Friendly Hackers to the Rescue: How Organizations Perceive Crowdsourced Vulnerability Discovery

5 authors, including:

Mortada Al-Banna
UNSW Sydney
3 PUBLICATIONS 3 CITATIONS
SEE PROFILE

Boualem Benatallah
UNSW Sydney
83 PUBLICATIONS 140 CITATIONS
SEE PROFILE

Daniel Schlagwein
The University of Sydney
48 PUBLICATIONS 266 CITATIONS
SEE PROFILE

Moshe Barukh
UNSW Sydney
32 PUBLICATIONS 66 CITATIONS
SEE PROFILE

Friendly Hackers to the Rescue: How Organizations Perceive Crowdsourced Vulnerability Discovery

Completed Research Paper

Mortada Al-Banna
UNSW Sydney
Australia
m.al-banna@unsw.edu.au

Boualem Benatallah
UNSW Sydney
Australia
boualem@unsw.edu.au

Daniel Schlagwein
UNSW Sydney
Australia
schlagwein@unsw.edu.au

Moshe C Barukh
UNSW Sydney
Australia
mosheb@cse.unsw.edu.au

Elisa Bertino
Purdue University
Indiana, USA
bertino@purdue.edu

Abstract

Over the past years, crowdsourcing has increasingly been used for the discovery of vulnerabilities in software. While some organizations have extensively used crowdsourced vulnerability discovery, other organizations have been very hesitant in embracing this method. In this paper, we report the results of a qualitative study that reveals organizational concerns and fears in relation to crowdsourced vulnerability discovery. The study is based on 36 key informant interviews with various organizations. The study reveals a set of pre-adoption fears (i.e., lacking managerial expertise, low quality submissions, distrust in security professionals, cost escalation, lack of motivation of security professionals) as well as the post-adoption issues actually experienced. The study also identifies countermeasures that adopting organizations have used to mitigate fears and minimize issues. Implications for research and practice are discussed.

Keywords: Crowdsourcing; Vulnerability Discovery; Security; Bug Bounty; Vulnerability Reward Program; Empirical Research

Introduction

This paper is about the use of crowdsourced vulnerability discovery by organizations that use software systems. Vulnerability discovery in software systems is an essential security tasks for organizations. Vulnerability discovery refers to the systematic search for bugs, backdoors, security holes and other vulnerabilities in systems (Liu et al. 2012). The purpose is preventing later exploitation by hackers, loss of data and corresponding negative business impacts. Increased system complexity, the web environment and reliance on third-party parts (e.g., cloud services, open APIs, external programming libraries) makes it difficult for in-house IT experts to perform sufficiently extensive and timely vulnerability discovery

Recently, a number of organizations have opted for crowdsourced approaches to vulnerability discovery. Crowdsourced vulnerability discovery benefits from the general advantages of crowdsourcing, such as diversity of participant skills, high scalability, fast speed and low cost (Brabham 2008)(Malone et al. 2010)(Li et al. 2015). In crowdsourced vulnerability discovery programs (also called bug bounty programs or vulnerability reward programs), organizations submit vulnerability discovery tasks as an open call to a community of security professionals (we will refer to the white hackers participating in crowdsourced vulnerability discovery tasks as security professionals). Several crowdsourcing platforms for vulnerability discovery have emerged (e.g., Bugcrowd, HackerOne, Synack). Individual organizations that use crowdsourced vulnerability discovery programs include

Google, Facebook, Microsoft and the US Department of Defense (U.S. Department of Defense 2016a). Facebooks reports that crowdsourced vulnerability discovery provides higher diversity of discovered vulnerabilities (Greene 2016). The US Department of Defense found that the costs of crowdsourced vulnerability discovery substantially lower than alternative approaches (U.S. Department of Defense 2016b)

Despite the benefits of crowdsourced vulnerability discovery, a surprisingly large number of organizations do not use it. For example, less than 6% of Forbes Global 2000 companies use it as of 2017 (HackerOne 2017), almost all of which run sizeable software systems. Organizations seem to be reluctant to embrace crowdsourced vulnerability discovery. Previous research investigated crowdsourced vulnerability discovery tasks in terms of benefit (M Finifter et al. 2013), effectiveness (Zhao et al. 2015), and crowd motivation(Laszka et al. 2016), but the underlying pre-adoption fears as well as the actual post-adoption issues have not been systematically studied.

At present, we do not understand: What are pre-adoption fears of organizations regarding crowdsourced vulnerability discovery? What are actual post-adoption issues experienced? What countermeasures are used to mitigate fears and minimize issues?

To answer the above research questions, we engaged in exploratory, empirical research. We-conducted semi-structured interviews (with 36 key informants), and qualitative data analysis (Ezzy 2002; Seaman 1999). In response to the research questions, we identify a set of pre-adoption fears that organizations have with respect to crowdsourced vulnerability discovery (i.e., lacking managerial expertise, low quality submissions, distrust in security professionals, cost escalation, lack of motivation of security professionals). We also identify a set of actual post-adoption issues adopting organizations face (i.e., low quality submissions, high cost of processing submissions, difficulties to maintain participants). Finally, we identified a set of countermeasures (i.e., learning, third party support, limiting participants, selective revealing, limiting scope and adjusting reward). These findings and the underlying analysis are reported in this paper.

The rest of this paper is organized as follows: in the next section we provide background information. In the subsequent section we present the research method. In the fourth section we summarize our findings. In the penultimate section we discuss our findings. We conclude the paper with a summary.

Background and Related Work

What is Crowdsourced Vulnerability Discovery?

A “vulnerability” is a security flaw that arises from system design, implementation or maintenance. Examples include SQL injection vulnerability, and logic vulnerabilities (e.g., allowing a discount code in an online market to be used multiple times till the cost of the purchase is zero). By exploiting these vulnerabilities, malicious parties could gain unauthorized access to protected resources (Krsul 1998). Relying on a single vulnerability discovery methods like manual penetration testing, static analysis, dynamic analysis has been proven to be not enough (Austin and Williams 2011). Additionally, there has been an increase in number of security threats and incidents in the past years (ISACA 2017). This has motivated the emergence of crowdsourced vulnerability discovery as a contemporary method of vulnerability discovery.

Crowdsourced vulnerability discovery works as follows. Organizations submit vulnerability discovery tasks to a community of security professionals. Security professionals upon discovering a specific vulnerability will make a submission of a vulnerability description report to the organization. The organization after verifying the legitimacy of the vulnerability will compensate the security professional with a reward. Crowdsourced vulnerability discovery has recently gained popularity as is evident from the increasing number of vulnerability discovery programs (Vijayan 2017).

Related Work

Researchers studied concerns, challenges, and issues of using crowdsourcing and how to mitigate them in various contexts. Lasecki et al. investigated different forms of threats from individuals and groups of workers extracting information from crowd-powered systems or manipulating these systems' outcomes (Lasecki et al. 2014). The authors also propose possible approaches to minimize and mitigate these threats. Daniel et al investigated quality control for crowdsourcing tasks and the methods to assess the quality attributes and the strategies that could be used to prevent and mitigate quality issues (Daniel et al. 2017). Wolfson et al. investigated the legal issues that may face organizations relying on crowdsourcing (Wolfson and Lease 2011). They identified several issues with employment laws, patent laws, data security, and copyrights. The authors provided recommendations to help address these legal issues. Stol and Fitzgerald investigated the challenges faced by organizations relying on crowdsourcing for software development (Stol and Fitzgerald 2014). They identified issues in regard to task decomposition, coordination and communication, planning and scheduling, quality assurance, and knowledge and intellectual property. Different strategies have been investigated by researcher to improve the quality of crowdsourced tasks. Filtering out the bad outcomes of the task is one strategy that has been investigated by researchers. Dow et al. filter outputs of the task based on expert's reviews (Dow et al. 2012), Marcus et al. filter according to the ground truth (Marcus et al. 2012), and Rao et al filter according to majority voting (Rao et al. 2013). Incentivizing the crowd to provide high quality output is another strategy investigated to enhance the quality of submissions. Ho et al. in their work investigate the effect of financial incentives on the quality of crowd output (Ho et al. 2015). Providing feedback about the performance of the crowd worker has been discovered to help the workers provide better quality results (Dow et al. 2012). Kulkarni et al. in their work illustrate the effect of the task requester's feedback on the quality of the output of the task (Kulkarni et al. 2012). Doroudi et al investigate the how to effectively teach the crowd by providing experts' examples for the crowd to learn from (Doroudi et al. 2016). And finally, Gamifying the task has proven to produce better results specially in complex tasks(Krause and Kizilcec 2015). Law et al showed that designing tasks that induce curiosity improve worker's retention in crowdsourcing (Law et al. 2016).

Kannan and Telang studied regulated and unregulated vulnerability markets (Kannan and Telang 2005). They found that it is socially beneficial to offer rewards for benign vulnerability discoverers. Algarni et al. examined the motivations and methods of security professionals participating in vulnerability discovery. They have identified multiple vulnerability discovery markets where exchanges between the discoverers and the buyers take place. They found that the majority of security professionals participating in the task of vulnerability discovery are from outside the software organizations and that their key motivation is monetary reward (Algarni and Malaiya 2014). Finifter et al. examined the characteristics of two crowdsourced vulnerability discovery programs (Matthew Finifter et al. 2013). They concluded that such programs appear economically efficient compared to the cost of hiring full-time security professionals. Zhao et al. conducted quantitative analyses for different vulnerability aspects of the Web ecosystem (Zhao et al. 2015). They found that monetary incentives have a significantly positive correlation with the number of vulnerabilities reported. They also investigated productivity and accuracy of the individuals (i.e. the number of valid submission against invalid submissions). Laszka et al. proposed a strategy that would incentivize Security professionals participating in task to self-assess their submissions and hence minimize invalid submissions (Laszka et al. 2016).

Analyzing and understanding the perception of organizations with respect to the fears surrounding crowdsourced-based vulnerability discovery tasks, the issues faced by the adopters of this approach and the strategies and techniques used by them to minimize or mitigate these fears and issues is paramount to understand the field. To the best of our knowledge, while previously identified as much "needed future work" by other researchers (Zhao et al. 2015), no existing work provides a rigorous analysis of fears, issues and countermeasures regarding vulnerability discovery programs as presented in this paper.

Research Method

Our research methodology consists of iterative phases of data collection and data analysis. We interviewed 36 key informants (personnel who hold authority in the field of security; oversee the security posture of organizations; and influence (directly or indirectly) the security strategy within organizations). We invited participants affiliated with organizations mentioned in a community-curated list of crowdsourced vulnerability discovery and disclosure programs, namely (Firebounty 2015). We sent direct e-mail invitations to 126 organizations with publicly available contact information. We advertised our study to information security groups on LinkedIn. We also used snowballing to get recommendations from security professionals about colleagues who would be interested to participate in our study (Research-office 2015). It was apparent that snowballing is the most effective approach since a more direct approach may often be considered as phishing –especially as phishing attacks are well known by security professionals who are much more cautious before accepting invitations from unknown parties. Another approach that proved effective is to directly approach the key informants during cyber security events (e.g., OWASP chapter meetings). This gave us the opportunity to explain the motivation of our research, and increased interest in the study.

The interviews were conducted by one of the authors according to an interview guide (Interview 2016). Summaries of the interviews were discussed to draw insights and identify the key emerging themes. The interviews lasted between 30 and 60 minutes and were conducted face-to-face, via Skype, tele-presence systems or phone calls. Audio was recorded when the interviewee consented, and notes were taken. The interviews were semi-structured based on the literature review and online content investigation, and interviewers could ask additional follow-up questions.

We followed the recommendations of Adler and Adler and aimed for a sample size between 12 and 60 participants (Baker and Edwards 2012). We used the criterion of theoretical saturation (i.e., no new insights were emerging from new instances) to determine the appropriate end point of our empirical data collection (Corbin and Strauss 1998; Ezzy 2002). When it was feasible we reinitiated contact with some of the interviewees to further clarify. To cater for diverse perspectives, we invited participants from different disciplines (e.g., finance, entertainment, and communications). The positions of the interview participants were diversified as well (e.g., CTOs, CISOs, IT managers, security analysts, security testing team leaders). The sample included small (25-100 employees), medium (100-500 employees), and large (over 500 employees) organizations. Participants are referred to by an anonymous identifier denoted as (P#). Among the interviewees, 10 participants (P3, P11, P13, P14, P17, P24, P25, P29, P33 and P34) have experience in crowdsourced vulnerability discovery programs.

For the data analysis, following each interview, we transcribed all interview data; we organized the transcripts and associated notes into easily retrievable sections. We also obtained increased familiarity with the data through reading and re-reading and writing down notes and summaries. We coded the data using the techniques of thematic analysis (Braun and Clarke 2006; Ezzy 2002): we first analyzed the data through open coding, developing a codebook that we refined over time. We aggregated relevant open codes into higher-level abstract codes (concepts) and analyzed the relationship between these concepts (taking into consideration existing theory and terminology, so as to connect our analysis to the ongoing academic discourse).

Findings

In this section, we summarize the fears, issues and countermeasures that emerged from our study. We discuss the findings from the interviews; and provide quotes to help explain how we derived our conclusions. We also explored the reasoning behind the answers. A summary of the findings from this study is illustrated in Figure 1.

Fears

We identified five themes in relation to fears that organizations have prior to using crowdsourcing for vulnerability discovery (Lacking managerial expertise, low quality submissions, distrust in security professionals, cost escalation, and lack of motivation of security professionals)

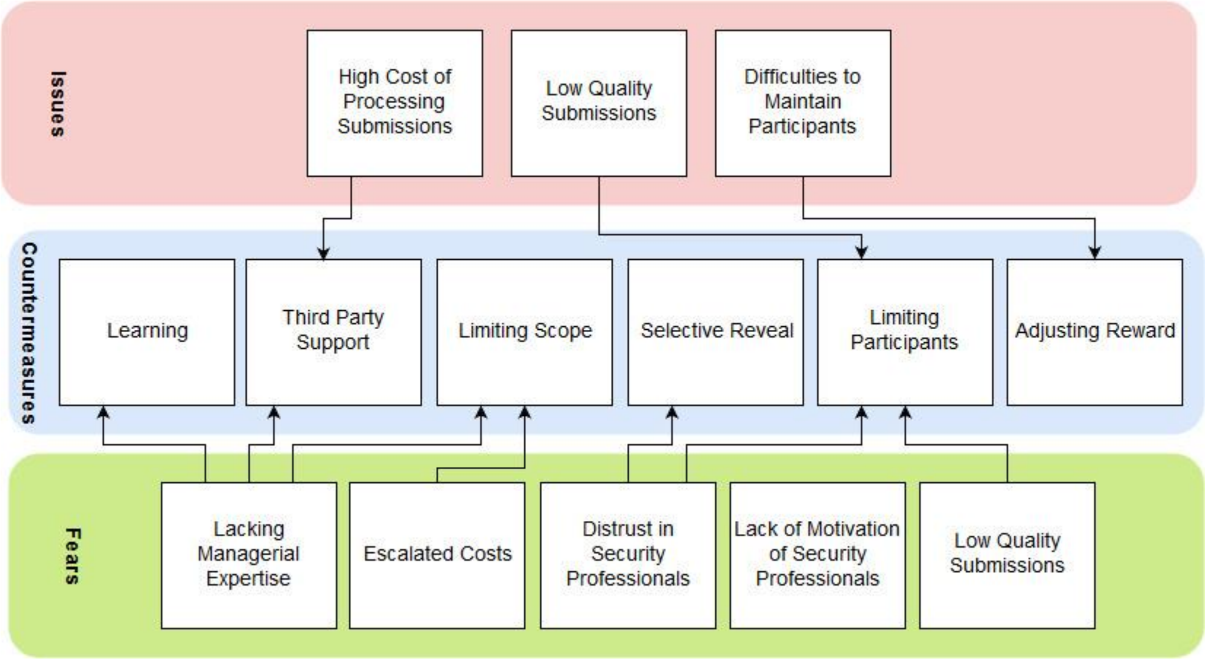

Figure 1 Summary of the Findings (arrows represent the relationship between the countermeasure and the fear or issue)

Lacking Managerial Expertise

Crowdsourcing vulnerability discovery is a relatively new paradigm. Being able to successfully plan, launch and manage the task require some knowledge and expertise. The effect on the success of the task due to lacking expertise is one of the fears of organizations. A product security evaluation lead in one of the global telecommunication equipment companies said "We have experts in all security disciplines but when it comes to running a crowdsourced vulnerability discovery program the skill-set is different. The field is relatively new, and it is not easy to recruit people with required experience" [P27]. It was also observed during the interviews that the key informants kept asking about the opinion of the authors and their recommendations.

Low-Quality Submissions

As security professionals have different backgrounds, skills and expertise, the outcomes may differ in quality. A low-quality submission may render worthless for the organization, since they will not be able to extract the required information to verify the legitimacy of the vulnerability and hence remediate it. A senior security lead in one of the business management companies shared his fears in regard to the quality of the report itself and whether it can be understood easily and proved useful "These hackers are from all around the world and so submitting reports that are not clear, not complete or plain ambiguous, would force our security staff to spend hours looking at it and in the end, it may or may not be useful" [P10].

Vulnerabilities differ in their impact if exploited, and they differ in the complexity of the exploitation. Several systems have thus been used to assign priorities to vulnerabilities (e.g., Common Vulnerability Scoring System (FIRST.Org 2015)). Organizations expressed the fear that the submitted vulnerabilities would be of low severity or acceptable risk. A security and systems engineer in a financial institute mentioned, "We have seen teenagers, hobbyists, and even housewives participating in programs, I doubt that the next Remote Code Execution vulnerability will come through them" [P25].

It was shown that organizations value actionable outcomes that would help them increase the level of security and that efficiency is an important aspect to consider. A penetration testing team leader in one of the security consultation companies mentioned "it is unreasonable to pay thousands of dollars for a pile of rubbish submissions only in hope to find some jewels inside" [P19].

Distrust in Security Professionals

Delegating a security task of vulnerability discovery requires a great deal of trust. Trust would be distributed between the security professionals and the platform that the organizations use to launch the task. Security professionals could be anywhere in the world and the organization would have a relatively limited information about them. A Chief Technology Officer of a security controls software company mentioned "To trust security professionals that we have only minimum information about such a critical task is somewhat difficult to absorb" [P20]. This could be due to the nature of security personnel as they are trained not to offer their trust easily.

Organizations also fear that the security professionals will participate in the task only to find vulnerabilities they can sell in the black market. This might be due to the perception that the vulnerability black market is more rewarding than the legal market (e.g., a grey market company in 2015 offered 1 million dollars for a vulnerability and working exploit for iOS). A senior security analyst in a SaaS provider mentioned "The grey market now is getting traction and selling there would be less risky and more profitable so how can we compete" [P26].

It was also reported that lacking the ability to hold legal accountability is one of the fears, since security professionals could be anywhere in the world where it is not possible to legally prosecute them if they act maliciously. An information security expert in a telecommunication company mentioned, "If security professionals act mischievously, who do we hold responsible for that? How can we pursue them legally? Or do we go after the platform that crowdsourced the task on our behalf?" [P7].

Costs Escalation

Crowdsourcing vulnerability discovery has a unique budgeting requirement. For crowdsourcing microtasks the organization can have a rough estimate about the cost (e.g., the task of transcribing 100 pages would mean multiplying 100 with the cost of one page and taking in consideration some management overhead and contingencies). Similarly, for competition-based tasks (e.g., ideas, software development, design) the organization announces a prize for the winner submitting the best solution and limits the costs of the prize. While for crowdsourcing vulnerability discovery, although it is also a competition-based task, the tasks can have multiple winners. As along as the security professional submits a valid vulnerability, the company has to abide by the rules for payments mentioned in the task description. Hence the number of vulnerabilities discovered cannot be determined and is related to the security of the product being tested and the number of security professionals participating in the task. A quality assurance team lead in a backup solutions company mentioned, "budgeting for a crowdsourced program is hard, we cannot predict how many vulnerabilities will be discovered and paid for" [P4]. A security lead in a SaaS provider described crowdsourcing vulnerability discovery as a potential "financial black hole" [P30].

Lack of Motivation of Security Professionals

Attracting security professionals to participate in the task is important especially in the presence of other crowdsourced vulnerability discovery tasks to compete with. A senior security engineer in a system integration service provider company mentioned "How can we know if we are paying the right amount for the discovered vulnerabilities? We do not want to be pay less and drive security professionals away or pay more than what we should" [P16]. Similarly, maintaining engagement and ensuring a stable flow of submissions will help organizations to have better coverage in terms of vulnerabilities discovered. A Chief Security Officer in a software provider company mentioned "When the program is running for a while and security professionals stop submitting vulnerabilities to us, does that mean our product is secure and there is nothing to find? Or is it just that security professionals picked the low hanging fruits and they are not interested anymore?" [P31].

Issues

Three issues were reported by organizations who have adopted crowdsourcing vulnerability discovery:

Low Quality Submissions

It has been reported that some organization had issues of low signal to noise ratio. Noise could be in the form of invalid reports, duplicates, spam, and out of scope. Organizations consider this issue to be a major one, as it would cause them to waste both time and resources to filter out the low-quality submissions. An internal security assessor in a charity organization mentioned, “we get bombarded by a huge number of reports that our team need to go through one by one and most of the time it is more noise than legitimate vulnerabilities” [P13]. The quality of the report is also important to the organization since the report is useless if organizations cannot reproduce and validate the vulnerability due to the report being incomplete or not clear. A security lead in a mobile gaming company mentioned, “It is time and effort consuming when we go back and forth asking for missing details, explanation, and proof of concept. This problem is amplified with reports that were prepared using services such as Google translate to translate from the original language of the security professional” [P14].

High Cost of Processing Submissions

When the organization receives a vulnerability report, they need to verify the legitimacy of the vulnerability discovered, making sure it was not reported before; determining the impact and severity of vulnerability; paying out the bounty to the security professional; and following-up with the development team to make sure that the vulnerability is remediated properly. This process when applied to a huge number of submissions would create burdens on the internal staff responsible for the process.

A security lead in a mobile gaming company mentioned “When we started our program we had some issues assigning tasks related to our crowdsourcing program to our internal team, we were putting too much pressure on our internal staff and that was affecting their daily tasks along with delays in processing the submitted vulnerabilities. We ended up hiring additional internal staff to take care of the vulnerability triage” [P14].

Difficulties to Maintain Participants

Security professionals who frequently submit vulnerability reports, in essence, spend more time on the same task and build some sort of loyalty to the task. If the organization does not manage to maintain engagement with security professionals, then the submissions for the task will dropped (Leyden 2016). A senior security consultant in a security consultation company mentioned “It is not easy to ensure engagement of security professionals and it is quite expected for submissions to drop dramatically after few weeks of launching the program when the low hanging fruits have already been picked up” [P3].

Competing with other organizations running crowdsourced vulnerability discovery tasks is also a challenge. A director of security architecture in a software provider company mentioned “After the low hanging fruits in our program is depleted, it is a matter of which organization keeps their program more profitable and interesting to keep the security professionals engaged” [P24].

Countermeasures

Five countermeasures were reported: Learning over time; relying on third party support; inviting only experts and vetted security professionals; selective reveal of information; and adjust rewards. It is to be noted that sometimes a countermeasure could be applied for more than one fear or issue (e.g., invite only experts and vetted security professionals can help with both issues of low quality submissions, as well as lack of trust in security professionals)

Learning

Organizations gain experience by running the crowdsourced vulnerability discovery task over a period of time. Organizations learn from the experiences they come across during the crowdsourced task. A security lead in a multimedia company mentioned, “We carefully assess every step we take while running our bug bounty program and we hold meetings to reflect and decide our next step” [P11]. Additionally, organizations learn from observing other organizations that is running a crowdsourced vulnerability discovery task. A security and system engineer in a financial institute mentioned “The stories and bug bounty program reviews done by other companies helped us, by knowing that other people walked the same path and how they handled some of the problems we are facing” [P25].

Third-Party Support

Organizations who did not have the knowledge and the expertise to launch and manage a crowdsourced vulnerability discovery task relied on professional services provided by third parties (e.g., Platforms for crowdsourcing vulnerability discovery). A security lead in a social media company mentioned, “We depended a lot on the platform to give us guidance how to design and run our program and we outsourced tasks of the triage process to the platform" [P17].

We identified that relying on the third-party support helped organizations to offload the burden of controlling the quality of submissions and managing the cost of processing the submissions. An internal security assessor in one of the charity organizations mentioned "We got help from the platform in managing our program and triaging the large number of submissions since we did not want to hire more people for a permanent position which would be costly" [P13].

Limiting Participants

Inviting only selected security professionals according to their expertise, past performance, or other criterion, is a strategy used by organizations to increase the quality of the outcome of the crowdsourced task of vulnerability discovery. It is a well-known strategy in the field of crowdsourcing vulnerability discovery and is often referred to as private programs (e.g., Onelogin private program (OneLogin 2017)) or invite only programs (e.g., Apple invite only bug bounty program (Conger 2016)). A senior product security engineer in a software provider company mentioned "We prefer to keep our program invite only since working with amateurs would be time consuming and will cause us to get less the output for double the effort" [P34].

We have also observed that selecting security professionals with verified expertise and identities is relied on by organizations to minimize the fear of untrustworthy security professionals participating in crowdsourced vulnerability discovery tasks. A product manager in a Bitcoin exchange services company mentioned "We prefer to work with a few selective vetted security professionals with good track record, since it is risky to work with amateurs in terms of tools they use that could be damaging and the maturity to know when to stop" [P29].

Selective Revealing

Organizations may share information with security professionals to help them discover vulnerabilities more effectively (e.g., source-code, test credentials, entry points, access to unreleased products). In order to minimize the concerns with respect to the trustworthiness of security professionals participating in crowdsourced tasks of vulnerability discovery, organization tend to control the reveal of some information. The director of security architecture in a software provider company mentioned "the source-code of our product is a very important asset to our company, so we do not share our source-code with security professionals even if it means we may minimize the efficiency of our bug bounty program" [P24]. Some organizations allow access to a staging environment with synthetic data to minimize the risk of accidental access to customer data. A security and systems engineer in a financial institute mentioned "Our program run in a staging environment to eliminate any chance that the security professional would get access to our customer data even if by accident during testing" [P25].

Limiting Scope

Each crowdsourcing vulnerability discovery task has a scope that helps security professionals understand clearly what the targets are (e.g., domains, services, applications), and what are the acceptable submissions (e.g., types of vulnerabilities the organization is interested in). Organizations also need to manage the expectations of security professionals in regard to the reward they will receive in compensation for investing their time to discover the vulnerabilities (e.g., range of monetary rewards, recognition). In order to manage the unpredictability of cost when a crowdsourced vulnerability discovery task is launched, organizations tend to start with small scope and reward scheme. A platform security lead in a multimedia company mentioned "We started small with only the main domains we have and providing only swag to the security professionals in return for their responsible disclosure of vulnerability, this way we can start with a smaller budget that the board would more likely approve and minimize the possibility of a spike in the payments" [P11].

We observed that limiting the scope and rewards was also used to minimize the flood of submissions and help the organization cope with the lack of expertise for managing a crowdsourced vulnerability discovery task. A security lead in a social media company mentioned "We took baby steps throughout our program and the scope was in the beginning only our mobile app, so we knew which team needs to be on alert and we had the flexibility to better absorb how the program is working" [P17].

Adjusting Rewards

If the crowdsourced task for vulnerability discovery was running for a long time and all the easy to find vulnerabilities were discovered, organizations need to motivate security professionals to keep engaged in the task. One way to motivate security professionals is through adjusting the rewards provided to them. A security lead in a social media company mentioned "In the beginning, we get a flood of simple, easy to find, spams, and duplicates, but after a while the submissions drop drastically, so we try to keep things interesting for security professionals by increasing the bounties and providing bonuses for high impact vulnerabilities and clever exploitation" [P17].

Discussion

We observed that organizations have some fears with respect to crowdsourced vulnerability discovery including,: lacking managerial expertise, low quality submissions, distrust in security professionals, cost escalation, lack of motivation of security professionals. We also observed that organizations that have adopted crowdsourced vulnerability discovery have reported issues they have faced including: low quality submissions, high cost of processing submissions and difficulties to maintain participants. Additionally, these organizations reported countermeasures they have relied on to minimize or mitigate these fears and issues: learning, third party support, limiting participants, selective revealing, limiting scope and adjusting reward.

It was evident from our findings that some fears are manifested into actual issues faced by the organizations (e.g., lack of motivation of security professionals, low quality submissions, cost escalation) while others did not (e.g., lacking managerial expertise and Lacking trust in security professionals). This could be the effect of using the countermeasures (such as, relying on third party support, selective reveal of information to security professionals and inviting only expert and vetted security professionals) as preventive measures to mitigate these fears before they occur.

Previous research investigated crowdsourcing vulnerability discovery: quantitatively by analyzing data sets of vulnerability submissions (Matthew Finifter et al. 2013; Maillart et al. 2017; Munaiah and Meneely 2016; Zhao et al. 2014, 2015), or qualitatively relying on input from security professionals participating in vulnerability discovery tasks (Al-Banna et al. 2016; Algarni and Malaiya 2013, 2014; Hafiz and Fang 2016). Some research proposed solutions to existing problems like incentivizing security professionals to submit valid vulnerabilities (Laszka et al. 2016), or helping organizations against the high rate of submissions through crowdsourcing vulnerability verification (Su and Pan 2016). To the best of our knowledge our work is the first to investigate (through a qualitative analysis of interview data) how organizations perceive crowdsourced vulnerability discovery. We think it is important in setting the foundation of a wholesome understanding of crowdsourced driven tasks on vulnerability discovery. We believe that a better understanding of the fears, issues surrounding the crowdsourced task of vulnerability discovery and the countermeasures currently relied on by organizations to mitigate or minimize these issues and fears, can help researchers investigate how to adopt countermeasures from other domains of crowdsourcing.

Improving the quality of the output of the crowdsourced task has been the focus of many researchers. Filtering out the bad outcomes of the task is one strategy that has been investigated by researchers. Dow et al. filter outputs of task based on expert's reviews (Dow et al. 2012), Marcus et al. filter according to the ground truth (Marcus et al. 2012), and Rao et al filter according to majority voting (Rao et al. 2013). Given the high noise in the output of the crowdsourced task of vulnerability discovery (BugCrowd 2017), filtering the noise (invalid, duplicates, out of scope…etc.) would be most desirable. Nevertheless, due to the complexity of the task and the sensitivity of the submission one plausible approach is to rely on expert reviews for filtering, such as the latest service offered by Hackerone called Human Augmented Signal (Russchen 2018), where the organization delegate the task of filtering noise to the staff of the platform. Also relying on machine learning (e.g., content analysis and entity extraction) could also be helpful to filter out duplicates and out of scope submission. Maleej and Nabil in their work relied on machine learning to classify app reviews to determine whether it is a bug report, a feature request or just a praise (Maalej and Nabil 2015). Platforms like Hackerone has a tool they call trigger action where it relies on text analysis to trigger an alert or an action in case it detects and expression (Hackerone 2018).

Incentivizing the crowd to provide high quality output is a method investigated to enhance the quality of submissions. Ho et al. in their work investigate the effect of financial incentives on the quality of crowd output (Ho et al. 2015). Laszka et al. proposed a strategy that would incentivize security professionals participating in the crowdsourced vulnerability discovery task to self-assess their submissions and hence minimize invalid submissions (Laszka et al. 2016).

Another way to improve the quality of the output of the task is to train the crowd and help them produce better output. Providing feedback about the performance of the crowd worker has been discovered to help the workers provide better quality results (Dow et al. 2012). Kulkarni et al. in their work illustrate the effect of the task requester's feedback on the quality of the output of the task (Kulkarni et al. 2012). Doroudi et al investigate the how to effectively teach the crowd by providing experts' examples for the crowd to learn from (Doroudi et al. 2016). Platforms like Hackerone offer security professionals access to some submitted vulnerability reports and by providing access to some online courses and materials.

The crowdsourced task description (aka program brief) contains the information the organization share with the security professionals so that they can effectively perform the task. Enhancing the clarity of the task description can help the Security professionals better understand what they need to do and what they would expect within the program (Kuehn and Mueller 2014). Additionally, making sure that the task description contains all the information needed by the security professional would also help in minimizing error (e.g., by mentioning what is out of the scope of the program the possibility that Security professionals would submit out of scope reports is minimized)(Bugcrowd 2016).

In regard to the engagement with the crowd, Finifter et al in their work highlights that offering extra reward top-ups and bonuses increase engagement with Security professionals. Similarly, Zhao et al. describe bonuses as a good way to incentivize new participants (Zhao et al. 2015). Additionally, researchers found that offering hiring opportunities for Security professionals who perform well in the crowdsourced vulnerability discovery task incentivize them and keep them engaged (Chatfield and Reddick 2017; Matthew Finifter et al. 2013; Zhao et al. 2014). Gamifying the task has proven to produce better results specially in complex tasks(Krause and Kizilcec 2015). Law et al showed that designing tasks that induce curiosity improve worker's retention in crowdsourcing (Law et al. 2016). Platforms like Bugcrowd, Hackerone and organizations like Google and Facebook acknowledge the importance of gamification and rely on ranking Security professionals and creating a reputation system to create competition between the participants.

We have observed that the lack of managerial expertise has been raised as one of the pre-adoption fears and that organizations who have adopted crowdsourced vulnerability discovery has tackled this fear actively with learning over time, rely on support from third party and limited scope and budget initiation. It would be interesting from a practical perspective to establish the skill set required for managing a crowdsourced vulnerability discovery task. Platforms for crowdsourcing vulnerability discovery like Hackerone have suggested some requirements for what they called a bug bounty leader (Bacchus 2016) (the person responsible for managing the crowdsourced task of vulnerability discovery). We believe that in addition for the technical security skills the candidate need to be familiar with crowdsourcing and the quality control assessment methods (e.g., rating, peer review and content analysis) and assurance strategies (e.g., data cleansing, task recommendation and task decomposition) (Daniel et al. 2017). There are also some industrial standards that the candidate need to be familiar with like ISO29147 Vulnerability Disclosure, ISO30111-Vulnerability handling process and control A.12.6.1 of ISO27001-Technical vulnerability management.

Our work will help organizations and crowdsourcing platforms for vulnerability discovery be aware of fears and issues perceived in regard to crowdsourcing vulnerability discovery and help design the task to minimize and mitigate these fears and issues. There is no 'one task fits all' solution for crowdsourcing vulnerability discovery and hence organizations need take into consideration the relevant fears and issues associated and the best countermeasures to minimize or mitigate them. By looking at current countermeasure strategies and how they are being used, organizations can adapt their own set of countermeasures, and platforms can work on enhancing these countermeasures.

There are some limitations to this study. The data was collected in 2015-2016 from a broad range for organizations. While this allowed us to draw some interesting insights from various perspectives, we did not study any one case over a long period of time. The in-depth analysis of particularly interesting cases over longer periods may reveal additional insights, as may the application of additional methodological lenses. We neither aim nor claim to provide a universal, timeless truth with the knowledge claims inductively developed here. Although we obtained feedback from a broad range of organizations, it is possible that there are some fields that we could not reach key informants working in these respective fields (e.g., government). Similarly, we tried to reach out to organizations that had a failed implementation of a crowdsourced vulnerability discovery task, but we were unsuccessful. This would help to gain even further insight. Nonetheless, we have aimed to mitigate this shortcoming by targeting various industries in order to obtain a broader view of possible perspectives. One of the authors of this paper also attended multiple security conferences (e.g., BSides, Ruxcon), to directly discuss with attendees. Although some feedback was acquired about the study from key informants, it was mostly an off-the-record approach. As the interviews were semi structured, another problem concerning the validity is that the study may suffer from confirmatory bias. In order to mitigate this concern, we started each interview with open questions about their perception in regard to crowdsourcing vulnerability discovery and wrote notes to review in regard to what they mentioned, and also closed the interview by asking if there are additional information they wanted to share with us. We also asked for clarification about each concern the interviewee may have in order to minimize any possible bias that may influence the interviewees to answer in agreement just to please the researchers.

Conclusions

In this paper, we reported the results from a qualitative study that sought to gain insight about organizational concerns, issues and countermeasures in relation to crowdsourced vulnerability discovery tasks. The study was based on 36 key informant interviews from various organizations. The study revealed a set of pre-adoption fears of using crowdsourced vulnerability discovery (lacking managerial expertise, low quality submissions, distrust in security professionals, cost escalation, lack of motivation of security professionals) as well as the actual issues faced by organizations that have adopted crowdsourced vulnerability discovery low quality submissions, high cost of processing submissions, difficulties to maintain participants). The study also identified countermeasures that organizations have used to mitigate or minimize these fears and issues (learning, third party support, limiting participants, selective revealing, limiting scope and adjusting reward). The implications for research and practice of these findings were discussed.

Acknowledgments

We would like to thank all the participants in the interviews for their time. We would like also to thank the editors and anonymous reviewers for their valuable feedback.

References

Al-Banna, M., Benatallah, B., and Barukh, M. C. 2016. “Software Security Professionals: Expertise Indicators,” in *2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC)*, IEEE, November, pp. 139–148.

Algarni, A., and Malaiya, Y. 2013. “Most Successful Vulnerability Discoverers: Motivation and Methods,” *Proceedings of the International Conference on Security and Management (SAM).*

Algarni, A., and Malaiya, Y. 2014. “Software Vulnerability Markets: Discoverers and Buyers,” *International Journal of Computer, Information Science and Engineering* (8:3), pp. 71–81.

Austin, A., and Williams, L. 2011. “One Technique Is Not Enough: A Comparison of Vulnerability Discovery Techniques,” in *2011 International Symposium on Empirical Software Engineering and Measurement*, IEEE, September, pp. 97–106.

Bacchus, A. 2016. “Bug Bounty Leader - Job Description Template.” (https://docs.google.com/document/d/1TWdTOl2FXLko2Eu5UGG1fFQxO7zFxxTaJWWnP8W

W2F8/edit, accessed March 5, 2018).

Baker, S. E., and Edwards, R. 2012. "How Many Qualitative Interviews Is Enough: Expert Voices and Early Career Reflections on Sampling and Cases in Qualitative Research," *National Centre for Research Methods Review Paper*, pp. 1–43.

Brabham, D. C. 2008. "Crowdsourcing as a Model for Problem Solving: An Introduction and Cases," *Convergence: The International Journal of Research into New Media Technologies* (14:1), Sage PublicationsSage UK: London, England, pp. 75–90.

Braun, V., and Clarke, V. 2006. "Using Thematic Analysis in Psychology," *Qualitative Research in Psychology* (3:2), pp. 77–101.

Bugcrowd. 2016. "Anatomy of a Bounty Brief." (https://pages.bugcrowd.com/hubfs/PDFs/Anatomy-Bounty-Brief.pdf?t=1514409250629, accessed February 3, 2018).

BugCrowd. 2017. "2017 State of Bug Bounty Report." (https://pages.bugcrowd.com/hubfs/Bugcrowd-2017-State-of-Bug-Bounty-Report.pdf).

Chatfield, A. T., and Reddick, C. G. 2017. "Cybersecurity Innovation in Government," in *Proceedings of the 18th Annual International Conference on Digital Government Research - Dg.o '17*, New York, New York, USA: ACM Press, pp. 64–73.

Conger, K. 2016. "Apple Announces Long-Awaited Bug Bounty Program | TechCrunch." (https://techcrunch.com/2016/08/04/apple-announces-long-awaited-bug-bounty-program/, accessed March 5, 2018).

Corbin, J. M., and Strauss, A. L. 1998. *Basics of Qualitative Research : Techniques and Procedures for Developing Grounded Theory*, Thousand Oaks, CA: Sage.

Daniel, F., Di Milano, P., Kucherbaev, P., Cappiello, C., Benatallah, B., and Allahbakhsh, M. 2017. "Quality Control in Crowdsourcing: A Survey of Quality Attributes, Assessment Techniques and Assurance Actions," *ACM Computing Surveys*.

Doroudi, S., Kamar, E., Brunskill, E., and Horvitz, E. 2016. "Toward a Learning Science for Complex Crowdsourcing Tasks," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, New York, New York, USA: ACM Press, pp. 2623–2634.

Dow, S., Kulkarni, A., Klemmer, S., and Hartmann, B. 2012. "Shepherding the Crowd Yields Better Work," in *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work - CSCW '12*, New York, New York, USA: ACM Press, p. 1013.

Ezzy, D. 2002. *Qualitative Analysis : Practice and Innovation*, Routledge. (https://books.google.com.au/books?hl=en&lr=&id=_YSMAQAAQBAJ&oi=fnd&pg=PR9&dq=Qualitative+Analysis:+Practise+and+Innovation&ots=rMklS3FvST&sig=EJLfhtBWkZe7MgLxV04s6jFqxC0#v=onepage&q=Qualitative Analysis%3A Practise and Innovation&f=false).

Finifter, M., Akhawe, D., Symposium, D. W.-U. S., and 2013, undefined. 2013. "An Empirical Study of Vulnerability Rewards Programs.," in *Proceedings of the 22Nd USENIX Conference on Security*, Washington, D.C: USENIX Association, pp. 273–288.

Finifter, M., Akhawe, D., and Wagner, D. 2013. "An Empirical Study of Vulnerability Rewards Programs," in *Proceedings of the 22Nd USENIX Conference on Security*, Washington, D.C.: USENIX Association, pp. 273--288.

Firebounty. 2015. "FireBounty | The Ultimate Bug Bounty List!" (https://firebounty.com/, accessed March 5, 2018).

FIRST.Org. 2015. "CVSS v3.0 Specification Document." (https://www.first.org/cvss/specification-document, accessed March 5, 2018).

Greene, C. 2016. "Bug Bounty 5 Years in – Collin Greene – Medium." (https://medium.com/@collingreene/bug-bounty-5-years-in-c95cda604365, accessed March 5, 2018).

Hackerone. 2018. “How Do We Manage Reports of Known Behavior? – HackerOne.” (https://support.hackerone.com/hc/en-us/articles/207858796-How-do-we-manage-reports-of-known-behavior-, accessed March 5, 2018).

HackerOne. 2017. “The Hacker-Powered Security Report 2017.” (https://www.hackerone.com/sites/default/files/2017-06/The Hacker-Powered Security Report.pdf).

Hafiz, M., and Fang, M. 2016. “Game of Detections: How Are Security Vulnerabilities Discovered in the Wild?,” *Empirical Software Engineering* (21:5), Springer US, pp. 1920–1959.

Ho, C.-J., Slivkins, A., Suri, S., and Vaughan, J. W. 2015. “Incentivizing High Quality Crowdwork,” in *Proceedings of the 24th International Conference on World Wide Web - WWW ’15*, New York, New York, USA: ACM Press, pp. 419–429.

Interview. 2016. “Interview Guide: Guiding Questions for the Interviews.” (https://docs.google.com/document/d/156lW3ipcW5QIE8XSeNGvIUJsewM3sBYjgr4HM1lK738/edit, accessed March 5, 2018).

ISACA. 2017. “State of Cyber Security 2017 State of Cyber Security 2017: Part 2: Current Trends in the Threat Landscape.” (http://www.isaca.org/Knowledge-Center/Research/Documents/state-of-cybersecurity-2017-part-2_res_eng_0517.PDF?regnum=428946).

Kannan, K., and Telang, R. 2005. “Market for Software Vulnerabilities? Think Again,” *Management Science*. (http://pubsonline.informs.org/doi/abs/10.1287/mnsc.1040.0357).

Krause, M., and Kizilcec, R. 2015. “To Play or Not to Play: Interactions between Response Quality and Task Complexity in Games and Paid Crowdsourcing,” *Third AAAI Conference on Human Computation and Crowdsourcing*.

Krsul, I. V. 1998. “SOFTWARE VULNERABILITY ANALYSIS,” Purdue University.

Kuehn, A., and Mueller, M. 2014. “Shifts in the Cybersecurity Paradigm: Zero-Day Exploits, Discourse, and Emerging Institutions,” in *Proceedings of the 2014 Workshop on New Security Paradigms Workshop - NSPW ’14*, New York, New York, USA: ACM Press, pp. 63–68.

Kulkarni, A., Can, M., and Hartmann, B. 2012. “Collaboratively Crowdsourcing Workflows with Turkomatic,” in *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work - CSCW ’12*, New York, New York, USA: ACM Press, p. 1003.

Lasecki, W., Teevan, J., and Kamar, E. 2014. “Information Extraction and Manipulation Threats in Crowd-Powered Systems,” *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*.

Laszka, A., Zhao, M., and Grossklags, J. 2016. *Banishing Misaligned Incentives for Validating Reports in Bug-Bounty Platforms*, Springer, Cham, pp. 161–178. (https://doi.org/10.1007/978-3-319-45741-3_9).

Law, E., Yin, M., Goh, J., Chen, K., Terry, M. A., and Gajos, K. Z. 2016. “Curiosity Killed the Cat, but Makes Crowdwork Better,” in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI ’16*, New York, New York, USA: ACM Press, pp. 4098–4110.

Leyden, J. 2016. “Fatigue Fears over Bug Bounty Programs • The Register.” (http://www.theregister.co.uk/2016/11/09/bug_bounty_fatigue_fears, accessed March 5, 2018).

Li, W., Tsai, W.-T., and Wu, W. 2015. *Crowdsourcing for Large-Scale Software Development*, Springer, Berlin, Heidelberg, pp. 3–23.

Liu, B., Shi, L., Cai, Z., and Li, M. 2012. “Software Vulnerability Discovery Techniques: A Survey,” in *2012 Fourth International Conference on Multimedia Information Networking and Security*, IEEE, November, pp. 152–156.

Maalej, W., and Nabil, H. 2015. “Bug Report, Feature Request, or Simply Praise? On Automatically Classifying App Reviews,” in *2015 IEEE 23rd International Requirements Engineering*

Conference (RE), IEEE, August, pp. 116–125.

Maillart, T., Zhao, M., Grossklags, J., and Chuang, J. 2017. "Given Enough Eyeballs, All Bugs Are Shallow? Revisiting Eric Raymond with Bug Bounty Programs," *Journal of Cybersecurity*.

Malone, T. W., Laubacher, R., and Dellarocas, C. 2010. "The Collective Intelligence Genome," *IEEE Engineering Management Review*, p. 38.

Marcus, A., Karger, D., Madden, S., Miller, R., Oh, S., Marcus, A., Karger, D., Madden, S., Miller, R., and Oh, S. 2012. "Counting with the Crowd," *Proceedings of the VLDB Endowment* (6:2), VLDB Endowment, pp. 109–120.

Munaiah, N., and Meneely, A. 2016. "Vulnerability Severity Scoring and Bounties: Why the Disconnect?," in *Proceedings of the 2nd International Workshop on Software Analytics - SWAN 2016,* New York, New York, USA: ACM Press, pp. 8–14.

OneLogin. 2017. "Private Bug Bounty Program" (https://www.onelogin.com/compliance/bug-bounty-program, accessed March 5, 2018).

Rao, H., Huang, S.-W., and Fu, W.-T. 2013. "What Will Others Choose? How a Majority Vote Reward Scheme Can Improve Human Computation in a Spatial Location Identification Task," *First AAAI Conference on Human Computation and Crowdsourcing*.

Research-office. 2015. "Guidelines: Active and Passive Snowballing," *Human Ethics Guide*, Sydney University. (http://sydney.edu.au/research_support/ethics/human/guidelines/snowballing.shtml, accessed April 20, 2016).

Russchen, M. 2018. "Double your signal, double your fun." (https://www.hackerone.com/blog/Double-your-signal-double-your-fun, accessed March 5, 2018).

Seaman, C. B. 1999. "Qualitative Methods in Empirical Studies of Software Engineering," *IEEE Transactions on Software Engineering* (25:4), pp. 557–572. (https://doi.org/10.1109/32.799955).

Stol, K., and Fitzgerald, B. 2014. "Two's Company, Three's a Crowd: A Case Study of Crowdsourcing Software Development," *Proceedings of the 36th International Conference on Software Engineering*.

Su, H.-J., and Pan, J.-Y. 2016. "Crowdsourcing Platform for Collaboration Management in Vulnerability Verification," in *2016 18th Asia-Pacific Network Operations and Management Symposium (APNOMS),* IEEE, October, pp. 1–4.

U.S. Department of Defense. 2016a. "Identifying Security Vulnerabilities in Department of Defense Websites – Hack the Pentagon | United States Digital Service." (https://www.usds.gov/report-to-congress/2016/hack-the-pentagon/, accessed March 5, 2018).

U.S. Department of Defense. 2016b. "Statement by Pentagon Press Secretary Peter Cook on DoD's 'Hack the Pentagon' Cybersecurity Initiative" (http://www.defense.gov/News/News-Releases/News-Release-View/Article/684106/statement-by-pentagon-press-secretary-peter-cook-on-dods-hack-the-pentagon-cybe).

Vijayan, J. 2017. "Bug Bounty Programs Are Growing Up Fast and Paying More," *DarkReading*. (https://www.darkreading.com/vulnerabilities---threats/bug-bounty-programs-are-growing-up-fast-and-paying-more/d/d-id/1328428?, accessed March 5, 2018).

Wolfson, S., and Lease, M. 2011. "Look before You Leap: Legal Pitfalls of Crowdsourcing," *Proceedings of the Association for Information Science and Technology*.

Zhao, M., Grossklags, J., and Chen, K. 2014. "An Exploratory Study of White Hat Behaviors in a Web Vulnerability Disclosure Program," in *Proceedings of the 2014 ACM Workshop on Security Information Workers - SIW '14,* New York, New York, USA: ACM Press, pp. 51–58.

Zhao, M., Grossklags, J., and Liu, P. 2015. "An Empirical Study of Web Vulnerability Discovery Ecosystems," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security - CCS '15,* New York, New York, USA: ACM Press, pp. 1105–1117.