Inter-Temporal Reward Strategies in the Presence of Strategic Ethical Hackers

Jing Hou, Xuyu Wang, *Member, IEEE*, and Amy Z. Zeng

***Abstract*— A skyrocketing increase in cyber-attacks significantly elevates the importance of secure software development. Companies launch various bug-bounty programs to reward ethical hackers for identifying potential vulnerabilities in their systems before malicious hackers can exploit them. One of the most difficult decisions in bug-bounty programs is appropriately rewarding ethical hackers. This paper develops a model of an inter-temporal reward strategy with endogenous e-hacker behaviors. We formulate a novel game model to characterize the interactions between a software vendor and multiple heterogeneous ethical hackers. The optimal levels of rewards are discussed under different reward strategies. The impacts of ethical hackers' strategic bug-hoarding and their competitive and collaborative behaviors on the performance of the program are also evaluated. We demonstrate the effectiveness of the inter-temporal reward mechanism in attracting ethical hackers and encouraging early bug reports. Our results indicate that ignoring the ethical hackers' strategic behaviors could result in setting inappropriate rewards, which may inadvertently encourage them to hoard bugs for higher rewards. In addition, a more skilled e-hacker is more likely to delay their reporting and less motivated to work collaboratively with other e-hackers. Moreover, the vendor gains more from e-hacker collaboration when it could significantly increase the speed or probability of uncovering difficult-to-detect vulnerabilities.**

***Index Terms*— Ethical hacker, vulnerability market, strategic behavior, bug bounty.**

I. Introduction

ETHICAL hackers commit their time and effort in vulnerability markets to uncover and report vulnerabilities to software developers or vendors. The software vendors then offer monetary rewards to the ethical hacker (e-hacker)[1] for helping and fixing the bugs, preventing data breaches and cyber-attacks. Many companies, including Mozilla, Google, and Microsoft, have relied on e-hackers to discover security vulnerabilities by launching the so-called bug-bounty programs [1].

Reward pricing is one of the most challenging decisions companies must make [2]. On the one hand, both the severity and the difficulty levels of bugs must be considered in compensating the e-hackers. If the rewards are the same for the obvious defects and those that are difficult to find, the e-hackers would only pick the low-hanging fruits and lack incentives to dig the hard ones continuously. On the other hand, as the vendor keeps patching the vulnerabilities reported by the e-hackers, the software gets harder to exploit. For example, in one of Google's bounty programs, the rate of critical bug reports slowed down after the program picked up, and Google had to add bonuses as the program progressed [3]. Therefore, to induce e-hackers to exert more time on those hard-to-detect bugs, inter-temporal pricing in which the reward grows over time has been proposed in the literature and applied in practice [2], [4]. One famous example is Donald E. Knuth's reward of initially 1.28 USD for each bug in his TEX typesetting system, which grows exponentially with the number of years the program is in use [5]. Indeed, it's quite common for bug bounty programs to increase rewards to motivate deeper security efforts. For instance, in 2019, Microsoft raised top award levels from $15K to $50K for the Windows Insider Preview bounty and from $15K to $20K for the Microsoft Cloud Bounty program [6].

However, the inter-temporal reward strategy also motivates strategic e-hackers to withhold their bug submission and wait for reward enhancement. In reality, it is very common that e-hackers hoard bugs for higher profits [3]. Yet the vendors cannot tell whether the e-hackers withhold the reports or have not discovered the bugs. There is constantly a cat-and-mouse game between software vendors, who hope to spend less money and to induce earlier vulnerability reports, and e-hackers, who wait for a higher reward. Consequently, companies are continuously enhancing their rewards to attract as many e-hackers and as early reports as possible, and the e-hackers are constantly modifying their report plans to earn as much as possible. Besides, the situation becomes more intricate for the e-hackers when they may not benefit from blindly waiting for a higher reward. It is now more likely that the other e-hackers will discover the same bug before them if they wait too long. In this case, the e-hackers will not be rewarded for late reports of soon-to-be-fixed bugs.

Motivated by the interactions between the software vendors and the e-hackers, this paper studies how inter-temporal reward

Manuscript received 17 November 2023; revised 21 May 2024; accepted 23 June 2024; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor E. E. Tsiropoulou. Date of publication 8 July 2024; date of current version 17 October 2024. This work was supported in part by NSF under Grant CNS-2321763, Grant CNS-2319343, and Grant CNS-2317190. The work of Jing Hou was supported by California State University San Marcos. *(Corresponding author: Jing Hou.)*

Jing Hou is with the Department of Computer Science and Information Systems, California State University San Marcos, San Marcos, CA 92096 USA (e-mail: jhou@csusm.edu).

Xuyu Wang is with the School of Computing and Information Sciences, Florida International University, Miami, FL 33199 USA (e-mail: xuywang@fiu.edu).

Amy Z. Zeng is with the Sawyer Business School, Suffolk University, Boston, MA 02108 USA (e-mail: azeng@suffolk.edu).

Digital Object Identifier 10.1109/TNET.2024.3422922

[1]For brevity, we will call the ethical hackers *e-hackers* in this article. We do not consider the malicious hackers.

1558-2566 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

strategy and the e-hackers' strategic behaviors, and particularly their interdependence, affect the bug-bounty program and e-hackers' utilities. Although the importance of strategic e-hacker behavior is widely recognized, little research has studied its implications on the reward strategy in vulnerability markets. Given this gap in the literature, our research aims to provide more effective solutions for companies working with strategic e-hackers to secure and improve their technologies.

We consider one or multiple e-hackers who can participate in a software vendor's bug-bounty program and get rewarded for vulnerability reports. The vendor adopts an inter-temporal reward strategy, which enhances the reward after a specific time during the program. Its decisions of p (regular reward) and s (enhanced reward) influence the e-hackers' to-report-or-to-wait decisions. To incorporate strategic e-hacker behavior, we allow the e-hackers to either submit their report once they discover the vulnerability or wait for a higher reward. If no one has reported the same vulnerability before, the e-hackers who wait may get a high reward; otherwise, they end up without the reward.

The research questions we are interested in are: (1) How do hackers of different skill levels react to the companies' reward strategies? (2) When e-hackers are strategic, i.e., they tend to withhold bug submission for reward enhancement, how can firms determine appropriate rewards that discourage bug hoarding? (3) What are the impacts if the e-hackers' strategic behaviors exist but are ignored? (4) Does an inter-temporal reward strategy benefit the vendor or the e-hackers? (5) How does the number of e-hackers affect the results? and (6) How do competition and collaboration between e-hackers influence their participation and report decisions? Additionally, when does the vendor benefit from their competition or collaboration? The problems are challenging due to the complex interrelationships among the decisions, the information asymmetry, and the uncertain nature of vulnerability discovery. In particular, the e-hackers' participation decisions are determined by their expected utility gain from the program, which is affected by their to-report-or-to-wait decisions and the vendors' reward decisions. The competition among multiple e-hackers makes these decisions even more complex. From the software vendor's perspective, an over-low reward would not attract potential e-hackers to join or stay in the program, while an over-high reward might induce a reporting delay. Such decisions are difficult to make as the existence of a particular type of bug is uncertain, and the vendor lacks knowledge about the actual state of e-hackers' discovery progress. Therefore, a systematic decision support model is needed to evaluate the effectiveness of the inter-temporal reward strategy and the impacts of e-hackers' strategic behaviors. To capture the interactions among multiple players with information asymmetry, we will model the game among the e-hackers as Bayesian games with incomplete information and the vendor's reward decisions in a Stackelberg game.

The main contributions of this paper are stated as follows:

- We take the first step towards formulating the software vendor's inter-temporal reward decisions when facing strategic e-hackers who may hoard the vulnerability and wait for a higher reward. A unified framework is provided by integrating the Bayesian game and the Stackelberg game model to account for the competition or collaboration among e-hackers with incomplete information and the cooperation between the software vendor and the e-hackers. We find that the vendor benefits more from e-hacker collaboration when it could significantly enhance the speed or likelihood of uncovering difficult-to-detect vulnerabilities.
- Although it is intuitively clear that e-hackers' strategic behaviors of hoarding bugs have negative effects on bug-bounty programs, the economic mechanisms underlying these effects are still unexplored. In contrast to most recent work, which presumes myopic e-hacker behavior, our work evaluates the impacts of strategic e-hacker behavior and the competition between them. We show that ignoring the e-hackers' strategic behaviors might lead to inappropriate rewards, which encourages them to hoard bugs for higher rewards.
- To investigate the behavior and decision-making of e-hackers with different skill levels, we incorporated the heterogeneity between e-hackers into our model. The equilibrium outcomes (i.e., rewards, e-hackers' participation and report decisions) and the conditions for the existence of the equilibrium are characterized. We find that less experienced hackers tend to report vulnerabilities immediately and have a stronger incentive to collaborate with peers. The impact of reward strategies on the differences between their expected utilities is also explored.
- Finally, we discuss three different reward strategies — static, increasing, and decreasing — to evaluate the effectiveness of an inter-temporal reward strategy. Our results indicate that if the vendor does not plan to enhance the reward afterward, it is advisable to set a higher initial reward. Overall, an inter-temporal reward strategy could save costs for the vendor in bug-bounty programs.

The remainder of this paper is organized as follows. Section II provides a literature review. Section III introduces the model setups. In Section IV, we analyze the game models and derive the equilibrium solutions with two competing e-hackers. Section V discusses the results of different reward strategies. In Section VI, the collaboration between the e-hackers is modeled, and Section VII extends the model to the case of multiple e-hackers. Finally, we briefly conclude in Section VIII.

II. Literature Review

As bug bounty programs enter the mainstream security practice in organizations [7], researchers are increasingly interested in the vulnerability markets. Existing studies have shown that one primary motivation of e-hackers is to receive rewards (others include making the product secure, helping users and developers, and receiving reputation) [8]. However, firms lack rigorous principles and explicit rules for setting bounty amounts in rewarding the e-hackers [9], [10]. Market incentive mechanisms are needed to change hackers' motivations and eventually give firms more control over the disclosure process [11]. To this end, literature and industrial practice have

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

proposed different rewarding policies and incentive mechanisms. The objectives include reducing the number of invalid reports [12], preventing adversarial copying and resubmission of bugs [10], incentivizing deep fixes instead of suppressing the symptoms [13], and fairness guarantees [14]. One of the most related works is [4] which proposes a dynamic reward strategy such that the reward increases continuously with time to free the firm from any need to understand the e-hackers' costs in order to set prices and maximize value.

To assess the effectiveness of the proposed reward strategies, we need a better understanding of the e-hackers' responses and behaviors. Therefore, a large body of empirical studies has been taken to investigate the behaviors of the e-hackers, such as competition [4] and switching programs [15]. However, a quantitative framework or decision-making model is lacking to support firms' inter-temporal reward decisions and evaluate the impacts of e-hackers' strategic behaviors. Although some quantitative studies have been conducted to analyze malicious hackers' attack behaviors, such as their hacking strategies [16], [17], [18] and information sharing strategies [19], their results cannot be applied to the bug-bounty programs as different types of hackers are motivated differently. Our work differs from the aforementioned literature in several ways. First, our focus is to quantitatively measure the effectiveness of software vendors' employing inter-temporal reward strategies in attracting e-hackers. Second, we explore the impact of the e-hackers' bug-hoarding behaviors on the bug-bounty programs, which is neglected in the literature. Finally, we explicitly examine how the vendors could make use of the competition and/or collaboration between the e-hackers to regulate their behaviors by optimizing the reward decision.

We have also reviewed papers on dynamic pricing in the presence of strategic customer behaviors in the marketing and operations research field. This stream of literature is devoted to modeling firms' optimal product pricing decisions when customers can choose to wait for a sale instead of purchasing the product at a regular price [20], [21], [22], [23]. The results of these studies cannot be applied to reward pricing decisions due to the uncertainty in bug hunting and the changes in the value of a bug report: First, all the customers make the to-purchase-or-to-wait decision, while only the e-hackers who have found the bug need to make the to-report-or-to-wait decision. Second, it is assumed that each customer is infinitesimally small, meaning that one customer's decision won't affect the others [22], while normally, there would not be many e-hackers that could find the same bug. Even if there are, as long as one successful e-hacker reports the bug, then the other e-hackers who report the same vulnerability later will not receive any reward. That being said, a single e-hacker's decision would affect all the other e-hackers' utilities. To support decision-making in vulnerability market practices, our research explains the underlying economics for the collaboration between software vendors and e-hackers. We optimize the vendors' reward decisions by exploring their interactions with multiple strategic e-hackers who compete with each other. Our results offer new insights into the role of e-hackers' behaviors in bug-bounty programs and provide guidelines for effective reward strategies in bug-bounty programs toward a safer software development environment.

Part of the work has been presented at the IEEE CNS 2023 conference in Orlando, FL. Compared to the conference version [24], this paper significantly extends the models and provides more insightful results, including incorporating the heterogeneity among e-hackers, a new model discussing collaboration between e-hackers, a framework for multiple e-hackers scenarios, and the analysis of different reward strategies.

III. Problem Formulation and Model Setup

A. Bug-Bounty Program

We consider a single software vendor that offers a bug-bounty program to e-hackers to discover and resolve critical vulnerabilities in its system. The bug-bounty program is operated over a finite time horizon before the software system's release time. After this period, no bug reports will be accepted or rewarded, though the vendor may initiate another bug-bounty program post-launch. This time-bound bug bounty helps identify issues early in the development lifecycle of new products and features, facilitating the implementation of architecture and design changes that would be more challenging later. For instance, Microsoft has offered a one-month pilot program to pay e-hackers to scour beta (preview) versions of Internet Explorer and Windows upgrades [3]. Similarly, the "Hack DHS" program launched by the Department of Homeland Security in 2021 had two initial phases of vulnerability identification and ethical hacking in select external systems, followed by a phase where lessons learned were identified [25]. This allowed the Department to address vulnerabilities not surfaced through other means. In compensating for the e-hackers' continuous effort, the vendor gradually enhances the bounty or reward over time, which at least covers the e-hacker's cost [2], [4]. We normalize the total length of the time horizon to one (time unit) and consider a two-period reward pricing policy [26], [27], [28]. During Period I, i.e., in $t \in [0, t_0]$ with $t_0 \in (0, 1)$ predetermined by the vendor, the reward for a valid vulnerability report is initialized at a low level p. If no valid report is submitted, the reward will be increased to s $(0 < p < s)$ after t_0. However, if a particular vulnerability has been reported in Period I, no further report will be rewarded on that vulnerability in Period II. This pricing strategy is reasonable because if a bug takes the e-hackers more time or effort, the vendor shall reward more to compensate the e-hackers.

The model's timing proceeds as follows:

Step 1. The vendor launches a bug-hunting program over a finite time horizon, which is divided into two periods. At the beginning of Period I, the vendor decides and broadcasts the reward p for any valid report submitted in Period I and reward s for the report submitted in Period II. The vendor aims to attract as many e-hackers as possible to participate in the program.

This assumption of pre-announced pricing at the beginning of the marketing season has also been widely discussed in the

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

economics and operations management literature [21], [22]. An alternate rationale for this pricing policy is that the vendor sets the compensation to the e-hacker based on the duration of their search for vulnerabilities. Our model depicts the reward function as a step-wise function, with a value of p if the time spent is less than t_0 and a value of s if the e-hacker spends more than t_0. It is natural to specify the values of the parameters (p, s) at the commencement of the bug-bounty program.

Step 2. The e-hackers then decide whether to participate or not. The vendor will provide the participants access to its bug-hunting system.

Step 3. Upon finding a bug in Period I, the e-hacker decides whether or not to report it immediately to the vendor. After collecting the report, the vendor will assess them and publicize their validity at the end of Period I. If the report is valid, the e-hacker gets a reward of p, and no further report of this bug is accepted in Period II.

Step 4. In Period II, if an e-hacker reports a bug that has not been reported in Period I, it will get a reward s.

B. Strategic E-Hackers

Our model departs from the classic reward model setup by introducing strategic e-hackers. Specifically, if e-hackers discover a bug during Period I and realize that a greater reward could be offered after time t_0, they may opt to delay reporting the bug and wait for the reward to increase to maximize their individual expected surplus. However, there is a risk that the bug will be found by other e-hackers and reported in Period I. In this situation, no reward will be given to the e-hacker who submits the report afterward.

We assume that the probability of an e-hacker finding out the bug is Q, and the probability of early detection, i.e., finding out the bug in Period I, is q_1. Denote $q_2 = Q - q_1$ as the detection probability in Period II. It is possible that a specific type of bug does not exist or the e-hacker could not find it during the given period, the probability of which is $1-Q$. The values of q_1 and Q depend on several factors, including the time allocated for bug hunting (t_0 and the overall duration), the e-hackers' skills, the system's characteristics, and the type of bug being addressed. Estimations for these values can be derived from data about similar systems' reliability, the bug types under investigation, and the vendor's assessment of e-hackers' skills and familiarity with the system based on their previous work experience. Similar assumptions regarding the presumed probability of discovering the vulnerability can also be found in literature such as [29].

For an e-hacker who participates in the bounty program, its utility depends on its to-report-or-to-wait decision. Its expected reward can be written as

$$U = \begin{cases} \gamma Q s & wait \\ q_1 p & submit \end{cases}, \tag{1}$$

where γ is the probability that no other e-hackers submit a report on the bug in Period I, which is determined by the possibility that the other e-hackers find the bug and their to-report-or-to-wait decisions.

We use the amount of time an e-hacker spends on the bug-hunting process as a measure of their cost (as in [12]). For illustration purposes, we'll adopt a step function to delineate the cost associated with each of these periods[2]

$$C = \begin{cases} c_1 & t \le t_0 \\ c_2 & t > t_0 \end{cases}, \tag{2}$$

where t is the time the e-hacker spent on vulnerability discovery and $c_2 > c_1$. That being said, if the e-hacker finds the bug in Period I, its expected cost will be c_1; otherwise, if it continues searching into Period II, its total expected cost, including the one during Period I, will rise to c_2 ($c_2 > c_1$). Note that if one e-hacker finds and reports the bug in Period I, the other e-hackers would stop working on this bug in Period II, and thus its total cost would be c_1. Therefore, measured in the probability of finding the bug, the expected reward, and the cost, the total expected utility of an e-hacker can be written as

$$f = U - q_1 c_1 - (1 - q_1)(1 - \gamma)c_1 - \gamma(1 - q_1)c_2. \tag{3}$$

We would like to examine how their participation and report decisions could be made facing the competition and uncertainty about the other e-hackers' bug-hunting results.

C. Software Vendor

For the software vendor, we are interested in the effectiveness of employing an inter-temporal reward strategy in attracting e-hackers to the bug-bounty program. The vendor needs to decide the values of the reward p and s with a multi-level objective: First, the vendor would like to attract as many e-hackers as possible to participate in the bug-bounty program as more participation contributes to higher productivity of the vulnerability discovery process [30]; Second, the vendor wants the e-hackers to report immediately once they find the bug rather than waiting for a higher reward later so that it could have more time to fix the bug before releasing the system; Finally, the objective is to maximize its expected utility:

$$\pi = \quad (r + R) \cdot Prob_1 + r \cdot Prob_2 - p \cdot N_1 - s \cdot N_2, \tag{4}$$

where r is the monetary benefit from discovering and fixing the bug for the vendor before the system is released into the market and R is the extra benefit of early detection (i.e., in Period I) as the effort and money required to resolve the issue is significantly lower during the earlier stages of development; $Prob_1$ and $Prob_2$ are the probabilities of getting bug reported in Period I and Period II respectively; and N_1 and

[2]This assumption of a simple step-wise cost function allows us to concentrate on the differences between two distinct periods without further subdividing the time. It is valid when each period is sufficiently short or differentiating the e-hacker's cost or discovery probability within each period is either unnecessary or impractical. If the periods are long enough, we can calculate the expected time cost in the following way: As we distinguish between q_1 and q_2 to account for difficulty differences of finding bugs early (in Period I) versus late (in Period II), we assume a uniform probability of discovering a bug *within* each period (similar assumption of uniform distribution can be found in [2]). With a linear cost $C = ct$ where c represents the monetary value per unit time or the opportunity cost of the e-hacker's time, given t_0 as the point dividing the timeline into Period I (before t_0) and Period II (after t_0), the average cost for an e-hacker who discovers the bug in different periods is calculated as $c_1 = \frac{ct_0}{2}$ and $c_2 = c(t_0 + \frac{1-t_0}{2}) = c\frac{1+t_0}{2}$, which are independent of q_1 and q_2.

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

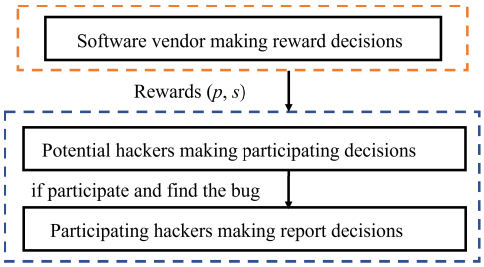


Fig. 1. Three-stage Stackelberg game between the vendor and the e-hackers.

N_2 are the numbers of valid reports in Period I and Period II, respectively.

IV. Competition Models of E-Hackers

In this section, we consider a scenario in which two potential e-hackers (H and L) would participate in the bug-bounty program and make decisions independently. The e-hackers are heterogeneous in terms of their levels of expertise, and consequently, their chances of finding vulnerabilities within the same timeframe differ. Without loss of generality, we assume e-hacker H is more experienced than e-hacker L with $q_{1H} > q_{1L}$ and $Q_H > Q_L$. These skill levels or expertise can often be inferred from their public profiles. For instance, on platforms like HackerOne, white hackers are awarded badges, which symbolize their accomplishments and skills [31]. Suppose the communication between the vendor and the e-hackers is managed through a platform on which information about the reward values, the report validation results, and the profiles of e-hackers are available to everyone. It is worth noting that we consider the scenario in which the vendor discloses the profiles of participating e-hackers to leverage the potential benefits of their competition. However, the true identities of the e-hackers are confidential information to themselves.

The interactions between the vendor and the e-hackers are formulated in a three-stage hierarchical order of decision-making as shown in Fig. 1: In Stage I, the vendor, as the leader in the Stackelberg game, determines the rewards (p, s) and post them on the platform. In Stage II, the e-hackers, as the followers, play a simultaneous game of participating. In Stage III, whenever a participating e-hacker finds a bug in Period I, it needs to determine whether or not to submit the report or wait for a higher reward in Period II. Adopting the backward induction, we first obtain the subgame-perfect Nash equilibrium of the e-hacker's decision on report submission and then analyze their participation decisions. Finally, we obtain the optimal reward decisions for the vendor.

A. Equilibrium Analysis of Participating E-hackers' Report Decisions

In the following analysis, we will focus on the case when both e-hackers participate in the bug-bounty program and first examine whether a pure-strategy equilibrium exists in their report decisions in Period I.

1) Existence of Pure-Strategy Equilibrium: Suppose e-hacker L has found the bug in Period I and needs to decide whether to report it immediately. Since e-hacker L does not know whether e-hacker H can find the bug in Period I or not, and only the probability distribution is commonly known, the game of reporting can be formulated as a Bayesian game with incomplete information [32]. If e-hacker H does not find the bug in Period I, the optimal action for e-hacker L is to wait. Otherwise, the utility matrix of the game faced by e-hacker L is illustrated in Table I (with e-hacker H's strategies listed in rows and e-hacker L's strategies listed in columns). The table lists the rewards but omits the costs already incurred, as these sunk costs do not influence their decisions.

TABLE I
Utility Table for the Game of Reporting When Both H and L Find the Bug in Period I

	report	wait
report	$p,\ p$	$p,\ 0$
wait	$0,\ p$	$s,\ s$

(1) If e-hacker H believes that e-hacker L prefers to wait, then e-hacker H will also wait for a higher reward. Similarly, if e-hacker L knows e-hacker H will wait, e-hacker L will also wait.

(2) If e-hacker H believes that e-hacker L will report immediately once it finds the bug, e-hacker H faces two choices: report immediately to secure the reward p, and wait. Given e-hacker L's probability of discovering the bug is q_{1L}, the likelihood of e-hacker H getting the reward s if is $1-q_{1L}$. Therefore, the expected reward of e-hacker H if it chooses to wait is $(1-q_{1L})s$.

Case 1: $p < (1-q_{1L})s$

E-hacker H always waits since the likelihood of e-hacker L finding the bug is sufficiently low that $q_{1L} < 1-\frac{p}{s}$. Therefore, the only pure-strategy Nash equilibrium is both e-hacker H and e-hacker L choose to wait.

Case 2: $p \geq (1-q_{1L})s$

In this case, since $p \geq (1-q_{1L})s \geq (1-q_{1H})s$, both e-hackers choose to report if they believe the other one reports immediately. Hence, there are two pure-strategy Nash equilibria: (report, report) and (wait, wait). Next, we will discuss the mixed-strategy equilibrium [33] regarding the probability of an immediate report for each e-hacker.

2) Analysis of Mixed Strategy Equilibrium: When $p \geq (1-q_{1L})s$, in the mixed-strategy Nash equilibrium solution, each e-hacker assigns a positive probability to every pure strategy. We suppose e-hacker H submits the report immediately w.p. m_H once it finds the bug, and e-hacker L reports w.p. m_L.

If e-hacker L finds the bug, the expected utility of e-hacker L if it decides to submit the report immediately is $f_L(report) = p$, and if it decides to wait, we have $f_L(wait) = q_{1H}(m_H \cdot 0 + (1-m_H)s) + (1-q_{1H})s$. As in mixed-strategy Nash equilibrium, $f_L(report) = f_L(wait)$, and a similar equation holds for e-hacker H, we have

$$m_H = (1-\frac{p}{s})/q_{1H}, \tag{5}$$

$$m_L = (1-\frac{p}{s})/q_{1L}. \tag{6}$$

We can see that e-hacker H, which has a higher chance of discovering the bug, is less likely to submit the report immediately than e-hacker L: $m_H \leq m_L$. This result is

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

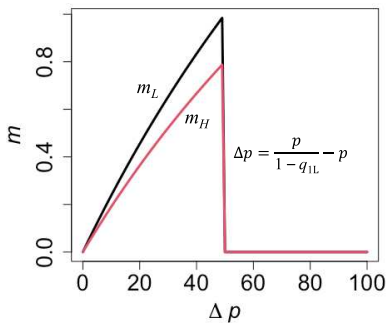


Fig. 2. The equilibrium report probability m in Period I vs. $\Delta p = s - p$.

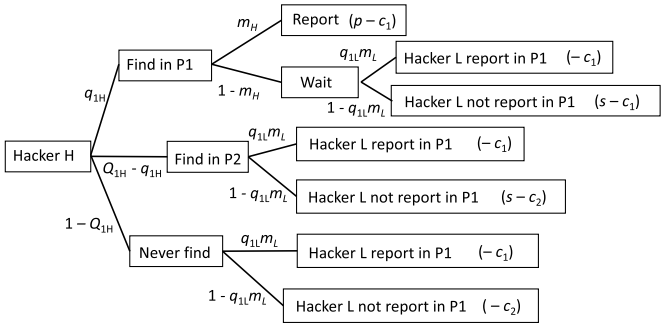


Fig. 3. Decision tree for e-hacker H.

consistent with our intuition that weaker hackers are more inclined to submit immediately because they face a higher risk of zero rewards if they choose to wait. Fig. 2 illustrates an example of how the values of m_H and m_L change with the gap between the rewards $\Delta p = s - p$ when $p = 200$, $q_{1H} = 0.35$, and $q_{1L} = 0.2$.

If $f(report) = f(wait)$, i.e., the expected utilities of reporting and waiting are identical, it is presumed that the e-hacker would prefer to report rather than wait. When $\Delta p \leq \frac{p}{1-q_{1L}} - p$, it is interesting to find that both e-hackers are more likely to report immediately under a higher reward s due to the competition between the e-hackers. The intuition behind this behavior is that as the enhancement in reward Δp or the reward s in Period II increases, the potential gains from waiting increase. To maintain indifference between the strategies (reporting and waiting) for another e-hacker, the probability assigned to waiting is decreased to be $m_H = (1 - \frac{p}{s})/q_{1H}$. Otherwise, if $m_H < (1 - \frac{p}{s})/q_{1H}$, e-hacker L always chooses to wait, and so does e-hacker H; on the other hand, if $m_H > (1 - \frac{p}{s})/q_{1H}$, both e-hackers choose to report immediately. When the increase in potential reward Δp is significant enough such that $\Delta p > \frac{p}{1-q_{1L}} - p$, waiting becomes always superior to reporting immediately, irrespective of the other e-hacker's choice to report immediately or wait. This decision is based on the justification that the potential for a larger future reward outweighs the risks of waiting and possibly receiving nothing. Therefore, in this scenario, both e-hackers choose to wait.

3) Equilibrium Utility Analysis: Fig. 3 represents different utility outcomes and the corresponding possibilities for e-hacker H in a decision-tree model, according to which we can write the following expected utility for e-hacker H:

$$\begin{aligned} f_H = {} & q_{1H}[m_H p + (1 - m_H)(1 - q_{1L}m_L)s - c_1] \\ & + (Q_H - q_{1H})[(1 - q_{1L}m_L)(s - c_2) - q_{1L}m_L c_1] \\ & + (1 - Q_H)[-c_1 q_{1L} m_L - (1 - q_{1L}m_L)c_2]. \end{aligned} \tag{7}$$

Case 1: $p \geq (1 - q_{1L})s$

By incorporating (5) into (7), we have e-hacker H's expected utility as: $f_H = pQ_H + \frac{p}{s}(1 - q_{1H})(c_1 - c_2) - c_1$ and similarly $f_L = pQ_L + \frac{p}{s}(1 - q_{1L})(c_1 - c_2) - c_1$. When (5) and (6) hold, by setting $c_1 = c_2 = 0$, we can derive the expected revenue for the e-hacker as $pQ_i (i \in \{H, L\})$. That is, the expected cost for the vendor is $p(Q_H + Q_L)$. It is interesting to notice that, due to the competition between the e-hackers and their strategic behaviors, the value of s does not affect the e-hackers' total expected reward or the cost for the vendor in the equilibrium solutions. As $Q_L < Q_H$, and $q_{1L} < q_{1H}$, it is easy to deduce that $f_L < f_H$, meaning that a more skilled e-hacker is anticipated to gain more from participating in the bug bounty program. Besides, the gap between their expected utilities $\Delta f = f_H - f_L = p(Q_H - Q_L) + \frac{p}{s}(q_{1H} - q_{1L})(c_2 - c_1)$ increases with p while decreases with s.

TABLE II
UTILITY TABLE FOR THE GAME OF PARTICIPATING

	in	out
in	f_H, f_L	$sQ_H - c_1 q_{1H} - c_2 q_{2H}$, 0
out	0, $sQ_L - c_1 q_{1L} - c_2 q_{2L}$	0, 0

Case 2: $p < (1 - q_{1L})s$

Both e-hackers always choose to wait, and their expected utilities can be written as $f_H = sQ_H - c_1 q_{1H} - c_2 q_{2H}$ and $f_L = sQ_L - c_1 q_{1L} - c_2 q_{2L}$, with their gap, $\Delta f = s * (Q_H - Q_L) - c_1 * (q_{1H} - q_{1L}) - c_2 * (q_{2H} - q_{2L})$, remains unchanged with p and grows as s increases.

B. Equilibrium Analysis of E-hackers' Participation Decisions

In the game of participating, each e-hacker decides whether or not to participate in the bug-bounty program before the program starts. We suppose $sQ_i - c_1 q_{1i} - c_2 q_{2i} > 0$ and $p > \frac{c_1}{Q_i}$ for $i \in \{H, L\}$; otherwise, there is no incentive for e-hackers to participate. From Section IV-A, we know that if $s > \frac{p}{1-q1L}$, then both e-hackers benefit from participating in the program, and both wait until Period II. If $s \leq \frac{p}{1-q1L}$, we obtain the utility for the e-hackers in the game of participating as shown in Table II.

(1) If $f_H > 0$ and $f_L > 0$, (in, in) is the only pure-strategy equilibrium. That being said, both e-hackers will participate.

(2) If $f_H > 0$ and $f_L < 0$, (in, out) is the only pure-strategy equilibrium, meaning e-hacker H will choose to participate, whereas L will opt out.

(3) If $f_H < 0$ and thus $f_L < 0$, there will be two pure-strategy equilibria: (in, out) and (out, in). In this case, under the mixed-strategy equilibrium, e-hacker H will participate w.p. $e_H = \frac{sQ_H - c_1 q_{1H} - c_2 q_{2H}}{sQ_H - c_1 q_{1H} - c_2 q_{2H} - f_L}$, and e-hacker L w.p. $e_L = \frac{sQ_H - c_1 q_{1H} - c_2 q_{2H}}{sQ_H - c_1 q_{1H} - c_2 q_{2H} - f_H}$.

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

TABLE III
EQUILIBRIUM RESULTS OF THE E-HACKERS' DECISIONS

Region	Participation Decisions	Report Decisions
I	H in, L in	both wait
II	participate w.p. e_H, e_L	report w.p. m_H, m_L
III	H in, L in	both wait
IV	H in, L out	H wait
V	H in, L in	both wait
VI	H in, L in	report w.p. m_H, m_L

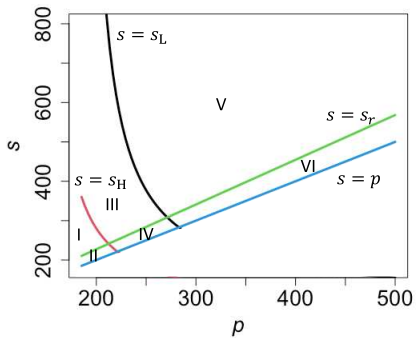


Fig. 4. Regions of e-hackers' optimal decisions with different reward p and s ($q_{1H} = 0.16, q_{2H} = 0.18, q_{1L} = 0.12, q_{2L} = 0.15, c_1 = 50, c_2 = 80$, $s_H = \frac{p(1-q_{1H})(c_2-c_1)}{pQ_H - c_1}, s_L = \frac{p(1-q_{1L})(c_2-c_1)}{pQ_L - c_1}, s_r = \frac{p}{1-q_{1L}}$).

By integrating the equilibrium results of the e-hackers' participation and report decisions, Fig. 4 plots the regions of the e-hackers' optimal decisions with different reward values p and s, and Table III summarizes the equilibrium results in these regions. Note that in the scenario where the e-hackers are homogeneous, i.e., $Q_H = Q_L$ and $q_{1H} = q_{1L}$, then in Fig. 4, the two curves of $s = s_H$ and $s = s_L$ overlap with each other. This results in only Regions I, II, V, and VI remaining without affecting the equilibrium outcomes.

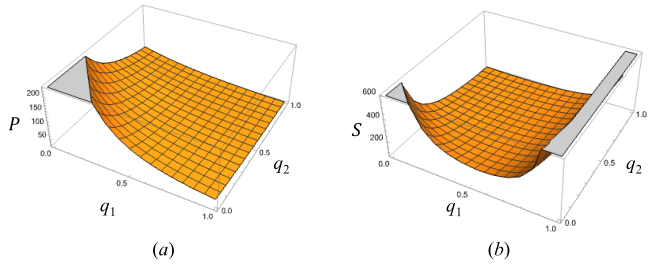
C. Vendor's Reward Decisions

(1) Incentive for the early report. One objective for the vendor is to encourage the participating e-hackers to submit their reports immediately instead of waiting for a higher reward. Two types of incentive compatibility (IC) can be implemented to induce the e-hackers to behave as the vendor wishes [22]: *weak IC* and *strong IC*.

- In *weak IC*, "both report immediately" is a possible outcome. According to Section IV-A, p and s must satisfy
$$s \le s_r = \frac{p}{1-q_{1L}}. \tag{8}$$
- In *strong IC*, "both report immediately" is the unique outcome, and the vendor needs to set the reward s so that $m_H = m_L = 1$, which means $s = s_r$.

Hence, to get early report in Period I, there are two options for the vendor:

- $s < min\{s_H, s_r\}$ (Region II in Fig. 4), which indicates that $p < \frac{(1-q_{1H})(c_2-c_1)+c_1}{Q_H}$. This condition reflects a scenario where the initial reward p is minimal, and there is only a slight increment in the reward for Period II, with the potential risk that the program may not attract any participants.



(a) (b)

Fig. 5. The optimal rewards vs. (q_1, q_2).

- $s_L < s \le s_r$ (Region VI in Fig. 4), This implies a substantial initial reward p and a moderate rise in reward during Period II.

(2) Incentive for participation. The other objective for the vendor is to attract both e-hackers to participate. To ensure that both e-hackers have the incentive to participate, it is required that $f_H > 0$ and $f_L > 0$ according to the equilibrium of e-hackers' participation decisions. That being said, if $s < s_r$, the reward s should be high enough that $s > s_L = \frac{p(1-q_{1L})(c_2-c_1)}{pQ_L - c_1}$. If $s = s_r$, We can obtain the following results regarding the vendor's optimal decision on rewards given to the e-hackers:

Proposition 1: To attract both e-hackers to participate in the bug-bounty program and maximize the chances of immediate reporting, the reward decisions for the vendor should satisfy $s^* = s_r$ and:

$$p^* > P = \frac{q_{1L}c_1 + (1-q_{1L})(1-q_{1H})c_2 + q_{1H}c_1(1-q_{1L})}{q_{1L} + q_{2L}\frac{1-q_{1H}}{1-q_{1L}}}, \tag{9}$$

$$s^* > S = \frac{q_{1L}c_1 + (1-q_{1L})(1-q_{1H})c_2 + q_{1H}c_1(1-q_{1L})}{q_{1L}(1-q_{1L}) + q_{2L}(1-q_{1H})}. \tag{10}$$

In the situation where there are two e-hackers of the same type, i.e., $q_1 = q_{1H} = q_{1L}$ and $q_2 = q_{2H} = q_{2L}$, we can obtain Corollary 1 in terms of how the rewards change with the detection probabilities:

Corollary 1: In the case of two homogeneous e-hackers, with other parameters unchanged, the lower bound of the vendor's first-period reward, P, decreases with q_1. However, the lower bound of second-period reward S has a non-monotonic relationship with q_1. Besides, both P and S decrease with q_2.

Proof: Since $P = \frac{q_1c_1 + c_2(1-q_1)^2 + q_1(1-q_1)c_1}{Q}$, $\frac{dP}{dq_1} < 0$, and $\frac{dP}{dq_2} < 0$, P is a decreasing function of q_1 and q_2, we can obtain the results in Corollary 1. □

Fig. 5 shows an example of how P and S change with q_1 and q_2.

Corollary 1 suggests that for bugs easily found in Period I (indicated by a large q_1), the vendor can offer a low initial reward p and a high subsequent reward s to induce immediate report as m increases with s. For bugs that are difficult to identify in Period I but are more likely to be discovered in Period II (small q_1 and large q_2), the vendor should increase the initial reward p moderately and then adjust it to a slightly higher s in Period II. When the bug is rather challenging to

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
VENDOR' REWARD DECISIONS WITH TWO HOMOGENEOUS E-HACKERS

Detection Probability	small q_1	large q_1
small q_2	high p, high s	low p, high s
large q_2	medium p, medium s	low p, high s

TABLE V
UTILITY TABLE FOR THE GAME OF PARTICIPATING WHEN E-HACKERS' STRATEGIC BEHAVIORS ARE IGNORED

	in	out
in	$f_{H,ign}$, $f_{L,ign}$	$(s-c_2)q_{2H}+(p-c_1)q_{1H}$, 0
out	0, $(s-c_2)q_{2L}+(p-c_1)q_{1L}$	0, 0

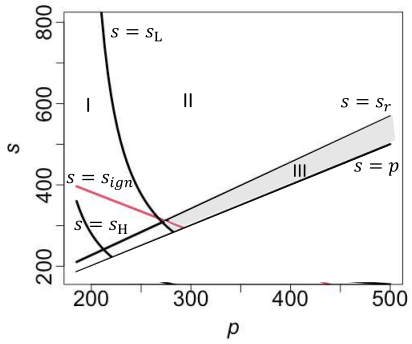


Fig. 6. Regions of vendor's decisions when e-hackers' strategic behaviors are ignored.

detect (with both q_1 and q_2 being small), the vendor should offer substantial rewards for both p and s to encourage e-hackers to join and stay in the program. Table IV outlines the vendor's reward decisions for different scenarios.

D. Impact of Ignoring the E-hackers' Strategic Behaviors

To emphasize the importance of considering the e-hackers' strategic behaviors in reward design, we examine the scenario where the vendor ignores the e-hackers' strategic behaviors, i.e., the vendor believes that the e-hackers will report immediately whenever they find the bugs. In the following analysis, we will use the subscript ign to represent the case. Table V summarizes the vendor's belief of e-hackers' utilities in the participating game, where $f_{H,ign} = pq_{1H} + sq_{2H}(1-q_{1L}) - c_1(q_{1H}+q_{1L}-q_{1H}q_{1L}) - c_2(1-q_{1H})(1-q_{1L})$ and $f_{L,ign} = pq_{1L} + sq_{2L}(1-q_{1H}) - c_1(q_{1L}+q_{1H}-q_{1L}q_{1H}) - c_2(1-q_{1L})(1-q_{1H})$.

The objective for the vendor is to ensure that both e-hackers are incentivized to participate, i.e., $f_{H,ign} > 0$ and $f_{L,ign} > 0$. Therefore, under the assumption that $q_{1H} > q_{1L}$ and $q_{2H} > q_{2L}$ the reward s should be high enough that

$$s > s_{ign} \tag{11}$$

where $s_{ign} = \frac{c_1(q_{1L}+q_{1H}-q_{1L}q_{1H})+c_2(1-q_{1L})(1-q_{1H})-pq_{1L}}{q_{2L}(1-q_{1H})}$.

Using the same input parameters values as in Fig. 4, we obtain the numerical results for the lower bound of s_{ign} when p increases from 185 to 500, as shown in Fig. 6. To attract both e-hackers, Regions I, II, and III above the line $s = s_{ign}$ represent the possible reward values (p, s) when the vendor ignores the e-hackers' strategic behaviors or views the e-hackers as myopic. The shaded region indicates the area where the vendor's decisions fall for weak IC when taking into account the strategic behaviors of e-hackers. Failure to take into account the e-hackers' behaviors has negative impacts as illustrated in two regions in Fig. 6: In both Regions (I) and Region (II), even though both e-hackers will participate in the program, they both choose to wait until Period II to submit a report. That being said, ignoring the e-hackers' strategic behaviors could result in setting the initial incentive p too low or the subsequent incentive s too high. This might motivate them to delay reporting the bug until Period II in anticipation of greater rewards. Therefore, to take the best advantage of a bug-bounty program, it is essential to incorporate e-hackers' strategic behaviors into the incentive mechanism.

TABLE VI
UTILITY TABLE FOR THE GAME OF PARTICIPATING WITH STATIC REWARD STRATEGY

	in	out
in	f_H, f_L	$(p-c_2)q_2+(p-c_1)q_1$, 0
out	0, $(p-c_2)q_2+(p-c_1)q_1$	0, 0

V. EXTENSION: DIFFERENT REWARD STRATEGIES

In our previous model, we analyzed the vendor's optimal reward decisions under $p < s$, i.e., an increasing reward strategy is used to compensate the e-hackers' continuous work for two periods. In this section, we discuss and compare the results under two different reward strategies: static reward strategy and decreasing reward strategies.

A. Static Reward Strategy

We first consider a static reward strategy scenario, where the vendor does not enhance the reward during the whole period, i.e., $s = p$. The e-hackers will have no incentive to wait until Period II if they discover the vulnerability in Period I. Table VI lists the e-hackers' utilities in the participating game,

$$\begin{aligned} f_H &= q_{1H}[p+(c_2-c_1)(1-q_{1L})] + q_{2H}p(1-q_{1L}) \\ &\quad - c_1q_{1L} - (1-q_{1L})c_2, \end{aligned} \tag{12}$$

$$\begin{aligned} f_L &= q_{1L}[p+(c_2-c_1)(1-q_{1H})] + q_{2L}p(1-q_{1H}) \\ &\quad - c_1q_{1H} - (1-q_{1H})c_2. \end{aligned} \tag{13}$$

To attract both e-hackers into the program, we have $f_H > 0$ and $f_L > 0$, or

$$p > P_s = \frac{q_{1L}c_1 + (1-q_{1L})(1-q_{1H})c_2 + q_{1H}c_1(1-q_{1L})}{q_{1L}+q_{2L}(1-q_{1H})}. \tag{14}$$

If we compare the value of P_s with the lower bounds P and S in (9) for the inter-temporal pricing strategy, it is interesting to find that $P < P_s < S$. That means if the vendor does not plan to enhance the reward afterward, the reward should be higher initially to attract the e-hackers. In this case, the vendor's expected utility under a static reward strategy is lower than that under an inter-temporal pricing strategy if the reward is set as $s = \frac{p}{1-q_{1L}}$.

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

B. Decreasing Reward Strategy

Noticing that under an increasing reward strategy with $p \leq s$, the participating e-hackers might hoard bugs for higher rewards, one might wonder whether a decreasing reward strategy works to encourage early reports. In this section, we explore the scenario of decreasing reward strategy: the vendor declares at the beginning of the program that a reward of s will be provided for each valid bug report. Additionally, if the report is submitted during an early phase (Period I), a bonus is added, raising the reward to p, where $p > s$.

In this scenario, if a bug is discovered during Period I, it would be reported right away, eliminating the game of reporting among the e-hackers in Period I. However, if the bug remains undiscovered by the end of Period I, the e-hackers will decide whether to proceed into Period II, taking into account the decrease in the reward. For example, given that e-hacker H didn't find the bug in Period I, the probability of it finding the bug in Period II is $\frac{q_{2H}}{1-q_{1H}}$. It will only choose to continue working into Period II if the condition $s\frac{q_{2H}}{1-q_{1H}} - (c_2 - c_1) > 0$ is satisfied. Therefore, to motivate both e-hackers to keep working through Period II, the vendor must set the value of s such that $s > s_m = max\{\Delta c\frac{1-q_{1H}}{q_{2H}}, \Delta c\frac{1-q_{1L}}{q_{2L}}\}$.

Next we consider the e-hackers' participation decisions. If the vendor would like to attract both e-hackers to participate in the program, we need $f_H = q_{1H}(p - c_1) + q_{2H}[(1 - q_{1L})(s-c_2)-q_{1L}c_1]-(1-Q_H)[q_{1L}c_1+(1-q_{1L})c_2] > 0$ and $f_L = q_{1L}(p - c_1) + q_{2L}[(1 - q_{1H})(s - c_2) - q_{1H}c_1] - (1 - Q_L)[q_{1H}c_1 + (1 - q_{1H})c_2] > 0$, or

$$
\begin{aligned}
& s > s_f \\
& = max\{\frac{(1-Q_H)(q_{1L}c_1+(1-q_{1L})c_2)-q_{1H}(p-c_1)}{q_{2H}(1-q_{1L})} \\
& \quad + \frac{q_{1L}c_1}{1-q_{1L}}, \\
& \quad \frac{(1-Q_L)(q_{1H}c_1+(1-q_{1H})c_2)-q_{1L}(p-c_1)}{q_{2L}(1-q_{1H})} \\
& \quad + \frac{q_{1H}c_1}{1-q_{1H}}\} + c_2. \qquad (15)
\end{aligned}
$$

Fig. 7 illustrates the regions of e-hackers' decisions under different values of p and s with the same parameters value as in Section IV-B: In Region I, s is not large enough, which leads to the situation that e-hackers lose motivation to continue working in Period II. In Region II, not all e-hackers are willing to participate in the program. The shaded Region III indicates the area where the vendor's decisions should fall if they want to attract both e-hackers to the program and ensure they are committed to working through both periods. By comparing this area with the vendor's decision under $p < s$ in Proposition 1, it can be found that the lowest value of p under decreasing reward strategy is higher than that under increasing reward strategy, indicating that if the vendor would like to reduce the reward s for Period II, it needs to enhance the reward p for Period I.

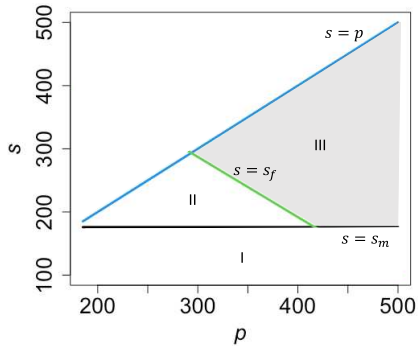


Fig. 7. Regions of vendor's decisions under decreasing reward strategy.

VI. Collaboration Model

In this section, we discuss the collaboration between two e-hackers when they could share resources and expertise as they work together to uncover vulnerabilities. In reality, companies encourage e-hackers to collaborate. For example, on the HackerOne platform, the e-hackers can set their collaboration preferences to invite collaborators and view all other hackers open to collaborating [34].

As "two heads are better than one", when the two e-hackers collaborate, the bugs will be more likely and earlier to be detected. If the e-hackers' skills complement each other very well, then the vulnerability discovery probability will be largely enhanced; otherwise, if they embrace similar skills or expertise, the probability might be slightly enhanced by collaboration. Therefore, we assume that the probability of finding the bug in Period I under collaboration is q_{1c} which is not smaller than the probability q_i when the e-hackers work independently: $q_{1c} \geq q_i$. Similarly, the likelihood of finding the bug in two periods will be enhanced, i.e., $Q_i \leq Q_c = q_{1c} + q_{2c}$. Besides, the collaborating e-hackers will need to split the rewards they receive. As our work focuses on the impacts of collaboration on the bug bounty programs, we will not discuss the bounty splitting rules in this paper, and the reward will be equally shared between the two e-hackers in our model.

In the case of collaboration, given the rewards p for Period I and s for Period II, if $p \leq s$, the e-hackers will always submit their report in Period II as competition does not exist, with an expected utility of $f_i = \frac{1}{2}Q_c s - q_{1c}c_1 - q_{2c}c_2$. Therefore, to encourage early reports, we consider an incentive mechanism in which the vendor would give out a larger reward, denoted by p_c ($p_c > s > p$), for the two collaborating e-hackers if they submit their report in Period I. The new timeline of the process is as follows:

- The vendor announces the rewards: p for individual reports in Period I, p_c for the collaborative report in Period I, and s for any report in Period II.
- The e-hackers then decide whether or not to collaborate in the bug bounty program. They can set up the collaboration channel through the vendor's platform. If they do not collaborate, each e-hacker will decide whether or not to participate by itself.
- The e-hackers submit their report once they find the bugs and get the rewards. If they collaborate, the reward will be split.

With the consideration of cooperative behavior, the e-hackers would evaluate the benefits of cooperation, and the

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

vendor needs to identify under what conditions to incentivize the e-hackers to collaborate instead of taking advantage of their competition as well as how to set p_c to facilitate collaboration. We assume that the expected cost remains the same for each e-hacker, and the expected utility for each hacker when they collaborate can be written as:

$$f_c = q_{1c}(\frac{1}{2}p_c - c_1) + q_{2c}(\frac{1}{2}s - c_2). \tag{16}$$

In the case of two heterogeneous e-hackers as in Section IV, our result shows that the vendor would set $s = \frac{p}{1-q_{1L}}$ to encourage participants to report bugs immediately when the e-hackers do not collaborate, and e-hacker H has an expected utility of $f_H = pQ_H + \frac{p}{s}(1-q_{1H})(c_1-c_2) - c_1$. Since $f_L \leq f_H$, e-hacker L has a stronger collaboration incentive. Suppose the e-hackers choose to collaborate when $f_c = f$, then the e-hackers would decide to collaborate if $f_c \geq f_H$. To enable the collaboration, for e-hacker H, p_c should satisfy:

$$p_c \geq p_{cH} = 2\frac{pQ_H - \Delta c(1-q_{1L})(1-q_{1H})}{q_{1c}} + T, \tag{17}$$

where $\Delta c = c_2 - c_1$, $T = 2c_1 - 2\frac{c_1 + q_{2c}(\frac{1}{2}s - c_2)}{q_{1c}}$, and $s = \frac{p}{1-q_1}$. Therefore, we have

$$p_c = max\{s, p_{cH}\} \tag{18}$$

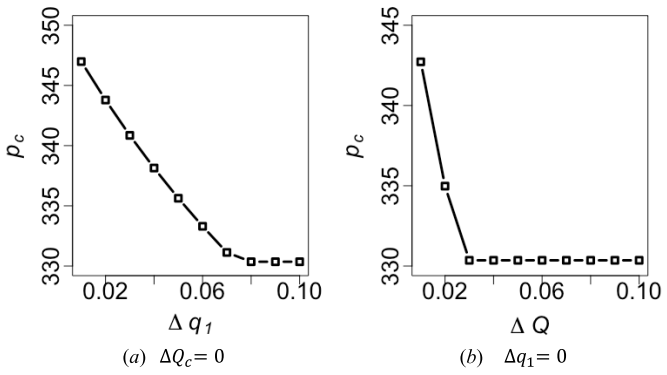
The first situation we are interested in is when collaboration enhances the early detection probability without affecting the overall detection probability for e-hacker H: $q_{1c} \geq q_{1H}$, $Q_c = Q_H$. It can be derived that if the increase in speed is substantial enough that

$$\Delta q_1 = q_{1c} - q_{1H} > \frac{Q_H(2p - s + 2c_2) - 2\Delta c[(1-q_{1L})(1-q_{1H}) + q_{1H}] - 2c_1}{2\Delta c}, \tag{19}$$

then there is no need for the vendor to offer a reward p_c greater than s to encourage collaboration. For illustration purposes, we assume that $q_{1H} = 0.22, q_{2H} = 0.2, q_{1L} = 0.18, q_{2L} = 0.08, c_1 = 50, c_2 = 80$, $p = 270.9$. The values of p_c under different values of Δq_1 are depicted in Fig. 8 (a) with $Q_c = 0.42$. It is interesting to notice that if collaboration could speed up the bug discovery largely ($\Delta q_1 > 0.073$ in this example), a static reward strategy with $p_c = s = 330.3$ would be enough to encourage collaboration between e-hackers; otherwise, the vendor needs to give out a larger reward to Period I report than to Period II report to facilitate e-hacker collaboration and early report: $p_c > s$.

A second situation we investigate is where collaboration may not expedite the finding of bugs, but it does improve the chances of success by the end of the two-period timeframe: $Q_c \geq Q_H$. It is derived that if

$$\Delta Q = Q_c - Q_H > \frac{Q_H(2p - s + 2c_2) - 2\Delta c[(1-q_{1L})(1-q_{1H}) + q_{1H}] - 2c_1}{s - 2c_2}, \tag{20}$$



(a) $\Delta Q_c = 0$ (b) $\Delta q_1 = 0$

Fig. 8. Reward p_c when collaboration (a) speeds up the detection process and (b) enhances the overall detection probability.

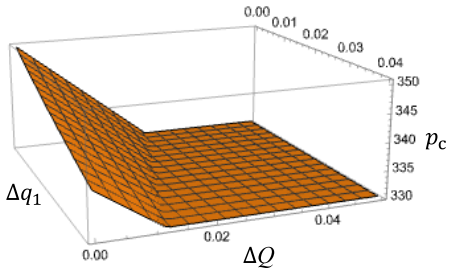


Fig. 9. Reward p_c versus Δq_1 and ΔQ.

then a reward of $p_c = s$ is adequate to incentivize collaboration. Fig. 8 (b) shows the results when $q_{1c} = 0.22$ and Q_c is increased, indicating that $p_c > s$ if $\Delta Q > 0.026$.

Next, we consider a more general situation when collaboration not only speeds up the bug-hunting process but also increases the overall success probability, offering dual benefits in terms of efficiency and effectiveness. Fig. 9 shows the value of p_c under different values of Δq_1 and ΔQ. The results indicate that a smaller reward is needed as collaboration raises the probability of detection, whether in Period I or Period II. The reason is that if the collaboration has minor effects on the bug discovery process, the e-hackers have little incentive to collaborate since they need to split the reward unless a larger reward is given. If the e-hackers have a higher chance of finding the bug or could find it much earlier than when they work independently, collaborating with others not only saves them time or cost but also mitigates the competition risk.

For the vendor, if the two e-hackers collaborate, its expected utility is $\pi_c = q_{1c}(r + R - p_c) + q_{2c}(r - s)$; otherwise, $\pi = r(1-(1-Q_H)(1-Q_L)) + R(q_{1H} + q_{1L} - q_{1H}q_{1L}) - p(q_{1H} + q_{1L}) - s[q_{2H}(1-q_{1L}) + q_{2L}(1-q_{1H})]$. Therefore, if $\Delta\pi = \pi_c - \pi > 0$, or the value of Δq and ΔQ satisfy:

$$\Delta q(R + s - p_c) + \Delta Q(r - s) > g(p_c), \tag{21}$$

then the vendor would benefits from their collaboration, where $g(p_c) = p_c q_{1H} + rQ_L(1-Q_H) + Rq_{1L}(1-q_{1H}) - p(q_{1H} + q_{1L}) - s[q_{2H}(1-q_{1L}) + q_{2L}(1-q_{1H}) - Q_H + q_{1H}]$. We use a numerical example to illustrate the benefit of having e-hackers collaborate: $\Delta\pi = \pi_c - \pi$. The shaded area in Fig. 10 shows the conditions under which the vendor could gain from the collaboration between e-hackers: (1) In Region I, a static

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

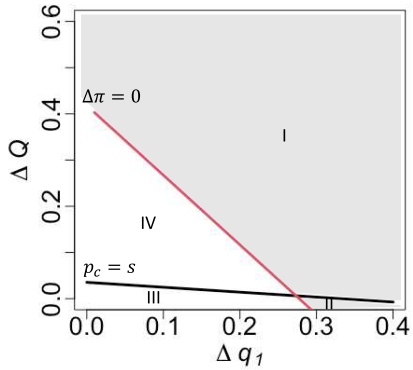


Fig. 10. Benefits of collaboration for vendor ($c_1 = 52, c_2 = 60, r = 350$, $R = 120$).

reward policy $p_c = s$ is sufficient to encourage collaboration; (2) In Region II, even though the chance of detection barely improves with collaboration and the vendor has to offer a reward $p_c > s$, the process of finding vulnerabilities is significantly faster, providing a benefit to the vendor; (3) In Region III where the collaboration results in only a slight improvement in either the speed of discovery or the overall detection probability, the vendor has to offer a reward p_c greater than s to promote cooperation among e-hackers. Therefore, it is advantageous for the vendor to utilize the competition between e-hackers to reduce costs; and (4) In Region IV, even though the vendor doesn't need to provide a reward p_c larger than s to facilitate collaboration between e-hackers, the vendor still benefits less from their collaboration than competition as it doesn't significantly enhance overall detection probability or speed. This result shows that the vendor benefits more from e-hacker collaboration if such joint efforts significantly enhance the speed of vulnerability detection or the likelihood of discovering potential bugs. It implies that vendors might focus on encouraging cooperative efforts to address hard-to-detect vulnerabilities, where the collective contribution is expected to have a significant impact.

VII. Models of Multiple E-Hackers

In this section, we extend the model of two e-hackers to the case of multiple heterogeneous e-hackers. The research questions we are interested in include: 1) With the same amount of rewards (p, s), how does the number of participating e-hackers impact their to-report-or-to-wait decisions? 2) Does an increase in the number of competitors decrease expected utilities for the e-hackers at the equilibrium? 3) For the vendor, does the possibility of getting an early report (in Period I) increase with the number of participating e-hackers? and 4) How would the reward s affect the participating e-hackers' to-report-or-to-wait decisions? We first construct the expected utility function of the e-hackers and analyze their equilibrium decisions. Then, we discuss the impact of the number of participants and rewards through a set of numerical examples.

A. Equilibrium Analysis of E-hacker's Decisions

Given p and s, we assume that there are a total of N participating e-hackers. Without loss of generality, we categorize these e-hackers into two types: high type (H) and low type (L), each having a number of n_i e-hackers, $i \in \{L, H\}$. These types differ in bug detection probabilities, with q_{1i} being the probability in Period I and q_{2i} in Period II. The total number of e-hackers across both types is $n_L + n_H = N$.

If an e-hacker of type i found the bug in Period I, it would decide whether to report it immediately and get a reward of p or wait for Period II with an expected utility of $sw_i - c_1$, where w_i is the probability that none of the other $n-1$ e-hackers submitting a report in Period I. We denote m_i as the probability of an e-hacker of type i submitting the report immediately and $1 - m_i$ as the probability of waiting until Period II. For instance, if the e-hacker is of type L, then $w_L = (1 - q_{1L}m_L)^{n_L - 1}(1 - q_{1H}m_H)^{n_H}$. The total expected utility of an e-hacker of type i ($i \in \{L, H\}$) can be written as

$$\begin{aligned} f_i = {} & q_{1i}[m_i p + (1 - m_i)w_i s - c_1] \\ & + q_{2i}[w_i(s - c_2) - (1 - w_i)c_1] \\ & - (1 - q_{1i} - q_{2i})[w_i c_2 + (1 - w_i)c_1]. \end{aligned} \tag{22}$$

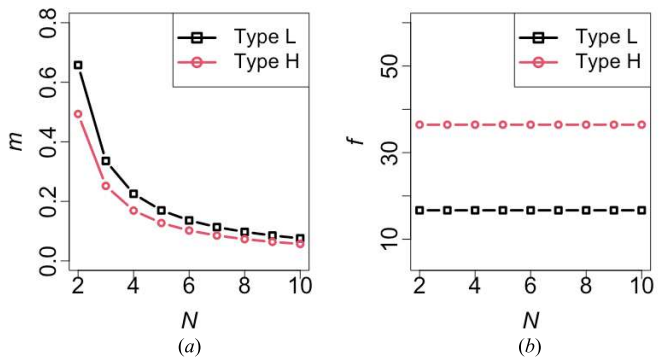
We are interested in the mixed-strategy Nash equilibrium, where the e-hacker would be indifferent between reporting immediately and waiting. That being said, at equilibrium, $p - c_1 = sw_i - c_1$ for both $i \in \{L, H\}$. Therefore, we can derive that

$$m_i = \frac{1}{q_{1i}}[1 - (\frac{p}{s})^{\frac{1}{N-1}}]. \tag{23}$$

Which indicates that a more capable e-hacker is more likely to delay their reporting.[3] However, when s is very large such that $s > s_d = \frac{p}{(1-q_{1L})^{n_L}(1-q_{1H})^{n_H - 1}}$, waiting is the dominant strategy for everyone. The threshold s_d decreases with n_L and n_H, indicating that the likelihood of all participating e-hackers delaying reports until Period II is higher when there are fewer competitors. When s is small enough to satisfy $m_i \leq 1$ in equation (23), as $\frac{dm_i}{dN} = \frac{1}{q_{1i}}(\frac{p}{s})^{\frac{1}{N-1}} ln(\frac{p}{s}) \frac{1}{(N-1)^2} < 0$, we observe a decrease in the immediate report probability m as the number of participants N increases, which may contradict our initial intuition. An example is shown in Fig. 11(a). From a game theoretical perspective, this behavior can be explained as follows: As the number of e-hackers rises, the benefits of waiting, represented by the value of sw_i, decreases. This requires a decrease in the value of m, so that other e-hackers remain indifferent between waiting and reporting immediately. The intuition behind this phenomenon is that the existence of more competitors enhances the chance that other e-hackers will find the bug, thus reducing its expected utility. Consequently, e-hackers are motivated to delay their report submission for a larger reward. That being said, at equilibrium, an increase in the number of participating e-hackers raises the likelihood of e-hackers choosing to wait for a higher reward to offset potential losses resulting from competition. This results in the probability that the vendor gets an early report in Period I, $Prob_1 = 1 - \prod_{i \in L,H}(1 - q_{1i}m_i)^{n_i}$, decreasing with n_i (as shown in Fig. 12(a)). But the chance of finding the bug in the

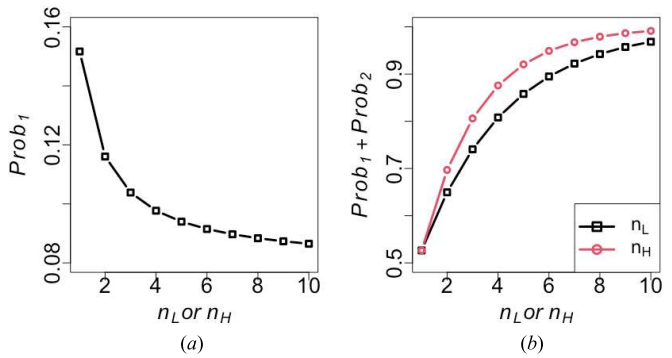
[3]Note that s should be small enough that $s < \frac{p}{(1-q_{1L})^{N-1}}$. Otherwise, waiting would be a possible dominant strategy for type H e-hackers.

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 11. Impacts of number of participants N ($q_{1H} = 0.16, q_{1L} = 0.12$, $p = 350, s = 380$).



(a) (b)

Fig. 12. The impacts of n_i on $Prob_1$ and $Prob_1 + Prob_2$ ($q_{2H} = 0.2, q_{2L} = 0.14$).

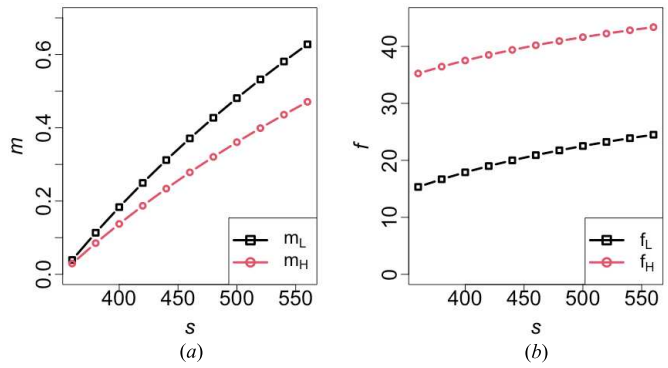
whole time horizon, $Prob_1 + Prob_2 = 1 - \prod_{i \in L,H}(1 - q_{1i} - q_{2i})^{n_i}$, increases with n_i (as shown in Fig. 12(b)). Another interesting observation is that the expected utility f_i of the e-hackers remains unchanged regardless of the number of participants N (as illustrated in Fig. 11(b)).

B. Vendor's Reward Decision

Firstly, we investigate how the value of reward s influences the participating e-hackers' to-report-or-to-wait decisions and their expected utility. Then, we will discuss how to decide the value of s for the vendor.

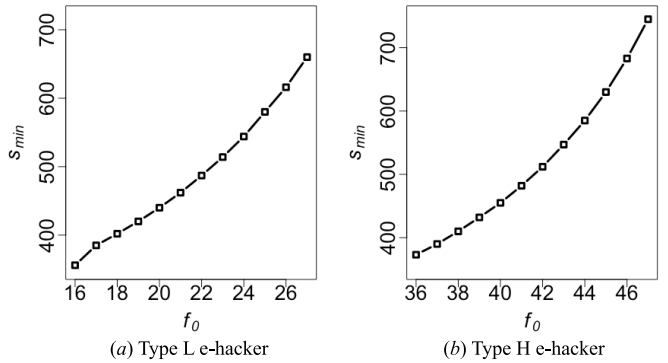
We change the value of s from 360 to 560 and show the equilibrium results in Fig. 13 for the case of $n_L = 5$ and $n_H = 2$. As we can see, as long as $s < \frac{p}{(1-q_{1L})^{N-1}}$, by enhancing the reward s, both types of e-hackers are more likely to submit reports immediately. As $Prob_1 = 1 - \prod_{i \in L,H}(1 - q_{1i} m_i)^{n_i}$ increases with m_i, we can derive that $Prob_1$ also increases with s, indicating that the vendor is expecting a higher chance of getting an early report. Besides, the participating e-hackers could also benefit more from the program, i.e., f_i increases with s.

Based on the equilibrium analysis, the number of participants has little impact on the equilibrium expected utility of e-hackers, while the value of s does. Our current focus is on the following question: If potential e-hackers are willing to participate as long as their expected utility f exceeds a threshold utility f_0, what is the minimum value of s, denoted as s_{min}, to ensure $f \geq f_0$ for both types of e-hackers and attract them to join the program? Considering different expectations of different types of e-hackers, we calculate the value of s_{min} under various values of f_0 for types H and L by solving $f_i = f_0$ with f_i in (22) and m_i in (23), respectively. Fig. 14 shows an example of s_{min} under different values of f_0. The result is intuitive that as the willingness-to-participate f_0 increases, the vendor needs to provide a larger reward s.



(a) (b)

Fig. 13. The impacts of s ($p = 350, n_L = 5, n_H = 2$).



(a) Type L e-hacker (b) Type H e-hacker

Fig. 14. s_l vs. f_0.

Now we have two limits of s: one lower limit denoted as s_{min} to ensure that $f \geq f_0$, and one upper limit denoted as $s_d = \frac{p}{(1-q_{1L})^{n_L}(1-q_{1H})^{n_H-1}}$ above which all e-hackers would delay their reports. The vendor aims to attract as many e-hackers as possible since with more participating e-hackers, there's a lower chance they all choose to delay reporting and a higher chance that the bug can be discovered. Given this objective, we can examine the vendor's strategy regarding s in two scenarios, providing interesting insights into working with e-hackers in bug bounty programs: (1) If e-hackers are content with modest expected rewards from the program with a modest value of f_0, i.e., $s_{min} < s_d$, the vendor can select a value of s that satisfies $s \in [s_{min}, s_d)$ to attract all the potential e-hackers, with early reports expected in Period I. (2) If the potential e-hackers are "greedy" and are driven by excessive profit expectations, by setting $s \geq s_{min} > s_d$ to attract all the potential e-hackers, the vendor should not anticipate early reports during Period I.

VIII. Conclusion

This article focuses on the role of e-hackers' strategic behaviors in software vendors' reward strategies for bug-bounty

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

programs. We formulate a novel game model between a software vendor and multiple e-hackers. The interaction between the vendor and the e-hackers is modeled as a Stackelberg game, and the competition between the e-hackers is modeled as Bayesian games with incomplete information. The equilibrium solutions are characterized by the e-hackers' participation decisions, bug report decisions, and the vendor's reward strategies.

The paper contributes to the existing literature by exploring the impacts of e-hackers' strategic behaviors, including hoarding bugs or delaying reports for higher rewards, as well as the competitive and cooperative interactions among peers. Furthermore, the study evaluates the benefits of employing an inter-temporal reward strategy in a bug-bounty program by comparing three different reward strategies: static, decreasing, and increasing reward strategies. The results of the study offer several insights into the vulnerability market. First, enhancing the final reward does not always induce the e-hackers to wait due to the uncertainty in bug hunting and the competition between the e-hackers. If the vendor does not plan to increase the reward afterward, a higher initial reward should be set to attract the e-hackers. Second, ignoring the strategic behaviors of e-hackers may lead to the initial reward being set too low and the subsequent reward too high, which could cause an increase in the delay of bug reporting. Finally, by carefully selecting the reward amounts for an inter-temporal reward strategy, the vendor can motivate potential e-hackers to engage in the bug bounty program, encourage timely reporting, and simultaneously reduces costs. Moreover, the vendor benefits more from e-hacker collaboration when their collective efforts significantly enhance the speed of vulnerability detection or the chances of their discovery.

Future research directions include considering the competition between multiple software vendors in recruiting e-hackers and the influence of e-hackers' irrational behaviors, such as sunk cost fallacy and risk-seeking behaviors. Considering the multi-dimensional motivations of e-hackers, which include financial rewards, self-improvement, and the commitment to defense [35], it would also be interesting to explore different reward schemes or incentive mechanisms. These include providing additional bonuses such as "badges" based on the quantity of previously successful submissions and using leaderboards to showcase e-hacker rankings [36], [37]. These strategies would help e-hackers find more effective collaboration partners and foster a competitive and engaging community. Besides, our study assumes that the e-hackers know the number of participants in the program. Such information may only be available if the vendor publicizes it. Therefore, the impact of asymmetric information on the collaboration between the vendor and the e-hackers, as well as the benefit of hiding the information for the vendor, needs to be studied further.

ACKNOWLEDGMENT

The authors would like to thank the editor and the anonymous referees for their very constructive comments to improve the quality of this article. Any opinions, findings, conclusions, or recommendations expressed in this article are those of the author(s) and do not necessarily reflect the views of NSF.

REFERENCES

[1] (2022). *Bug Bounty Program*. Accessed: Oct. 24, 2022. [Online]. Available: https://en.wikipedia.org/wiki/Bug_bounty_program

[2] R. Böhme, "A comparison of market approaches to software vulnerability disclosure," in *Proc. Int. Conf. Emerg. Trends Inf. Commun. Secur.* Cham, Switzerland: Springer, 2006, pp. 298–311.

[3] N. Perlroth, *This is How They Tell Me World Ends: The Cyberweapons Arms Race*. London, U.K.: Bloomsbury Publishing USA, 2021.

[4] S. Schechter, "How to buy better testing using competition to get the most security and robustness for your dollar," in *Proc. Int. Conf. Infrastruct. Secur.* Cham, Switzerland: Springer, 2002, pp. 73–87.

[5] R. Böhme, "Vulnerability markets," in *Proc. 22C3*, vol. 27, 2005, p. 30.

[6] MSRC. *Microsoft Bounty Program Updates: Faster Bounty Review, Faster Payments, and Higher Rewards*. Accessed: May 2, 2024. [Online]. Available: https://msrc.microsoft.com/blog/2019/04/microsoft-bounty-program-updates-faster-bounty-review-faster-payments-and-higher-rewards/

[7] S. S. Malladi and H. C. Subramanian, "Bug bounty programs for cybersecurity: Practices, issues, and recommendations," *IEEE Softw.*, vol. 37, no. 1, pp. 31–39, Jan. 2020.

[8] H. Hata, M. Guo, and M. A. Babar, "Understanding the heterogeneity of contributors in bug bounty programs," in *Proc. ACM/IEEE Int. Symp. Empirical Softw. Eng. Meas. (ESEM)*, Nov. 2017, pp. 223–228.

[9] A. Laszka, M. Zhao, A. Malbari, and J. Grossklags, "The rules of engagement for bug bounty programs," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Cham, Switzerland: Springer, 2018, pp. 138–159.

[10] L. Breidenbach, P. Daian, F. Tramèr, and A. Juels, "Enter the hydra: Towards principled bug bounties and $\$Exploit - Resistant\$$ smart contracts," in *Proc. 27th USENIX Secur. Symp. (USENIX Secur.)*, 2018, pp. 1335–1352.

[11] A. Ahmed, A. Deokar, and H. C. B. Lee, "Vulnerability disclosure mechanisms: A synthesis and framework for market-based and non-market-based disclosures," *Decis. Support Syst.*, vol. 148, Sep. 2021, Art. no. 113586.

[12] M. Zhao, A. Laszka, and J. Grossklags, "Devising effective policies for bug-bounty platforms and security vulnerability discovery," *J. Inf. Policy*, vol. 7, pp. 372–418, Feb. 2017.

[13] M. Rao, D. F. Bacon, D. C. Parkes, and M. I. Seltzer, "Incentivizing deep fixes in software economies," *IEEE Trans. Softw. Eng.*, vol. 46, no. 1, pp. 51–70, Jan. 2020.

[14] L. Badash, N. Tapas, A. Nadler, F. Longo, and A. Shabtai, "Blockchain-based bug bounty framework," in *Proc. 36th Annu. ACM Symp. Appl. Comput.*, Mar. 2021, pp. 239–248.

[15] T. Maillart, M. Zhao, J. Grossklags, and J. Chuang, "Given enough eyeballs, all bugs are shallow? Revisiting Eric Raymond with bug bounty programs," *J. Cybersecurity*, vol. 3, no. 2, pp. 81–90, Oct. 2017.

[16] Y. Wu, H. Xiao, T. Dai, and D. Cheng, "A game-theoretical model of firm security reactions responding to a strategic hacker in a competitive industry," *J. Oper. Res. Soc.*, vol. 73, no. 4, pp. 716–740, Apr. 2022.

[17] D. Dey, A. Lahiri, and G. Zhang, "Hacker behavior, network effects, and the security software market," *J. Manage. Inf. Syst.*, vol. 29, no. 2, pp. 77–108, Oct. 2012.

[18] X. Gao and W. Zhong, "Information security investment for competitive firms with hacker behavior and security requirements," *Ann. Oper. Res.*, vol. 235, no. 1, pp. 277–300, Dec. 2015.

[19] K. Hausken, "A strategic analysis of information sharing among cyber hackers," *JISTEM-J. Inf. Syst. Technol. Manage.*, vol. 12, no. 2, pp. 245–270, May/Aug. 2015.

[20] X. Su, "Intertemporal pricing with strategic customer behavior," *Manage. Sci.*, vol. 53, no. 5, pp. 726–741, May 2007.

[21] X. Su and F. Zhang, "Strategic customer behavior, commitment, and supply chain performance," *Manage. Sci.*, vol. 54, no. 10, pp. 1759–1773, Oct. 2008.

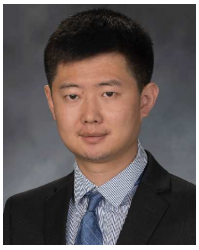
[22] Y. Song and X. Zhao, "Strategic customer behavior facing possible stockout: An experimental study," *Int. J. Prod. Econ.*, vol. 180, pp. 57–67, Oct. 2016.

[23] Z. Huang, L. Huang, Y. Zhao, and X. Meng, "Money-back guarantee in the presence of strategic customer behavior," *Int. J. Prod. Econ.*, vol. 239, Sep. 2021, Art. no. 108191.

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.

[24] J. Hou, X. Wang, and A. Z. Zeng, "Inter-temporal reward decisions with strategic ethical hackers," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Oct. 2023, pp. 1–9.
[25] (2022). *Hack DHS Program Successfully Concludes First Bug Bounty Program*. Accessed: May 7, 2024. [Online]. Available: https://www.dhs.gov/news/2022/04/22/hack-dhs-program-successfully-concludes-first-bug-bounty-program
[26] S. Shum, S. Tong, and T. Xiao, "On the impact of uncertain cost reduction when selling to strategic customers," *Manage. Sci.*, vol. 63, no. 3, pp. 843–860, Mar. 2017.
[27] J. Sheng, S. Du, T. Nie, and Y. Zhu, "Dynamic pricing vs. pre-announced pricing in supply chain with consumer heterogeneity," *Electron. Commerce Res. Appl.*, vol. 62, Nov. 2023, Art. no. 101311.
[28] J. Chaab and G. Zaccour, "Dynamic pricing in the presence of social externalities and reference-price effect," *Omega*, vol. 122, Jan. 2024, Art. no. 102963.
[29] J. Zhou and K.-L. Hui, "Bug bounty programs, security investment and law enforcement: A security game perspective," in *Proc. 2019 Workshop Econ. Inf. Secur. (WEIS), Boston, MA, USA*, pp. 3–4, 2019.
[30] M. Zhao, J. Grossklags, and K. Chen, "An exploratory study of white hat behaviors in a Web vulnerability disclosure program," in *Proc. ACM Conf. Comput. Commun. Secur. (SIW)*, 2014, pp. 51–58.
[31] (2024). *Badges*. Accessed: Mar. 5, 2024. [Online]. Available: https://docs.hackerone.com/en/articles/8390060-badges
[32] R. B. Myerson, *Game Theory: Analysis of Conflict*. Cambridge, MA, USA: Harvard Univ. Press, 1997.
[33] M. Dresher, *Games of Strategy: Theory and Applications*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1961.
[34] (2024). *Collaboration*. Accessed: Mar. 22, 2024. [Online]. Available: https://docs.hackerone.com/en/articles/8457618-collaboration?q=colla
[35] Hackerone. (2023). *Hacker-Powered Security Report*. Accessed: May 14, 2024. [Online]. Available: https://www.hackerone.com/resources/reporting/7th-annual-hacker-powered-security-report-2023
[36] (2024). *Hackerone Leaderboards*. Accessed: Apr. 30, 2024. [Online]. Available: https://hackerone.com/leaderboard
[37] (2024). *Leaderboard*. Accessed: Apr. 30, 2024. [Online]. Available: https://bug-bounty.com/leaderboard/

Jing Hou received the first Ph.D. degree in systems engineering from Southeast University, China, in 2011, and the second Ph.D. degree in computer science and software engineering from Auburn University, AL, USA, in 2021. She is currently an Assistant Professor with the Department of Computer Science and Information Systems, California State University San Marcos, CA, USA. Her research interests include network economics, machine learning, wireless communications and networking, security, and privacy.



Xuyu Wang (Member, IEEE) received the B.S. degree in electronic information engineering and the M.S. degree in signal and information processing from Xidian University, Xi'an, China, in 2009 and 2012, respectively, and the Ph.D. degree in electrical and computer engineering from Auburn University, Auburn, AL, USA, in August 2018. He is currently an Assistant Professor with the Knight Foundation School of Computing and Information Sciences, Florida International University, Miami, FL, USA. His research interests include wireless sensing, the Internet of Things, wireless localization, smart health, wireless networks, and deep learning. He received the NSF CRII Award in 2021. He was a co-recipient of the ACM FAcct 2023 Best Paper Award; the 2022 Best Journal Paper Award of IEEE ComSoc eHealth Technical Committee; the IEEE INFOCOM 2022 Best Demo Award; the IEEE ICC 2022 Best Paper Award; the IEEE Vehicular Technology Society 2020 Jack Neubauer Memorial Award; the IEEE GLOBECOM 2019 Best Paper Award; the IEEE ComSoc MMTC Best Journal Paper Award in 2018; the IEEE PIMRC 2017 Best Student Paper Award; the IEEE SECON 2017 Best Demo Award; and the Second Prize of the Natural Scientific Award of the Ministry of Education, China, in 2013.

Amy Z. Zeng has been the sixth and also the first female Dean of the Sawyer Business School, Suffolk University, Boston, MA, USA, since July 2020. She has brought to the school a passion and distinction for experience-based learning, cross-disciplinary and cross-sectoral collaborations, and global engagement. She is a Scholar recognized in the fields of supply chain management and global logistics. She is the award-winning author of more than 100 publications, including journal articles, book chapters, conference proceedings, and teaching cases. She has delivered more than 120 speeches spanning a wide range of locations and occasions, and secured nearly $\$1M$ grants for teaching innovations, research projects, and corporate sponsorships for student projects.

Authorized licensed use limited to: FLORIDA INTERNATIONAL UNIVERSITY. Downloaded on July 16,2025 at 21:44:24 UTC from IEEE Xplore. Restrictions apply.