An Exploratory Study of White Hat Behaviors in a Web Vulnerability Disclosure Program

Mingyi Zhao
College of Information Sciences and Technology
Pennsylvania State University
muz127@ist.psu.edu

Jens Grossklags
College of Information Sciences and Technology
Pennsylvania State University
jensg@ist.psu.edu

Kai Chen*
State Key Laboratory of Information Security, Institute of Information Engineering
Chinese Academy of Sciences
chenkai010@gmail.com

ABSTRACT

White hats are making significant contributions to cybersecurity by submitting vulnerability discovery reports to public vulnerability disclosure programs and company-initiated vulnerability reward programs. In this paper, we study white hat behaviors by analyzing a 3.5-year dataset which documents the contributions of 3254 white hats and their submitted 16446 Web vulnerability reports. Our dataset is collected from Wooyun, the predominant Web vulnerability disclosure program in China. We first show that Wooyun is continuously attracting new contributors from the white hat community. We then examine white hats' contributions along several dimensions. In particular, we provide evidence about the diversity inside Wooyun's white hat community and discuss the importance of this diversity for vulnerability discovery. Our results suggest that more participation, and thereby more diversity, contributes to higher productivity of the vulnerability discovery process.

Categories and Subject Descriptors

K.6.5 [**MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS**]: Security and Protection; D.2.5 [**SOFTWARE ENGINEERING**]: Testing and Debugging

Keywords

Vulnerability discovery; vulnerability disclosure; behavior.

1. INTRODUCTION

Undisclosed vulnerabilities in publicly and privately deployed software systems are an important contributing factor to potentially highly damaging security incidents. For example, the recent Heartbleed vulnerability discovered in OpenSSL has caused panic on a massive scale in the user community [3]. Malicious hackers (i.e., black hats) keep searching for unknown zero-day software vulnerabilities and attempt to derive (monetary) benefit by either exploiting such vulnerabilities to steal data and damage service availability, or even by selling information about such vulnerabilities on black markets [11, 20]. To reduce the number of vulnerabilities in their products, software vendors are utilizing various testing and auditing approaches. Yet, such efforts cannot eliminate all vulnerabilities because of, for example, economic constraints and technical complexity.

Ameliorating this state of affairs, benign hackers (i.e., white hats) hunt for vulnerabilities and may notify important stakeholder organizations directly, or may communicate their findings to public vulnerability disclosure programs (VDPs). Public VDPs, such as the BugTraq mailing list [1] that emerged more than 20 years ago, have been an important source for companies and the public to receive vulnerability reports from white hats. In addition, some companies such as Facebook, Google and Mozilla have established vulnerability reward programs (VRPs) that *pay white hats to hack*. A study based on the Google VRP and Mozilla VRP has shown that harvesting vulnerabilities from the white hat community is cost effective, and compares favorably to hiring full-time vulnerability researchers [10]. More recently, we begin to see start-up companies (e.g., BugCrowd, Synack and CrowdCurity) that act as brokers between white hats and software companies. Some of these companies such as BugCrowd and CrowdCurity are specialized on Web service vulnerabilities, because organizations' Web services have become a major attack surface.

This trend clearly shows that the white hat community is an important force to improve cybersecurity. Part of the reason is that some white hats are very skilled security researchers. In addition, involving the white hat community allows to harvest their diversity in expertise, approaches, etc. that may successfully complement the relatively narrow view of a company's internal security team. In fact, recent work suggests the importance of diversity in vulnerability discovery [8, 10].

However, while plenty of work has studied vulnerability disclosure data to understand trends and patterns of software vulnerabilities [6, 7, 18, 19, 21, 23], very little work has focused on the white hats that discover vulnerabilities [8, 10]. And these studies are either based on data from small controlled studies [8], or focused on company-initiated

*The work was done when Dr. Kai Chen was a visiting scholar at Pennsylvania State University.

SIW'14, November 7, 2014, Scottsdale, Arizona, USA.

http://dx.doi.org/10.1145/2663887.2663906.

VRPs [10]. As far as we know, no previous work has focused on the large white hat community in a representative public VDP.

We complement existing related work by conducting an exploratory study of white hat behaviors for vulnerability discovery. We collected a 3.5-year dataset from Wooyun[1], the predominant Web vulnerability disclosure program in China. We also chose this VDP because its data is publicly available and contains detailed information including authorship, bug type, submission time etc., for each vulnerability report. Through our analysis, we make the following contributions:

- We identify several trends of Web vulnerability disclosure and white hat behaviors on Wooyun. The VDP is continuously attracting more white hats, more vulnerability reports, and it captures an increasingly broader range of websites. In addition, white hats find more high severity vulnerabilities; however, their focus increasingly shifts to less popular websites.
- We explore white hat behaviors along the following three dimensions: (1) vulnerability counts, (2) vulnerability types, and (3) vulnerability discovery strategies. Our exploratory analysis provides further evidence for the existence of diversity, and the importance of diversity inside the white hat community [10].

Our findings constitute additional steps towards a comprehensive understanding and modeling of white hats, and have implications for designing better vulnerability disclosure and reward programs.

The rest of this paper is organized as follows. Section 2 discusses related work on studying the ecosystem of vulnerability discovery. Section 3 describes the dataset we obtained. In Section 4, we present our analysis results. In Section 5, we discuss the implications of our work, present ideas for future work, and offer concluding remarks.

2. RELATED WORK

2.1 Understanding the Behavior of Vulnerability Discoverers

Most research on vulnerability discovery and disclosure only focuses on vulnerabilities or affected software products, rather than the vulnerability discoverers. However, some efforts have been made to investigate the human side of this process.

Edmundson et al. conducted a code review experiment with 30 subjects. One of their findings is that none of the participants were able to find all 7 Web vulnerabilities embedded in the test code, but a random sample of half of the participants could cover all vulnerabilities with a probability of about 95% [8]. This conclusion indicates that diversity is important in vulnerability discovery. Finifter et al. analyzed the behavioral characteristics of participants in Google Chrome VRP and Mozilla Firefox VRP [10], and showed that VRPs are more cost-effective compared to hiring full-time security researchers. The authors suggest that "an increase in the number of researchers looking for vulnerabilities yields an increase in the diversity of vulnerabilities discovered."

[1]http://www.wooyun.org/

We similarly explore this aspect in our work (Section 4.3.2). However, our study considers a much larger dataset in terms of white hat contributors and vulnerability reports. Further, the vulnerabilities considered by the two cited studies above are mainly browser vulnerabilities, while our data set is about Web vulnerabilities.

Another interesting study analyzed vulnerability discoverers and vulnerability markets by interviewing the most prolific white hats [4]. The researchers found that top discoverers mostly rely on their expertise and insight, rather than automated vulnerability discovery tools.

2.2 Vulnerability Disclosure Programs

There has always been a debate on whether vulnerability disclosure programs are beneficial to society [9]. On the one hand, Rescorla showed that the pool of vulnerabilities in a software product is essentially infinite with respect to the effort and potential impact of white hats, and that vulnerabilities are found in no particular order. Therefore, black hats are likely to discover different vulnerabilities when compared to white hats' contributions. He suggests that the vulnerability discovery efforts do not provide much social benefit [21]. On the other hand, this conclusion is challenged by Ozment, who showed that the pool of vulnerabilities in OpenBSD 2.2 is being depleted and vulnerability rediscovery is common. Therefore, he gives the opposite conclusion, i.e., vulnerability hunting by white hats is socially beneficial [18].

Conceptual work has discussed different approaches to organize and design vulnerability markets [5]. For example, Ozment proposed a vulnerability auction mechanism that allows a software company to measure its software quality as well as encourage vulnerability discovery at an acceptable cost [17]. Further, previous research has been aimed to identify different existing markets [4].

Vulnerability disclosure data also provides a unique angle for the study of the dynamics of vulnerability discovery. Ozment et al. reported that the identification rate of vulnerabilities in OpenBSD is decreasing [19]. Shahzad et al. conducted a large-scale vulnerability life cycle study [23]. Their study showed that monthly vulnerability disclosures are decreasing since 2006 and that the complexity of the identified vulnerabilities is increasing. These researchers concluded that the security of software in general is increasing.

Clark et al. studied the impact of code familiarity on vulnerability discovery. They found that the time for finding the first vulnerability of a software product after release is primarily determined by familiarity with the code [7].

3. DATASET

3.1 Background

Wooyun, which launched in May 2010, is the predominant Web vulnerability disclosure program (VDP) in China. Until the end of 2013, it has attracted 3254 white hats to submit 16446 vulnerability reports. The websites that were found vulnerable by the white hats include many famous Chinese websites (see Section 4.1.3 and Section 4.3.1 for more details). Wooyun has also received significant media coverage during several security incidents [24, 2]. And partly due to the success of Wooyun, several leading Web companies have created their own vulnerability reward programs to utilize the great potential of white hats for improving their security.

As a VDP, Wooyun roughly works as follows. When a white hat finds a website vulnerability, she can submit a report to Wooyun. After an inspection of the report, Wooyun will inform the administrators of the vulnerable website about the vulnerability and give them 2 months to fix it. Then, the vulnerability report will be disclosed to the public. The motivation for disclosing the vulnerability is knowledge sharing and community learning.

3.2 Data Description

We have collected all vulnerability reports published on Wooyun from May 2010 to December 2013. Table 1 gives an overview of the data set. We also provide a graphical representation of the growth of the platform in Figure 1.

Variable	Value
Number of white hats	3254
Number of vulnerabilities	16446
Number of websites	4269

Table 1: Dataset Overview

For each vulnerability report available on Wooyun, we collected the following data types: (1) white hat's registration name, (2) target website, (3) target website's Alexa rank, (4) vulnerability type, (5) severity, (6) submission time. We further explain selected data types below.

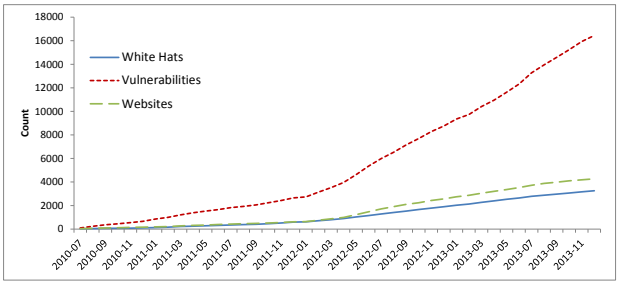

Figure 1: Growth of the Wooyun VDP visualized by overall number of white hats who joined the platform, contributed vulnerability reports, and websites included in reports.

The white hat's registration name is used as a unique identifier for a particular white hat. We believe this to be a reasonable assumption, however, there are the following caveats. First, a white hat could register several names on Wooyun, so that our study would falsely consider them as multiple white hats. We believe this to be uncommon because one of the major motivations for white hats to publish on Wooyun is reputation, and registering multiple names would dilute reputation. Second, there may be groups of white hats who publish under joint registration names. We are unaware of any such case on Wooyun. Further, Wooyun provides a separate page for team registration.

The target website name is contained in each report. We also collect the website's Alexa rank, which indicates its popularity. However, since the Alexa Top Sites service[2] requires the domain name of a website, we have to map the Wooyun data to domain names. To achieve this, we wrote a script that searches the website name on Google, and takes the first query result as the domain name. Subsequently, we retrieved the Alexa rank of all websites from the Alexa Top Sites service. Since websites on Wooyun are Chinese websites, we use the Chinese Alexa rank, rather than the global rank.

The vulnerability type of a report is chosen by white hats from a predefined list. We directly translated this list into English. In Table 2a, we list the Top 10 of the vulnerabilities types on Wooyun. The list contains major Web vulnerability types such as SQL injection and cross-site scripting (XSS). It is similar to the list of the 2013 Top 10 vulnerability types published by the Open Web Application Security Project (OWASP); see Table 2b [16] for comparison.

Type	Count
SQL Injection	3225
Cross-Site Scripting (XSS)	2659
Logic Erros/Design Flaws	1901
Command Execution	1245
Unauthorized Access	1114
Privilege Bypass	1005
Sensitive Information Leakage	737
Service Misconfiguration	639
Intrusion	573
File Upload for Arbitrary Code Execution	547

(a) Top 10 vulnerability types in Wooyun.

Type
Injection
Broken Authentication and Session Mgmt.
Cross-Site Scripting (XSS)
Insecure Direct Object References
Security Misconfiguration
Sensitive Data Exposure
Missing Function Level Access Control
Cross-Site Request Forgery (CSRF)
Using Components with Known Vulnerabilities
Unvalidated Redirects and Forwards

(b) Top 10 vulnerability types; OWASP 2013 [16].

Table 2: Top 10 vulnerability types from the Wooyun dataset and from OWASP 2013.

Upon submission, Wooyun adminstrators classify the severity level of a particular report as either high, medium or low.

There is other information available in the vulnerability reports, such as the technical description of the vulnerability and comments made by white hats. However, we defer the analysis of this additional information to future work.

4. RESULTS

4.1 Trends

In the following, we investigate potential trends regarding vulnerability disclosure on Wooyun from three different perspectives.

4.1.1 White Hat Community Engagement

In this subsection, we investigate white hat community engagement; that is, how many white hats have participated in Wooyun and how many of them are active per month. In Figure 1, we can observe that Wooyun has been success-

[2]http://aws.amazon.com/alexatopsites/

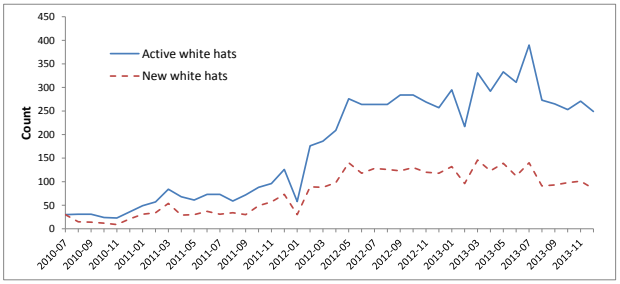

Figure 2: Monthly figures for the number of new white hats and active white hats.

fully attracting new white hats during the entire period of observation.

To understand new participation and general activity in more detail, we can break down the data into monthly figures. In Figure 2, we plot the number of new white hats for each one-month window, as well as the total number of white hats that have been active in the same month. A white hat is considered new if she submitted her first report in that month. We count as active white hats anybody who posts a vulnerability report in a particular month (i.e., this figure includes the newcomers). We observe that (on average) both measures have increased since the beginning. At the end of 2013, there were approximately 100 new white hats per month (while the active white hats count at the same time was about 250), which shows that Wooyun was steadily attracting new white hats from the outside pool of capable individuals.

We can also observe an obvious increase in both the number of active white hats and the number of new white hats in the beginning of 2012. We hypothesize that this increase in participation is caused by heightened publicity about Wooyun. At the end of 2011, a massive user information leakage incident occurred in China [24], and some related vulnerabilities of several important websites were published on Wooyun that increased its status in the community.

4.1.2 Number and Severity of Vulnerabilities

We next examine how the figures for vulnerability report submissions and their severity have evolved. Figure 3 shows the monthly count of vulnerability reports. The total number of vulnerability reports per month increased from 92 in July 2010 to 512 in December 2013. However, the peak number of vulnerability reports per month is 959, which occurred in July 2013. We can also observe a sharp increase in the beginning of 2012 caused by the sudden increase in the number of white hats on Wooyun (see Figure 2).

By breaking down the number of vulnerability reports according to severity, we observe that submissions of high and medium severity vulnerabilities keep increasing during the period of observation. However, for low severity vulnerabilities, the number has been decreasing since May 2012. Several potential explanations may contribute to the latter effect. Wooyun administrators can reject submissions that they deem unsuitable which could have impacted the number of published reports with low severity in a noticeable fashion. Further, a reduced pool of low severity vulnerabilities on Chinese websites, or a different vulnerability search behavior of Wooyun contributors could potentially explain the data.

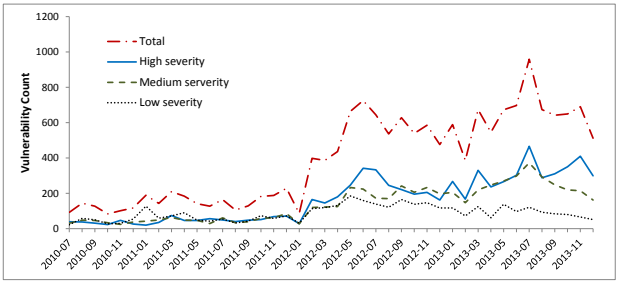

Figure 3: Monthly total number of vulnerability reports and number of reports broken down by level of severity.

4.1.3 Target Websites

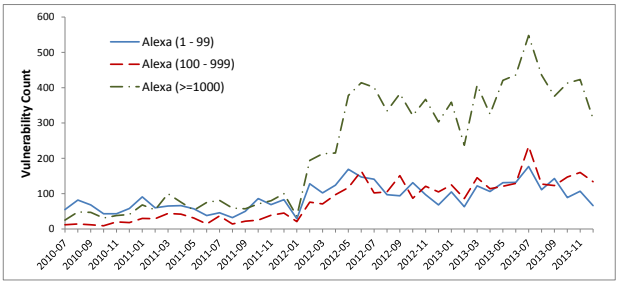

Figure 4: Monthly number of vulnerability reports for different Alexa groups.

We next analyze how white hats select target websites over time. Particularly, we are interested in whether the fame of a website is related to their decisions. In principle, it would be more rewarding to "hack" a website with millions of visitors than an unpopular website. To research this question, we categorize vulnerability reports based on the fame of the target websites, i.e., we use a website's Alexa rank to assess its popularity and divide websites into three groups: Alexa rank $1-99$ (high-alexa), Alexa rank $100-999$ (medium-alexa) and Alexa rank ≥ 1000 (low-alexa).[3]

Figure 4 shows the trends for the number of vulnerability reports in each group. In the beginning, the majority of the submitted vulnerabilities concern the famous websites (even though it is a much smaller group). Gradually, while the number of vulnerability reports in all three groups increased, reports for low-alexa websites started to outnumber vulnerabilities of the other two groups.

There could be multiple reasons that help to explain this trend. First, since there are increasingly more white hats, the competition will drive white hats to look for new targets. Second, the pool of vulnerabilities in the high-alexa group may be depleted which increases the difficulty of finding unknown problems. Some white hats may therefore shift their

[3]Websites for which we could not find an Alexa rank were added to the low-alexa group because they presumably have low popularity.

attention to low-alexa websites, which are likely less secure and more vulnerable. Third, increased white hat participation may lead to more diversity in the community and, therefore, may expand the set of target websites.

4.2 Behaviors

Next, we will investigate the vulnerability discovery behavior of white hats from three perspectives.

4.2.1 Number of Submitted Reports

In Figure 5, we plot the number of vulnerability reports submitted by individual white hats, and notice that the curve follows a very skewed distribution. During the observation time, only a small number of white hats were productive and published more than 50 vulnerabilities, while most submitted only very few reports. The maximum submission number is 291 and the average is 4.8 reports. Our observation is consistent with previous work which shows a similar pattern for the number of vulnerability reports (rated high or critical) by each participant in the Chrome and Firefox VRPs [10].

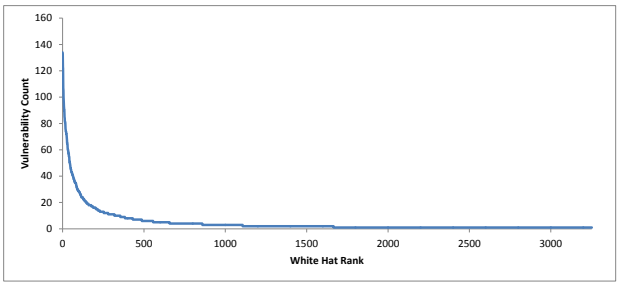

Figure 5: Distribution of number of vulnerability reports for individual white hats.

This distribution is similar to the power law distribution in many complex systems [14], i.e., $p(x) = Cx^{-\alpha}$. Fitting our data to this distribution, we obtain $\alpha = 2.143$ with $\sigma = 0.044$. Interestingly, the value of α corresponds approximately to Lotka's law [13], which states that the authors' frequency of publication activity in a scientific field follows a power law distribution; and in many fields, $\alpha \approx 2$. This observation might hint at parallels between the processes for vulnerability discovery and scientific discovery. In future work, we are interested in developing a model to explain the origin of the power law distribution in vulnerability discovery [14].

4.2.2 Vulnerability Types

Since discovering specific types of Web vulnerabilities usually requires particular knowledge and skills, we expect that white hats' vulnerability discovery behaviors exhibit such specialization. We follow a clustering approach to investigate whether such specialization is represented in our data where each white hat is represented as a vector (i.e., each dimension corresponds to a vulnerability type and the value is the number of vulnerabilities in that type found by the white hat). For the analysis, we focus on white hats with more than 10 discoveries and vulnerability types with more than 100 reports, which leaves us with 355 white hats and 23 vulnerability types. Next, we normalize each sample and apply k-means clustering.[4] The size and labeling of the 7 identified clusters is listed in Table 3.

White hats in the same cluster are similar in terms of their specialization. In particular, we observe a high degree of specialization on SQL injection as well as cross-site scripting (XSS) vulnerabilities. Further, vulnerability types in the same cluster likely share common aspects, which could improve our understanding of vulnerabilities and help with classification.

Cluster	Size	Labels
1	103	SQL Injection
2	84	XSS
3	45	CSRF, Design Flaws/Logic Errors, Insufficient Account Control
4	42	Sensitive Information Leakage, Arbitrary File Traversal, Weak Password, URL Redirect, Intrusion, File Upload for Arbitrary Code Execution
5	39	Application Misconfiguration, Command Execution
6	22	Unauthorized Access, Privilege Bypass
7	20	Service Misconfiguration, Untimely Service Patch

Table 3: Identified clusters of white hat specialization

4.2.3 Vulnerability Discovery Strategies

White hats do not follow a random process during their hunt for vulnerabilities. Rather, they likely follow certain strategies, which will be reflected in the sequence of vulnerability discoveries. One possible strategy is to search for a specific type of vulnerability during a period of time, because each type of vulnerability requires a unique set of skills. Another possible strategy is to thoroughly explore one particular website for a period of time, because familiarity with the website's architecture and code will facilitate vulnerability discovery [7].

We use the following simple model to identify whether these basic white hat strategies are utilized. The vulnerability submission record of a white hat is a sequence $v_1, v_2, ..., v_n$, where each element v_i is a vulnerability report and has two fields, the target website $v_i.site$ and the vulnerability type $v_i.type$. We define a transition from v_i to v_{i+1} as a *same site transition* if $v_i.site = v_{i+1}.site$, and a *same type transition* if $v_i.type = v_{i+1}.type$. We calculate the percentage of same site transitions, p_s, and the percentage of same type transitions, p_t, as follows:

$$p_s = \frac{\sum_{i=1...n-1} I(v_i.site = v_{i+1}.site)}{n-1}$$

$$p_t = \frac{\sum_{i=1...n-1} I(v_i.type = v_{i+1}.type)}{n-1}$$

Where I is an indicator function.

[4]To determine the most appropriate number of clusters, k, we test k from 2 to 9, and find that 7 clusters is associated with the highest silhouette value [22]. To label these clusters, we analyze the centroid of each cluster and assign the vulnerability type to the cluster whose centroid has the highest value on that dimension.

Since a white hat could use a combination of these two strategies, we use (p_s, p_t) to represent a white hat's overall vulnerability discovery strategy. We plot these points for all white hats with more than 15 vulnerability submissions in Figure 6.

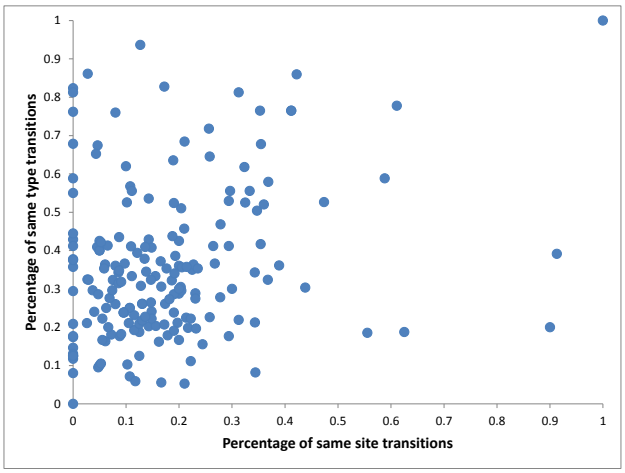

Figure 6: White hat strategy scatterplot.

From Figure 6, we observe that p_s and p_t for most white hats are both non-zero; that is, most white hats use both strategies during vulnerability discovery. We further notice that *same type transition* is more frequent than *same site transition*, because most dots are located above the diagonal. This suggests that focusing on the same website provides benefits from familiarity, but moving to a new website (with potentially similar vulnerabilities) also contributes to the chance of discovery. Interestingly, we found one white hat with $p_s = p_t = 1$, meaning that his whole submission sequence only concerns a single type of vulnerability on one particular website.

4.3 Diversity

We have examined the existing diversity of white hat behaviors in Section 4.2. Now, we would like to further understand the relationship between diversity and vulnerability discovery. Previous work suggests that a larger white hat population yields better vulnerability discovery results [10, 8]. And a unique advantage of VDPs and VRPs is the utilization of the diversity in a large white hat community.

In this subsection, we first analyze the correlation between white hat diversity and the number of vulnerability reports. We also analyze the relationship between white hat diversity and vulnerability report diversity.

4.3.1 Diversity and Productivity

By comparing Figures 2 & 3, we observe that the trend for the total number of vulnerability reports and the trend for the number of active white hats are very similar, which suggests that the number of active white hats is critical to the identification of vulnerabilities.

To gain further insights, we select the five websites with the most reported vulnerabilities in the data set. In Table 4, we collect the number of vulnerability reports for these websites, the number of white hats who submitted the reports and data about the top contributor. We observe that the number of white hats for these websites is close to the number of vulnerabilities discovered. This indicates that the community as a whole, rather than merely very few expert white hats, plays a key role for vulnerability discovery.

Website Name	Vul. count	White hat count	Top contributor
Baidu	595	471	23
Tencent	1005	781	30
163	306	235	12
Sina	760	535	16
Sohu	264	196	9

Table 4: Statistics for five representative websites.

The top contributor column lists the number of the contributed vulnerability reports by those individuals. In all cases, they constitute only a small fraction of all vulnerability reports. While rewarding top contributors may be beneficial, attracting more white hats to participate is equally helpful.

To further study this effect, we select all websites with more than 20 vulnerabilities and correlate the number of submitted vulnerability reports with the number of active white hats for each website. There are such 85 data points in total and the Pearson correlation coefficient is 0.987 (p-value = 0.000). Such high positive correlation further indicates that increasing the number of participants, and thereby diversity, is very important for vulnerability discovery.

4.3.2 Head vs. Tail

Previous work proposes the hypothesis that "an increase in the number of researchers looking for vulnerabilities yields an increase in the diversity of vulnerabilities discovered" [10]. In the following, we conduct a preliminary investigation of this hypothesis by comparing two groups which we call the head group and the tail group. The *head group* contains the most productive white hats, who discovered a lot more vulnerabilities than the average contributor. In Figure 5, they are on the left side of the graph. Members of this group likely represent experts in vulnerability discovery. (Companies like Google and Mozilla have hired top white hats through their VRPs [10].) The remaining white hats form the *tail group* which appears as the long tail in Figure 5. Those individuals published much fewer vulnerability reports and the size of this group is much larger than the first one. Due to this large size, we also assume that the tail group contains more diversity than the head group.

We divide all white hats into these two groups in the following way. We first sort all white hats based on their vulnerability report numbers in descending order. We then find the cutoff point such that the number of vulnerabilities discovered by white hats before the point (head group) and after the point (tail group) is roughly the same. Basic statistics of these two groups are listed in Table 5, for example, head group members discovered 43 vulnerabilities on average, while tail group members only published 3 reports on average.

Figure 7 shows that the two groups exhibit similar trends in terms of the monthly number of reported vulnerabilities. Figure 8 shows that the distribution of vulnerability types

Variable	Head	Tail
Number of white hats	191	3063
Number of vulnerabilities	8226	8220
Average Productivity	43	3

Table 5: Basic statistics of head and tail white hat groups.

is also relatively similar. The head group discovers slightly more SQL injection and XSS vulnerabilities, while the tail group discovers slightly more logic errors, command execution and privilege bypass vulnerabilities.

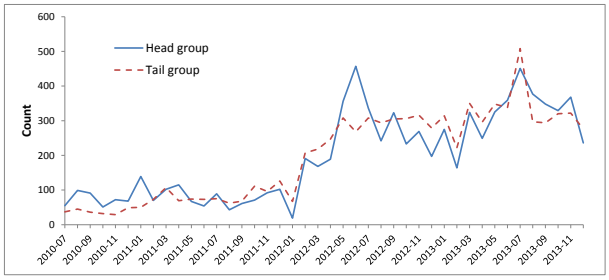

Figure 7: Comparing vulnerability trends of the head group and the tail group.

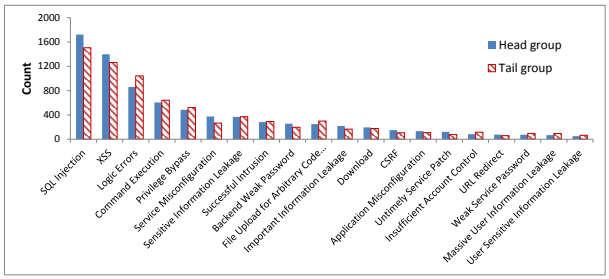

Figure 8: Comparing the vulnerability type distribution of the head group and the tail group.

Table 6 lists additional statistics for comparison. We first compare the number of target websites for which a group has found at least one vulnerability. The result shows that the tail group has a wider target range than the head group. This means that the tail covers more potentially vulnerable websites which can be considered as an advantage of them. The groups do not differ substantially, when we break down the data by Alexa popularity. Finally, when comparing the severity of vulnerabilities found by these two groups, we see that the head group produces slightly more high severity vulnerabilities than the tail group, which can be interpreted as a sign of expertise.

5. DISCUSSION AND CONCLUDING REMARKS

In this work, we have studied white hat behaviors for Web vulnerability discovery based on a Wooyun data set. We show that this VDP is continuously attracting more white hats and receiving more vulnerability reports. We examine the diversity in white hat behaviors along the following three

Variable	Head	Tail
Number of websites	2031	2914
Alexa high	28.7%	28.7%
Alexa medium	26.4%	23.9%
Alexa low	44.9%	47.3%
Severity high	45.7%	39.1%
Severity medium	33.1%	37.4%
Severity low	21.3%	23.5%

Table 6: Comparison of head and tail group.

dimensions: (1) vulnerability counts, (2) vulnerability types, and (3) vulnerability discovery strategies. Our analysis provides further evidence for the existence of diversity, and the importance of diversity inside the white hat community.

Our work is an exploratory study of white hat behaviors and several opportunities for further work exist. First, the current study is mostly focused on the quantity of vulnerability reports and does not investigate the details of the reports. Since Wooyun discloses certain technical aspects of the vulnerabilities, it is possible to analyze the white hat behaviors with respect to this more detailed information.

Second, it would also be interesting to examine the interactions between white hats, for example, how one vulnerability report influences other white hats' behaviors.

A third direction is to more closely examine the temporal characteristics of white hat behaviors. An example research question is how the vulnerability report type of a white hat changes over time.

Fourth, the white hat data would be applicable to model the dynamics of vulnerability discovery, and to potentially make predictions about future vulnerability disclosure trends similar to [6]. We also plan to consider other vulnerability disclosure data sources such as the Open Sourced Vulnerability Database (OSVDB) [15], which partly contains authorship information.

Suggestions for VDPs and VRPs. From the preliminary results in our work, we suggest that managers of VDPs and VRPs should not only focus on the top contributors, but also try to attract as many white hats as possible as contributors. More participation would likely translate in more diversity during the search process and more discoveries. However, this insight might require the design of new mechanisms for organizing and rewarding white hats. For example, a potential downside of including more white hats is the increasing probability of rediscovery of the same vulnerability. Since only the first discoverer gets the credit, the effort of other participants is ignored. The issue of rediscovery has been observed in previous work [18]. A good VRP/VDP mechanism probably should reduce the chance of multiple discoveries (or multiple overlapping reports); perhaps through better search functions.

Another possible suggestion is to disclose more technical details of past vulnerabilities, so that white hats can learn from others' findings. Wooyun's full disclosure model, which allows the reading of the white hats' comments in the vulnerability reports, likely helps new and even experienced white hats to learn. We plan to investigate the impact of disclosure models in future work.

Comparing Wooyun and VRPs. Inspired by Wooyun and sometime embarrassed by the vulnerability reports on Wooyun, some Chinese website companies have established

their own vulnerability reward programs. These VRPs usually offer attractive rewards, but require the white hats not to disclosure the vulnerability to the public, while Wooyun asks the white hats to disclose certain vulnerability details for the purposes of knowledge sharing and learning in the white hat community. It would be interesting to examine how these two kinds of vulnerability programs compete and complement each other.

In summary, we suggest that research should not only focus on the study of the technical details of vulnerabilities and the design of new vulnerability discovery tools. It is also important to understand how vulnerability hunters make their discoveries. After all, previous work [12, 4] suggest that vulnerability discovery today still largely relies on the knowledge and experience of the contributors.

Acknowledgments

We thank the anonymous reviewers for their helpful comments. The authors would also like to thank Dr. Peng Liu, Yue Zhang, Chen Zhong and Jun Wang for their valuable comments on an earlier version of this paper.

6. REFERENCES

[1] Bugtraq. Background information available at: `http://en.wikipedia.org/wiki/Bugtraq`, 2014. [Online; accessed 03-September-2014].

[2] China's no.1 online travel firm ctrip hit by security scare. Available at: `http://english.cntv.cn/program/bizasia/20140324/103772.shtml`, Mar. 2014. [Online; accessed 03-September-2014].

[3] Heartbleed. Available at: `http://en.wikipedia.org/wiki/Heartbleed#Affected_services`, 2014. [Online; accessed 03-September-2014].

[4] A. Algarni and Y. Malaiya. Software vulnerability markets: Discoverers and buyers. *International Journal of Computer, Information Science and Engineering*, 8(3):71–81, 2014.

[5] R. Böhme. A comparison of market approaches to software vulnerability disclosure. In G. Müller, editor, *Emerging Trends in Information and Communication Security*, volume 3995 of *Lecture Notes in Computer Science*, pages 298–311. Springer Verlag, 2006.

[6] M. Bozorgi, L. Saul, S. Savage, and G. Voelker. Beyond heuristics: Learning to classify vulnerabilities and predict exploits. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 105–114, Washington, DC, July 2010.

[7] S. Clark, S. Frei, M. Blaze, and J. Smith. Familiarity breeds contempt: The honeymoon effect and the role of legacy code in zero-day vulnerabilities. In *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC)*, pages 251–260, Austin, TX, Dec. 2010.

[8] A. Edmundson, B. Holtkamp, E. Rivera, M. Finifter, A. Mettler, and D. Wagner. An empirical study on the effectiveness of security code review. In J. Jürjens, B. Livshits, and R. Scandariato, editors, *Engineering Secure Software and Systems*, pages 197–212. Springer Verlag, 2013.

[9] S. Egelman, C. Herley, and P. van Oorschot. Markets for zero-day exploits: Ethics and implications. In *Proceedings of the 2013 New Security Paradigms Workshop (NSPW)*, pages 41–46, Banff, Canada 2013.

[10] M. Finifter, D. Akhawe, and D. Wagner. An empirical study of vulnerability rewards programs. In *Proceedings of the 22nd USENIX Security Symposium*, Washington, DC, Aug. 2013.

[11] S. Frei, D. Schatzmann, B. Plattner, and B. Trammell. Modeling the security ecosystem - The dynamics of (In)Security. In T. Moore, D. Pym, and C. Ioannidis, editors, *Economics of Information Security and Privacy*, pages 79–106. Springer Verlag, 2010.

[12] S. Heelan. Vulnerability detection systems: Think cyborg, not robot. *IEEE Security & Privacy*, 9(3):74–77, May-June 2011.

[13] A. Lotka. The frequency distribution of scientific productivity. *Journal of Washington Academy Sciences*, 16(12):317–323, 1926.

[14] M. Newman. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics*, 46(5):323–351, September-October 2005.

[15] Open Sourced Vulnerability Database (OSVDB). Available at: `http://osvdb.org/`.

[16] Open Web Application Security Project (OWASP). 2013 Top 10 list. Available at: `https://www.owasp.org/index.php/Top_10_2013-Top_10`, 2013. [Online; accessed 03-September-2014].

[17] A. Ozment. Bug auctions: Vulnerability markets reconsidered. In *Proceedings of the Third Workshop on the Economics of Information Security (WEIS)*, Minneapolis, MN, May 2004.

[18] A. Ozment. The likelihood of vulnerability rediscovery and the social utility of vulnerability hunting. In *Proceedings of the Fourth Workshop on the Economics of Information Security (WEIS)*, Cambridge, MA, June 2005.

[19] A. Ozment and S. Schechter. Milk or wine: Does software security improve with age? In *Proceedings of the 15th USENIX Security Symposium*, Vancouver, Canada, July-August 2006.

[20] J. Radianti. Eliciting information on the vulnerability black market from interviews. In *Proceedings of the 4th International Conference on Emerging Security Information Systems and Technologies (SECURWARE)*, pages 154–159, Venice, Italy, July 2010.

[21] E. Rescorla. Is finding security holes a good idea? *IEEE Security & Privacy*, 3(1):14–19, January-February 2005.

[22] P. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, Nov. 1987.

[23] M. Shahzad, M. Shafiq, and A. Liu. A large scale exploratory analysis of software vulnerability life cycles. In *Proceedings of the 2012 International Conference on Software Engineering*, pages 771–781, Zurich, Switzerland, June 2012.

[24] Wikipedia. 2011 User Information Leakage Incident in Chinese Websites. Available at: `goo.gl/0UwgkW`. [Online; accessed 03-September-2014].