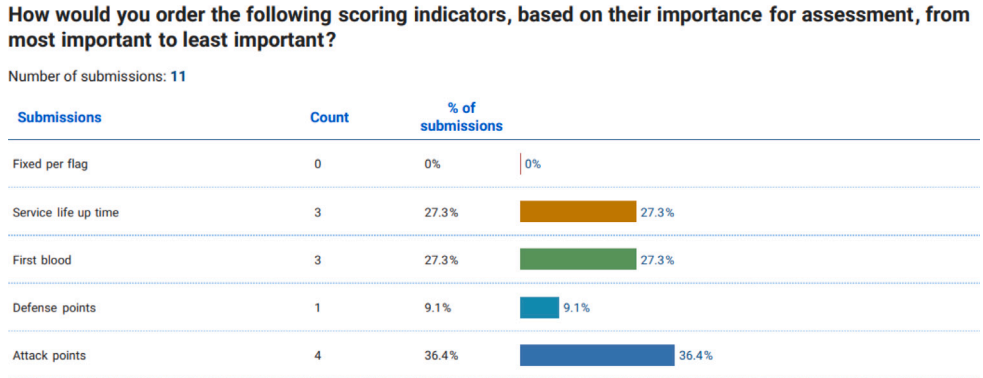


Fig. 4. Survey results about important factors in attack defense CTF.

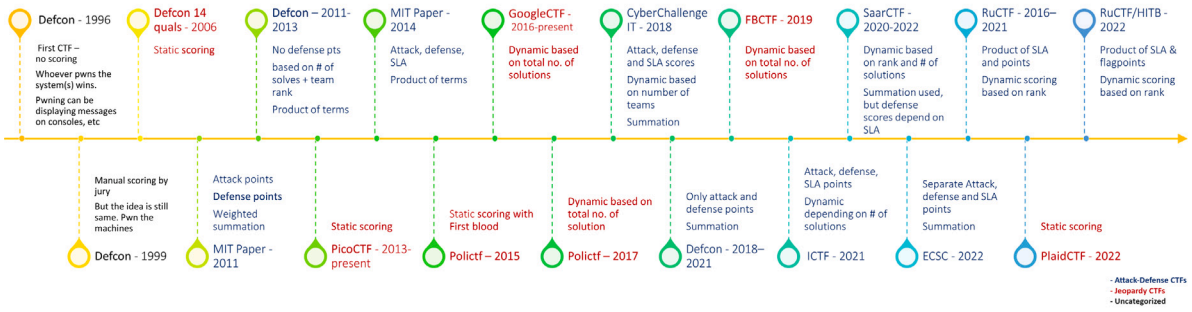


Fig. 5. Jeopardy and Attack-defense scoring timeline.

Table 1
Scoring configurations in Jeopardy-style CTFs.

CTF	Year	Static	Dynamic - total # of solutions	Dynamic - previous solutions	Dynamic - time based	Jury/ Manual	First blood	Hints deduction	Wrong flag deduction
Defcon 7 (DEF, 2022)	1999					Y			
Defcon 14 (CTF, 2024)	2006	Y							
PicoCTF (Chapman & Brumley, 2013; picoCTF, 2022)	2013 - **	Y							
PoliCTF (policTF, 2022; PoliCTF, 2024)	2015	Y					Y		
Google CTF (rules, 2022)	2016 - **		Y						
PoliCTF (policTF, 2022)	2017		Y						
FBCTF (Facebook, 2022)	2019		Y						
CODE CTF (CODE, 2024)	2019		Unclear	Unclear					
Trend Micro Quals (Capture The Flag, 2024b)	2021	Y					Y		Y (AL)
Trend Micro Finals (Capture The Flag, 2024b)	2021		Y				Y		

to the account being locked (Capture The Flag, 2024a). This section will further detail each of the identified scoring algorithms used in Jeopardy-style CTFs, including their advantages and disadvantages.

5.1.1. Static scoring

In static scoring, the challenges are assigned scores that remain constant for the duration of the event. Usually, the scores are based on the perceived difficulty of the challenges/tasks, which means that the tasks perceived to be harder by the CTF organizer have more points and vice versa. Since it is the simplest scoring method to implement and understand, it is fairly common in CTF events even though more advanced scoring algorithms exist. picoCTF (2022) and PlaidCTF (Plaidiverse, 2022) are two famous CTF competitions that use static scoring.

One benefit of static scoring is that it is the simplest to explain, understand, and implement. This is the reason we still see it being used in 2022. However, on the downside, the challenge scores may not depict the challenge difficulty correctly as perceived by the participants. In addition, tie-breakers might be needed since more teams can end up with the same points. The other option is to use the time of the last solution to find a way around this issue. Furthermore, static scoring is susceptible to flag hoarding.

Flag hoarding is a technique used by CTF participants to hide their true scores by not submitting the flags until the very last moment so that other teams put in less effort by being under the assumption they have a lead in the game. It can also be used to blindside other teams because the number of solves of a challenge is an indicator of the difficulty of a challenge. Hence, teams are less likely to try to solve a challenge with fewer solutions to prevent wasting time on difficult/unsolvable challenges.

5.1.2. Dynamic scoring based on the total number of solutions

In all kinds of dynamic scoring, a maximum score is set for each challenge at the beginning of the event that drops based on some predecided factor as the event progresses. The maximum score is typically

set to the same value for all challenges but can also vary depending on the difficulty of different challenges. A score decay formula, which can vary depending on the implementation, decides the rate and slope of decreasing the scores. It is also a common practice to have a minimum score so that solving the challenges does not become completely worthless.

For dynamic scoring based on the total number of solutions, the challenge scores decay based on the number of people able to solve that challenge. This decay also applies to the teams that have previously solved that challenge. The final score for any task is calculated at the end depending on the total number of teams that solved that task. This kind of scoring solves one of the problems with the previous scoring, i.e., awarding points based on challenge difficulty. However, it is still susceptible to flag hoarding. This scoring for Jeopardy-style CTFs is getting more and more common. Google CTF 2022 (rules, 2022) and PoliCTF 2017 (polictf, 2022) are famous examples where this scoring is used in a CTF.

An implementation of dynamic scoring can be explained by the maximum points, minimum points, and score decay formula. The scores from a challenge start with a maximum value when the number of solves is zero. The scores can decrease based on the score decay formula depending on the number of solves up to the minimum points. The score decay formula defines the relationship between the number of solves and challenge points. As an example, CTFd uses the following formula for dynamic scoring (Dynamic Value, 2022):

$$f(x) = \frac{b-a}{s^2}x^2 + a$$

In the above formula, x is the number of solves for the challenge, a is the maximum points, b is the minimum points, and s is the solve threshold (i.e., the number of solves after which the score reaches the minimum value). If the result of the above formula is less than the minimum, then the score is set to the minimum points instead.

5.1.3. Dynamic scoring based on the number of previous solutions

This is similar to the previous one. The difference is that the teams are awarded points equal to the challenge worth at that time, and the score for the task is decreased for subsequent teams that solve this task later. This makes it more tricky for the players to prioritize which tasks to solve first to maximize their scores. Again, the rate of decreasing the score can vary depending on the implementation of the score decay formula.

Like the previous scoring method, the points at the end of the event accurately depict the task difficulty. One improvement is that it is not susceptible to flag hoarding since submitting a flag early results in more points, so the teams that hoard flags will be penalized for doing so. On the negative side, it is vulnerable to a different attack where participants can create multiple accounts to submit flags. In this way, they decrease points for other teams solving other challenges at that time (ECSC, 2022). This type of manipulation with the scoring is more adverse for other players participating in the CTF than flag hoarding.

5.1.4. Dynamic scoring - time-based

In time-based dynamic scoring, the tasks lose their value with time. Unlike previously mentioned algorithms, the score decay formula depends on elapsed time instead of the number of solutions. In this way, the team that solves the challenges first gets the most points and the succeeding teams get fewer points with time. This is similar to the previous scoring in that they both incentivize speed. However, time or speed measures a team's skills instead of the number of solutions. Just like the previous scoring method, it automatically discourages flag hoarding. Since this method does not depend on how often a problem has been solved, it is not vulnerable to participants creating fake users to decrease a problem's value. One downside is that it is less practical for international CTF events where some participants will always be penalized due to time zone differences. Furthermore, it may overcomplicate prioritizing the challenges to solve from a player's perspective. Due to these reasons, it is very rarely seen in actual CTFs.

5.1.5. Manual/Jury scoring

In this kind of scoring, a jury awards points based on the quality of each solution. Clearly, it is not scaleable for large events. That is one of the reasons it is not very common these days. However, it still has its application where the tasks are more open-ended, and the success/failure of a participant cannot be ascertained entirely based on the submission of a flag.

Jury scoring has become obsolete, especially for Jeopardy-style CTFs. However, it is still used in other CTF formats. It has two serious problems concerning Jeopardy-style CTFs. It is infeasible to scale for events with large participation. In addition, the subjective nature of scoring can make it questionable for the participants.

5.1.6. Some other scoring configurations

- First blood — It means awarding extra points or giving another prize to the first team that solves a challenge. It is a good way to break ties while also incentivizing speed, but the incentive is lost as soon as someone gets the first blood. As an example, PoliCTF (2022) awards 6%, 4%, and 2% bonuses to the first three teams.
- Negative points for hints — Organizers can decide to have hints for the CTF tasks to make them beginner-friendly. However, these hints usually come at a cost where some points are deducted from the participant for unlocking a hint. For example, Hints (2022) can give hints either free or in exchange for some points that the organizer can adjust.
- Negative points for wrong solutions — Some CTF platforms deduct scores for submitting wrong flags. Here, the aim can be to discourage guesses leading to the right solution.

5.2. Attack-defense

For attack-defense CTFs, the different configurations were identified based on analyzing different studies, CTF platforms, and events. One common thing in all attack-defense CTFs is the attack points, irrespective of how the scoring is implemented. However, the attack points can also be implemented in various ways, as described in this section. Defense points and SLA are treated more like optional, and some CTF organizers skip these. Table 2 summarizes these scoring traits found in different CTF events. In the following sections, we will go through the different scoring traits identified in attack-defense CTFs one by one.

5.2.1. Attack points

Attack points can be distributed statically or dynamically. Static scoring in attack-defense, similar to Jeopardy, works by awarding fixed scores for all flags captured throughout the CTF. Defcon 2018–2022 (OOO, 2022) is an example of this scoring. In Defcon, 1 point is added for every stolen flag and subtracted for every lost flag.

The idea behind dynamic scoring in attack-defense is also similar to jeopardy. Attack points can be dynamically calculated based on the number of flag captures, team ranking, or both. These configurations are described in more detail in Sections 5.2.3 and 5.2.4 below.

5.2.2. Defense points

A vast majority of the CTF competitions and platforms have defense points. The defense points work by either awarding points for "good" defense or subtracting points for "bad" defense. For any round, points are awarded for flags not captured by other teams, or points are subtracted for flags captured by other teams. It is now almost universal to have defense scores in an attack-defense CTF. However, historically, some CTF events, e.g., Defcon 2011–2013 (Diutinus Defense Techonologies Corp, 2022) had no points for defense.

Defense points can be calculated separately from attack points or as a negative offset equal to the attack points earned by the attacking

Table 2
Attack-defense CTF scoring configurations.

Paper/CTF	Year	Attack points	Defense points	Dynamic based on number of solutions	Dynamic based on ranking	SLA	Sum of terms	Product of terms
The MIT LL CTF exercise (Werther et al., 2011)	2011	Y	Y			Y	Y	
Defcon finals (Diutinus Defense Techonologies Corp, 2024)	2011–2013	Y		Y		Y		Y
The fun and future of CTF (Davis et al., 2014)	2014	Y				Y		Y
RuCTF (RuCTF 2021 rules, 2024)	2016–2021	Y	Y		Y	Y		Y
CyberChallengeIT (ctf-ad-rules/README.md, 2024)	2018	Y	Y	Y		Y	Y	
Defcon finals (OOO, 2024)	2018–2021	Y	Y				Y	
iCTF (iCTF2021, 2024)	2021	Y	Y	Y		Y	Y	
Saarland CTF (saarCTF, 2022)	2021	Y	Y	Y	Y	Y	Y	
ECSC (Scoring, 2022)	2022	Y	Y	Y		Y	Y	
RuCTF (RuCTF, 2024)/HITB SECCONF CTF (HITB, 2022)	2022	Y	Y		Y	Y		Y

teams, indicating a compromise of flags. For example, FAUST CTF 2022 (Rules, 2022) has a separate formula for calculating defense scores, resulting in fewer points being deducted for the defense than earned by the attack. Whereas others like iCTF 2021 (iCTF2021, 2024) and RuCTF 2022 (RuCTF, 2024) subtract the points equal to what the attacking teams are awarded.

5.2.3. Scoring based on number of flag-captures

This kind of scoring is similar to dynamic scoring based on the total number of solutions described in the Jeopardy Section 5.1. For a particular flag, the points are divided among all teams that capture that flag. Here, the idea is to award points based on the difficulty of capturing a flag. In the same way, defense points are divided among all the teams that have those services intact after the end of the round. SaarCTF 2020–2022 (saarCTF, 2022), FAUST CTF 2022 (Rules, 2022) and iCTF 2021 (iCTF2021, 2024) use this kind of scoring technique. FAUST CTF has a relatively straightforward implementation of this, as shown below:

$$Offense = N + \sum_{i=1}^{N} \left(\frac{1}{C_i}\right)$$

where N is the number of flags captured by the team, and C_i is the number of teams that captured a particular flag.

5.2.4. Scoring based on team ranking

Some CTF competitions award points based on differences in the team positions of the attacker and the victim. As a result, teams are encouraged to attack high-ranking teams. A direct effect of this is that low-ranking teams earn more points than high-ranking teams for successful attacks. This also, to some extent, discourages the stronger teams from accumulating too many points and keeps the game interesting for weaker teams.

Scoring based on teams' relative ranking is similar to the ELO rating system where the points depend on the teams' relative ratings (Elo rating system, 2022). The most famous example is RuCTF, where rank-based scoring has been used since 2016. In addition, SaarCTF 2020–2022 (saarCTF, 2022), HITB SECCONF CTF 2022 (HITB, 2022) and Defcon 2011–2013 (Diutinus Defense Techonologies Corp, 2022) implemented this technique in their scoring algorithms. For example, the formula used in HITB SECCONF CTF is added below:

$$Motivation(teamA, teamB) = \left\{ \begin{array}{ll} 1 & \text{if } pos_A \geq pos_B \\ 1 - \frac{pos_B - pos_A}{teams\_count - 1} & \text{else} \end{array} \right\}$$

$$Price = flag.base\_price^{motivation(teamA, teamB)}$$

The formula is taken from HITB (2022). We can see the flag.base_price is raised to the power equal to a factor named motivation that depends on the teams' relative positions on the scoreboard.

5.2.5. SLA points

It would not be wrong to say that SLA is the backbone of the attack-defense CTF competition. Without SLA, teams can just shut down their services to keep them from being attacked. Clearly, this is never the goal of a CTF event. That is why points are awarded for the time the services are kept working during the event. Service checkers are deployed periodically to test the basic intended functionality of the service for giving SLA points.

Some choose to calculate SLA points separately for each round based on a formula (that usually relies on services that are up and the number of teams in the CTF) and add to the total scores. In contrast, others calculate the average SLA since the start of the CTF and multiply it by the attack & defense points. However, examples like Defcon (OOO, 2022) do not consider SLA points in their scoring. One example where the SLA was calculated for each tick is the SaarCTF 2022 (saarCTF, 2022) where the SLA formula used is as follows:

$$SLA = \left\{ \begin{array}{ll} 1 & \text{if status=up} \\ 0 & \text{else} \end{array} \right\} * \sqrt{num\_online\_teams}$$

In some CTFs, other intermediates service states (e.g., recovering) are also defined to provide a finer gradation of service states (Rules, 2022).

5.2.6. Sum or product

This section deals with the way different components of scoring (i.e., SLA, attack score, defense score) are combined to get the total score. Adding vs multiplying the scores for each metric also affects the game. By multiplying the terms, each of the three aspects of scoring is valued. Whereas, if the terms are added, the teams that are exceptionally good at either of the aspects of the game can, in theory, end up as the top-scoring team. Hence, the choice between additive and multiplicative scoring methods can significantly influence gameplay dynamics and strategic decision-making, shaping the overall experience for the players.

As an example of multiplying the terms, the RuCTF uses the following formula (RuCTF, 2024):

$$score = \sum_{i=1}^{N} SLA(team, service_i) \times FlagPoints(team, service_i)$$

In contrast, FAUST CTF has a formula where the terms are added (Rules, 2022):

$$total = \sum_{i=1}^{N} offense(service_i) + defense(service_i) + sla(service_i)$$

where N is the number of services.

5.2.7. Defense points dependent on SLA

This kind of scoring was only observed in SaarCTF 2022 (saarCTF, 2022), where the defense points depend on the SLA points by including a multiplication factor in the formula for defense points. In this way, the importance of SLA is increased for both SLA points and defense points.

Table 3
Occurrence of scoring traits in CTFs.

Scoring trait	Frequency
Attack points	12
SLA	11
Defense points	10
Dynamic	9
Sum of terms	9
Dynamic based on # of solutions	7
Dynamic based on team rank	3
Product of terms	3
Fixed total points	1

This is also a good way of ensuring that the service is up and running to get any defense points. The defense formula used in SaarCTF is as follows:

$$Defense\_points = \sum_{flag \in stolen\_flags} -\frac{(\frac{num\_all\_captures\_of(flag)}{num\_online\_team})^{0.3} * SLA}{flags\_per\_round}$$

In the above formula, the defense score is first calculated by dividing the number of flags captured by the total number of teams and raising it to a power of 0.3. This score is normalized by the total number of flags in a round. Finally, the defense score is multiplied by SLA.

5.2.8. Other configurations

- Same or different defense points:
  Some CTFs have separate formulas for attack and defense point calculations. Others subtract the sum of points received by all the attackers who could attack a particular service in a round. iCTF2021 (2024) and RuCTF (2024) are examples where the same points are subtracted from defending teams. In contrast, Rules (2022) has a separate formula for defense points calculation.
- Constant sum of total scores:
  A unique trait was mentioned in the iCTF scoring (iCTF2021, 2024) where the sum of total scores awarded to all teams in any given round was constant. The scores are divided among all the teams depending on their progress. If a team has a service up and unexploited, it will receive 50 points. If it has a service up but is exploited, 50 points are divided among each team that has exploited this service. If the service is down, then 50 points will be divided among all teams having that service up. The following formula gives the total number of points in any round:

  $$points\_per\_round = 50 \times num\_teams \times num\_services$$

5.2.9. Scoring traits used in different CTFs

We surveyed 12 attack-defense CTF competitions, including some platforms available on GitHub, to analyze their use of different scoring traits. The list includes Defcon 2021, SaarCTF 2022, ENOWARS 6 2022, FAUST CTF, ECSC, CyberChallengeIT, iCTF 2021, RuCTF 2022, Stay CTF, ADCTF platform, ctf01d, and InCTF. The results are shown in Fig. 6 and Table 3. The attack points and SLA are the most common traits, followed by defense points. Then, we see a lot of variance in other things like the use of dynamic scoring or rank-based scoring, etc.

6. Simulating identified algorithms

Scoring simulators for Jeopardy and attack-defense CTFs were developed as part of this study to evaluate and understand the effect of scoring methods on CTF results. The simulators' source code is freely available on GitHub (ncr-no/ctf-scoring-simulator, 2023). The idea was to apply different scoring methods on CTF submission logs from real CTFs and then analyze their effect on player rankings. Separate simulators were developed to analyze jeopardy and attack defense scoring because of the difference in the structure of these CTF styles.

Table 4
Comparison with scoring-playground using Defcon Quals 2019 dataset, left: scoring-playground from OOO, right: our jeopardy scoring simulator.

Score	Solves	Time	Team	Player-id	Score	Solves
3591	22	186	PPP	3	3591	22
3371	22	163	HITCONBFKinesiS	34	3371	22
3035	19	223	Shellphish	18	3035	19
2877	20	248	Sauercloud	17	2877	20
2863	19	205	Samurai	7	2863	19
2860	20	208	A*0*E	113	2860	20
2695	19	110	SeoulPlusBadAss	32	2695	19
2659	19	206	Tea	77	2659	19
2547	18	166	CGC	14	2547	18

6.1. Jeopardy simulator

The Jeopardy scoring simulator has a modular design, with the core functionality developed in C++. Accompanying Matlab scripts are also provided that were used to convert CTF logs obtained from different sources like Defcon Quals (o-o-overflow, 2023) and pwn2win CTF (pwn2winctf/nizkctf-audit-trail, 2023) to a format that is compatible with the C++ simulator. This tool is different from the scoring-playground (o-o-overflow, 2023) developed by the organizers of Defcon 2018–2021. While the scoring-playground only supports dynamic scoring based on the total number of solutions, this tool also supports static and dynamic scoring based on previous solutions. In addition, this tool can be used to output scores of players in text files that can be plotted for further analysis using a Matlab script provided in the repository.

The simulation process can be divided into three steps, i.e., initialization, processing, and output. The submissions file is parsed and stored in the internal database in the initialization phase. Other information, like the number of players and challenges, are also extracted from the submissions. In the next step, i.e., the processing phase, we iterate over the parsed submissions and add these to the players while assigning them scores. The scores are calculated according to the selected scoring algorithm and decay formula (if applicable). The final phase, i.e., the output phase, is marked by displaying the top-10 leaderboard and writing players' scores to output files.

To verify the results, the outputs were compared with the "scoring-playground" from Order of the overflow (results shown in Table 4). Datasets from Defcon Quals 2018–2021 were used for this purpose. In addition, the output from the pwn2win dataset was also compared and found to be in line with the pwn2win scoreboard available at the pwn2win website (Scoreboard, 2023). Figs. 7 and 8 show the plots generated from the simulation and the original scoreboard, respectively.

6.1.1. Simulation results

The datasets from Defcon Quals 2019–2021 (o-o-overflow, 2023) and pwn2win 2022 (pwn2winctf/nizkctf-audit-trail, 2023) were used in the simulation and testing of different CTF configurations by varying scoring algorithms and scoring decay functions. The resulting comparisons were made based on the top 20 positions of the leaderboard. Two experiments were conducted to estimate the effect of changing score-decay functions and scoring algorithms. The leaderboards were then compared using three metrics, i.e., a dissimilarity index, the percentage difference in ranks, and the percentage difference in scores. The results are presented in Tables 5, 6, and 7. In these tables, "Dynamic-P" and "Dynamic-T" abbreviations are used for dynamic based on previous solutions and dynamic based on the total number of solutions.

Before going over the results, it is important to explain the metrics used for comparing the two scoreboards. The first metric, referred to as **Leaderboard dissimilarity index**, is inspired by the similarity index presented in How much do football (2023). The dissimilarity index can be explained as the percentage of teams whose rankings have changed from one leaderboard to another. The second metric, the **percentage**

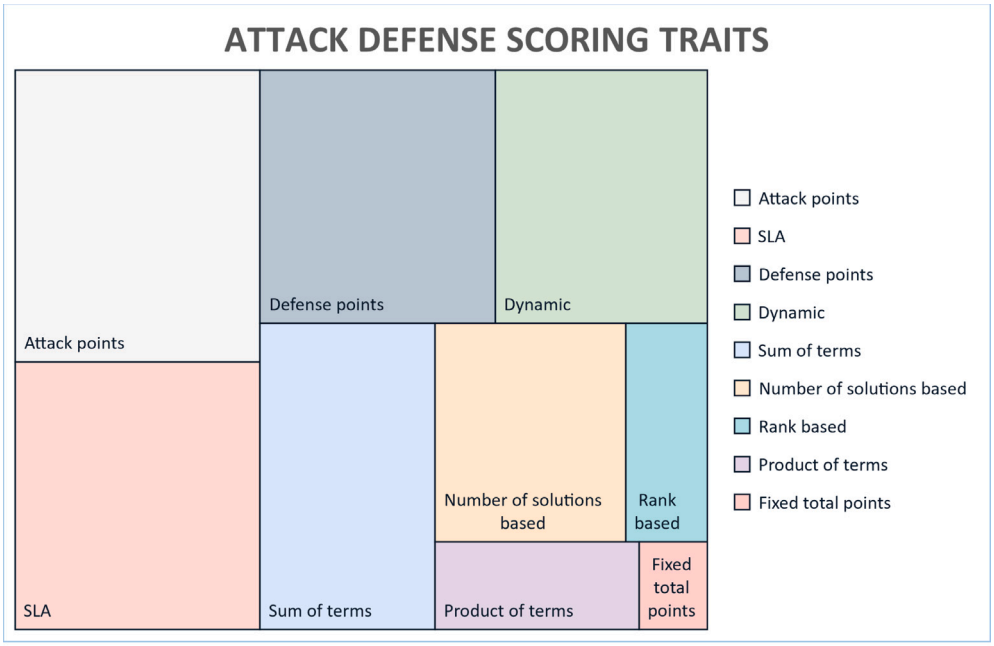


Fig. 6. Attack-defense scoring traits treemap.

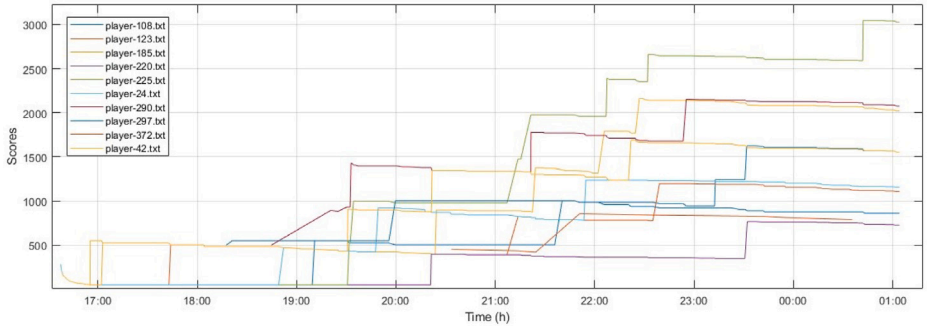


Fig. 7. Pwn2win graph from simulated scores.

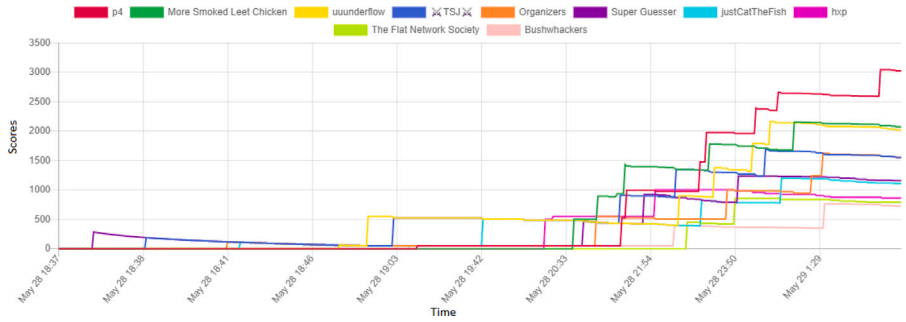


Fig. 8. Pwn2win original scoreboard taken from Scoreboard (2023).

difference in ranks, is the average percentage change in ranks of all teams between the two leaderboards. Similarly, **percentage difference in scores** is the average percentage change in scores of all teams between the two leaderboards.

The tables on the left present results from comparing different score decay formulas, while the tables on the right show the results from different scoring algorithms. The results from the leaderboard dissimilarity index tables show the variance in leaderboards is quite similar in both test cases. However, Table 6 shows the percentage difference to be more for changing scoring algorithms compared to the score-decay functions.

6.2. Attack-defense simulator

A similar process was adopted for simulating different scoring algorithms on attack-defense data. But here, the approach was changed because, in contrast to jeopardy, an attack defense event typically involves more than one metric. Different scoring methods were developed for attack-defense in separate Python scripts also available on

Table 5
Leaderboard dissimilarity index.

	DEFCON	pwn2win	ctfd	watevr		Static	Dynamic-T	Dynamic-P
DEFCON	0	46.25	65.6	62.5	Static	0	59.27	76.04
pwn2win	46.25	0	61.9	54.37	Dynamic-T	59.27	0	72.7
ctfd	65.62	61.9	0	53.8	Dynamic-P	76.04	72.7	0
watevr	62.5	54.37	53.8	0				

Table 6
Difference in positions.

	DEFCON	pwn2win	ctfd	watevr		Static	Dynamic-T	Dynamic-P
DEFCON	0	1.3	2.4	2.8	Static	0	5.2	15.2
pwn2win	1.4	0	3.2	2.5	Dynamic-T	4.1	0	8.6
ctfd	2.3	3.2	0	1.4	Dynamic-P	13.3	9.0	0
watevr	2.3	2.5	1.3	0				

Table 7
Difference in scores.

	DEFCON	pwn2win	ctfd	watevr		Static	Dynamic-T	Dynamic-P
DEFCON	0	43.1	10.8	28.5	Static	0	−23.2	36.1
pwn2win	−29.9	0	−21.8	−9.3	Dynamic-T	36.2	0	79.5
ctfd	−8.2	31.6	0	16.1	Dynamic-P	−21.5	−42.5	0
watevr	−21.2	12.8	−14.0	0				

Table 8
Ids of top 10 teams from enowars5 and after fixing the bug.

Enowars5	After bugfix
34	34
5	5
60	60
15	15
68	68
45	48
48	45
67	67
66	2

Table 9
Top 10 IDs by applying different scoring rules to ECSC data.

ECSC	CyberChallengeIt	Defcon	Saarland	iCTF
7	7	7	7	6
16	16	16	11	11
11	11	28	6	7
28	28	11	16	28
6	6	6	28	16
29	10	29	10	10
10	29	10	29	22
22	22	22	22	29
9	9	20	1	9
20	20	9	30	1

GitHub (abza1, 2023). To our knowledge, no such tool is available for simulating attack-defense scoring algorithms, making it the first of its kind. CTF players can also use this tool to audit CTF results after a competition.

The scoring methods were applied to two datasets from ECSC 2022 and Enowars5 2021 (enowars/ctf-dumps, 2023). The data consists of SQL dumps of the logs generated from the CTF. The relevant tables from the SQL dumps were exported as JSON to be processed by the scoring simulator. The scoring methods implemented are from FAUST, CyberChallengeIt, Defcon, SaarCTF, and iCTF CTF competitions. For verification, the same approach was applied here as in Jeopardy, where first, the data was simulated using the original CTF scoring method, and results were compared to verify the working. This was followed by adapting the code to implement other scoring methods.

6.2.1. Effect of a bug on scoring

While verifying the output, a bug was discovered in the EnoEngine (enowars/EnoEngine, 2023) from Enowars. The output from our tool did not match the scores on the Enowars website. It was found upon debugging that this behavior was due to a bug in the organizers' implementation of the scoring algorithm. The issue was coordinated with the maintainers of the EnoEngine repository, who verified our findings. The bug affected not only the scores of the players but also their team rankings. The change in rankings was observed starting as early as position 6, depicting the crucial role played by scoring systems in CTFs and the need for methods to verify their working. Ids for the top 10 positions from the CTF website (buggy) and the original algorithm are shown in Table 8.

6.2.2. Simulation results

The scripts' working was verified by comparing the output to the official CTF results. The results obtained with EnoEngine are already explained in the last section. The same procedure was applied to ECSC data, and the results were found to be the same as the scoreboard present in the SQL dump. Furthermore, we applied different scoring formulas to the dataset to see the effect on the result. The top 10 team IDs obtained by applying various algorithms on the ECSC dataset are shown in Table 9.

Now, to compare different scorings, we use the same metrics as defined in Section 6.1.1. The results are presented in Tables 11, 12, and 13. Dissimilarity indices in Table 11 show the effect of changing scoring formulas on the leaderboard. We can see that all variations change the leaderboard considerably, even the ones that are quite similar. In the same way, Table 13 shows the percentage change in the scores of a team by varying the scoring method. The results here might seem drastic at first, but they are expected because some algorithms might give all teams fewer points overall. This gets accumulated and results in a very high absolute change in scores.

Table 12 shows the percentage difference in ranks on the same dataset. Here, we see apparent differences in scores between iCTF-scoring and the other scoring methods. This is expected if we look at the ranks in Table 9, where the top 3 teams for iCTF differ entirely from other scoring methods. These results are expected as the top teams affect this percentage score more. Another thing to note is that the SaarlandCTF scoring results are similar to the other. This may seem counterintuitive initially because SaarlandCTF scoring has rank-based scoring, which is missing in others. Also, SaarlandCTF scoring has

Table 10
Top 10 scores by applying SaarlandCTF scoring (left) and ECSC scoring (right).

TeamId	Attack	Defense	SLA	Total	TeamId	Attack	Defense	SLA	Total
7	26 552	−376	10 059	36 235	7	25 736	−2485	10 202	33 453
16	26 112	−429	10 375	36 058	16	25 316	−2824	10 455	32 948
11	25 450	−334	10 260	35 375	11	24 552	−2427	10 208	32 333
28	25 386	−416	10 283	35 253	28	24 539	−3054	10 231	31 716
6	23 067	−316	10 197	32 948	6	22 329	−2235	10 254	30 348
10	17 957	−322	9 955	27 590	29	17 592	−2127	9 840	25 306
29	18 187	−344	9 628	27 471	10	17 372	−2420	9 829	24 781
22	16 101	−293	9 456	25 263	22	15 614	−2456	9 484	22 642
9	14 266	−367	9 685	23 584	9	13 794	−3032	9 605	20 367
20	14 555	−475	9 306	23 386	20	14 088	−4439	8 979	18 627

Table 11
ECSC scoring comparison with dissimilarity index.

	ECSC	CyberChallengeIt	Defcon	Saarland	iCTF
ECSC	0	55.88	64.71	52.94	73.53
CyberChallengeIt	55.88	0	73.53	35.29	73.53
Defcon	64.71	73.53	0	61.76	79.41
Saarland	52.94	35.29	61.76	0	67.65
iCTF	73.53	73.53	79.41	67.65	0

Table 12
Scoring comparisons by percentage difference in ranks.

	ECSC	CyberChallengeIt	Defcon	Saarland	iCTF
ECSC	0	5.57	8.2	7.35	22.85
CyberChallengeIt	5.34	0	7.39	2.49	22.65
Defcon	7.56	7.35	0	6.75	27.31
Saarland	6.58	2.4	6.67	0	23.57
iCTF	26.82	28.06	34.66	30.43	0

Table 13
Scoring comparisons by percentage difference in scores.

	ECSC	CyberChallengeIt	Defcon	Saarland	iCTF
ECSC	0	26.92	42.8	55.59	607.48
CyberChallengeIt	18.28	0	52.35	20.08	462.99
Defcon	96.71	156.23	0	218.07	1408.8
Saarland	30.03	15.85	58.94	0	378.01
iCTF	85.24	81.78	91.28	78	0

reduced weightage for defense. But this makes sense because the contribution of rank-based scoring to attack scores is minimal. Furthermore, the defense scores contribute a small percentage to the overall scores, so reducing them has little impact. This is shown in Table 10 that contains individual scores of the top 10 teams by applying ECSC scoring and saarlandCTF scoring.

7. Suggested algorithm design

This section will present the suggested algorithms for Jeopardy and attack-defense scoring. The proposed algorithms aim to enhance the competition's fairness and transparency and the participants' overall gameplay experience.

7.1. Jeopardy

For Jeopardy-style CTFs, the choices are somewhat limited. There are four major kinds of algorithms. Even though, technically, there can be an infinite number of decay functions for the dynamic algorithms, they all can be divided into three general categories.

7.1.1. Scoring algorithm

As described in Section 5.1, it can be seen that there are trade-offs for choosing different algorithms. Static scoring has the benefit that it is not susceptible to any manipulation by the participants (other than being susceptible to another kind of behavior known as flag hoarding). However, the vast majority of the community (as shown in the survey) chooses to give up static scoring for dynamic scoring based on the total number of solutions because the final scores correctly depict the difficulty of challenges. Also, the possibility of manipulating scores by registering more teams is considered to be generally limited in this kind of scoring because it affects one's team, too. However, dynamic scoring based on previous solutions is far more susceptible to this manipulation because only other teams are affected by this behavior.

Time-based scoring may be a good option in cases where the aim is to judge the team's ability to prioritize tasks where the most important task needs to be solved first. However, generally, in Jeopardy-style CTFs, it is counterintuitive that all teams lose points only because they are working on some other tasks first. So, in light of this discussion and considering the widely used practice and the survey results, dynamic scoring based on total solutions is the best option for Jeopardy-style CTFs.

7.1.2. Decay functions

As seen in the simulation results in Section 6.1.1, choosing the decay function also affects the outcome of a CTF. However, the effect is generally lesser than choosing the scoring algorithm. Generally speaking, there are three kinds of decay functions i.e. with decreasing slope (or convex decreasing), increasing slope (or concave decreasing), and constant slopes (or linear). These types are shown in Fig. 9.

The argument favoring a concave function is that the score has a slow drop at the start so that demanding challenges retain their value. However, its downside is that concave functions (like CTFd) depend on a decay factor set according to the number of players. The decay factor tells the number of solves, after which the score will go to the minimum. Since it depends on the number of players, it is open to manipulation by registering fake users. This effect can be seen in Fig. 10. Besides, the decreasing rate of a convex function can be adjusted to have a slower drop at the start.

From the discussion above and the feedback obtained in the survey and expert review, we propose using dynamic scoring based on the total number of solutions. In addition, we propose the use of a convex decay function. The decay function used in Aachen 34C3 CTF (https, 2023) (also suggested in the survey) is an example of such a decay function:

$$int(round(30 + 470/(1 + (max(0, solves - 1)/11.92201) ** 1.206069)))$$

7.2. Attack-defense

We propose a generalized scheme of weighted sums for attack-defense scoring where the weights can be adjusted according to the desired purpose of the gameplay. The final formula is based on the results obtained from running simulations while considering the survey results. In addition, the proposed formula was shared with experts for comments; this feedback was also incorporated into the final design.

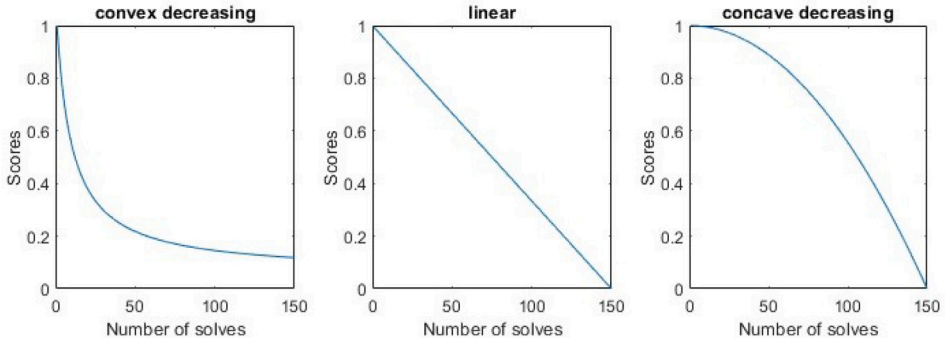


Fig. 9. Jeopardy decay function types (Sydsæter & Hammond, 2008).

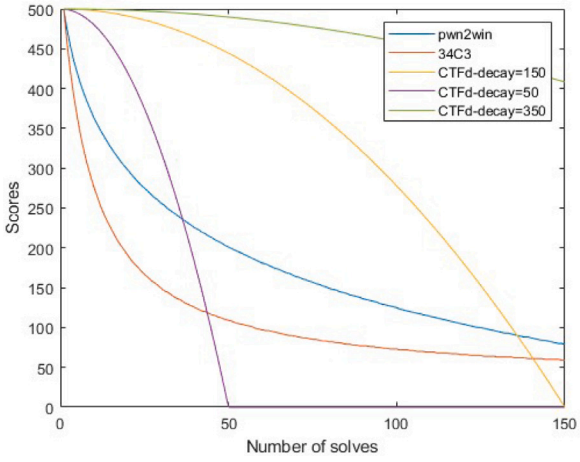


Fig. 10. Jeopardy — different score decay functions.

7.2.1. Attack

The following formula gives attack points. For each flag captured by a team, the attack scores are incremented by:

$$Attack = base\_score + w_d * \frac{1}{flag\_capture\_count} + w_r * \left\{ \begin{array}{ll} 1 & \text{if } pos_A \geq pos_V \\ \frac{1}{500}(30 + \frac{470}{1+(pos_V - pos_A/11.92)^{1.2}}) & \text{else} \end{array} \right\} \quad (1)$$

Here, w_d and w_r are weights of dynamic and rank-based parts respectively, the flag_capture_count is the number of captures of the victim's flag by all teams, and pos_A and pos_V are ranks of attacker and victim teams at the beginning of the round respectively.

The attack score consists of a base score, a dynamic part depending on the number of teams that capture a particular flag, and the victim and the attacker ranks. The base score is universal for all teams and flags. It can be thought of as the number of flags captured by the team. The second part, similar to the dynamic part in jeopardy scoring, measures the difficulty of capturing a flag. The last part is similar to the Elo rating system, and awards points based on the difference in victim and attacker ranks.

The rank-based part is added to support weaker teams, as suggested in the survey results. The literature review also identified this practice as explained in 5.2.4. The decay functions have more significance for the rank-based part because these scores are not common for all teams, unlike the dynamic part, which depends on the number of solves and is the same for all teams. We experimented with different decay functions and weights and analyzed various parts' attack scores and contributions in the total scores for all teams. The plots of different variations of decay functions are shown in Fig. 11.

As shown in Fig. 11, the inverse function has a steep drop. As a result, the contribution from this part reduces to 20% when the difference in rankings reaches 5. The expert feedback suggested that the inverse function is unsuitable for the rank-based part due to its drastic drop, as it will result in very few scores for the stronger teams. The following two plots show the inverse function with decreased slopes. A disadvantage of the generalized inverse function with reduced slope is that the slopes depend on the number of teams involved. So, by registering a large number of fake teams, it can be made to behave like an almost horizontal line. The same argument stands against a linear function that also depends on the number of teams. The next two plots are of the CTFd formula with varying score-decay thresholds. These threshold values should also be set according to the number of participants. Lastly, the decay formula from 34C3 CTF is used in the proposed algorithm. It has a steady drop and does not rely on the number of participants.

The weights and the resulting contribution of different parts to attack scores were also simulated on the existing dataset. The Table 15 has attack scores sorted in decreasing order, including the contribution from different components to the total attack score when the weights are set to 1. As can be seen from the table, the scores are fairly divided, with the top-scoring teams having a lesser contribution from the rank-based part, whereas the low-scoring teams have relatively more contribution (up to 50%) from the rank-based part. The top-scoring teams' scores were mainly based on the base score, whereas the low-scoring teams had up to 50% contribution from the Elo-based points. Also, it can be observed that several teams that have fewer base scores (e.g., teams 8 and 17) than teams ranked lower than them due to the other two components.

The rank-based part in the initially proposed attack scores formula consisted of an inverse function as shown in Eq. (2). However, it was replaced with a less steep decay function as per the feedback received from experts. The results obtained by simulating both formulas on ECSC 2022 data are added in Tables 14 and 15. It can be seen from the results that the high-ranked teams have relatively less contribution from Elo-Part in the case of the inverse decay function (Table 14), and the opposite in the case of the Aachen 343C CTF (Table 15). Since this factor decides the difficulty of the CTF, the user is left to determine the rank-based scoring as per the intended gameplay.

$$Rank - based score = \left\{ \begin{array}{ll} 1 & \text{if } pos_A \geq pos_V \\ \frac{1}{(pos_V - pos_A)} & \text{else} \end{array} \right\} \quad (2)$$

Weights of different components should be based on some reasoning according to the desired gameplay and expected outcomes. As an example of its importance, by setting the weight of the dynamic part to 10, one could make the team with ID 6 win, which is now ranked 5. But doing this without any reasoning shall be viewed with suspicion and can be termed unfair. Hence, it is crucial to set the weights before the event has started for transparency and fairness.

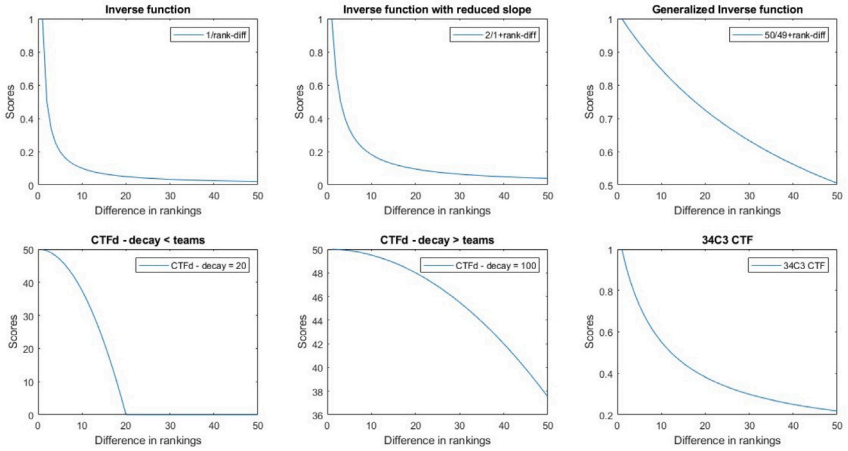


Fig. 11. Plot showing decay functions for AD rank-based score.

7.2.2. SLA

The SLA points increment for working services each round.

$$SLA = \begin{Bmatrix} 1 + w_d + w_r & \text{if } service\_is\_running \\ 0 & \text{else} \end{Bmatrix}$$

Now, it is essential that the score for SLA is comparable to attack and defense scores and is weighted more per flag. As highlighted in the survey results, if SLA gives fewer scores than attack and defense, a team might still win even if they have intentionally turned their services off. The scoring formula should discourage such behavior because it is unfair to other teams, and if all teams resort to such behavior, then no team would be able to attack other teams. On the contrary, it is counterproductive if SLA contributes the most to the total score because a team will score many points just by keeping the services on, even if it has a weak attack and defense. So, the points received for SLA are set to the maximum score one can obtain by a successful attack. However, since we know that SLA is only obtained for one's services, whereas attack scores are obtained for all other teams, attack scores contribute the most to the total scores. By setting the weights for dynamic and rank-based parts to 1, we get the following formula for SLA:

$$SLA = \begin{Bmatrix} 3 & \text{if } service\_is\_running \\ 0 & \text{else} \end{Bmatrix}$$

7.2.3. Defense

The defense points increment every round by the following formula for each successfully defended service:

$$Defense = (2 + 1 * \frac{1}{num\_defended\_teams}) * \begin{Bmatrix} 1 & \text{if } service\_is\_running \\ 0 & \text{else} \end{Bmatrix}$$

Here, the maximum score obtained by a successful defense is set to 3, equal to the maximum score for attack and SLA. One important thing to note is that the defense scores are multiplied by the SLA scores in that round to ensure that the defense scores are never more than the SLA scores. Here, the aim again is to discourage the behavior of shutting down services.

7.2.4. Total

The total score for each team is a sum of all components and is given by the following:

$$Total = \sum_{a=1}^{N_a} Attack_a + \sum_{r=1}^{N_r} (Defense + SLA)$$

where N_a is the number of successful attacks by the team, and N_r is the total number of rounds. As pointed out in the literature review section, summation is used instead of multiplication to avoid having interdependence in attack and defense scores.

7.3. Real world deployment

The proposed attack-defense algorithm was implemented and used in the European Cybersecurity Competition 2023. Documentation for this can be found at ECSC Documentation.[4] This algorithm underwent rigorous evaluation and testing by ENISA's steering committee and jury members. The comprehensive testing process confirmed that the algorithm met all specified requirements as expected. This validation demonstrates the algorithm's effectiveness in a competitive cybersecurity environment.

In addition to the attack-defense algorithm, the Jeopardy algorithm[5] used in ECSC was a variation of a convex decay function as proposed in this research. Furthermore, the convex decay algorithm has a proven track record and is frequently employed in CTF competitions. One such example is its use in the events documented in the 35C3 CTF FAQ.[6] The widespread adoption of the Jeopardy algorithm underscores its reliability and the CTF community's trust in its performance.

8. Discussion & conclusion

The study shows the vast array of CTF scoring algorithms currently being used, non-systematic or unordered, which can leave the players unsure about the strategies to adopt while participating in a CTF. This research attempted to gather the different flavors of CTF scoring in one place and explain the desired behavior while highlighting their commonalities and differences for the CTF organizers and players alike. The results from the survey provided the key requirements and properties that should be present in a CTF scoring algorithm. They helped understand the community's expectations of the CTF scoring. The opinions shared in the survey were key in shaping the final proposed algorithm.

Furthermore, the importance of CTF scoring in a CTF outcome was confirmed by running simulations. The change in teams' rankings by changing the scoring algorithm supports our proposition about the importance of scoring algorithms and the need for having a fair and balanced scoring algorithm. It can be argued that the simulations are a bit unfair because the players might have played differently had they been aware of this new scoring during the CTF. We believe the premise of this argument further proves our point. However, this can

[4] https://ecsc.no/docs/scoring.
[5] https://ecsc.no/docs/scoring.
[6] https://archive.aachen.ccc.de/35c3ctf.ccc.ac/faq/index.html.

Table 14
Attack score components — Inverse decay in Elo part.

Team-ID	Base points	Elo-Part	Dynamic-Part	Attack total
7	22 384	2974.01	3351.99	28 710
16	22 270	2818.31	3046.12	28 134.43
28	21 504	2953.18	3034.9	27 492.08
11	21 356	2356.36	3195.59	26 907.95
6	18 426	3175.56	3902.94	25 504.5
29	15 914	3106.63	1677.99	20 698.62
10	15 353	3193.3	2018.84	20 565.13
22	13 645	3819.77	1968.73	19 433.49
20	12 908	3419.49	1180.13	17 507.62
9	12 262	2737.69	1532.11	16 531.8
24	9 779	2715.21	1343.66	13 837.87
2	9 122	2922.1	1043.96	13 088.06
3	8 423	3819.1	734.49	12 976.59
8	8 373	3184.06	1176.83	12 733.89
14	9 113	2749.33	767.09	12 629.42
30	7 496	4018.38	744.83	12 259.21
27	7 731	3116.73	971.96	11 819.69
21	6 968	3932.8	509.79	11 410.59
23	5 561	2781.55	366.07	8708.61
33	5 418	2708.09	486.51	8612.6
13	4 917	3045.41	330.81	8293.21
31	5 087	2545.39	425.15	8057.54
25	4 520	2982.12	459.7	7961.82
17	4 017	3380.5	329.21	7726.71
5	4 084	2941.65	262.46	7288.11
18	3 705	2957.14	277.3	6939.45
34	4 218	2211.14	269.81	6698.95
15	3 429	2947.27	228.16	6604.43
19	3 783	2539.25	219.93	6542.17
32	1 656	1563.83	86.47	3306.31
26	1 412	1396	85.28	2893.28
4	1 230	1192.42	75.08	2497.49
12	714	712.83	43.14	1469.98
1[a]	0	0	0	0

[a] Team-id 1 is not a real team (NOP). It is there just to check exploits.

Table 15
Attack score components — Aachen decay in Elo part.

Team-ID	Base points	Elo-Part	Dynamic-Part	Attack total
7	22 384	11 193.46	3351.99	36 929.45
16	22 270	10 973.39	3046.12	36 289.51
28	21 504	10 949.33	3034.9	35 488.23
11	21 356	10 163.54	3195.59	34 715.13
6	18 426	9932.87	3902.94	32 261.8
29	15 914	9068.47	1677.99	26 660.46
10	15 353	8783.12	2018.84	26 154.95
22	13 645	8726.1	1968.73	24 339.82
20	12 908	8131.4	1180.13	22 219.53
9	12 262	7429.83	1532.11	21 223.94
24	9 779	6242.76	1343.66	17 365.42
2	9 122	6144.39	1043.96	16 310.35
14	9 113	6058.75	767.09	15 938.84
3	8 423	6578.15	734.49	15 735.63
8	8 373	6045.45	1176.83	15 595.27
30	7 496	6164.13	744.83	14 404.95
27	7 731	5681.24	971.96	14 384.2
21	6 968	5907.85	509.79	13 385.63
23	5 561	4505.75	366.07	10 432.81
33	5 418	4304.2	486.51	10 208.71
31	5 087	4086.86	425.15	9599.01
13	4 917	4299.53	330.81	9547.34
25	4 520	4034.35	459.7	9014.05
17	4 017	3889.67	329.21	8235.88
5	4 084	3734.21	262.46	8080.67
34	4 218	3418	269.81	7905.8
18	3 705	3526.5	277.3	7508.81
19	3 783	3349.32	219.93	7352.25
15	3 429	3336.17	228.16	6993.33
32	1 656	1645.01	86.47	3387.48
26	1 412	1410.56	85.28	2907.83
4	1 230	1225.15	75.08	2530.22
12	714	713.86	43.14	1471
1[a]	0	0	0	0

[a] Team-id 1 is not a real team (NOP). It is there just to check exploits.

be further investigated by conducting full-fledged CTF exercises and studying their outcomes.

We realize that the meanings of "fairness" and "balanced" for a scoring algorithm can be interpreted differently in different scenarios. The proposed algorithms were our attempt at creating a scoring that can be used in most of the general capture-the-flag events where a balanced approach is used, and the focus is not on any particular aspect of the game. This may not work in some exercises where the aim is to put more emphasis on one aspect, e.g., a blue-teaming exercise will have defense as the most important factor in determining the score. However, we believe that the approach presented in this paper can be adapted to implement the needs of these kinds of cybersecurity exercises in general.

Despite the progress made in understanding the scoring algorithms in CTFs, there are several avenues for future research and improvements in this field. This work mainly focussed on two CTF types, i.e., Attack-defense and Jeopardy. Future work could also analyze the other two CTF formats. Moreover, future work can also address the limitations of the survey conducted during this research (as described in Section 4.2). It can be extended to include participants from places other than Europe. It can also be extended to CTF players but in a controlled way, taking care of the limitations of this approach. The survey can be further improved by reviewing the literature and using that knowledge to prepare more specific questions.

CRediT authorship contribution statement

Abdullah Zafar: Literature review, Implementation, Writing – original draft. **Muhammad Mudassar Yamin:** Idea, Literature review, Data collection, Review, Funding acquisition. **Basel Katt:** Idea, Data collection, Review, Funding acquisition. **Espen Torseth:** Data collection, Review.

Declaration of competing interest

The authors declare no conflict of interest in publishing the article.

Data availability

Data will be made available on request.

Acknowledgments

We want to express our sincere gratitude for the generous funding provided by the Center for Cyber and Information Security (CCIS), Norway. Their support has played a crucial role in successfully executing this work. Additionally, we would like to thank the European Union Agency for Cybersecurity (ENISA) for their valuable support throughout the project.

Appendix. Tables

See Tables 14 and 15.

References

abza1/AD-scoring-simulator: attack defense scoring algorithms simulator. (2023). https://github.com/abza1/AD-scoring-simulator. (Accessed on 11 March 2023).

Capture the flag - hackathon - hacking contest | trend micro (US). (2024a). https://www.trendmicro.com/en_us/campaigns/capture-the-flag.html?modal=s6d-btn-event-rules-2012fe. (Accessed on 15 May 2024).

Capture the flag - hackathon - hacking contest | trend micro (US). (2024b). https://www.trendmicro.com/en_us/campaigns/capture-the-flag.html?modal=s6d-btn-event-rules-2012fe. (Accessed on 16 May 2024).
Chapman, Peter, & Brumley, David (2013). *picoctf: Teaching 10,000 high school students to hack*.
Childers, Nicholas, Boe, Bryce, Cavallaro, Lorenzo, Cavedon, Ludovico, Cova, Marco, Egele, Manuel, et al. (2010). Organizing large scale hacking competitions. In *International conference on detection of intrusions and malware, and vulnerability assessment* (pp. 132–152). Springer.
CODE CTF 2019 the 5th element : CONCORDIA. (2024). https://www.concordia-h2020.eu/blog-post/code-ctf-2019-the-5th-element/. (Accessed on 16 May 2024).
Creswell, John W. (1999). Mixed-method research: Introduction and application. In *Handbook of educational policy* (pp. 455–472). Elsevier.
CTF 2006 qualifiers final scores. (2024). https://web.archive.org/web/20090407023103/http://kenshoto.com/ctf06/quals_final.html. (Accessed on 16 May 2024).
Ctf-ad-rules/readme.md at master · cyberchallengeit-ve/ctf-ad-rules. (2024). https://github.com/cyberchallengeit-ve/ctf-ad-rules/blob/master/README.md. (Accessed on 16 May 2024).
CTF events — ENISA. (2022). https://www.enisa.europa.eu/publications/ctf-events. (Accessed on 09 November 2022).
Ctftime.org / about. (2022). https://ctftime.org/about/. (Accessed on 07 November 2022).
Ctftime.org / all about CTF (capture the flag). (2023). https://ctftime.org/event/list/?year=2023. (Accessed on 30 March 2023).
Davis, Andy, Leek, Tim, Zhivich, Michael, Gwinnup, Kyle, & Leonard, William (2014). The fun and future of $\{CTF\}$. In *2014 USeNIX summit on gaming, games, and gamification in security education (3GSE 14)*.
DEF con® hacking conference - CTF history. (2022). https://defcon.org/html/links/dc-ctf-history.html. (Accessed on 07 November 2022).
Diakoumakos, Jason, Chaskos, Evangelos, Kolokotronis, Nicholas, & Lepouras, George (2021). Cyber-range federation and cyber-security games: A gamification scoring model. In *2021 IEEE international conference on cyber security and resilience* CSR, (pp. 186–191). IEEE.
Diutinus defense techonologies corp. / CTF history. (2022). https://web.archive.org/web/20160315221404/http://ddtek.biz/about-ctf.html. (Accessed on 10 November 2022).
Diutinus defense techonologies corp. / CTF history. (2024). http://ddtek.biz/about-ctf.html. (Accessed on 16 May 2024).
Dynamic value | CTFd docs. (2022). https://docs.ctfd.io/docs/custom-challenges/dynamic-value. (Accessed on 19 November 2022).
ECSC 2020 analysis report — ENISA. (2022). https://www.enisa.europa.eu/publications/ecsc-2020-analysis-report. (Accessed on 12 November 2022).
Elo rating system - wikipedia. (2022). https://en.wikipedia.org/wiki/Elo_rating_system. (Accessed on 19 November 2022).
Enowars/ctf-dumps. (2023). https://github.com/enowars/ctf-dumps. (Accessed on 26 February 2023).
Enowars/EnoEngine. (2023). https://github.com/enowars/EnoEngine. (Accessed on 26 February 2023).
Facebook CTF. (2022). https://web.archive.org/web/20190531061639/https://www.fbctf.com/. (Accessed on 16 May 2024).
Hints | CTFd docs. (2022). https://docs.ctfd.io/docs/challenges/hints. (Accessed on 12 November 2022).
HITB SECCONF CTF 2022. (2022). https://2022.ctf.hitb.org/hitb-ctf-singapore-2022/rules. (Accessed on 10 November 2022).
Hoffman, Lance J, Rosenberg, Timothy, Dodge, Ronald, & Ragsdale, Daniel (2005). Exploring a national cybersecurity exercise for universities. *IEEE Security & Privacy*, *3*(5), 27–33.
How much do football leaderboards change throughout a season? | linkedin. (2023). https://www.linkedin.com/pulse/how-much-do-football-leaderboards-change-throughout-season-pisello/. (Accessed on 20 February 2023).
https://archive.aachen.ccc.de/34c3ctf.ccc.ac/faq/index.html. (Accessed on 27 March 2023).
Ictf2021. (2024). https://web.archive.org/web/20230531004500/https://shellphish.net/ictf/archive/ictf_2021/competition_website/howto.html. (Accessed on 16 May 2024).
Internet archive: Wayback machine. (2022). https://archive.org/web/. (Accessed on 07 November 2022).
Ncr-no/ctf-scoring-simulator. (2023). https://github.com/ncr-no/ctf-scoring-simulator. (Accessed on 26 May 2023).
O-o-overflow/scoring-playground: Tool to test different CTF scoring algorithms on real data. (2023). https://github.com/o-o-overflow/scoring-playground. (Accessed on 23 January 2023).
OOO — DEF con CTF. (2022). https://oooverflow.io/dc-ctf-2021-finals/. (Accessed on 09 November 2022).
OOO — DEF con CTF. (2023). https://oooverflow.io/. (Accessed on 23 January 2023).
OOO philosophy | OOO — DEF con CTF. (2024). https://oooverflow.io/philosophy.html. (Accessed on 16 May 2024).
Picoctf - CMU cybersecurity competition. (2022). https://picoctf.org/. (Accessed on 10 November 2022).
Plaidiverse. (2022). https://plaidctf.com/rules. (Accessed on 10 November 2022).
Polictf 2015. (2022). https://2015.polictf.it/instructions. (Accessed on 12 November 2022).
Polictf 2017 | instructions. (2022). https://2017.polictf.it/instructions.html. (Accessed on 12 November 2022).
Polictf 2015. (2024). https://2015.polictf.it/instructions. (Accessed on 16 May 2024).
Price, Benjamin, Zhivich, Michael, Thompson, Michael, & Eagle, Chris (2018). House rules: Designing the scoring algorithm for cyber grand challenge. *IEEE Security & Privacy*, *16*(2), 23–31.
Pwn2winctf/nizkctf-audit-trail: NIZKCTF audit trail. (2023). https://github.com/pwn2winctf/nizkctf-audit-trail. (Accessed on 26 January 2023).
Ructf. (2024). https://ructf.org/2022/en/rules. (Accessed on 16 May 2024).
Ructf 2021 rules. (2024). https://web.archive.org/web/20210922134244/https://ructf.org/rules.html. (Accessed on 16 May 2024).
Rules. (2022). https://capturetheflag.withgoogle.com/rules. (Accessed on 12 November 2022).
Rules | FAUST CTF 2022: Hack to the future. (2022). https://2022.faustctf.net/information/rules/. (Accessed on 10 November 2022).
Saarctf. (2022). https://ctf.saarland/rules. (Accessed on 10 November 2022).
Scoreboard - NIZKCTF. (2023). https://pwn2.win/ranking. (Accessed on 19 March 2023).
Scoring - ECSC 2022. (2022). https://docs.ecsc2022.eu/scoring/. (Accessed on 07 November 2022).
Swann, Matthew, Rose, Joseph, Bendiab, Gueltoum, Shiaeles, Stavros, & Li, Fudong (2021). Open source and commercial capture the flag cyber security learning platforms-a case study. In *2021 IEEE international conference on cyber security and resilience* CSR, (pp. 198–205). IEEE.
Sydsæter, Knut, & Hammond, Peter J. (2008). *Essential mathematics for economic analysis*. Pearson Education.
Werther, Joseph, Zhivich, Michael, Leek, Tim, & Zeldovich, Nickolai (2011). Experiences in cyber security education: The $\{MIT\}$ lincoln laboratory $\{Capture-the-Flag\}$ exercise. In *4th workshop on cyber security experimentation and test (CSET 11)*.

Abdullah Zafar is currently doing his Master of Information Security from Norwegian University of Science and Technology. He is also working as a research assistant in Norwegian cyber range and is involved in various research activities.

Muhammad Mudassar Yamin is currently working as an Associate Professor at the Department of Information and Communication Technology at the Norwegian University of Science and Technology (NTNU). He is a member of the system security research group, and the focus of his research is on system security, penetration testing, security assessment, and intrusion detection. Before joining NTNU, Mudassar worked as an Information Security consultant and served multiple government and private clients. He holds multiple cybersecurity certifications, such as OSCE, OSCP, LPT-MASTER, CEH, CHFI, CPTE, CISSO, and CBP.

Basel Katt is currently working as a Professor at the Department of Information and Communication Technology at the Norwegian University of Science and Technology. He is the technical project leader of Norwegian cyber range. Focus of his research Is:

* Software security and security testing.
* Software vulnerability analysis.
* Model driven software development and model driven security.
* Access control, usage control and privacy protection.
* Security monitoring, policies, languages, models and enforcement.

Espen Torseth is working as a senior advisor in Norwegian University of Science and Technology. He is the administrator of Norwegian cyber range and handle its daily operations and cyber security exercises.